Advanced Intelligent Control through Versatile Intelligent Portable Platforms

Edited by

Luige Vladareanu

MDPI

Advanced Intelligent Control through Versatile Intelligent Portable Platforms

Advanced Intelligent Control through Versatile Intelligent Portable Platforms

Editor

Luige Vladareanu

Contents

About the Editor

Luige Vladareanu is Head of Robotics and Mechatronics Department of Institute of Solid Mechanics, Romanian Academy. His scientific work is focused on real time control in solid mechanics applied, in robot trajectory control, hybrid position – force control, multi-microprocessor systems for robot control, acquisition and processing of experimental physical data, experimental methods and signal processing, nano-micro manipulators, semi-active control of the mechanical system vibrations, semi-active control of the magnetorheological dissipaters systems, complex industrial automations with programmable logical controllers in distributed and decentralized structure. He has published over 35 books and book chapters, 11 edited books, and over 200 papers in journals, proceedings, and conferences in these areas. He has served as director and coordinator of over 15 grants of international and national research development programs in the last 5 years, 15 invention patents, and the development of 17 advanced work methods resulting from applicative research activities and more than 60 research projects. He is the author of the *virtual projection method* known as Vladareanu–Munteanu method, of the *robot neutrosophic control RNC method* known as the Vladareanu–Smarandache method, and the method of the *extended (extenics) hybrid force/position control eHFPC* of robots, applied as the Vladareanu–Smarandache–Sandru method. He is the winner of two prizes and Gold for Excellence in Research 2000, SIR 2000, of the Romanian Government and the Agency for Science, Technology and Innovation as well as 9 International Invention and Innovation Competition Awards and Gold in the World Exhibition of Inventions, Geneva, 2007–2016, in addition to 9 International Invention Awards and Gold at the Brussels, Zagreb, and Bucharest International Exhibitions. He received the "TraianVuia" (2006) award of the Romanian Academy, Romania's highest scientific research forum, for a group of scientific papers published on real time control in solid mechanics. He is a Corresponding Member of the American Romanian Academy and a member of the International Institute of Acoustics and Vibration (IIAV), Auburn University, USA (2006), ABI's Research Board of Advisors, American Biographical Institute (2006), World Scientific and Engineering Academy Society, WSEAS (2005), International Association for Modelling and Simulation Techniques in Enterprises (AMSE), France (2004), National Research Council from Romania (2003–2005), etc. He is a PhD advisor in the field of mechanical engineering and robotics at the Romanian Academy. He has served as organizer of numerous international conferences, including as the General Chair of four WSEAS International Conferences (http://www.wseas.org/conferences/2008/romania/amta/index.html), chaired Plenary Lectures at Houston 2009; Harvard, Boston, 2010, and Penang, Malaysia, 2010; Paris 2011; Florence 2014; Tenerife 2015 WSEAS International Conferences, is team leader of the WSEAS scientific research project "Mechanics and Robotics Systems", and currently serves on the board of various other conferences and academic societies.

Editorial

Advanced Intelligent Control through Versatile Intelligent Portable Platforms

Luige Vladareanu

Department of Robotics and Mechatronics, Institute of Solid Mechanics of the Romanian Academy,
010141 Bucharest, Romania; luigiv2007@gmail.com

Abstract: Deep research and communicating new trends in the design, control and applications of the real time control of intelligent sensors systems using advanced intelligent control methods and techniques is the main purpose of this research. The innovative multi-sensor fusion techniques, integrated through the Versatile Intelligent Portable (VIP) platforms are developed, combined with computer vision, virtual and augmented reality (VR&AR) and intelligent communication, including remote control, adaptive sensor networks, human-robot (H2R) interaction systems and machine-to-machine (M2M) interfaces. Intelligent decision support systems (IDSS), including remote sensing, and their integration with DSS, GA-based DSS, fuzzy sets DSS, rough sets-based DSS, intelligent agent-assisted DSS, process mining integration into decision support, adaptive DSS, computer vision based DSS, sensory and robotic DSS, are highlighted in the field of advanced intelligent control.

Keywords: intelligent control; robot control intelligent sensor systems; intelligent decision support systems; versatile intelligent portable platforms; new technologies; adaptive sensor networks; virtual and augmented reality; intelligent remote control and communication

1. Introduction

Advanced intelligent control is a rapidly developing, complex and challenging field with great practical importance and potential, which is addressed by the authors to foster the advance of science and technology and provide the theoretical and practical considerations of intelligent control techniques and their application using intelligent sensors, integrated through versatile intelligent portable platforms.

Intelligent control is the control method which imitates human intelligence in learning, decision-making and problem solving. Human characteristics consist of experience, learning, adapting and changing methods of approach and solving problems. Intelligent control techniques allow for the development of an environment which leads to recreating the advantages of natural intelligence with artificial intelligence. Advances in sensors, actuators, computation technology and communication networks help provide the necessary tools for the implementation of intelligent control hardware. Practical applications using intelligent sensors for this control method, emerged from artificial intelligence and computer-controlled systems as an interdisciplinary field, are aimed toward a variety of relevant scientific research fields on machine learning, including deep learning, bio-inspired algorithms, petri nets, recurrent neural networks, neuro-fuzzy control, bayesian control, genetic control, intelligent agents (cognitive/conscious control), extensions to traditional techniques—such as Neutrosophic logic—Extenics control, and artificial intelligence in general.

Creating new technologies using advanced intelligent control through versatile intelligent portable platforms involves complex multidisciplinary research covering: enhanced IoT technologies and applications in the 5G densification era; bio-inspired techniques in future manufacturing enterprise

control; cyber-physical systems approach to cognitive enterprise; developing the IT Industry 4.0 concept; industrial systems in the digital age; cloud computing; robotics and automation with applications such as human aid mechatronics moving in unstructured and uneven environments; rescue robots; firefighting robots; rehabilitation robots; robot-assisted surgery; domestic robots.

2. Review of the Contributions in This Special Issue

Advanced intelligent control is an inter-disciplinary field which combines and extends theories and methods from control theory, computer science and operations research areas with the aim of developing controllers which are highly adaptable to significant unanticipated changes.

Illustrative wide palette approaches in advanced intelligent control through versatile intelligent portable platforms are presented in this Special Issue.

"The Intelligent Cyber Enterprise" starts from new emerging paradigms, such as the Internet of Things and Cyber-Physical Systems, focused on adopting new technologies in order to become agile, safe and productive, and capable of interoperating with smart manufacturing applications [1]. The paper introduces the concept of the intelligent cyber-enterprise using information and knowledge dynamics focusing on the importance of appropriately adapting external and internal perceptions of an enterprise through a new generation of sensorial systems—the perceptive interfaces which led to new concepts on intelligent interface instance registry, intelligent interface repository, semantic routing systems and middleware ontology, etc.

The main characteristics of Intelligent Cyber Enterprise, in order to become agile, safe and productive, are identified: Perception; Mobility of Systems; Human–Machine Interaction; Agility of Industrial Systems; Embodied AI and Data Generation for Manufacturing; Collaboration; Safety Performance. Utilizing these characteristics allow machines interoperate with smart manufacturing applications and to better evolve. How data acquired from sensors or other system components can be analyzed and used in determining interface behaviors are demonstrated in this paper.

The Intelligent Cyber Enterprise model integrates key functions, such as processing, perception, communication, learning, pattern recognition, data mining facilitating the system adaptation to a dynamic working environment and illustrating the advantages in relation to complex system behavior modeling.

Object segmentation masks instance classification, together with YOLOv3 with improved design. 3D object reconstruction and prediction with an extended YOLOv3 network is addressed through intelligent versatile applications using full 3D, depth-based two streams, especially in the scenarios of intelligent remote control and communications, where virtual and augmented reality will soon become outdated and are forecast to be replaced by point-cloud streams, providing explorable 3D environments of communication and industrial data [2]. A hybrid artificial neural network for reconstructing polygonal meshes using a single RGB D frame and prior knowledge is proposed. The method entails the requirement of a priori information for the captured object reconstruction and a need for a large well-labelled element dataset. The training data consist exclusively of synthetically generated datasets which use ShapeNetCore, a subset of the ShapeNet dataset that provides 3D object models spanning 55 categories.

The real-life data acquired by the Intel Realsense ZR300 and Intel Realsense D435i (Intel Corp., Santa Clara, CA, USA) devices were used for visual validation, being impossible to measure objectively without having a 3D artist recreating a 1:1 replica of said objects. The modified hybrid artificial neural networks have improved the reconstruction results by 8.53%, which allows for the much more precise filling of occluded object sides and a reduction in noise during the process.

The Special Issue continues with the presentation of six papers that deepen the scope and discuss new trends in the design, control and applications of the real-time control of robots, mechatronic systems, and human aid mechatronics or HAM (Human Adaptive Mechatronics) using advanced intelligent control methods and techniques.

In the field of the intelligent rehabilitation robots, two papers aim to help patients in achieving complete active rehabilitation training.

New active rehabilitation training on lower limb robot motion measurement is approached, based on the dynamic modeling of human–machine coordination, using the static torque sensors for detecting the patient leg motion intention [3]. Using the dynamic relationship between the patient leg and robot leg mechanism and the variation of the leg mechanism joint torques, an innovative modeling is analyzed by which the robot completes the patient's motion intention for active rehabilitation training. The mechanism and the hardware control system design of the LLR-Ro, the patient's lower limb motion intention acquisition, the angular position and angular velocities of each rod centroid, each joint torque and the contact force between the patient's leg and the leg mechanism were identified, modeled and determined. Based on the variations in the joint torques, the principle of detecting the volunteer motion intention is clear and feasible.

The patient leg motion intention and the active rehabilitation training, which improve the patient's training initiative and accelerate the rehabilitation process, are detected based on the variation of torque sensors installed on the leg mechanism joint axis, LLR-Ro, the active training control strategy, using the joint static torque sensors and motion intension acquisition, correlated with biomechanics concepts.

An alternative to the process of rehabilitation training for stroke patients, compared to most of the methods, which process EMG signals or oxygen consumption for patients' participation measurements, uses high cost and high complexity robotic devices, a multi-sensor system robot with torque and six-dimensional force sensors integrated in advanced intelligent control, applying the support vector machines [4]. The support vector classifiers and regression machines were used to predict the degree of the patient's task participation, taking into account the small sample and non-linear data of the patients' training and questionnaire data. The C and σ parameters, reported to the patients' participation, are optimized by the hybrid quantum particle swarm optimization and support vector machines (Hybrid QPSO-SVM) algorithm. QPSO optimizes two key parameters, the C and σ of the MLSSVM, and the optimization goal is minimizing the fitness (σ, γ) function.

A further challenge is advancing intelligent control on improving upon such clinical trial data. The task difficulty can be judged and predicted online, and the assistant force adjusted in real time, for active and optimal training.

The intelligent haptic robot-glove (IHRG), for the monitoring or control of human behavior, is well described by the fractional order model (FOM) operators for the rehabilitation of patients that have a diagnosis of a cerebrovascular accident [5]. An exoskeleton architecture ensures the mechanical compliance of human fingers. The modelling is based on Lyapunov techniques, the methods that derive from Yakubovici–Kalman–Popov lemma, the frequency criteria that ensure asymptotic stability of the closed loop system and an observer control for the complex models, exoskeleton and sensors.

The use of the dynamics of an exoskeleton hand through fractional order proposes intelligent control solutions for a larger class of complex systems, such as hyper-redundant systems. An IHRG versatile intelligent portable platform, attached to the exoskeleton, supports the human hand activities by using the driving and skin sensor system, including an intelligent control for dexterous grasping and manipulation, mimicking the mechanical compliance of human fingers and determining comfortable and stable grasping functions.

Advanced intelligent control opens new approaches for high precision positioning, based on the adaptive Kalman fusion algorithm for multiple cameras and fiducial markers using multiple sensor data in complex infrastructures, in which the issue of proper information implies complex considerations with respect to system dynamics, flexibility, efficiency, safety and in defining the emergent interaction with a highly dynamic and sparsely defined environment [6]. The use of solid target alignment, in which multiple optical diagnostics are positioned using motorized 3–6 DOF manipulators, utilizes multiple instruments which need to be precisely positioned relative to each other during the experiment. The method could achieve the relative and simultaneous positioning of multiple fiducial markers in the development of advanced applications.

The Connected Bike concept combines several technologies—both hardware and software—to provide a modern alternative solution for the training and processing of training data via an internet server, leading to smart connected bike for personalized training, using the interaction of future-oriented mobility and state-of-the-art data technology [7]. Taking into account the multiple IoT specific architectures, the most suitable one for this research proved to be a hybrid open architecture, including sensors, microcontrollers, web applications, GPS module, wireless and infrared communications.

Developed as an IoT system, the Connected Bike system has a server for the management of the MQTT broker, monitoring intermediation of messages between clients, and the Back-End web application for holding the data transmitted from the bicycle and providing functionality to the Front-End. The Connected Bike system uses a wide range of technologies, starting from the electronics and hardware side, to the web and mobile applications.

A portable air scanning system was developed using a quadcopter, equipped with an air scanning sensor to perform air quality measurements. Using the Computational Fluid Dynamic (CFD) simulation, the vortex field generated by the propeller was analyzed to determine the best place for sensor mounting in order to increase the response and the accuracy of the sensor-collected data [8]. The grid pattern, with a point source and non-point source, and a wind algorithm were integrated into the gas measurement process. The DM+V kernel algorithm is used for the analysis of gas dispersion, measured by the quadcopter using convolution with a two-dimensional Gaussian kernel.

A versatile intelligent portable platform permits the quadcopter to perform optimally, reaching the target point set through the GPS coordinates. Quadcopter flight behavior, in the form of altitude, speed and measurement pattern, includes an open source web application on the ground station, which allows for remote controlling to force the quadcopter, in case of an emergency, to make a crash-landing or to fly back to the home coordinate. The flight controller is required to maintain the stability of the maneuvers and the sniffer system for performing air scanning, and also for saving data on a memory card. The analysis using normal dispersion and ANOVA were essential to obtaining increased accuracy in terms of the gas concentration and gas source position.

Among intelligent portable platforms, smartphones are some of the most ubiquitous [9]. They provide primary access to the internet and modern amenities, hold our private data and are becoming one of the primary means of attack against the user, be it through power viruses (or other means to consume resources) or more ordinary malware menaces (calling or texting tolled numbers, installing unwanted software, sending the attacker private information about the device or its owner, spying on the owner using the camera or microphone, etc.). The steps involved in obtaining a set of relevant data sources and the accompanying method using software-based sensors to detect anomalous behavior in modern smartphones, based on machine-learning classifiers, are described.

The purpose of this study was to assess if anomalous behavior could be detected through machine-learning classifiers, based on input data sources from a variety of sensors within the device. Three types of classifiers for the machine-learning application—logistic regression, a shallow neural network for pattern recognition, and SVMs—are investigated. The three are evaluated on several metrics, the most important of which being the F1 score on the test set. The full details of the design, implementation, and evaluation of the learning algorithms are presented step by step in their respective sections.

The results show that all the three investigated algorithms perform reasonably well, with SVMs having a slight edge. The dataset split procedure, discussed at length throughout the paper, give the model a good ability to generalize as yet unseen data.

The effectiveness of several deep learning models of facial expression recognition, such as SSD and fast R-CNN for human–robot interaction is very interesting, deeply analysed and verified [10]. Based on an innovative end-to-end pipeline method that applies two optimized CNNs, the face recognition (FR) and the facial expression recognition (FER), using deep convolutional neural networks (CNN) for the growth real-time inference speed of the entire process is achieved, leading to a high level of advanced intelligent control in the interaction between humans and a NAO robot.

The paper focuses on enhancing the performances of different types of convolutional neural networks (CNN), in terms of accuracy, generalization and inference speed, using several optimization methods (including rectified Adam), such as FER2013 database augmentation with images from other databases and asynchronous threading at inference time. For emotion recognition, transfer learning and the fine-tuning of three CNN models (VGG, Inception V3 and ResNet) have been used. The outcomes prove improvements over 10% when using two serialized CNN, instead of when using only the FER CNN, while the recent optimization model, called rectified adaptive moment optimization (RAdam), lead to a better generalization and accuracy improvement of 3–4% on each emotion recognition CNN. The innovative end-to-end pipeline uses deep convolutional neural networks for training real-time accurate models, which can be applied to human–machine interactions on humanoids robots or other intelligent portable platforms in order to obtain advanced intelligent control.

Advanced intelligent control in micro-nano technologies is addressed in the following papers.

The development of sensors that will lead to new trends in real-time intelligent sensor systems through advanced intelligent control methods and techniques can be achieved through an innovative approach in the growing of nanowires on silicon for ultraviolet photodetectors [11]. The effect of silver catalysts to enhance the growing of Ga_2O_3 nanowires and the sensitivity of β-Ga_2O_3 nanowires for UV detection were investigated. Semiconductor nanowires exhibit improved material properties compared to thin-film semiconductors, becoming an ideal candidate for visible-blind UV-light sensors, such as power electronics, solar-blind UV detectors and devices for harsh environments.

The results led to highly-oriented, dense and long Ga_2O_3 nanowires that can be grown directly onto the surface of silicon, forming a pn heterojunction with rectifying characteristics and excellent UV photo-response.

The advanced intelligent control methods and techniques, which could lead to new concepts and designs in the development of the real-time intelligent sensory control systems, is approached in a novel PDMS-based sensor system for MPWM measurements of picoliter volumes in microfluidic devices [12]. An automatic microinjection system, by integrating a sensor based on image processing of the fluid that flows through microchannels using the microwire molding technique realized with the technique known as microwire-molding, was designing and achieved, validating the concept of the sensor that measures fluid volumes at picolitric levels or lower. The microfluidic devices have wide applications in biological and medical analysis and in the detection, control and manipulation of biological samples and cell biology research, such as in the analysis of unpurified blood samples, analysis of complex mixtures and molecules (especially DNA and proteins), DNA sequencing, single cell manipulation, electrophoretic separations, drug screening, screens for protein crystallization conditions, cell culture studies and reproductive cell selection.

A group of two papers focuses on advanced intelligent control in human health monitoring.

In total, 94 studies, summarize the state of research into the use of technology with a focus on teaching people with Asperger's syndrome, taking into account the 13 aspects of user experience, usability and accessibility. An in-depth review shows how the use of technology in 12 educational contexts helps people with autism spectrum disorder (ASD) to develop several skills [13].

The research methodology was based on systematic literature review using the Kitchenham' process: planning, conducting and publicizing in terms of Research Questions, Data Sources and Search Strategies, Search Strings, etc.

The use of technological advancements such as virtual agents, artificial intelligence, virtual reality and augmented reality, allows for an environment of comfort and constant learning for people with Asperger's syndrome.

Smart offices, enhanced and developed through assistive technologies using Brain Computer Interface (BCI) for the adaptive control of the lighting and temperature in working spaces, sensing the environment's temperature and lighting, can respond of the user's comfort needs [14]. The BCI acquires alpha, beta and theta powers extracted from the EEG signals, representing the worker's comfort level. Advanced intelligent control systems, considering many factors which influence EEG, such as

emotional state, fatigue, sleepiness, age, body temperature, and blood oxygen saturation, will be able to lead to the best comfort and engagement environment using artificial intelligence techniques.

A flexible automatic distortion rectification methodology, for the automatic distortion rectification of wide-angle and fisheye lens camera models, with a comprehensive mathematical model that can refine the outliers simultaneously, optimizing the best-fit parameters with minimum error possible, is proposed [15]. An iterative optimization was used with the refinement of the outliers from the pool of robust line-member set, the identification of the plumbline angular cumulative loss over refined line-member set and an investigation into the significance through an ablation approach. The system is congruent quantitative vs. accuracy, practical significance and qualitative vs. adaptability and processing time, in relation to the real/synthetic, public and private datasets.

Relevant experiments for image quality, stretching and pixel-point error on various metrics, such as PSNR, SSIM, S3 and LPC-SI, with greater precision regarding distortion compensation and maintaining pixel consistency in the context of employing wide-angle lens models for applications on advanced driver-assistance system (ADAS) and video surveillance, were extensively exploited to validate the automatic distortion rectification of wide-angle images.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Dumitrache, I.; Caramihai, S.; Moisescu, M.; Sacala, I.; Vladareanu, L.; Repta, D. A Perceptive Interface for Intelligent Cyber Enterprises. *Sensors* **2019**, *19*, 4422. [CrossRef] [PubMed]
2. Kulikajevas, A.; Maskeliūnas, R.; Damaševičius, R.; Ho, E. 3D Object Reconstruction from Imperfect Depth Data Using Extended YOLOv3 Network. *Sensors* **2020**, *20*, 2025. [CrossRef] [PubMed]
3. Feng, Y.; Wang, H.; Vladareanu, L.; Chen, Z.; Jin, D. New Motion Intention Acquisition Method of Lower Limb Rehabilitation Robot Based on Static Torque Sensors. *Sensors* **2019**, *19*, 3439. [CrossRef] [PubMed]
4. Yan, H.; Wang, H.; Vladareanu, L.; Lin, M.; Vladareanu, V.; Li, Y. Detection of Participation and Training Task Difficulty Applied to the Multi-Sensor Systems of Rehabilitation Robots. *Sensors* **2019**, *19*, 4681. [CrossRef] [PubMed]
5. Ivanescu, M.; Popescu, N.; Popescu, D.; Channa, A.; Poboroniuc, M. Exoskeleton Hand Control by Fractional Order Models. *Sensors* **2019**, *19*, 4608. [CrossRef] [PubMed]
6. Popescu, D.; Dumitrache, I.; Caramihai, S.; Cernaianu, M. High Precision Positioning with Multi-Camera Setups: Adaptive Kalman Fusion Algorithm for Fiducial Markers. *Sensors* **2020**, *20*, 2746. [CrossRef] [PubMed]
7. Catargiu, G.; Dulf, E.; Miclea, L. Connected Bike-Smart IoT-Based Cycling Training Solution. *Sensors* **2020**, *20*, 1473. [CrossRef] [PubMed]
8. Kuantama, E.; Tarca, R.; Dzitac, S.; Dzitac, I.; Vesselenyi, T.; Tarca, I. The Design and Experimental Development of Air Scanning Using a Sniffer Quadcopter. *Sensors* **2019**, *19*, 3849. [CrossRef] [PubMed]
9. Vladareanu, V.; Voiculescu, V.; Grosu, V.; Vladareanu, L.; Travediu, A.; Yan, H.; Wang, H.; Ruse, L. Detection of Anomalous Behavior in Modern Smartphones Using Software Sensor-Based Data. *Sensors* **2020**, *20*, 2768. [CrossRef] [PubMed]
10. Melinte, D.; Vladareanu, L. Facial Expressions Recognition for Human—Robot Interaction Using Deep Convolutional Neural Networks with Rectified Adam Optimizer. *Sensors* **2020**, *20*, 2393. [CrossRef] [PubMed]
11. Alhalaili, B.; Vidu, R.; Islam, M. The Growth of Ga_2O_3 Nanowires on Silicon for Ultraviolet Photodetector. *Sensors* **2019**, *19*, 5301. [CrossRef] [PubMed]
12. Ardeleanu, M.; Popescu, I.; Udroiu, I.; Diaconu, E.; Mihai, S.; Lungu, E.; Alhalaili, B.; Vidu, R. Novel PDMS-Based Sensor System for MPWM Measurements of Picoliter Volumes in Microfluidic Devices. *Sensors* **2019**, *19*, 4886. [CrossRef] [PubMed]
13. Valencia, K.; Rusu, C.; Quiñones, D.; Jamet, E. The Impact of Technology on People with Autism Spectrum Disorder: A Systematic Literature Review. *Sensors* **2019**, *19*, 4885. [CrossRef] [PubMed]

14. Al-Hudhud, G.; Alqahtani, L.; Albaity, H.; Alsaeed, D.; Al-Turaiki, I. Analyzing Passive BCI Signals to Control Adaptive Automation Devices. *Sensors* **2019**, *19*, 3042. [CrossRef] [PubMed]
15. Kakani, V.; Kim, H.; Lee, J.; Ryu, C.; Kumbham, M. Automatic Distortion Rectification of Wide-Angle Images Using Outlier Refinement for Streamlining Vision Tasks. *Sensors* **2020**, *20*, 894. [CrossRef] [PubMed]

Article

New Motion Intention Acquisition Method of Lower Limb Rehabilitation Robot Based on Static Torque Sensors

Yongfei Feng [1,2,3], Hongbo Wang [2,*], Luige Vladareanu [3,*], Zheming Chen [1] and Di Jin [1]

[1]   Faculty of Mechanical Engineering & Mechanics, Ningbo University, Ningbo 315211, China

[2]   Parallel Robot and Mechatronic System Laboratory of Hebei Province and Key Laboratory of Advanced Forging & Stamping Technology and Science of Ministry of Education, Yanshan University, Qinhuangdao 066004, China

[3]   Robotics and Mechatronics Department, Institute of Solid Mechanics of the Romanian Academy, 010141 Bucharest, Romania

*   Correspondence: hongbo_w@ysu.edu.cn (H.W.); luige.vladareanu@vipro.edu.ro (L.V.)

Abstract: The rehabilitation robot is an application of robotic technology for people with limb disabilities. This paper investigates a new applicable and effective sitting/lying lower limb rehabilitation robot (the LLR-Ro). In order to improve the patient's training initiative and accelerate the rehabilitation process, a new motion intention acquisition method based on static torque sensors is proposed. This motion intention acquisition method is established through the dynamics modeling of human–machine coordination, which is built on the basis of Lagrangian equations. Combined with the static torque sensors installed on the mechanism leg joint axis, the LLR-Ro can obtain the active force from the patient's leg. Based on the variation of the patient's active force and the kinematic functional relationship of the patient's leg end point, the patient motion intention is obtained and used in the proposed active rehabilitation training method. The simulation experiment demonstrates the correctness of mechanism leg dynamics equations through ADAMS software and MATLAB software. The calibration experiment of the joint torque sensors' combining limit range filter with an average value filter provides the hardware support for active rehabilitation training. The consecutive variation of the torque sensors from just the mechanism leg weight, as well as both the mechanism leg and the patient leg weights, obtains the feasibility of lower limb motion intention acquisition.

Keywords: lower limb; rehabilitation robot; motion intention acquisition; static torque sensor

---

1. Introduction

Cerebral vascular disease, hemiplegic, and paraplegia may cause limb motor dysfunction. For patients with limb dysfunction, the quality of life depends on the level of limb damage. Based on nerve rehabilitation theory, patients can recover through specialized rehabilitation training [1–3]. The lower limb rehabilitation robot is an application of robotic technology for people with lower limb disabilities [4]. In recent years, research on the lower limb rehabilitation robots has become an active topic [5,6]. Several kinds of lower limb rehabilitation robots have been developed [7]. These can be divided into the single degree-of-freedom gait trainers [8], wearable gait trainers [9,10], suspended gait trainers [11–14], and sitting/lying gait trainers [15,16]. Switzerland has developed a suspended gait trainer, Lokomat, whose left and right mechanism legs can assist patients to simulate the walking gait of normal people and restore the control ability of the nervous system to walk [17,18]. M. Bouri et al. developed a new rehabilitation robot, Lambda. Based on two translational articulations and one rotational for ankle mobilization, the patient's hip, knee, and ankle can conveniently be mobilized in

order to carry out rehabilitation, fitness, or high-level sport training [19]. Carleton University made a virtual gait rehabilitation robot (ViGRR) for bed-ridden stroke patients. It can provide average gait motion training as well as other targeted exercises, such as leg press, stair stepping, and motivational gaming [20]. Yildiz University of Science and Technology in Turkey made a sitting/lying gait trainer, Physiotherabot, helping patients to do passive training and active training [21]. A new applicable and effective sitting/lying lower limb rehabilitation robot (the LLR-Ro) is proposed in this paper.

The outstanding feature of the intelligent rehabilitation robots is that they can help the patient complete active rehabilitation training [22–25]. Active rehabilitation training is a high-level rehabilitation training method. It can improve a patient's training initiative and accelerate the rehabilitation process, realizing patient-led training to replace traditional robot-led training. The most important part of active rehabilitation training is the acquisition of patient lower limb motion intention. There are many excellent achievements to recognize the limb motion intention. *Zhang* et al. developed a sitting/lying lower limb rehabilitation robot, named iLeg [26]. It employs the surface electromyography (EMG) signals from muscle groups to obtain the Cartesian torque/force. Based on the Cartesian torque/force, iLeg can detect the patient's motion intention and assist the patient to achieve active rehabilitation training. Leonard et al. proposed a novel EMG-driven hand exoskeleton for stroke patients' bilateral rehabilitation through grasping motion [27]. Yepes et al. used the optimal frame of the EMG signal to obtain the motion intention of knee joints, and adaptively derived the necessary moment to follow the motion of the knee joint based on the patient's random motion [28]. Khoshdel et al. collected electromyogram signals near the four muscles of the lateral femoral muscle, rectus femoris, medial femoral muscle, and biceps femoris of the lower extremity through multiple channels to identify the motion and state of the knee joint [29]. However, the EMG-driven robot has its shortcomings. The EMG signal strength is changed with the patient's leg rehabilitation and it is difficult to build the relationship between EMG signals and patient leg motion intention to realize continuing control. Present human motion intention detection is mainly designed based on biomechanical signals [30,31]. Based on the Inertial Measurement Units, Wittmann et al. proposed an arm tracking method which is used in the home environment [32]. Hwang et al. installed a number of pressure sensors on the contact surface between the three-degrees-of-freedom standing lower limb rehabilitation robot and human lower limbs, and collected human–computer interaction force information as a quantitative active motion intention [33]. A tactile control method based on plantar force signals was proposed by Berlin University of Technology, Germany, to meet the training needs of patients with arbitrary trajectories in the daily activities of lower limbs [34]. Most researchers could detect the patient motion intention based on relationships between the patient leg end forces and patient leg joint torques. However, the joint torques of the patient leg are difficult to obtain and most of them are estimated for use. This paper proposes the robot, the LLR-Ro, whose torque sensors are installed on the joint axis. Based on the dynamics modeling of human–machine coordination and the impact on the mechanism leg torques from the patient leg end force, the patient motion intention can be obtained through the variation of torque sensors installed on the mechanism leg.

2. Materials and Methods

2.1. Mechanism and Hardware Control System Design of the LLR-Ro

For most of the sitting/lying gait trainers, their structures consist the left mechanism leg model, right mechanism leg model, chair, and so on. The structure of this trainer is just like a ring, which is difficult for patients to sit or lie on the robot. So the sitting/lying gait trainers need an auxiliary device to transfer the patient to the robots. It will cause the originally narrow rehabilitation space to become more crowded in rehabilitation institutions or hospitals. Based on the modular principle, the LLR-Ro is composed of the movable seat, the left mechanism leg model, the right mechanism leg model, the control box, and the touch screen monitor. The LLR-Ro was designed as shown in the Figure 1. There are four universal wheels under the base of the movable seat, so the movable seat can separate from

the LLR-Ro to transfer the patient on the robot easily without the help of another auxiliary device. The innovative design of the movable seat can also help the patient realize sitting or lying on it fairly easily.
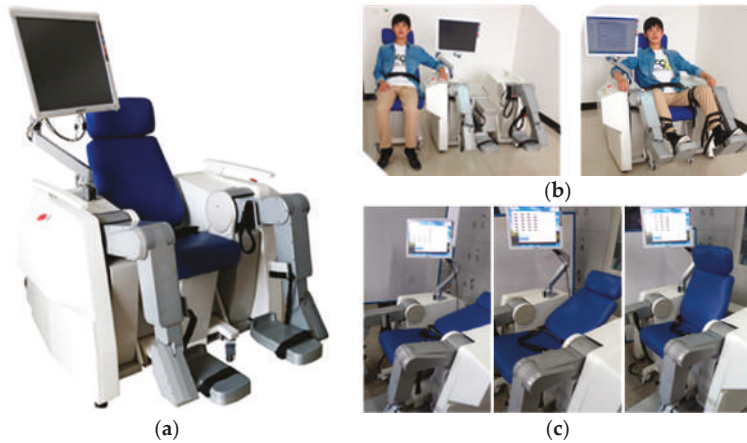
Figure 1. The prototype of the LLR-Ro: (**a**) The prototype of the LLR-Ro; (**b**) The movable seat separated from and grouped into the LLR-Ro; (**c**) The back angle of the movable seat altered from 110° to 170°.

The most important part of the LLR-Ro is the mechanism leg as shown in Figure 2. The left mechanism leg module and the right mechanism leg module are bilaterally symmetrical. Each module has a mechanism leg, which has the hip, knee, and ankle joint in the human body sagittal. Based on Man–machine Engineering and the innovative design for the mechanism, the length of the mechanism leg can be automatically adjusted to fit patients with different heights through the motor-driven pushrod. The mechanism leg contains sensors to estimate the torque and force produced by patients. It also contains sensors to measure joint rotation and motors to drive the three joints. Both the hip joint and knee joint adapt a mechanical structure where the torque sensors are installed on the joint axis. Though it increases the difficulty of the mechanism leg design, it avoids the transmission errors of the joint torques.
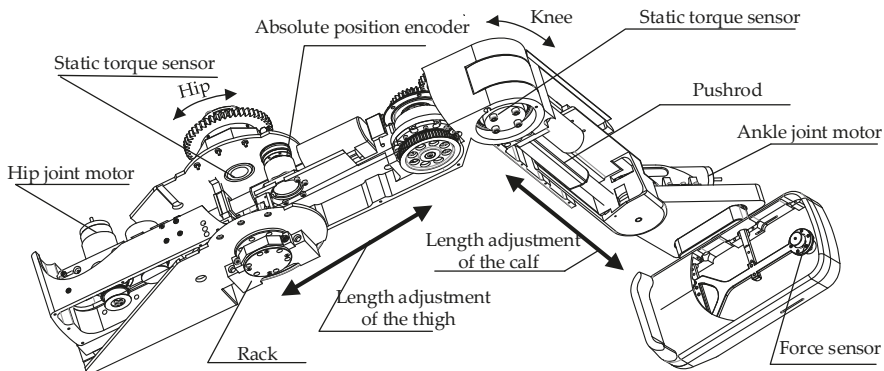
Figure 2. Design of the mechanism leg.

The torque sensors for the hip joint and knee joint of the mechanism leg were manufactured by Shijiazhuang Baisen Instruments and Technology Co., Ltd. in China. The profile of the sensor and the

sensor's detailed parameters are shown in Figure 3, including the rated output, range, bridge voltage, and output signal.

Main Technical Specifications

- Rated Output:1.9mV/V
- Range:200Nm
- Bridge Voltage:10VDC
- Output Signal:0~40mV
- Working Temperature:-20~+65°C
- Safe Overload:150%F.S
- Hysteresis:±0.1%FS
- Creep(30min):±0.1%FS
- Zero Balance: 1%FS
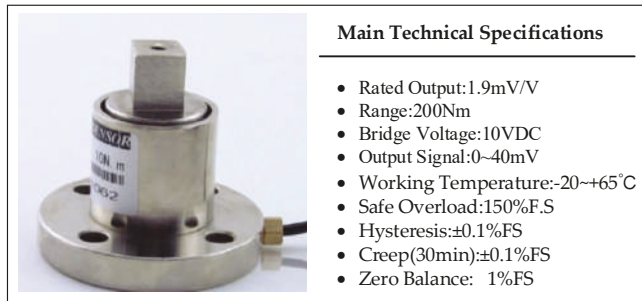
Figure 3. The profile and detailed parameters of the torque sensor.

Based on the functions of the LLR-Ro, the hardware control system contains the central control module, the human–machine interactive system, the sensor feedback system, and the motion control system, as shown in Figure 4. The central control module mainly runs the control software and receives the operational order from the human–machine interactive system. The human–machine interactive system displays the control software interface and feeds back the training conditions. The motion control system receives the motion control commands from the central control module, realizes the motor closed-loop control, and feeds back the joint real motion condition to the central control module. The sensor feedback system acquires the sensor information and achieves the sensor state.
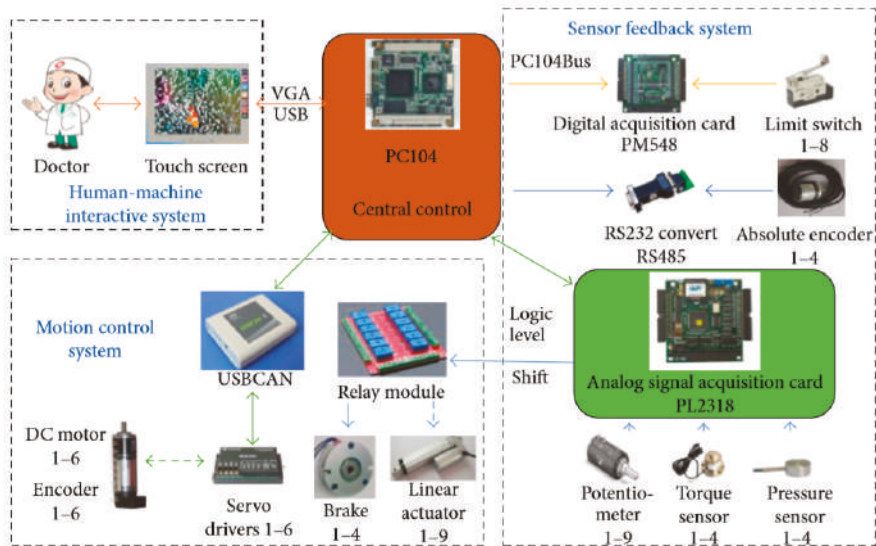
Figure 4. Design of the hardware control system.

2.2. Patient Lower Limb Motion Intention Acquisition

Based on the innovative design of the robot mechanism leg, the joint angular torques, angular positions, and angular velocities of the LLR-Ro can be measured in real time. Through the dynamics relationship between the patient leg and robot mechanism leg, and the variation of the mechanism leg

joint torques, the robot can conclude the patient motion intention and help the patient realize active rehabilitation training. The flow chart of patient lower limb motion intention acquisition is shown in Figure 5. When the patient's foot is placed on the mechanism leg without active force, the value τ_m is collected by the joint torque sensors. Then, the patient can begin active rehabilitation training, and the patient starts to exert force on the mechanism leg. At this time, the torques of each joint are recorded, which is called the actual torque τ_C. By comparing τ_C and τ_m, the patient's movement intention is determined by the rehabilitation robot, and then the speed of motor movement is controlled. Considering the uncertainty of the patient motion intention and the unsteadiness of the patient's illness, the patient's safety is the most important factor to be considered. The variation of the torque sensor is processed through the recursion median filtering. The sampling frequency is set as 200 Hz. The sampling period is set as five samples per period. The deviation value $\Delta\tau$ is calculated through the average difference between data in the current moment and data in the previous moment. The control speed is obtained through the proportional control function $S = H \times \Delta\tau$. The proportionality coefficient H is set large at the start of active training or when the patient's lower limb has low strength. It can be adjusted by the therapist to assist the patient as needed. The optimal coefficient H should also be determined through the clinical trial to meet patients with different levels of rehabilitation.
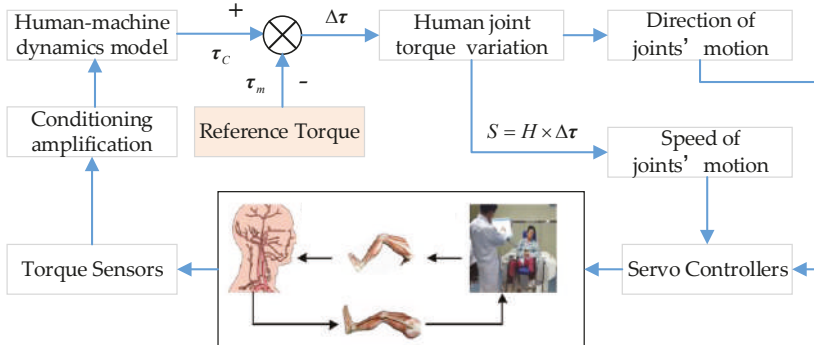
Figure 5. The motion intention acquisition flow diagram of the patient's lower limb.

The mechanism legs were simplified into a linkage model, as shown in Figure 6. *O*, *A*, and *B* represent the hip joint, knee joint, and ankle joint of mechanism leg, respectively. The hip joint angle and knee joint angle are expressed through θ_1 and θ_2, while the ankle joint angle is a constant equaling 90°. $l_i (i = 1, 2, 3)$ represents the length of each segment, $C_i (i = 1, 2, 3)$ represents the rod centroid, and $R_i (i = 1, 2, 3)$ represents the length from centroid to joint axis.
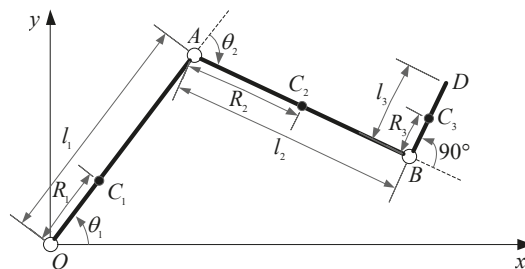
Figure 6. The linkage model of the mechanism leg.

We obtained the coordinate position of point D:

$$\begin{cases} x_D = l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) - l_3 \sin(\theta_1 + \theta_2) \\ y_D = l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2) \end{cases}. \tag{1}$$

The angular position and angular velocities of each rod centroid were calculated:

$$\begin{cases} x_{C1} = R_1 \cos\theta_1 \\ y_{C1} = R_1 \sin\theta_1 \\ x_{C2} = l_1 \cos\theta_1 + R_2 \cos(\theta_1 + \theta_2) \\ y_{C2} = l_1 \sin\theta_1 + R_2 \sin(\theta_1 + \theta_2) \\ x_{C3} = l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) - R_3 \sin(\theta_1 + \theta_2) \\ y_{C3} = l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2) + R_3 \cos(\theta_1 + \theta_2) \\ \dot{x}_{C1} = -R_1 \dot{\theta}_1 \sin\theta_1 \\ \dot{y}_{C1} = R_1 \dot{\theta}_1 \cos\theta_1 \\ \dot{x}_{C2} = -\dot{\theta}_1 l_1 \sin\theta_1 - (\dot{\theta}_1 + \dot{\theta}_2)R_2 \sin(\theta_1 + \theta_2) \\ \dot{y}_{C2} = \dot{\theta}_1 l_1 \cos\theta_1 + (\dot{\theta}_1 + \dot{\theta}_2)R_2 \cos(\theta_1 + \theta_2) \\ \dot{x}_{C3} = -\dot{\theta}_1 l_1 \sin\theta_1 - (\dot{\theta}_1 + \dot{\theta}_2)l_2 \sin(\theta_1 + \theta_2) - (\dot{\theta}_1 + \dot{\theta}_2)R_3 \cos(\theta_1 + \theta_2) \\ \dot{y}_{C3} = \dot{\theta}_1 l_1 \cos\theta_1 + (\dot{\theta}_1 + \dot{\theta}_2)l_2 \cos(\theta_1 + \theta_2) - (\dot{\theta}_1 + \dot{\theta}_2)R_3 \sin(\theta_1 + \theta_2) \end{cases}. \tag{2}$$

Then the total kinetic energy of the mechanism leg E_k was:

$$E_k = E_{k1} + E_{k2} + E_{k3} = \begin{cases} \frac{1}{2}\dot{\theta}_1^2 \begin{cases} m_1 R_1^2 + m_2(l_1^2 + R_2^2) + m_3(l_1^2 + l_2^2 + R_3^2) + (I_1 + I_2 + I_3) + \\ 2(m_2 R_2 l_1 + m_3 l_1 l_2)[\sin\theta_1 \sin(\theta_1 + \theta_2) + \cos\theta_1 \cos(\theta_1 + \theta_2)] + \\ m_3 l_1 R_3[\sin\theta_1 \cos(\theta_1 + \theta_2) - \cos\theta_1 \sin(\theta_1 + \theta_2)] \end{cases} + \\ \frac{1}{2}\dot{\theta}_2^2 [m_2 R_2^2 + m_3(l_2^2 + R_3^2) + (I_2 + I_3)] + \\ \dot{\theta}_1\dot{\theta}_2 \begin{cases} m_2 R_2^2 + m_3(l_2^2 + R_3^2) + (I_2 + I_3) + \\ (m_2 R_2 l_1 + m_3 l_1 l_2)[\sin\theta_1 \sin(\theta_1 + \theta_2) + \cos\theta_1 \cos(\theta_1 + \theta_2)] + \\ m_3 l_1 R_3[\sin\theta_1 \cos(\theta_1 + \theta_2) - \cos\theta_1 \sin(\theta_1 + \theta_2)] \end{cases} \end{cases}, \tag{3}$$

where $E_{ki}(i = 1, 2, 3)$ represents the kinetic energy of the rod i, $m_i(i = 1, 2, 3)$ represents the weight of the rod i and I_i represents the rotational inertia of the rod i.

The total potential energy of the mechanism leg E_p was:

$$E_P = E_{P1} + E_{P2} + E_{P3} = \left[\begin{array}{l} (m_1 g R_1 + m_2 g l_1 + m_3 g l_1) \sin\theta_1 + \\ (m_2 g R_2 + m_3 g l_2) \sin(\theta_1 + \theta_2) + m_3 g R_3 \cos(\theta_1 + \theta_2) \end{array} \right], \tag{4}$$

where $E_{pi}(i = 1, 2, 3)$ represents the potential energy of the rod i.

Lagrange function is defined as the difference between the total kinetic energy (K) and the total potential energy (P) of the mechanical system [35]:

$$L = K - P. \tag{5}$$

By using Lagrange function, the system dynamics equation (the second Lagrange equation) [35] was:

$$\tau = \frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}} - \frac{\partial L}{\partial \theta}, \tag{6}$$

where θ represents the generalized coordinates of the kinetic energy and potential energy system, $\dot{\theta}$ represents the generalized velocity of the system, and τ represents the driving torque vector.

As the potential energy E_P did not contain $\dot{q}$ obviously, the dynamics equation was transformed into [35]:

$$\tau = \frac{d}{dt}\frac{\partial E_k}{\partial \dot{q}} - \frac{\partial E_k}{\partial q} + \frac{\partial E_p}{\partial q}. \tag{7}$$

Each joint torque was calculated:

$$\tau_1 = \frac{d}{dt}\frac{\partial E_k}{\partial \dot{\theta}_1} - \frac{\partial E_k}{\partial \theta_1} + \frac{\partial E_p}{\partial \theta_1} = \left\{ \begin{array}{l} \ddot{\theta}_1 \left\{ \begin{array}{l} m_1 R_1{}^2 + m_2(l_1{}^2 + R_2{}^2) + m_3(l_1{}^2 + l_2{}^2 + R_3{}^2) + (I_1 + I_2 + I_3) + \\ 2(m_2 R_2 l_1 + m_3 l_1 l_2)[\sin\theta_1 \sin(\theta_1 + \theta_2) + \cos\theta_1 \cos(\theta_1 + \theta_2)] + \\ m_3 l_1 R_3[\sin\theta_1 \cos(\theta_1 + \theta_2) - \cos\theta_1 \sin(\theta_1 + \theta_2)] \end{array} \right\} + \\ \ddot{\theta}_2 \left\{ \begin{array}{l} m_2 R_2{}^2 + m_3(l_2{}^2 + R_3{}^2) + (I_2 + I_3) + \\ (m_2 R_2 l_1 + m_3 l_1 l_2)[\sin\theta_1 \sin(\theta_1 + \theta_2) + \cos\theta_1 \cos(\theta_1 + \theta_2)] + \\ m_3 l_1 R_3[\sin\theta_1 \cos(\theta_1 + \theta_2) - \cos\theta_1 \sin(\theta_1 + \theta_2)] \end{array} \right\} + \\ \dot{\theta}_1 \dot{\theta}_2 \left\{ \begin{array}{l} 2(m_2 R_2 l_1 + m_3 l_1 l_2)[\sin\theta_1 \cos(\theta_1 + \theta_2) - \cos\theta_1 \sin(\theta_1 + \theta_2)] - \\ m_3 l_1 R_3[\sin\theta_1 \sin(\theta_1 + \theta_2) + \cos\theta_1 \cos(\theta_1 + \theta_2)] \end{array} \right\} + \\ \dot{\theta}_2{}^2 \left\{ \begin{array}{l} (m_2 R_2 l_1 + m_3 l_1 l_2)[\sin\theta_1 \cos(\theta_1 + \theta_2) - \cos\theta_1 \sin(\theta_1 + \theta_2)] - \\ m_3 l_1 R_3[\sin\theta_1 \sin(\theta_1 + \theta_2) + \cos\theta_1 \cos(\theta_1 + \theta_2)] \end{array} \right\} + \\ (m_1 g R_1 + m_2 g l_1 + m_3 g l_1) \cos\theta_1 + \\ (m_2 g R_2 + m_3 g l_2) \cos(\theta_1 + \theta_2) - m_3 g R_3 \sin(\theta_1 + \theta_2) \end{array} \right\}, \tag{8}$$

$$\tau_2 = \frac{d}{dt}\frac{\partial E_k}{\partial \dot{\theta}_2} - \frac{\partial E_k}{\partial \theta_2} + \frac{\partial E_p}{\partial \theta_2} = \left\{ \begin{array}{l} \ddot{\theta}_2 \left[m_2 R_2{}^2 + m_3(l_2{}^2 + R_3{}^2) + (I_2 + I_3) \right] - \\ \frac{1}{2}\dot{\theta}_1{}^2 \left\{ \begin{array}{l} 2(m_2 R_2 l_1 + m_3 l_1 l_2)[\sin\theta_1 \cos(\theta_1 + \theta_2) - \cos\theta_1 \sin(\theta_1 + \theta_2)] - \\ m_3 l_1 R_3[\sin\theta_1 \sin(\theta_1 + \theta_2) + \cos\theta_1 \cos(\theta_1 + \theta_2)] \end{array} \right\} + \\ \ddot{\theta}_1 \left\{ \begin{array}{l} m_2 R_2{}^2 + m_3(l_2{}^2 + R_3{}^2) + (I_2 + I_3) + \\ (m_2 R_2 l_1 + m_3 l_1 l_2)[\sin\theta_1 \sin(\theta_1 + \theta_2) + \cos\theta_1 \cos(\theta_1 + \theta_2)] + \\ m_3 l_1 R_3[\sin\theta_1 \cos(\theta_1 + \theta_2) - \cos\theta_1 \sin(\theta_1 + \theta_2)] \end{array} \right\} + \\ (m_2 g R_2 + m_3 g l_2) \cos(\theta_1 + \theta_2) - m_3 g R_3 \sin(\theta_1 + \theta_2) \end{array} \right\}. \tag{9}$$

The dynamics equation was simplified as follows [35]:

$$H(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + G(\theta) = \tau, \tag{10}$$

where, $H(\theta)$ represents the inertia matrix, $C(\theta, \dot{\theta})$ represents the centrifugal force and the Coriolis force matrix, and $G(\theta)$ represents the gravity matrix.

The contact force between the lower limb of the patient and the end of the mechanism leg changed in real time with the movement of the mechanism leg. When we just considered the effect on the joint torque of the mechanism leg from patient leg weight, the joint torque τ_p generated by the endpoint force was obtained based on the force Jacobin formula [35]:

$$\tau_p = J^T(\theta)F_0, \tag{11}$$

where $J^T(\theta)$ represents the force Jacobin matrix of the mechanism legs and F_0 represents the contact force between the patient's leg and the mechanism leg while the patient does not exert active force.

Combined with the mechanism leg's kinematics, the expression $J^T(\theta)$ was obtained:

$$J^T(\theta) = \begin{bmatrix} -l_1 \sin\theta_1 - l_2 \sin(\theta_1 + \theta_2) & l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) \\ -l_2 \sin(\theta_1 + \theta_2) & l_2 \cos(\theta_1 + \theta_2) \end{bmatrix}. \tag{12}$$

Combined with the robot dynamics equation, the dynamics modeling of human–machine coordination was obtained when the patient's leg was put on the rehabilitation robot:

$$H(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + G(\theta) = \Delta\tau = \tau - J^T(\theta)F_0. \tag{13}$$

3. Results

3.1. Verification of the Mechanism Leg Dynamics Equations

The dynamics equations of the LLR-Ro were so complicated and the solution procedure was tedious. It is very necessary to prove the correctness of the derivative results. The verification of the dynamics equation was conducted through ADAMS software and MATLAB software. ADAMS software, designed by American MSC Company, was used for the multi-body dynamics simulation. Figure 7 shows the simulation model developed through ADAMS.
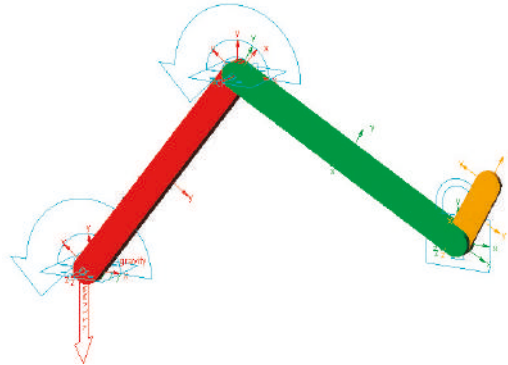
Figure 7. Simulation model developed through ADAMS.

The model is a three linkage manipulator. It has three joints moving in xy-plane. The first joint is the hip joint and the second is the knee joint. The third joint is the ankle joint and it is locked. The detailed parameters of the three linkage manipulator are given as below in Table 1.

Table 1. The parameters of the three linkage manipulator.

Parameters	Thigh	Calf	Foot
Segment Length	390 mm	400 mm	100 mm
Distance from Centroid to the Joint Axis	50 mm	250 mm	50 mm
Segment Rotational Intertia	1.5 kg·m^2	0.2 kg·m^2	0.02 kg·m^2

The simulation time was $t = 8$ s, and the driving function of the hip joint is given as below:

$$\begin{cases} \theta_{11}(t) = 10 + 4.935t^2 - 0.603t^3 + 0.011t^4 \ (0 \le t \le 4) \\ \theta_{12}(t) = 53.216 + 13.384t_1 - 1.234t_1{}^2 - 0.425t_1{}^3 + 0.066t_1{}^4 \ (t_1 = t - 4; 4 \le t \le 8) \end{cases}, \quad (14)$$

where θ_{11} and θ_{12} are the hip joint angular position at the times $0 \le t \le 4$ and $4 \le t \le 8$, respectively. The driving function of the knee joint is given as below:

$$\begin{cases} \theta_{21}(t) = -11.390t^2 + 1.967t^3 - 0.098t^4 \ (0 \le t \le 4) \\ \theta_{22}(t) = -81.350 - 21.691t_1 + 2.848t_1{}^2 + 0.407t_1{}^3 - 0.081t_1{}^4 \ (t_1 = t - 4; 4 \le t \le 8) \end{cases}, \quad (15)$$

where θ_{21} and θ_{22} are the knee joint angular position at the times $0 \le t \le 4$ and $4 \le t \le 8$, respectively.

Then we obtained the hip joint and knee joint actual driving torque through ADAMS, as shown in Figure 7. Meanwhile, we also achieved the theoretical driving torques based on the dynamics

Equations (8) and (9). According to the driving function of joints, the velocity and accelerations of the hip and knee joints were obtained as below:

$$
\begin{cases}
\dot{\theta}_{11}(t) = 9.870t - 1.809t^2 + 0.044t^3 \quad (0 \le t \le 4) \\
\ddot{\theta}_{11}(t) = 9.870 - 3.618t + 0.132t^2 \quad (0 \le t \le 4) \\
\dot{\theta}_{12}(t) = 13.384 - 2.468t_1 - 1.275t_1{}^2 + 0.264t_1{}^3 \quad (t_1 = t - 4, 4 \le t \le 8) \\
\ddot{\theta}_{12}(t) = -2.468 - 2.550t_1 + 0.792t_1{}^2 \quad (t_1 = t - 4, 4 \le t \le 8) \\
\dot{\theta}_{21}(t) = -22.780t + 5.901t^2 - 0.392t^3 \quad (0 \le t \le 4) \\
\ddot{\theta}_{21}(t) = -22.780 + 11.802t - 1.176t^2 \quad (0 \le t \le 4) \\
\dot{\theta}_{22}(t) = -21.691 + 5.696t_1 + 1.221t_1{}^2 - 0.324t_1{}^3 \quad (t_1 = t - 4, 4 \le t \le 8) \\
\ddot{\theta}_{22}(t) = 5.696 + 2.442t_1 - 0.972t_1{}^2 \quad (t_1 = t - 4, 4 \le t \le 8)
\end{cases}
\tag{16}
$$

The theoretical driving torques were also obtained through MATLAB, as shown in Figure 8. Based on the contrast curves, the theoretical curves basically fit with the actual curves. So we could conclude that the dynamics equations of the LLR-Ro are correct.
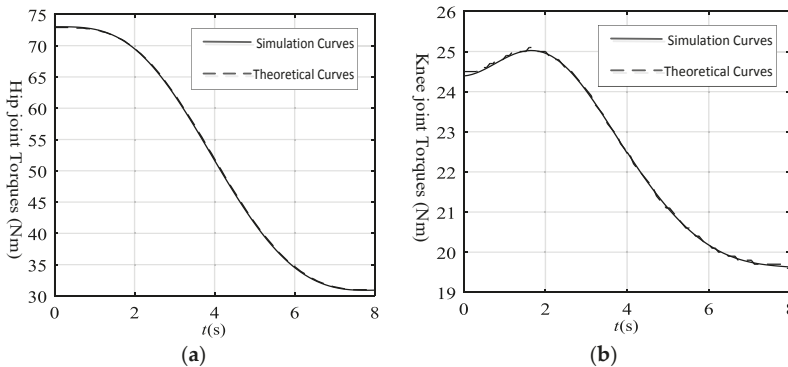
Figure 8. The simulation curves and theoretical curves of the mechanism leg joints: (**a**) The simulation curves and theoretical curves of the hip joint; (**b**) The simulation curves and theoretical curves of the knee joint.

3.2. Calibration Experiment of the Joint Static Torque Sensors

In active training, the torque sensors are important for the whole control. The calibration experiment of torque sensors was conducted. The voltage values of the torque sensors were obtained through the analogue acquisition PL2318. The voltage values were processed through the combing limit range filter with an average value filter. The merit of this filtering method is that it can overcome accidental jamming and the curve of the voltage value is smooth. The calibration of the hip joint torque sensor was similar to the calibration of the knee joint torque sensor. The detailed calibration process of the hip joint torque is introduced in Figure 9.

Through data processing, the calibration curves of the hip and knee joint torque sensors were obtained, as shown in Figure 10. The curves can be described through the below expression:

$$
\begin{cases}
M_1 = 125.3V_1 \\
M_2 = 134.5V_2
\end{cases}
\tag{17}
$$

where M_1 and M_2 represent the hip joint torque and the knee joint torque, and their units are Nm, and V_1 and V_2 represent the voltage value of hip joint torque and knee joint torque, and their units are *V*.
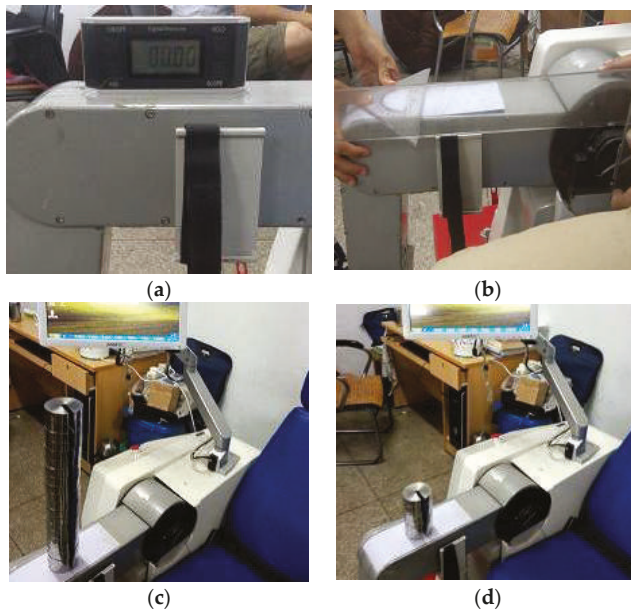
Figure 9. The calibration experiment of torque sensors: (**a**) The thigh of the mechanism leg is set at the horizontal position; (**b**) One point is marked from the hip joint axis 585 mm; (**c**) The analytical weights (each weight is 2.5 kg) are put on the marked point one by one until the weight equals 17.5 kg; (**d**) The weights start to be unloaded one by one until it equals 0 kg. The steps above are repeated three times, and the voltage values are processed through the combing limit range filter and average value filter.
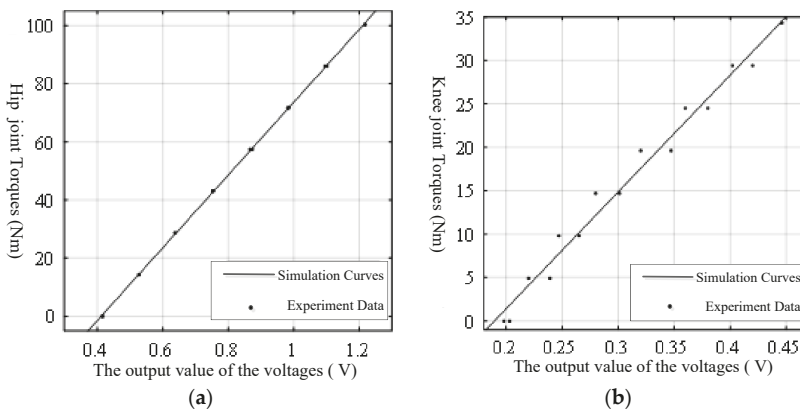
Figure 10. The calibration curves of the joint torque sensors: (**a**) The calibration curves of the hip joint torque sensors; (**b**) The calibration curves of the knee joint torque sensors.

3.3. Verification Experiment of the Motion Intension Acquisition Based on Biomechanics

Based on the calibration experiment of the joint torque sensors, there existed errors between the actual torques and the theoretical torques. It was necessary for the joint torques to set a given threshold values. The given threshold values were obtained through the experiment, as shown in Figure 11. Then theoretical torques and the actual torques just from the mechanism leg weight were also obtained, as shown in Figure 12. The experiment curves were similar to the theoretical curves. However, the errors

were very large at the start of the experiment, because of the heavy mechanism leg and the mechanism assembly error. The maximum errors of the hip joint and the knee joint from the mechanism leg weight were 9.82 Nm and 3.89 Nm, respectively. So the given threshold values of the hip joint torque and the knee joint torque were set to 10 Nm and 5 Nm, respectively.

Mechanism Leg Parameters :

m_1=35.5kg; $m_2 = 7.6kg$; $m_3 = 1.6kg$; l_1=385mm; $l_2 = 390mm$; $l_3 = 80mm$; R_1=20mm; $R_2 = 240mm$; $R_3 = 50mm$; I_1=1.69$kg \cdot m^2$; I_2=0.712$kg \cdot m^2$; I_3=0.024$kg \cdot m^2$
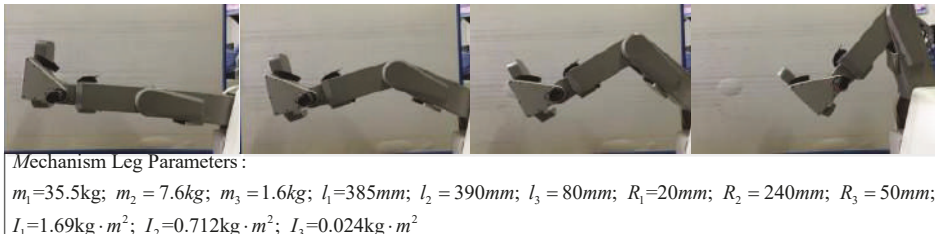
Figure 11. The experiment to obtain the joint torques from the mechanism leg weight.
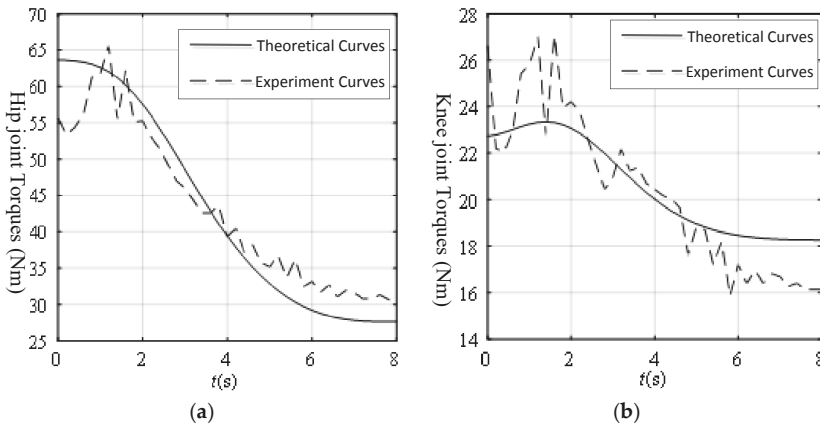
Figure 12. The calibration curves of the joint torque sensors: (**a**) Hip torque just from the mechanism leg weight; (**b**) Knee torque just from the mechanism leg weight.

One volunteer participated in the experiment and his leg was put on the mechanism leg, as shown in Figure 13.

Volunteer Leg Parameters : *Male*，25 years old；body weight 65*kg*

m_1=9.147kg; $m_2 = 2.811kg$; $m_3 = 0.795kg$; l_1=385mm; $l_2 = 390mm$; $l_3 = 80mm$; R_1=233.33mm; $R_2 = 207.97mm$; $R_3 = 39.56mm$
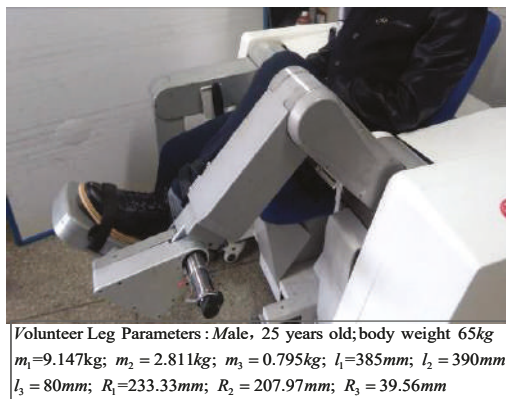
Figure 13. The verification experiment of active training without patient active force.

Figure 14 shows the theoretical and experiment joint torque curves from the mechanism leg and volunteer leg weight. The maximum error of the hip joint without active force was 9.08 Nm and the maximum error of the knee joint without active force was 3.91 Nm.
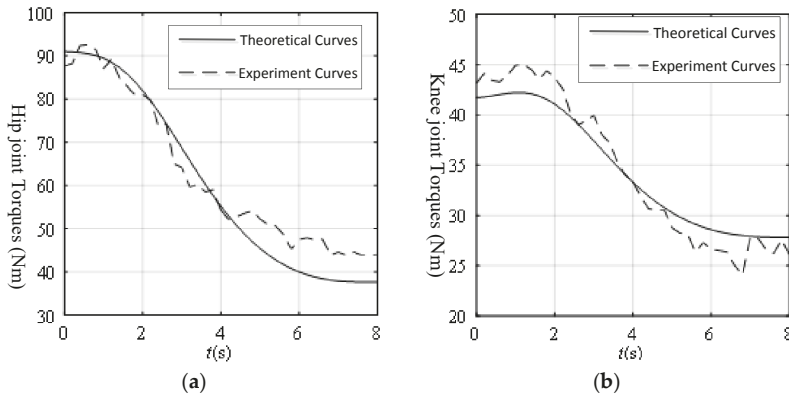
Figure 14. Joint torques from the mechanism leg and patient leg weights: (**a**) Hip torque from the mechanism leg and patient leg weights; (**b**) Knee torque from the mechanism leg and patient leg weights.

4. Discussion

From the calibration curves of the joint torque sensors in Figure 10, the actual test points were closely distributed around the fitting curves, reflecting that the torque sensor had better linear characteristics. However, because of the complex structure of the components around the torque sensor, and the processing accuracy, installation error, and the wear and amplification error of the conditioning board, there was a deviation between the function relationship between the voltage value of the torque sensor in the real state and the ideal state.

At the start, the errors in Figure 14 were smaller than the errors in Figure 12 because the volunteer's leg weights were just like a pre-tightening force, making the transmission error on the mechanism leg smaller. It was demonstrated that the proposed threshold values of the joint torques were satisfied for future experiments. All the errors in Figure 14 were in the scope of the threshold values. From Figures 12 and 14, although the volunteer's leg weight was much lighter than the mechanism leg weight, the variations of the joint torques were prominent. If the patient exerted his active force on the mechanism leg, the variations of joint torques could be much larger. Based on the variations of the joint torques, the principle of detecting the volunteer motion intention is clear and feasible.

Compared with other methods, like EMG, for obtaining patient motion intention, this method may be more suitable for continuous active rehabilitation training control. This is because the EMG signals would be reduced with the patient limb becoming stronger, while the variations of the LLR-Ro torque sensors would be increased. In the future, the research team will continue conducting clinical trials to verify the excellent effect of the active rehabilitation training based on the biomechanics and differential analysis through biomechanics signals and EMG signals on the LLR-Ro.

5. Conclusions

This paper investigates a new applicable and effective sitting/lying multi-joint lower limb rehabilitation robot. In order to improve patient's training initiative and accelerate the rehabilitation process, a new motion intention acquisition method based on biomechanics is proposed. The simulation experiment demonstrates the correctness of the mechanism leg dynamics equations, the calibration experiment of the joint torque sensors provides the hardware support for active rehabilitation training, and the consecutive variation of the torque sensors from just the mechanism leg weight and both

the mechanism leg and patient leg weights obtains the feasibility of lower limb motion intention acquisition. In the future, new active rehabilitation training for the LLR-Ro will be proposed on the basis of the motion intention acquisition method in this paper. Meanwhile, the patients' recovery efficiency through the future active rehabilitation training method will be verified in clinical trials.

References

1. Shi, L.; Duan, F.; Yang, Y.; Sun, Z. The effect of treadmill walking on gait and upper trunk through linear and nonlinear analysis methods. *Sensors* **2019**, *19*, 2204. [CrossRef] [PubMed]
2. Wang, S.; Wang, L.; Meijneke, C.; Van Asseldonk, E.; Hoellinger, T.; Cheron, G.; Ivanenko, Y.; La Scaleia, V.; Sylos-Labini, F.; Molinari, M.; et al. Design and control of the mindwalker exoskeleton. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2015**, *23*, 277–286. [CrossRef] [PubMed]
3. Kwakkel, G.; Kollen, B.J.; van der Grond, J.; Prevo, A.J. Probability of regaining dexterity in the flaccid upper limb: Impact of severity of paresis and time since onset in acute stroke. *Stroke* **2003**, *34*, 2181–2186. [CrossRef] [PubMed]
4. Koceska, N.; Koceski, S.; Durante, F.; Zobel, P.B.; Raparelli, T. Control architecture of a 10 dof lower limbs exoskeleton for gait rehabilitation. *Int. J. Adv. Robot. Syst.* **2013**, *10*, 68. [CrossRef]
5. Veneman, J.F.; Kruidhof, R.; Hekman, E.E.; Ekkelenkamp, R.; Van Asseldonk, E.H.; Van Der Kooij, H. Design and evaluation of the lopes exoskeleton robot for interactive gait rehabilitation. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2007**, *15*, 379–386. [CrossRef] [PubMed]
6. Iñaki, D.; Gil, J.J.; Emilio, S. Lower-limb robotic rehabilitation: Literature review and challenges. *Int. J. Robot.* **2011**, 1–11. [CrossRef]
7. Meng, W.; Liu, Q.; Zhou, Z.D.; Ai, Q.S. Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation. *Mechatronics* **2015**, *31*, 132–145. [CrossRef]
8. Chen, K.; Zhang, Y.; Yi, J.; Liu, T. An integrated hysical-learning model of physical human-robot interactions with application to pose estimation in bikebot riding. *Int. J. Robot. Res.* **2016**, *35*, 1459–1476. [CrossRef]
9. Gan, D.; Qiu, S.; Guan, Z.; Shi, C.; Li, Z. Development of an exoskeleton robot for lower limb rehabilitation. In Proceedings of the Conference on Advanced Robotics and Mechatronics, Macau, China, 18–20 August 2016; pp. 312–317. [CrossRef]
10. Esquenazi, A.; Talaty, M.; Packel, A.; Saulino, M. The rewalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury. *Am. J. Phys. Med. Rehabil.* **2012**, *91*, 911–921. [CrossRef]
11. Fleerkotte, B.M.; Koopman, B.; Buurke, J.H.; van Asseldonk, E.H.; van der Kooij, H.; Rietman, J.S. The effect of impedance-controlled robotic gait training on walking ability and quality in individuals with chronic incomplete spinal cord injury: An explorative study. *J. Neuroeng. Rehabil.* **2014**, *11*, 498–500. [CrossRef]
12. Meuleman, J.; van Asseldonk, E.; van Oort, G.; Rietman, H.; van der Kooij, H. LOPES II—Design and evaluation of an admittance controlled gait training robot with shadow-leg approach. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2016**, *24*, 352–363. [CrossRef] [PubMed]
13. Koenig, A.; Riener, R. The human in the loop. In *Neurorehabilitation Technology*; Reinkensmeyer, D.J., Ed.; Springer-Verlag: London, UK, 2016; pp. 161–181.

14. Alcobendas-Maestro, M.; Esclarín-Ruz, A.; Casado-López, R.M.; Muñoz-González, A.; Perez-Mateos, G.; Gonzalez-Valdizan, E.; Martin, J.L. Lokomat robotic-assisted versus overground training within 3 to 6 months of incomplete spinal cord lesion: Randomized controlled trial. *J. Neuroeng. Rehabil.* **2012**, *26*, 1058–1063. [CrossRef] [PubMed]

15. Patrick, M.; Mohamed, B. Improvement of rehabilitation possibilities with the motionmaker TM. In Proceedings of the 1st IEEE/RAS-EMBS International Conf. on Biomedical Robotics and Biomechatronics, Pisa, Italy, 20–22 February 2006; pp. 359–364. [CrossRef]

16. Wang, H.; Shi, X.; Liu, H.; Li, L.; Hou, Z.; Yu, H. Design, Kinematics, Simulation and Experiment for a Lower Limb Rehabilitation Robot. *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.* **2011**, *225*, 860–872. [CrossRef]

17. Mayr, A.; Quirbach, E.; Picelli, A.; Kofler, M.; Smania, N.; Saltuari, L. Early robot-assisted gait retraining in non-ambulatory patients with stroke: A single blind randomized controlled trial. *Eur. J. Phys. Rehabil. Med.* **2018**, *54*, 819–826. [CrossRef] [PubMed]

18. Aurich-Schuler, T.; Gut, A.; Labruyere, R. The freed module for the lokomat facilitates a physiological movement pattern in healthy people—A proof of concept study. *J. Neuroeng. Rehabil.* **2019**, *16*, 1–13. [CrossRef]

19. Bouri, M.; Gall, B.L.; Clavel, R. A new concept of parallel robot for rehabilitation and fitness: The lambda. In Proceedings of the IEEE International Conference on Robotics and Biomimetics, Guilin, China, 12–23 December 2009; IEEE: New York, NY, USA, 2009; pp. 2503–2508. [CrossRef]

20. Chisholm, K.J.; Klumper, K.; Mullins, A.; Ahmadi, M. A task oriented haptic gait rehabilitation robot. *Mechatronics* **2014**, *24*, 1083–1091. [CrossRef]

21. Akdogan, E.; Adli, M.A. The design and control of a therapeutic exercise robot for lower limb rehabilitation: Physiotherabot. *Mechatronics* **2011**, *21*, 509–522. [CrossRef]

22. Srivastava, S.; Kao, P.; Kim, S.; Stegall, P. Assist-as-needed robot-aided gait training improves walking function in individuals. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2014**, *23*, 956–963. [CrossRef]

23. Wang, H.; Zhang, D.; Lu, H.; Feng, Y.; Xu, P.; Mihai, R.V.; Vladareanu, L. Active training research of a lower limb rehabilitation robot based on constrained trajectory. In Proceedings of the International Conf. on Advanced Mechatronic Systems, Beijing, China, 22–24 August 2015; pp. 24–29. [CrossRef]

24. Hussain, S.; Xie, S.Q.; Jamwal, P.K. Adaptive impedance control of a robotic orthosis for gait rehabilitation. *IEEE Trans. Cybernet.* **2013**, *43*, 1025–1034. [CrossRef]

25. Hussian, S.; Jamwai, P.K.; Ghayesh, M.H.; Xie, S.Q. Assist-as-needed control of an intrinsically compliant robotic gait training orthosis. *IEEE Trans. Ind. Electron.* **2016**, *99*, 1–10. [CrossRef]

26. Zhang, F.; Hou, Z.G.; Cheng, L.; Wang, W.; Chen, Y.; Hu, J.; Peng, L.; Wang, H. iLeg—A lower limb rehabilitation robot: A proof of concept. *IEEE Trans. Hum.-Mach. Syst.* **2016**, *46*, 761–768. [CrossRef]

27. Leonardis, D.; Barsotti, M.; Loconsole, C.; Solazzi, M.; Troncossi, M.; Mazzotti, C.; Castelli, V.P.; Procopio, C.; Lamola, G.; Chisari, C.; et al. An EMG-controlled robotic hand exoskeleton for bilateral rehabilitation. *IEEE Trans. Haptics* **2015**, *8*, 140–151. [CrossRef] [PubMed]

28. Yepes, J.C.; Portela, M.A.; Saldarriaga, Á.J.; Pérez, V.Z.; Betancur, M.J. Myoelectric control algorithm for robot-assisted therapy: A hardware-in-the-loop simulation study. *Biomed. Eng. Online* **2019**, *18*, 3. [CrossRef] [PubMed]

29. Khoshdel, V.; Akbarzadeh, A.; Naghavi, N.; Sharifnezhad, A.; Souzanchi-Kashani, M. sEMG-based impedance control for lower-limb rehabilitation robot. *Intell. Serv. Robot.* **2017**, *11*, 97–108. [CrossRef]

30. Wittmann, F.; Lambercy, O.; Gassert, R. Magnettometer-based drift correction during rest in IMU arm motion tracking. *Sensors* **2019**, *19*, 1312. [CrossRef] [PubMed]

31. Riener, R.; Koenig, A.; Bolliger, M.; Wieser, M.; Duschau-Wicke, A.; Vallery, H. Bio-cooperative robotics: Controlling mechanical, physiological and mental patient states. In Proceedings of the IEEE International Conference on Rehabilitation Robotics, Tokyo, Japan, 23–26 June 2009; pp. 407–412. [CrossRef]

32. Wang, X.; Lu, T.; Wang, S.; Gu, J.; Yuan, K. A patient-driven control method for lower-limb rehabilitation robot. In Proceedings of the 2016 IEEE International Conference on Mechatronics and Automation, Harbin, China, 7–10 August 2016; pp. 908–913. [CrossRef]

33. Hwang, B.; Jeon, D. Estimation of the user's muscular torque for an over-ground gait rehabilitation robot using torque and insole pressure sensors. *Int. J. Control Autom.* **2018**, *16*, 275–283. [CrossRef]

34. Alia, C.; Spalletti, C.; Lai, S.; Panarese, A.; Lamola, G.; Bertolucci, F.; Vallone, F.; Di Garbo, A.; Chisari, C.; Micera, S.; et al. Neuroplastic changes following brain ischemia and their contribution to stroke recovery: Novel approaches in neurorehabilitation. *Front. Cell. Neurosci.* **2017**, *11*, 76. [CrossRef] [PubMed]
35. Craig, J.J. *Introduction to Robotics*, 3rd ed.; Pearson Education International: Upper Saddle River, NJ, USA, 2005; pp. 201–316.

Article

Detection of Participation and Training Task Difficulty Applied to the Multi-Sensor Systems of Rehabilitation Robots

Hao Yan [1], Hongbo Wang [1,2,*], Luige Vladareanu [3,*], Musong Lin [1], Victor Vladareanu [3] and Yungui Li [1]

1   Parallel Robot and Mechatronic System Laboratory of Hebei Province, Yanshan University, Qinhuangdao 066004, China; yh@stumail.ysu.edu.cn (H.Y.); lms19910704@163.com (M.L.); lyg@stumail.ysu.edu.cn (Y.L.)
2   Academy for Engineering & Technology, Fudan University, Shanghai 200433, China
3   Robotics and Mechatronics Department, Institute of Solid Mechanics of Romanian Academy, 010141 Bucharest, Romania; victor.vladareanu@imsar.ro
*   Correspondence: hongbo_w@ysu.edu.cn (H.W.); luige.vladareanu@vipro.edu.ro (L.V.)

Abstract: In the process of rehabilitation training for stroke patients, the rehabilitation effect is positively affected by how much physical activity the patients take part in. Most of the signals used to measure the patients' participation are EMG signals or oxygen consumption, which increase the cost and the complexity of the robotic device. In this work, we design a multi-sensor system robot with torque and six-dimensional force sensors to gauge the patients' participation in training. By establishing the static equation of the mechanical leg, the man–machine interaction force of the patient can be accurately extracted. Using the impedance model, the auxiliary force training mode is established, and the difficulty of the target task is changed by adjusting the K value of auxiliary force. Participation models with three intensities were developed offline using support vector machines, for which the C and σ parameters are optimized by the hybrid quantum particle swarm optimization and support vector machines (Hybrid QPSO-SVM) algorithm. An experimental statistical analysis was conducted on ten volunteers' motion representation in different training tasks, which are divided into three stages: over-challenge, challenge, less challenge, by choosing characteristic quantities with significant differences among the various difficulty task stages, as a training set for the support vector machines (SVM). Experimental results from 12 volunteers, with tasks conducted on the lower limb rehabilitation robot LLR-II show that the rehabilitation robot can accurately predict patient participation and training task difficulty. The prediction accuracy reflects the superiority of the Hybrid QPSO-SVM algorithm.

Keywords: rehabilitation robot; human–robot interaction; training task; multi-sensor system; quantum particle swarm optimization; support vector machines

1. Introduction

Neuromuscular injury can lead to disability or inconvenient movements, such as stroke and spinal cord injury, which have become important problems in the world [1]. Nowadays, there are more than 33 million stroke patients in the world [2], the mortality rate is as high as 80%, and 75% of the survivors are disabled [3]. The necessity to develop rehabilitation robots has made it one of the research hotspots in the world [4,5]. As a robot that is in direct contact with the patient, the rehabilitation robot shoulders the responsibility of helping the patient recover smoothly and safely. The human–computer interaction strategy, the energy interaction and role distribution control, are very important [6].

Clear detection of human–computer interaction and patient intention are the basis of flexible robot control. Most rehabilitation robots use force sensors to feedback mechanical information from patients, such as Hongbing Tao [7], Victor G [8]. Hwang et al. judged human motion intention by collecting pressure sensor data placed at the contact point between the standing posture rehabilitation robot and stroke patients [9]. Wolf S et al. connected the elastic element in series with the driving part and named it the Serial elastic actuator. By detecting the deformation of the elastic element, the joint moment can be detected and the motion intention of the patient can be judged [10]. Kim et al. used only one pressure sensor to realize the assistant force of the robot in the motion of the patient's elbow joint [11]. Some of them rely on current changes of the joint motors to detect motion intentions, such as Kim [12]. A few researchers use surface EMG (electromyography) signals and EEG (electroencephalogram) signals to predict the patients' motor intentions [13,14], such as Edward [15], Magdalena [16], and Tang [17]. Yepes et al. use electromyogram signals to determine the required moment of the knee joint [18]. Some researchers apply EMG signals in motion modal recognition of the prosthesis [19] and Cooperative Robot [20]. All these measurement methods have their own advantages, but EEG is susceptible to noise interference [21]. The EMG signal is not easy to collect when the skin surface changes [22,23]. Relatively speaking, it does not increase the cost of the robot and the complexity of the system. In this paper, the joint torque sensor and the hardware detection system of the six-dimensional force sensor on the sole are used.

Meanwhile, the rehabilitation effect is not only related to scientific rehabilitation training methods and reasonable training planning, but also has a great impact on the patients' active participation and active sports intention, which has been proven by clinical studies [24]. In order to improve the active participation of patients during the training process, it is necessary to provide assistance to patients according to the interaction situation during the training process [25] and to maximize the patient's independent tasks. Introducing patients' movement characteristics and physical fitness into the control strategy has a positive impact on the rehabilitation effect on the patients [26]. Yatsenko et al. controlled and adjusted the movement speed of the robotic arm according to the amplitude ratio of the EMG signal of the affected limb [27], and the patient could quickly adapt to control the movement of the prosthesis [28], but it was inconsistent with the movement characteristics of the human body. Many researches have introduced velocity field and virtual channel technology in the specific trajectory of rehabilitation training [29,30]. Cai uses impedance control to construct the velocity field in different directions of expected trajectories and provide correction force within a certain range of trajectories, with the correction force being a rigid force outside the interval threshold [31], but the threshold size setting is not given. In order to provide more accurate training for patients, radial basis function (RBF) neural network has excellent analytical ability in the patient's motor ability analysis, which was researched by Wolbrecht [32] and Pehlivan [33]. However, during the training period, the rehabilitation robot frequently interferes with the training of patients, which easily causes the side effect of relying on the machine, and fails to motivate the active participation of patients. In order to introduce the training state of patients into the control loop more accurately, many studies have judged the patient's psychological-level participation by collecting the patient's EMG signal, EEG signal, and other physiological information [34–36]. At the same time, in order to stimulate patients to actively participate in training, most rehabilitation robots currently use interesting games [37] or virtual reality technology [38].

In summary, this paper proposes a lower limb rehabilitation robot using joint torque sensors and six-dimensional force sensors on the foot soles. In the training task, man–machine interaction force information is collected, from which can be extracted characteristic quantities to predict the task difficulty by using support vector machines. The rest of this paper is organized as follows: the second section introduces the rehabilitation robot structure of multi-sensor system and human–machine interaction mechanical model. In the third section, a multi-difficulty rehabilitation training task is proposed. Under the model of impedance control, a support vector machines algorithm is used to establish the model of the patients' active participation and task difficulty detection. The fourth

section analyses the characteristic quantity of 10 healthy volunteers during different difficulty training tasks, using support vector machines (SVM) to predict the participation and task difficulty of two other volunteers.

2. LLR-II Rehabilitation Robot

2.1. Structural Design of LLR-II

In order to adapt to the patients in the early stage of rehabilitation, the lower limb rehabilitation robot LLR-II designed in this paper can be trained in two postures, so as to prevent the mechanical leg from squeezing the patient [39,40]. As to the hardware platform of the rehabilitation system, LLR-II adopts a modular design and consists of five sub-modules: lower limb mechanical leg, main control system, sensor system, multi-function seat, and mechanical limit adjustment frame, as shown in Figure 1, to validate that the rehabilitation robot can accurately predict subjects participation and training task difficulty.

Figure 1. The LLR-II Rehabilitation.

The mechanical leg is a planar three-degree-of-freedom serial mechanism, similar to the three joints of human leg, including hip, knee, and ankle. In order to solve the problem of excessive driving power of the hip joint, a self-balancing design is adopted. The knee drive component is installed on the back of the hip joint rotation axis to balance the weight of part of the mechanical leg, reduce the driving power of the hip joint, and improve the dynamic performance of the mechanical leg. The addition of an electric pushrod in the mechanical leg can automatically adapt to patients with a height of 1500 mm to 1900 mm. In order to realize the safety of sitting and lying posture training for patients, variable joint limitation consisting of fastened limit groove and driven limit groove was designed, as shown in Figure 2.

Torque sensors are mounted inside the hip and knee joint of the LLR-II sagittal plane to detect the dynamic torque characteristics of the patient's training state in real time. The dynamic torque characteristic of the ankle joint is detected by a six-dimensional force sensor mounted on the sole of the foot. The torque sensor is manufactured by Sunrise Instruments Company in China, and the six-dimensional force sensor is manufactured by Junde Technology Co., Ltd. in China. The profile of the sensor and the sensor's detailed parameters are shown in Figure 3. The output side of the reducer increases the sensitivity and accuracy of the mechanical information detection and uses this information to complete the patient's motion intention detection.

Based on the LLR-II rehabilitation training function, its electrical control system is divided into Control Center system, Movement Control system, Signal Feedback system, Human–Computer Interaction system, as shown in Figure 4. The control center system use a variety of sensors to monitor the human–computer interaction state, and uses a variety of signals to complete the planning and

training tasks; the robotic arm receives the instructions and drives the affected limbs to perform the multi-mode advanced rehabilitation training under the guidance of the driving system.
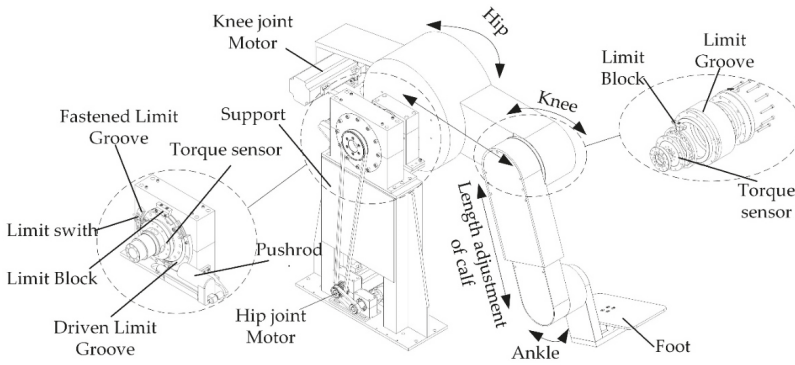
Figure 2. The detailed design of LLR-II leg mechanism.

Figure 3. The detailed design of LLR-II leg mechanism.
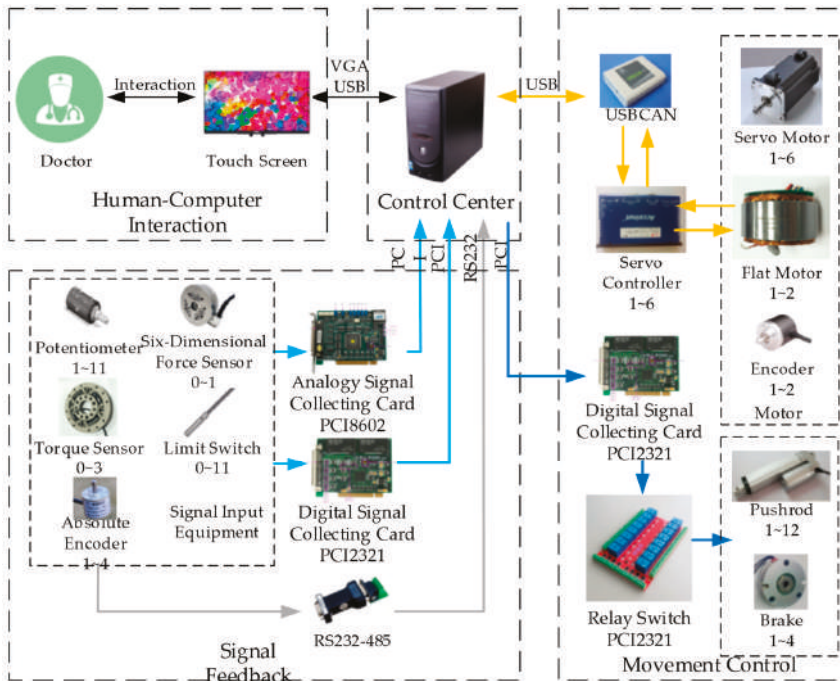
Figure 4. The sensors system composition of LLR-II.

2.2. Man–Machine Interaction Mechanics Model of LLR-II

The joint no-load moment in LLR-II man–machine coupled motion is affected by the weight of the mechanical leg and the patient's leg, and it can be expressed as a nonlinear function of joint variables [41].

$$M_{n\times1} = F(\theta_{n\times1}) \tag{1}$$

where, $M_{n\times1}$ is the column vector of joint no-load torque, $F(\bullet)$ is the mapping function, and $\theta_{n\times1}$ is the joint variable.

According to its own structure, it can be simplified as a planar three-link series mechanism. It should be noted that, considering the large weight of the leg, in order to increase the stability of the hip joint, the self-balancing design concept was introduced in the design process. The specific model can be shown in Figure 5.
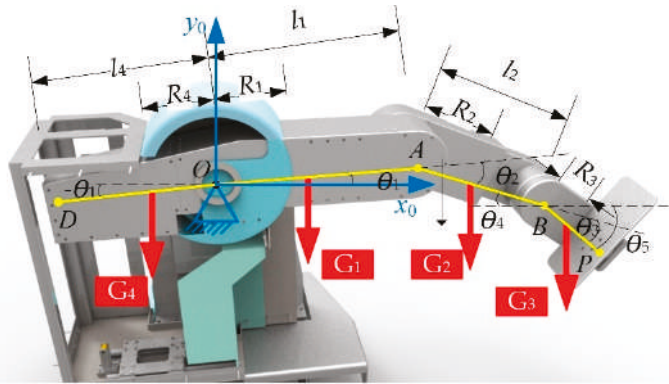
Figure 5. Leg model of lower limb rehabilitation robot.

In the Figure, l_1–l_3 represent the length of the thigh, calf, and sole, respectively; l_4 represents the length of the self-balancing part; O, A, B represent the hip, knee, and ankle joints, respectively; D, P respectively represent the first and last two endpoints; G_1, G_2, G_3 represent the weight of the machine and the patient's thigh, calf, and foot, respectively; G_4 represents the weight of the self-balancing part; R_1–R_4 represent the lengths from the center of gravity of each part to the node; $\theta_1, \theta_2, \theta_3$ represent the joint variables, in a counter-clockwise positive direction; θ_4, θ_5 represent the intermediate quantity introduced.

The joint no-load moment equation is obtained as

$$\begin{bmatrix} M_1 \\ M_2 \\ M_3 \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & \cos\theta_4 & \cos\theta_5 \\ 0 & \cos\theta_4 & \cos\theta_5 \\ 0 & 0 & \cos\theta_5 \end{bmatrix} \begin{bmatrix} G_3l_1 + G_2l_1 + G_1R_1 - G_4R_4 \\ G_3l_2 + G_2R_2 \\ G_3R_3 \end{bmatrix} \tag{2}$$

In combination with Equation (2), the above equation can be modified to:

$$\begin{bmatrix} M_1 \\ M_2 \\ M_3 \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & \cos(\theta_1 + \theta_2 + \theta_3) & \cos(-\theta_2 - \theta_1) \\ 0 & \cos(\theta_1 + \theta_2 + \theta_3) & \cos(-\theta_2 - \theta_1) \\ 0 & 0 & \cos(-\theta_2 - \theta_1) \end{bmatrix} \begin{bmatrix} G_3l_1 + G_2l_1 + G_1R_1 - G_4R_4 + f_1 \\ G_3l_2 + G_2R_2 + f_2 \\ G_3R_3 + f_3 \end{bmatrix} \tag{3}$$

It can be abbreviated as:

$$M_{3\times1} = L_{3\times3}(\theta) \bullet C_{3\times1} \tag{4}$$

In the formula, the joint no-load torque term is represented by $M_{3\times1}$, and the joint variable term is $L_{3\times3}(\theta)$, and $C_{3\times1}$ is a characteristic parameter term.

The characteristic parameter item $C_{3\times1}$ is associated with patient information, which is unique to any patient, and needs to be solved for each patient. Since joint variables L_i and θ_i can be measured by the sensor system on the robot body, but G_i cannot be directly measured by the weight of the patient's leg, the measured torque M_s is the sum of applied torque M_h and no-load torque M:

$$M_s = M_h + M \tag{5}$$

And can be obtained from Equation (4) as,

$$C_{3\times1} = L_{3\times3}(\theta)^{-1} \bullet M_{3\times1} \tag{6}$$

First, the ankle joint is moved at a small speed V, the foot pressure value f_{zd} at this time is recorded at intervals Δt, the joint angles θ_1, θ_2, and θ_3 are calculated, and a total of k times are recorded. Then the knee joint is rotated k times in the same manner, knee joint torque value M_2 and angle values θ_1, θ_2 and θ_3 are recorded, then the hip joint is rotated to record k hip joint torque values M_1 and angle values θ_1, θ_2 and θ_3, then f_{zd}, θ_3, θ_2, θ_1 are converted into M_1, θ_4, θ_5. C_{31}, C_{21} and C_{11} are calculated according to the following formulae.

$$M_{3i} = \cos\theta_{5i} C_{31i} \quad (i = 1 - k)$$
$$\overline{C}_{31} = \frac{1}{k} \sum_{i=1}^{k} \frac{M_{3i}}{\cos\theta_{5i}} \tag{7}$$

$$M_{2i} = \cos\theta_{4i} C_{21i} + \cos\theta_{5i}\overline{C}_{31} \quad (i = 1 - k)$$
$$\overline{C}_{21} = \frac{1}{k} \sum_{i=1}^{k} \frac{M_{2i} - \cos\theta_{5i}\overline{C}_{31}}{\cos\theta_{4i}} \tag{8}$$

$$M_{1i} = \cos\theta_{1i} C_{11i} + \cos\theta_{4i}\overline{C}_{21} + \cos\theta_{5i}\overline{C}_{31} \quad (i = 1 - k)$$
$$\overline{C}_{11} = \frac{1}{k} \sum_{i=1}^{k} \frac{M_{1i} - \cos\theta_{5i}\overline{C}_{31} - \cos\theta_{4i}\overline{C}_{21}}{\cos\theta_{4i}} \tag{9}$$

$$C_{3\times1} = \begin{bmatrix} \overline{C}_{11} & \overline{C}_{21} & \overline{C}_{31} \end{bmatrix}^T \tag{10}$$

The force exerted by the patient's active intention is the main feature to be identified in rehabilitation training. We judge the rehabilitation effect of the patient by identifying the force that the patient can produce actively. In the training process, the actual measurement of human–machine interaction force is the data measured by the sensor system of the robot. The following equation is the established equivalent terminal mechanical model of human patients.

$$f_{2\times1} = H(\theta_{3\times1}, M_{3\times1}, M_{s3\times1}) \tag{11}$$

In the formula, $f_{2\times1}$ is static terminal forces in the plane of motion, $\theta_{3\times1}$ is current position lower joint variable, $M_{3\times1}$ is three-joint no-load torque, $M_{s3\times1}$ is measured force/torque at the three joints, and $H(\bullet)$ is the mapping function.

In the process of human–machine motion, since the force exerted by the patient mainly acts on the pedal, a six-dimensional force sensor is placed in the middle of the bottom of the pedal. Due to the influence of the arch structure of the foot, the force at the heel and the forepaw is simplified to two points: B and P. The force at the heel generates torque mainly at the hips and knees, and the force at the forepaw generates pressure on the foot pedal, as shown in Figure 6.
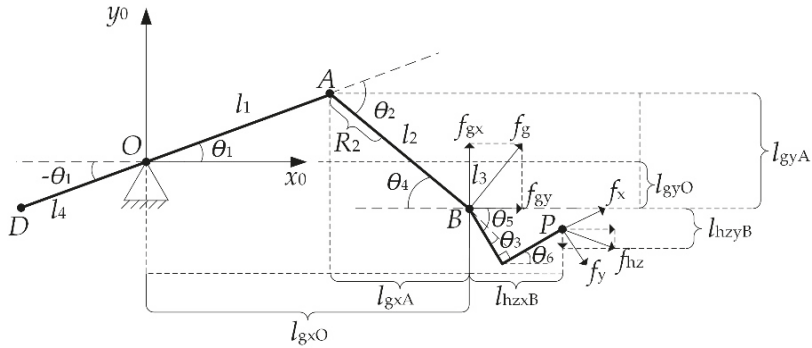
Figure 6. End applied force model.

In Figure 6, f_g represents the patient applying force at point B; f_{gx}, f_{gy} and denote the horizontal and vertical resolution of f_g, respectively; f_{hzx}, f_{hzy} represent the horizontal and vertical resolution of foot forepaw forces, respectively; l_{gxO}, l_{gxA} represent the moment arms generated by f_{gx} at point A and O, respectively; l_{gyO}, l_{gyA} represent the moment arm generated by f_{gy} at point A and O, respectively; l_{hzxB}, l_{hzyB} represent the moment arms generated by f_{hzx} and f_{hzy} at point **B**; f_x and f_y represent the force measured by the sensor mounted on the sole of the robot foot, respectively.

The moment of hip and knee joint can be expressed as:

$$\begin{bmatrix} M_{h1} \\ M_{h2} \end{bmatrix} = \begin{bmatrix} l_2 \sin\theta_5 + l_1 \sin\theta_1 & l_2 \cos\theta_5 + l_1 \cos\theta_1 \\ l_2 \sin\theta_5 & l_2 \cos\theta_5 \end{bmatrix} \begin{bmatrix} f_{gx} \\ f_{gy} \end{bmatrix} \tag{12}$$

The patient's heel force f_g can be expressed as:

$$f_g = \begin{bmatrix} f_{gx} \\ f_{gy} \end{bmatrix} = \begin{bmatrix} l_2 \sin\theta_5 + l_1 \sin\theta_1 & l_2 \cos\theta_5 + l_1 \cos\theta_1 \\ l_2 \sin\theta_5 & l_2 \cos\theta_5 \end{bmatrix}^{-1} \begin{bmatrix} M_{h1} \\ M_{h2} \end{bmatrix} \tag{13}$$

From which can be obtained

$$f_g = \frac{\begin{bmatrix} l_2 \cos\theta_5 & -(l_2 \cos\theta_5 + l_1 \cos\theta_1) \\ -l_2 \sin\theta_5 & l_2 \sin\theta_5 + l_1 \sin\theta_1 \end{bmatrix} \begin{bmatrix} M_{s1} - M_1 \\ M_{s2} - M_2 \end{bmatrix}}{((l_2 \sin\theta_5 + l_1 \sin\theta_1)(l_2 \cos\theta_5) - (l_2 \sin\theta_5)(l_2 \cos\theta_5 + l_1 \cos\theta_1))} \tag{14}$$

The force exerted on the patient's forefoot is collected by a six-dimensional force sensor on the sole of the foot that has the same axial direction as the pedal, so the end force f_{hz} can be decomposed as follows:

$$f_{hz} = \begin{bmatrix} f_{hzx} \\ f_{hzy} \end{bmatrix} = \begin{bmatrix} f_x \cos\theta_6 + f_y \sin\theta_6 \\ f_x \sin\theta_6 - f_y \cos\theta_6 \end{bmatrix} \tag{15}$$

Then, the equivalent terminal force of the patient can be calculated as:

$$f = f_g + f_{hz} = \begin{bmatrix} F_x \\ F_y \end{bmatrix} \tag{16}$$

where F_x, F_y are the horizontal and vertical components of terminal force;

By combining Equations (13) and (15), the terminal static component can be expressed as follows:

$$F_x = \frac{l_2\cos\theta(M_{s1}-M_1)-(l_2\cos\theta+l_1\cos\theta_1)(M_{s2}-M_2)}{\left((l_2\sin\hat{\theta}+l_1\sin\theta_1)(l_2\cos\hat{\theta})-(l_2\sin\hat{\theta})(l_2\cos\hat{\theta}+l_1\cos\theta_1)\right)}+$$
$$f_x\cos(\theta_1+\theta_2+\theta_3+\pi/2)+f_y\sin(\theta_1+\theta_2+\theta_3+\pi/2)$$

(17)

In the formula, $\hat{\theta}$ is intermediate variable of joint Angle, $\hat{\theta}=-\theta_1-\theta_2$.

$$F_y = \frac{-l_2\sin\hat{\theta}(M_{s1}-M_1)+(l_2\sin\hat{\theta}+l_1\sin\theta_1)(M_{s2}-M_2)}{\left((l_2\sin\hat{\theta}+l_1\sin\theta_1)(l_2\cos\hat{\theta})-(l_2\sin\hat{\theta})(l_2\cos\hat{\theta}+l_1\cos\theta_1)\right)}+$$
$$f_x\sin(\theta_1+\theta_2+\theta_3+\pi/2)-f_y\cos(\theta_1+\theta_2+\theta_3+\pi/2)$$

(18)

Equations (17) and (18) can completely solve the mapping relationship between terminal force and joint variables, no-load torque and measured torque mentioned in Equation (11), and provide the entry parameters for the following judgment of patients' motion intention and control strategy.

3. Participation Detection of LLR-II

3.1. Assist Force Training Control

According to the change of the patient's participation in the training process, the assist mode and the active training mode are divided into different grades to ensure that the patient completes the training and maximize the patient's training enthusiasm and task completion. Using the impedance control model, the human–computer interaction force is represented by the end position offset, and is magnified by game in the task. With the participation of the robot's assistant force, the task difficulty is classified, to ensure that patients can find suitably challenging rehabilitation tasks. In order to improve the level of the patients' active participation, according to the recognized level of physical participation, the size of the auxiliary force is adjusted in real time to ensure that patients maintain a high level of participation for training.

With the progress of rehabilitation training, the patient gradually has a certain ability to control the affected limb, but when it is not enough to fully control it, it is necessary to introduce assistance training, in which the robot obtains the patient's motion intention through the force/torque sensor, and then drives the affected limb for training. In order to improve the coordination ability of each joint, the patient needs to complete the trajectory training, such as the circular trajectory and the linear trajectory. In many cases, the patient does not have the ability to perform the trajectory training task independently, and the robot needs to assist in suppressing the wrong movement. The assistance training control mode introduces the impedance model as shown in Figure 7. According to the current joint actual position θ_a, the position positive solution is compared with the current desired position of the training trajectory, the auxiliary force calculation is performed, the auxiliary force F_T is obtained, the patient force F_a is summed, and the result is sent to the impedance controller to obtain the end position control amount P_d. It then inversely solves the position, calculates the desired joint position θ_d, and transmits it to the position controller to realize the assist control.
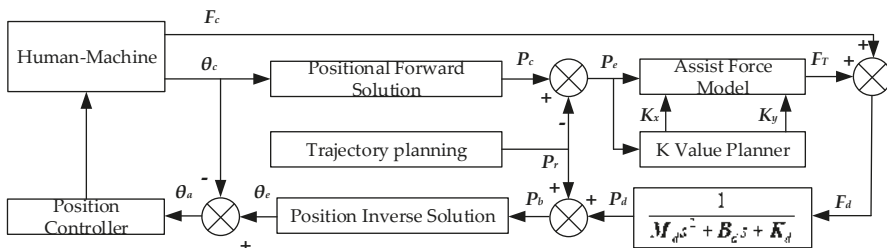

Figure 7. Assistance training control block diagram.

In order to make patients intuitively understand the movement track of their affected limbs, a task game was designed in which the patient operated the virtual mice to walk in the safe area between the red lines. The position of the mouse on the screen reflects the position of the patient's limb end in the motion plane. Participation scores increased with the time the mice spent walking in the safe area, and did not increase when the mice were outside the safe area. The trajectory of the safe passage can be selected according to the length of the patient's limb, and the width of the safe passage is related to the parameters of the impedance model. The impedance model parameters are as follows:

$$M = \begin{bmatrix} 0.0625 \\ 0.0625 \end{bmatrix}, B = \begin{bmatrix} 5 \\ 5 \end{bmatrix}, K = \begin{bmatrix} 1000 \\ 1000 \end{bmatrix} \tag{19}$$

In order to make the width of the safety passage between the red lines challenging for most patients, and not tedious at the same time, 70 kg is selected as the standard reference value for the weight of patient, and the positional offset of the standard reference value is selected as the width of the safety channel. The standard reference value of 70 kg was selected as appropriate to the lot of volunteers that the machine was tested on, and it is used to provide an approximate starting point to the initial conditions of the algorithm. During the training task, patients need to resist the weight of their limbs and control the virtual mice to walk at a constant speed in the safe passage. If the virtual mice touch the red line during the task, the physical strength of the mice will decline until the end of the training mission cycle. According to the size of the auxiliary force, the task difficulty is divided into nine levels, with K values ranging from 0 to 0.8, respectively. The degree to which patients participate in training is related to the degree of the patient's recovery and individual physical strength. These parameters are difficult to quantify. Therefore, the degree of the patients' participation in training is quantified in stages by means of an experimental questionnaire. In the course of the experiment, patients are asked to try nine different difficulty training tasks, and they are asked to accept questionnaires to determine the current situation. Task difficulty is appropriate for each patient. Tasks with different difficulty can be divided into three states: under-challenge, challenge, and over-challenge, which are expressed by $-1, 0, 1$.

3.2. Patient Participation and Training Task Difficulty Prediction Model

In order to predict the degree of the patient's task participation, a mathematical model based on support vector classifiers and regression machines was established according to the characteristics of the small sample and nonlinear data of a small number of patients' training data and questionnaire data. The characteristic parameters were extracted from the training data, and the data was analyzed. The implicit mathematical relationship between input value and output value predicts the actual participation of patients, so as to achieve the goal of selecting the appropriate task difficulty.

Using the characteristic quantity of patient training data and task difficulty states, a QPSO-MLSSVM (quantum particle swarm optimization and multi-output least squares support vector machine) model can be established and tested. This model is based on the LS-SVM (least squares support vector machines) model, which is a class of kernel-based learning methods normally used for regression and classification problems. The main distinction in LS-SVM is solving for a set of linear equations, rather than the quadratic programming problem in classical SVMs [42]. The QPSO (quantum particle swarm optimization) algorithm is used to optimize the key parameters in the model to make the model performance better [43,44]. The sample set is $\{(x_i, y_i), i = 1, 2, \ldots, l\}$, where $x_i \in R^n$ is the input value of the *ith* sample, $y_i \in R$ is the output value of the *nth* sample. The assumption is

$$f_i(x) = \omega^T \bullet \Phi(x_i) + b_i, i = 1, 2, \ldots, l \tag{20}$$

where, $\Phi(x_i)$ is the spatial conversion variable function, ω is the weight vector, b is the adjustment parameter.

We optimize the confidence interval under this condition, and transform the optimization problem into the minimum value problem according to the principle of structural risk minimization [45]:

$$
\begin{cases}
\min & \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{l}\xi_i \\
s.t. & y_i \bullet f_i(x) \geq 1 - \xi_i, i = 1,2,\ldots,l \\
& \xi_i \geq 0, i = 1,2,\ldots,l
\end{cases}
\tag{21}
$$

where, C is the weight coefficient; ξ_i is the relaxation factor.

The first item in the optimization problem reflects the generalization ability and the model complexity, the second item reflects the model error, and the parameter C adjusts the weight of these two items. Introducing the Lagrange equation into the above formula:

$$
L = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\left(\xi_i + \xi_i^*\right) - \sum_{i=1}^{l}a_i\left(\varepsilon + \xi_i - y_i + \left(\omega^T\bullet\Phi(x_i) + b\right)\right)
$$
$$
- \sum_{i=1}^{l}a_i^*\left(\varepsilon + \xi_i^* + y_i - \left(\omega^T\bullet\Phi(x_i) + b\right)\right) - \sum_{i=1}^{l}\left(\eta_i\xi_i + \eta_i^*\xi_i^*\right)
\tag{22}
$$

where $\xi^{(*)}$, $\alpha^{(*)}$ and $\eta^{(*)}$ represent ξ, α, η with * and without *, $\alpha^{(*)}$ and $\eta^{(*)}$ are Lagrange multipliers. A relaxation variable is introduced $\xi_i, \xi_i^* \geq 0, i = 1,2,\ldots,l$, ξ is the insensitive.

The radial basis function is selected to calculate the spatial inner product of the kernel function in the support vector machine model. The result obtained by the above formula is inserted back into the Lagrange equation to obtain the dual equation of the optimization function:

$$
\max W\left(a_i, a_i^*\right) = -\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\left(a_i - a_i^*\right)\left(a_j - a_j^*\right) \times K\left(x_i, x_j\right) + \sum_{i=1}^{l}\left(a_i - a_i^*\right)y_i - \sum_{i=1}^{m}\left(a_i + a_i^*\right)\varepsilon_i
\tag{23}
$$

The constraint of this dual equation is:

$$
\begin{cases}
\sum_{i=1}^{l}\left(a_i - a_i^*\right) = 0 \\
a_i, a_i^* \in (0, C)
\end{cases}
\tag{24}
$$

where, $C > 0$ is the Penalty parameter

When $0 < \alpha_i < C$, $\xi_i = 0$; when $0 < \alpha_i^* < C$, $\xi_i^* = 0$, the corresponding sample is the standard support vector, and expresses the reliability of the calculation. In general, the b value of the standard support vector is calculated respectively, and then the average value is calculated.

$$
b = \frac{1}{N_{NSV}}\left\{\sum_{0<\alpha_i<C}\left[y_i - \sum_{x_i\in SV}\left(\alpha_i - \alpha_i^*\right)\Phi(x_i)\bullet\Phi(x_i) - \varepsilon\right]\right.
$$
$$
\left. + \sum_{0<\alpha_i^*<C}\left[y_i - \sum_{x_i\in SV}\left(\alpha_i - \alpha_i^*\right)\Phi(x_i)\bullet\Phi(x_i) + \varepsilon\right]\right\}
\tag{25}
$$

In order to eliminate local optima problems, the QPSO algorithm and the SVM algorithm are mixed the Hybrid QPSO-SVM algorithm. The formula of QPSO is:

$$
\begin{cases}
m_{best} = \frac{1}{M}\sum_{i=1}^{M}P_i \\
P_{C_{ij}} = \phi P_{ij} + (1-\phi)P_{gj} \\
x_{ij}(t+1) = P_{C_{ij}} \pm \alpha\left|m_{bestj} - x_{ij}(t)\right|\ln\left(\frac{1}{u}\right)
\end{cases}
\tag{26}
$$

And the particle swarm velocity formula is:

$$v_{ij}(t+1) = \omega \bullet v_{ij}(t) + c_1 r_{1j}\left[P_{ij}(t) - x_{ij}(t)\right] + c_2 r_{2j}\left[P_{gj}(t) - x_{gj}(t)\right] \tag{27}$$

where, p_{ij}, p_{gj}, p_{ij} are the optimal positions of the i particle and the g particle in the j dimension, respectively; m_{best}, m_{bestj} are center points of the current best position of all individuals on all dimensions; M is the particle swarm size; p_i is the current best position of the i particle; $p_{c_{ij}}$ is the random position between p_{ij}, p_{gj}; α is the control coefficient.

QPSO optimizes two key parameters, C and σ of the MLSSVM, and the optimization goal is minimizing the *fitness*(σ, γ) function. The sample mean square error (MSE) is selected as the particle swarm fitness function.

$$fitness(\sigma, \gamma) = \frac{1}{M}\sum_{i=1}^{M}\left(y_i - \hat{y}_i\right)^2 \tag{28}$$

where, y_i is the actual value and $\hat{y}_i$ is the predicted value. When *fitness* reaches its minimum, the optimal solution is obtained.

4. Experiment

In order to verify the effectiveness of this method for patients, 12 groups of healthy volunteers were tested. All subjects gave their informed consent for inclusion before they participated in the study. The study was conducted in accordance with the Declaration of Helsinki, and the protocol was approved by the Ethics Committee of Yanshan University. Because the speed of physical expenditure for stroke patients is different in difficult tasks and due to the changes of the assistant force parameter K, maintaining the foot in a safe area requires a different level of initiative. With the increase of the difficulty coefficient, the K value of the assisted force parameter decreases gradually. Meanwhile, the patients' goal participation will increase during the training process, which may lead to a rapid decline in the patients' physical strength. So the human–computer interaction exerted by the patient at the end of the robotic chain is related to the degree to which the patient participates in the task. The degree of the patients' participation in tasks is also closely related to the difficulty of the tasks. To verify this point, the force/moment of three joints is calculated as the terminal force in robot coordinates. Secondly, the obtained data yields an observation of the changes of the calculated end force in the training cycle and a comparison of the position of the end of the robot under different parameter K values of the assisted force.

Figure 8 shows the end force after transforming the data collected by the sensor into the end force of the robot coordinate after eliminating the self-weight of the robot. Under the assists force with K = 0.4 task difficulty, the human–robot interaction force keeps at a relatively low level for a period of time at the beginning of training. At 380 s, the volunteer is too weak to bear his own weight to complete the task, and the Human–robot interaction has reached its first peak. At 400 s, the volunteer challenges himself again and strives to achieve the goal of the task, so the human–robot interaction force declines rapidly because the volunteer takes the initiative to bear the weight of their limbs. However, the second phase of maintaining a lower level is shorter than the first phase, and the second peak of the interaction force appears. At the end of the training, it is difficult for the volunteer to bear part of the body weight again in order to achieve the goal of the task, so the interaction force, which is almost entirely composed of the volunteer's limb weight, keeps at a high level. The variations in the time period of repeated challenges for a volunteer at different task difficulties are shown in Figure 9.
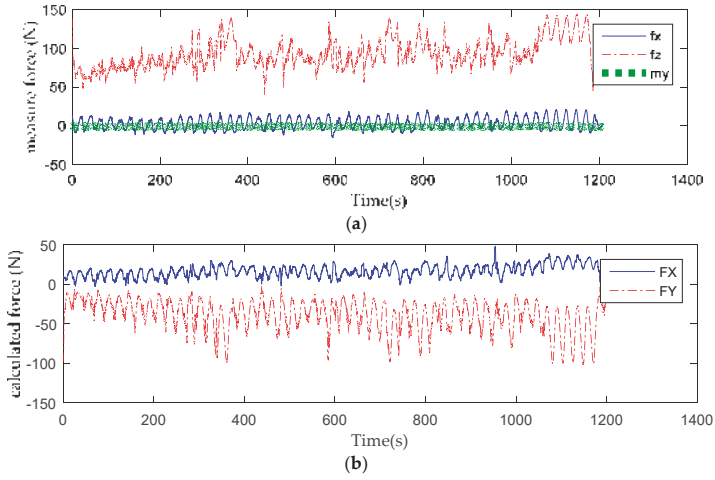
Figure 8. Sensors measuring terminal force during training: (**a**) Human–computer interaction of six-dimensional force acquisition under the training task of assistant force parameter K = 0.4; (**b**) the calculated terminal force in robot coordinates under the training task of assistant force parameter K = 0.4.
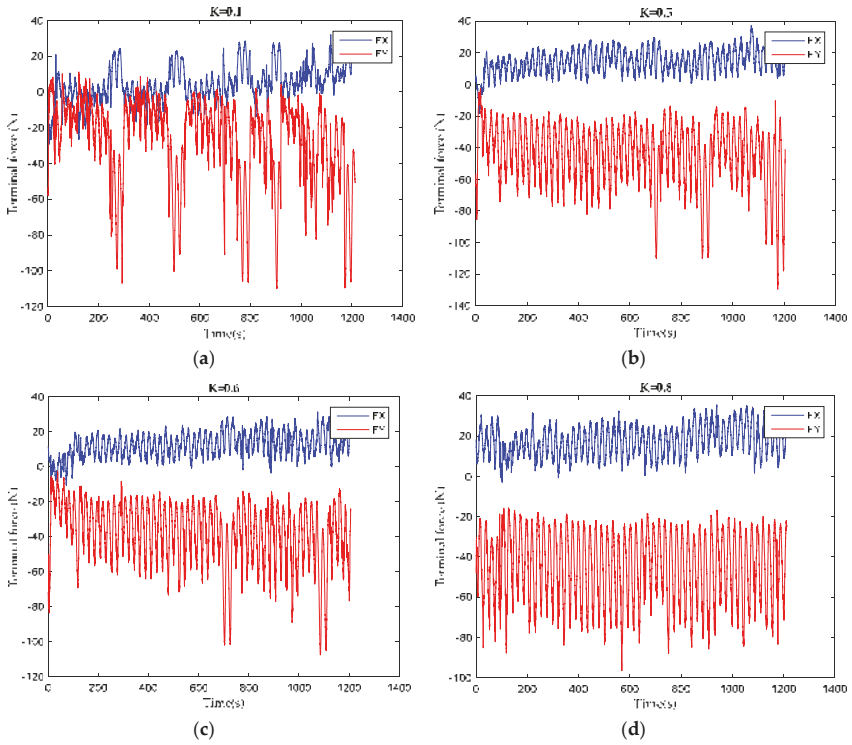
Figure 9. Terminal force during different difficulty tasks training: (**a**) the training task of assistant force parameter K = 0.1; (**b**) the training task of assistant force parameter K = 0.5; (**c**) the training task of assistant force parameter K = 0.6; (**d**) the training task of assistant force parameter K = 0.8.

The picture above is the terminal force of the first volunteer in training tasks with different K values of the assist force. After a questionnaire survey of the volunteer, the K = 0.1 of assist force task for the patient is "over–challenge", and the K = 0.5, K = 0.6 of the assist force task are "challenge", with the K = 0.8 of the assist force task being "under-challenge". Under the over-challenge task, there being heavier limb weights to load, the volunteer's physical exertion is fast and there appear many peaks in the human–robot interaction force. With the increasing participation of assistive forces in training, volunteers need less initiative to achieve task goals. This phenomenon can be clearly seen by observing the relationship between the end position and the safe passage.

Figure 10 shows the relationship between the terminal position of the robot and the safe passage of the target in the different difficulty training tasks for the first volunteer. The blue line is the rehabilitation robot terminal position, and the green line is the target terminal trajectory, while the red line is the safe passage. The difficulty of K = 0, K = 0.1, K = 0.2, K = 0.3 assist force tasks are "over-challenge", K = 0.4, K = 0.5, K = 0.6 assist force tasks are "challenge", and K = 0.7, K = 0.8 assist force tasks are "under-challenge". It is difficult for the volunteer to complete the task goal in the over-challenged task. In the under-challenged task, it's easy for volunteers to reach the goal position.
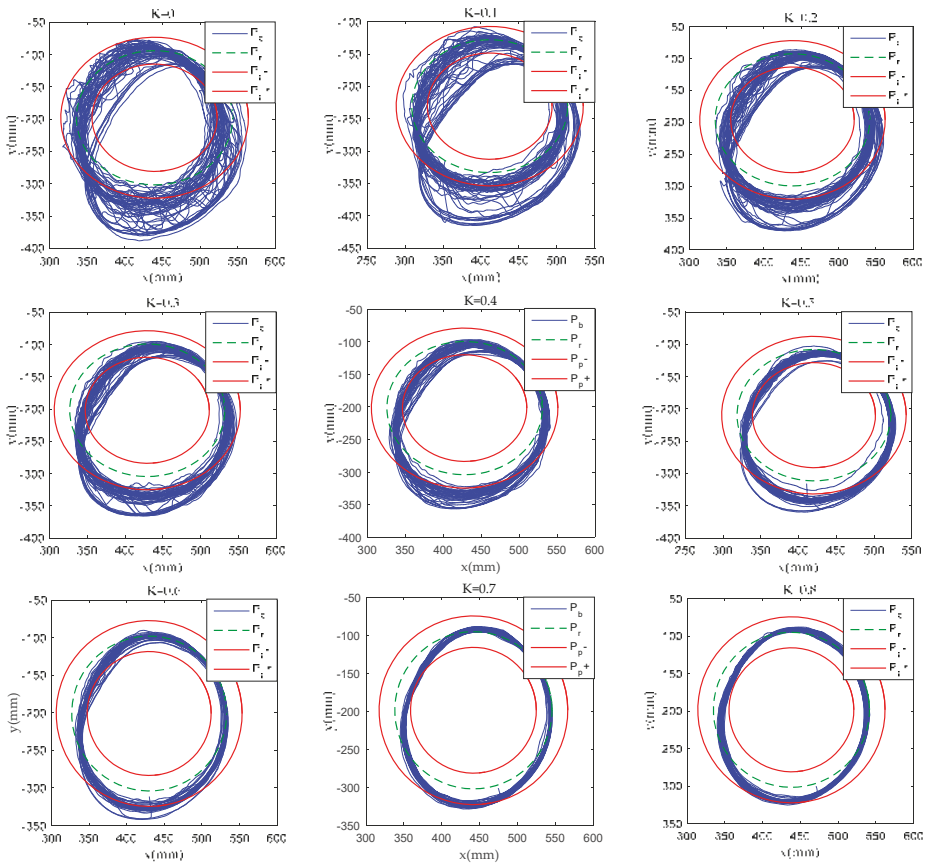
Figure 10. Terminal position during different difficulty tasks training.

The degree of patient participation under different task difficulties is reflected in the fluctuation of human–machine interaction mechanical signals, and the feature fluctuation represents the fluctuation of signal data in this period. The greater the fluctuation of the feature, the greater the degree of

dispersion, as it is more sensitive to signal fluctuation, and it is more suitable to be used as an input parameter of the detection model for patient participation. The sample data is processed by using four indicators, describing the degree of dispersion of the signals, the interquartile range, and the variance. The above features in the data are statistically analyzed to judge their significance and correlation under different volunteer states. Significance analysis is undertaken to compare the feature data of "under-challenge" and "over-challenge" volunteer states with that of the "challenge" state, in order to judge that the feature data have significant differences in the three states. The correlation is done to compare the insignificant features with the degree of volunteer participation, whether and if there is correlation. This feature will still be used as an input parameter to train the support vector machine. This paper extracted the preliminary feature variables in Table 1.

Table 1. Characteristic parameters of volunteer participation.

| Type	Description
P_{RMSE}	Mean square error of position
P_{STD}	Position standard deviation
T_{SCA}	The proportion of time outside the safe passage
F_Q	Inter-quartile range of terminal force
U_{MAX}	Maximum value in frequency domain of terminal force
f_{MAX}	Peak frequency in frequency domain of terminal force
F_D	Component at frequency 0 in frequency domain of terminal force
F_{VAR}	Variance of terminal force
P_{OR}	Offset range of position
Uh_{MAX}	Maximum value in frequency domain of volunteer motivation
fh_{MAX}	Peak frequency in frequency domain of volunteer motivation
P_{MAE}	Mean absolute error of position

Ten volunteers were selected to carry out the experimental verification of the participation and task difficulty detection. Each volunteer was trained in 10 difficult tasks that lasted from 15 min to 20 min. In order to eliminate the influence of physical energy consumption between each experiment, they were conducted one day apart. After the end of the experiment, a questionnaire survey was conducted on the difficulty of the task, which is divided into three participation levels: "under-challenge", "challenge", and "over-challenge". As part of the experiment, the data of hip and knee joint torque, plantar six-dimensional force, and terminal trajectory were collected. With the different participation of assistive force, there are different performances of the terminal force and position. The characteristic quantities were extracted, as shown in Table 2, from the training data. The training data characteristic quantities of volunteers were then compared to their classification, according to the predicted task difficulty. The pairwise t-test comparisons of the characteristic quantities were statistically analyzed to verify whether the characteristic quantities are significantly different under different task difficulties. Comparisons of the characteristics of each two difficult tasks, using one-way repeated measure ANOVA, were done separately. Table 2 shows the results of the significance analysis of the characteristic quantity in the difficulty of the three tasks. p value is the test probability, F value is the effect of random error. When p value is less than 0.05, the characteristic quantity has significant difference under different difficult tasks.

The significance analysis of the characteristic quantity from the training data of 10 volunteers shows that there are obvious differences when comparing the P_{RMSE}, P_{STD}, T_{SCA}, P_{MAE} among different volunteers. The p value of F_Q, U_{MAX}, F_D are greater than 0.05, only in the case of the difficult and medium groups. For F_Q, U_{MAX}, F_D there are obvious differences in other groups, as it can distinguish the difficulty of the under-challenge tasks. Although P_{OR} has significant differences, its value is rough and its stability is not high. Accordingly, P_{RMSE}, P_{STD}, T_{SCA}, F_Q, U_{MAX}, F_D, P_{MAE} are used as feature inputs for volunteer participation and task difficulty classification. Figure 11

shows more intuitively the difference in training characteristics among three difficulty levels for each volunteer.

Table 2. Significance comparison of characteristic quantities.

Comparison	P_{RMSE}		P_{STD}		T_{SCA}	
	P	**F**	**P**	**F**	**P**	**F**
Difficult/Medium	3.19×10^{-4}	1.09×10^{-11}	1.5×10^{-4}	3.76×10^{-10}	1.44×10^{-3}	5.9×10^{-4}
Difficult /Easy	2.74×10^{-7}	3.2×10^{-14}	4.26×10^{-8}	1.08×10^{-12}	2.06×10^{-11}	1.9×10^{-4}
Medium/ Easy	8.08×10^{-9}	0.1	1.26×10^{-8}	0.112	6.13×10^{-10}	0.634
Comparison	F_{Q}		U_{MAX}		f_{MAX}	
	P	**F**	**P**	**F**	**P**	**F**
Difficult/Medium	0.844	0.0904	0.136	0.07	0.39	0.382
Difficult /Easy	9.97×10^{-3}	0.01	6.37×10^{-4}	0.028	0.028	0.007
Medium/ Easy	4.7×10^{-3}	0.339	0.0147	0.65	0.154	0.066
Comparison	F_{D}		F_{VAR}		P_{OR}	
	P	**F**	**P**	**F**	**P**	**F**
Difficult/Medium	0.735	2×10^{-5}	0.255	0.001	7.59×10^{-6}	0.0035
Difficult /Easy	0.033	0.221	0.961	0.072	3.07×10^{-13}	5.66×10^{-11}
Medium/ Easy	0.0013	0.002	0.193	0.212	2.922×10^{-8}	1.78×10^{-5}
Comparison	Uh_{MAX}		fh_{MAX}		P_{MAE}	
	P	**F**	**P**	**F**	**P**	**F**
Difficult/Medium	0.0013	0.009	0.001	0.6789	0.0005	5.38×10^{-12}
Difficult /Easy	0.1478	0.003	0.072	0.5531	3.19×10^{-7}	4.29×10^{-13}
Medium/ Easy	0.5216	6.06×10^{-7}	0.929	0.8565	2.32×10^{-9}	0.339
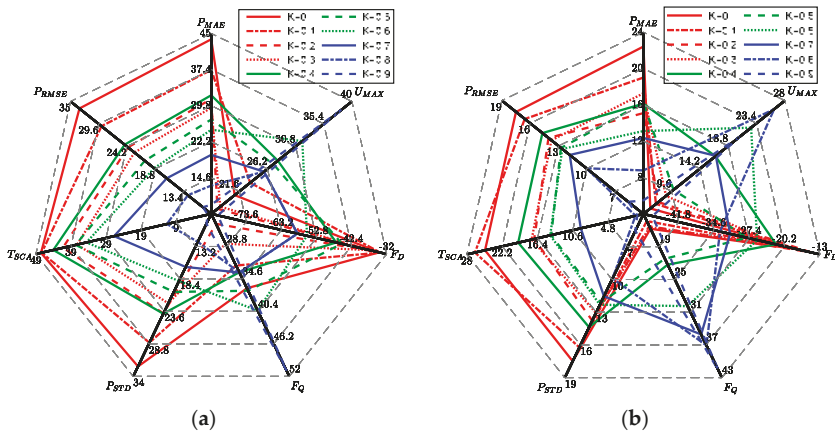
Figure 11. Characteristic quantity of training data under different task difficulties: (**a**) characteristic quantity of Volunteer 1[#] training data; (**b**) characteristic quantity of Volunteer 2[#] training data.

Figure 11 shows the characteristics of two volunteers under different task difficulty. Among them, the red line is under the task difficulty of over-challenging and difficult; the green line is under the task difficulty of challenging; the blue line is under the task difficulty of under-challenge. P_{RMSE}, P_{STD}, T_{SCA}, F_{D}, P_{MAE} are positively correlated with the task difficulty evaluation, and F_{Q}, U_{MAX} are negatively correlated with the task difficulty evaluation.

The training data of 100 groups of 10 volunteers receiving the test were used as training sample data. At the same time, another two volunteers were randomly selected as the predictive group. Two volunteers in the predictive group were trained in all tasks with different difficulty levels, and questionnaires were conducted on task difficulty. Their training data is used as predictive sample data. Extracted feature quantities X_s as input from 100 sets of experimental sample data and the training set known category information Y_s (Task Difficulty of Patient Evaluation) were taken as output. The prediction model based on QPSO-MLSSVM hybrid optimization algorithm and the two comparison models based on MLSSVM algorithm and a neural network algorithm were established. QPSO is an iterative optimization that optimizes the parameters C and σ in the MLSSVM algorithm to improve the generalization ability and prediction accuracy of the model. The 20 sets of data in the test set are similar to the ones in the training set, and seven feature quantities X_l are extracted as inputs of the existing model, and the predicted output Y_p is obtained by the model operation. The accuracy of the model was evaluated by a minimum mean square deviation operation between Y_l (Task Difficulty of Patient Evaluation) and Y_p.

In the analysis of the results, the training samples obtained from the mathematical model cannot directly reflect the prediction ability of the model. Further model evaluation can be achieved by comparing the prediction data of the training samples with the real data. Common evaluation indexes of the model include MSE, RMSE, correlation coefficient, and so forth. In this paper, the mean square error and the correlation coefficient are used to evaluate the model.

The data from 100 datapoints of the volunteers' participation status were input into the prediction model to train the model, and the prediction tested on 20 datapoints. Correlation analysis was conducted on the actual values and predicted values of the data, and the linear fitting results are shown in Figure 12.
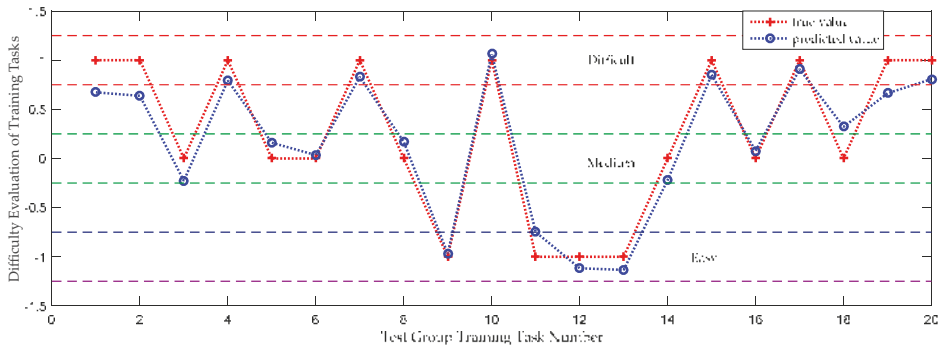
Figure 12. Comparison between task difficulty prediction and reality of test group.

In order to further analyze the classification effect of the QPSO-MLSSVM support vector machine on the state of volunteer participation, the minimum mean square error (MSE) and mean absolute error (MAE) as well as Standard Deviation (MAPE) of the predicted and true values of various volunteer participation states was obtained, as shown in Table 3

Table 3. Significant comparison of characteristic quantities.

	MSE	MAE	STD
Matching degree	0.0428	0.1822	0.1006

Training concentration simply divides task difficulty evaluation into −1, 0, and 1. Because patients have different evaluation criteria for difficulty, the dynamic trend in training data is different, and the

results after algorithm testing will be distributed around three values. If the test results are graded according to the difficulty of (−1.5, −0.75), (−0.25, 0.25), (0.75, 1.25), the accuracy of the test can reach 100%. In order to eliminate the result of slightly larger offset, the determination range is reduced to half of the original one that are (−1.25, −0.5), (−0.5, 0.5), (0.5, 1.5). And the test accuracy can still reach 80%. The matching result of task difficulty evaluation shows that the predicted value of task difficulty is close to the real value, which also verifies that volunteers' evaluation of the difficulty of training tasks can be obtained from training data.

5. Discussion

Early rehabilitation training for stroke patients is very important and effective. While the rehabilitation robot can be used in the later stages of recuperation and even as a workout enhancer, the research work is aimed at the early stages of post-trauma rehabilitation. The aim is to re-train the nerve control and brain–body associations for typical movements of the affected limb. Overall, the success of training is measured in how quickly and effectively a patient regains normal control of their limbs. In order to make patients take the initiative to participate fully, while not letting the patient's physical strength drop rapidly, dispelling the enthusiasm of patient training, is a complex problem. It is very important to choose the appropriate training task difficulty for patients. Therefore, this paper determines whether the current task difficulty is suitable for patient training to achieve optimal training effect based on the data of the patient in the training task. The final experiment in the paper proves that the matching degree of task difficulty evaluation of the two volunteers in the test group was worse than that of the test difficulty evaluation of the volunteers in the experimental group from the fitness curve. This is due to the volunteer's subjective persistence and subjective evaluation of the difficulty of the task. However, the support vector machine task difficulty judgment model still has a prediction accuracy of 80% for the volunteer task difficulty evaluation of the test group. As the training data continues to increase and a variety of training information is introduced, the prediction accuracy of the judgment model will become higher and higher.

As can be seen from Figures 8 and 10, when volunteers perform multiple training tasks with different difficulty, with the decrease of difficulty and the increase of the proportion of assistant power, the strength needed by volunteers to achieve the goal task position and the speed of physical consumption will be reduced. During clinical trials, most volunteers have emotional issues when performing challenging tasks. Most of them have low mood, and some are irritated. They need to continue their psychological counselling and speech encouragement to support their task training. When the volunteers were under challenging tasks, most of them felt bored and emotionally stable. It may have an effect on the experimental results for the frequency of speech encouragement during training. In this clinical trial, speech encouragement was given four times in each training process to minimize the influence of this factor. In the future, the research team will use a variety of measurements to study the emotional and physical characteristics of volunteers to verify their impact on rehabilitation training.

In this paper, the performance of volunteers in training tasks at different levels of difficulty is investigated in order to determine whether the task difficulty is appropriate and to verify and judge the past data. But it also proves the validity and universality of the assistant training strategy. This control strategy can maximize the ability of patients to actively participate in training. In the future, the research team will continue to study and improve upon such clinical trial data. It is expected that the task difficulty can be judged and predicted online, and then the assistant force can be adjusted in real time, so that patients can participate in training actively and optimally.

6. Conclusions

This paper studies a seated and reclining training lower limb rehabilitation robot with a multi-joint sensing system. In order to make the patient participate actively in the training task, an assistive force training control strategy and corresponding task difficulty are proposed. The multi-joint mechanical

sensing system is used to solve the more accurate end mechanical model, and then the human–computer interaction force is detected. Clinical trials of 10 volunteers were conducted, and each volunteer underwent nine levels of difficulty training. Through the optimized support vector machines algorithm, quantitative features in the training data are taken as the input set, and the volunteer's evaluation of the task difficulty is taken as the output set, and a task difficulty judgment model based on the volunteer training data is obtained. The training difficulty of two other volunteers, not in the original 10 persons training set, was predicted. It was verified that the difficulty judgment model of the task was universal and could exclude the influence of body size and weight. By comparing the prediction results of various algorithm models, the accuracy and convergence speed of the optimization algorithm are verified.

Future work will concentrate on extending the research to alternative models, such as described in the introduction, with a detailed comparison providing possible improvements to the data pipeline. The application will also benefit from a continuous expansion of the dataset, as more patient trials become available. This will also lead to the training data being judged and predicted online, and the difficulty of the task being adjusted in real time to optimize the rehabilitation effect of the patient in the future. As discussed throughout the paper, the patient's perception of the difficulty of the training exercise influences their mood, behavior, and performance. As such, matching the patient's perception is an important task in itself, even if the mechanical ground truth may be misrepresented. The desired end result for the rehabilitation robot, including future research, is a real-time online assessment which includes individual patient profiles, which should make patient subjectivity less relevant.

Author Contributions: Mechanical design and prototype debug, H.Y. and M.L.; conceptualization and supervision, H.W., L.V. and V.V.; data analysis and validation H.Y. and Y.L.; the main content of this manuscript was created and written by H.Y. and reviewed by all authors.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dong, Q.; Guo, Q.; Luo, B.; Xu, Y. Expert consensus on post-stroke cognitive management. *Chin. J. Stroke* **2017**, *12*, 519–531.
2. Feigin, V.L.; Forouzanfar, M.H.; Krishnamurthi, R.; Mensah, G.A.; Connor, M.; Bennett, D.A.; Moran, A.E.; Sacco, R.L.; Anderson, L.; Truelsen, T. Global and regional burden of stroke during 1990–2010: Findings from the Global Burden of Disease Study 2010. *Lancet* **2014**, *383*, 245–255. [CrossRef]
3. Xiang, S.D.; Meng, Q.L.; Yu, H.L.; Meng, Q.Y. Research status of compliant exoskeleton rehabilitation manipulator. *Chin. J. Rehabil. Med.* **2018**, *33*, 461–465.
4. Xie, S. Advanced Robotics for Medical Rehabilitation. *Springer Tracts Adv. Robot.* **2016**, *108*, 1–357.
5. Hou, Z.G.; Zhao, X.G.; Cheng, L.; Wang, Q.N.; Wang, W.Q. Recent advances in rehabilitation robots and intelligent assistance systems. *Acta Autom. Sin.* **2016**, *42*, 1765–1779.
6. Losey, D.; McDonald, C.; Battaglia, E.; O"Malley, M.K. A review of intent detection, arbitration, and communication aspects of shared control for physical human-robot interaction. *Appl. Mech. Rev.* **2018**, *70*, 1. [CrossRef]
7. Tao, H.B.; Yu, Y.; Zhang, Q.; Sun, H.Y. A control system based on MCU for wearable power assist legs. In Proceedings of the IEEE International Conference on Robotics & Biomimetics, Karon Beach, Phuket, Thailand, 7–11 December 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 2193–2198.
8. Victor, G.; Svetlana, G.; Bram, V.; Dirk, L.; Carlos, R.G. Multi-Axis force sensor for human-robot interaction sensing in a rehabilitation robotic device. *Sensors* **2017**, *17*, 1294.

9. Hwang, B.; Jeon, D. Estimation of the user's muscular torque for an over-ground gait rehabilitation robot using torque and insole pressure sensors. *Int. J. Control Autom.* **2018**, *16*, 275–283. [CrossRef]

10. Wolf, S.; Grioli, G.; Eiberger, O.; Friedl, W.; Alin, A.S. Variable stiffness actuators: Review on design and components. *IEEE/ASME Trans. Mechatron.* **2016**, *21*, 2418–2430. [CrossRef]

11. Bongsu, K.; Aurelien, R.; Deshpande, A.D. Impedance control based on a position sensor in a rehabilitation robot. In Proceedings of the ASME 2014 Dynamic Systems and Control Conference, San Antonio, TX, USA, 22–24 October 2014; pp. 1–7.

12. Kim, B.; Deshpande, A.D. Controls for the shoulder mechanism of an upper-body exoskeleton for promoting scapulohumeral rhythm. In Proceedings of the IEEE International Conference on Rehabilitation Robotics, Singapore, 11–14 August 2015; IEEE: Piscataway, NJ, USA, 2015.

13. Novak, D.; Riener, R. A survey of sensor fusion methods in wearable robotics. *Robot. Auton. Syst.* **2015**, *73*, 155–170. [CrossRef]

14. Nguyen, T.H.; Chung, W.Y. Detection of driver braking intention using EEG signals during simulated driving. *Sensors* **2019**, *19*, 2863. [CrossRef] [PubMed]

15. Washabaugh, E.P.; Guo, J.; Chang, C.K.; Remy, C.D. A portable passive rehabilitation robot for upper-extremity functional resistance training. *IEEE Trans. Biomed. Eng.* **2019**, *66*, 496–508. [CrossRef] [PubMed]

16. Magdalena, Z.; Malgorzata, S.; Celina, P. Use of the surface electromyography for a quantitative trend validation of estimated muscle forces. *Biocybern. Biomed. Eng.* **2018**, *38*, 243–250.

17. Tang, Z.C.; Shouqian, S.; Sanyuan, Z.; Chen, Y.; Li, C.; Chen, S. A Brain-Machine Interface Based on ERD/ERS for an Upper-Limb Exoskeleton Control. *Sensors* **2016**, *16*, 2050. [CrossRef]

18. Yepes, J.C.; Portela, M.A.; Saldarriaga, Á.J.; Pérez, V.Z.; Betancur, M.J. Myoelectric control algorithm for robot-assisted therapy: A hardware-in-the-loop simulation study. *Biomed. Eng. Online* **2019**, *18*, 3. [CrossRef]

19. Farina, D.; Jiang, N.; Rehbaum, H.; Holobar, A.; Graimann, B.; Dietl, H. The extraction of neural information from the surface emg for the control of upper-limb prostheses: Emerging avenues and challenges. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2014**, *22*, 797–809. [CrossRef]

20. Ison, M.; Vujaklija, I.; Whitsell, B.; Farina, D.; Artemiadis, P. High-density electromyography and motor skill learning for robust long-term control of a 7-dof robot arm. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2016**, *24*, 424–433. [CrossRef]

21. Engemann, D.A.; Gramfort, A. Automated model selection in covariance estimation and spatial whitening of MEG and EEG signals. *NeuroImage* **2015**, *108*, 328–342. [CrossRef]

22. Fougner, A.; Scheme, E.; Chan, A.D.C.; Englehart, K.; Stavdahl, J. Resolving the limb position effect in myoelectric pattern recognition. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2011**, *19*, 644–651. [CrossRef]

23. Peng, L.; Hou, Z.G.; Chen, Y.X.; Wang, W.Q.; Tong, L.N.; Li, P.F. Combined use of sEMG and accelerometer in hand motion classification considering forearm rotation. In Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Osaka, Japan, 3–7 July 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 4227–4230.

24. Lo, A.C.; Guarino, P.D.; Richards, L.G.; Jodie, K.H.; George, F.W.; Daniel, G.F.; Robert, J.R.; Todd, H.W.; Hermano, I.K.; Bruce, T.V.; et al. Robot-assisted therapy for long-term upper-limb impairment after stroke. *N. Engl. J. Med.* **2010**, *362*, 1772–1783. [CrossRef]

25. Reinkensmeyer, D.J.; Wolbrecht, E.; Bobrow, J. A computational model of human-robot load sharing during robotassisted arm movement training after stroke. In Proceedings of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Lyon, France, 22–26 August 2007; IEEE: Piscataway, NJ, USA, 2007; pp. 4019–4023.

26. Qian, Z.Q.; Lv, D.Y.; Lv, Y. Modeling and quantification of impact of psychological factors on rehabilitation of stroke patients. *IEEE J. Biomed. Health Inform.* **2019**, *23*, 683–692. [CrossRef] [PubMed]

27. Yatsenko, D.; Mcdonnall, D.; Guillory, K.S. Simultaneous, proportional, multi-axis prosthesis control using multichannel surface EMG. In Proceedings of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Lyon, France, 22–26 August 2007; IEEE: Piscataway, NJ, USA, 2007; pp. 6133–6136.

28. Lenzi, T.; De Rossi, S.M.M.; Vitiello, N.; Carrozza, M.C. Intention-based EMG control for powered exoskeletons. *IEEE Trans. Biomed. Eng.* **2012**, *59*, 2180–2190. [CrossRef] [PubMed]

29. Duschau-Wicke, A.; Zitzewitz, J.V.; Caprez, A.; Lunenburger, L.; Riener, R. Path control: A method for patient-cooperative robot-aided gait rehabilitation. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2010**, *18*, 38–48. [CrossRef] [PubMed]

30. Krebs, H.I.; Palazzolo, J.J.; Dipietro, L.; Ferraro, M.; Krol, J.; Rannekleiv, K. Rehabilitation robotics: Performance-based progressive robot-assisted therapy. *Auton. Robot.* **2003**, *15*, 7–20. [CrossRef]

31. Cai, L.L.; Fong, A.J.; Liang, Y.Q.; Burdick, J.; Edgerton, V.R. Assist-as-needed training paradigms for robotic rehabilitation of spinal cord injuries. In Proceedings of the 2006 IEEE International Conference on Robotics and Automation, Orlando, FL, USA, 15–19 May 2006; IEEE: Piscataway, NJ, USA, 2006; pp. 3504–3511.

32. Wolbrecht, E.T.; Chan, V.; Reinkensmeyer, D.J.; Bobrow, J.E. Optimizing compliant, model-based robotic assistance to promote neurorehabilitation. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2008**, *16*, 286–297. [CrossRef]

33. Pehlivan, A.U.; Losey, D.P.; Ormalley, M.K. Minimal assist-as-needed controller for upper limb robotic rehabilitation. *IEEE Trans. Robot.* **2016**, *32*, 113–124. [CrossRef]

34. Kleinsmith, A.; Bianchi-Berthouze, N. Affective body expression perception and recognition: A survey. *IEEE Trans. Affect. Comput.* **2013**, *4*, 15–33. [CrossRef]

35. Khosrowabadi, R.; Quek, C.; Ang, K.K.; Wahab, A. ERNN: A biologically inspired feedforward neural network to discriminate emotion from EEG signal. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *25*, 609–620. [CrossRef]

36. Xu, G.; Gao, X.; Pan, L.; Chen, s.; Wang, Q.; Zhu, B. Anxiety detection and training task adaptation in robot-assisted active stroke rehabilitation. *Int. J. Adv. Robot. Syst.* **2018**, 15. [CrossRef]

37. Ozkul, F.; Palaska, Y.; Masazade, E.; Erol-Barkana, D. Exploring dynamic difficulty adjustment mechanism for rehabilitation tasks using physiological measures and subjective ratings. *IET Signal Process.* **2019**, *13*, 378–386. [CrossRef]

38. Fundaro, C.; Maestri, R.; Ferriero, G.; Chimento, P.; Taveggia, G.; Casale, R. Self-selected speed gait training in parkinson's disease: Robot-assisted gait training with virtual reality versus gait training on the ground. *Eur. J. Phys. Rehabil. Med.* **2019**, *55*, 456–462. [CrossRef] [PubMed]

39. Feng, Y.F.; Wang, H.B.; Yan, H.; Wang, X.; Jin, Z.; Vladareanu, L. Research on safety and compliance of a new lower limb rehabilitation robot. *J. Healthc. Eng.* **2017**, *2017*, 1523068. [CrossRef] [PubMed]

40. Feng, Y.; Wang, H.; Vladareanu, L.; Wang, X.C.; Jin, Z.N.; Vladareanu, L. New Motion Intention Acquisition Method of Lower Limb Rehabilitation Robot Based on Static Torque Sensors. *Sensors* **2019**, *19*, 3439. [CrossRef]

41. Wang, H.B.; Feng, Y.F.; Yu, H.N.; Wang, Z.H.; Vladareanuv, V.; Du, Y.X. Mechanical design and trajectory planning of a lower limb rehabilitation robot with a variable workspace. *Int. J. Adv. Robot. Syst.* **2018**, 15. [CrossRef]

42. Suykens, J.A.; Vandewalle, J. Least squares support vector machine classifiers. *Neural Process. Lett.* **1999**, *9*, 293–300. [CrossRef]

43. Zhuang, J.X.; Jiang, H.Y.; Liu, L.L.; Wang, F.F.; Tang, L.; Zhu, Y. Parameters optimization of rice development stage model based on individual advantages genetic algorithm. *Sci. Agric. Sin.* **2013**, *46*, 2220–2231.

44. Gou, X.; Peng, C.; Zhang, S.; Yan, J.; Duan, S.; Wang, L. A novel feature extraction approach using window function capturing and QPSO-SVM for enhancing electronic nose performance. *Sensors* **2015**, *15*, 15198–15217.

45. Zhao, Y.T.; Shan, Z.Y.; Chang, Y.; Chen, Y.; Hao, X.C. Soft sensor modeling for cement fineness based on least squares support vector machine and mutual information. *Chin. J. Sci. Instrum.* **2017**, *38*, 487–496.

A Perceptive Interface for Intelligent Cyber Enterprises

Ioan Dumitrache [1], Simona Iuliana Caramihai [2], Mihnea Alexandru Moisescu [2,*],
Ioan Stefan Sacala [1], Luige Vladareanu [3] and Dragos Repta [2]

[1]   Automatic Control and Systems Engineering Department Bucharest, University Politehnica of Bucharest,
      999032 Bucharest, Romania; ioan.dumitrache@acse.pub.ro (I.D.); ioan.sacala@acse.pub.ro (I.S.S.)
[2]   Automatic Control and Industrial Informatics Department Bucharest, University Politehnica of Bucharest,
      999032 Bucharest, Romania; simona.caramihai@aii.pub.ro (S.I.C.); repta_dragos@yahoo.com (D.R.)
[3]   Institute of Solid Mechanics of the Romanian Academy, 010141 Bucharest, Romania;
      luige.vladareanu@vipro.edu.ro
[*]   Correspondence: mihnea.moisescu@upb.ro

Abstract: Large scale, complex, networked enterprises, as may be considered (trans)national energy systems, multi-national manufacturing enterprises, smart cities a.s.o. are structures that can be characterized as systems of systems (SoS) and, as such, require specific modelling paradigms and control architectures to ensure their successful running. Their main characteristic is the necessity of solving practically one-of-a-kind problems with respect to the external context and internal configuration, thus dealing with dynamically evolving flows of data and information. The paper introduces the concept of intelligent cyber-enterprise, as an integrating paradigm that uses information and knowledge dynamics, in order to model and control SoS, especially focusing on the importance of appropriately adapt external and internal perception of an enterprise through a new generation of sensorial systems—the perceptive interfaces. The authors analyze sensing and perception in relation to intelligent cyber enterprise model and propose an implementation for a perceptive system interface.

Keywords: cyber-physical systems; intelligent cyber enterprise; systems interface

---

1. Introduction

New technological developments have helped solve many scientific and socio-economic problems over the years. Nowadays we can produce goods in every economic sector, and we can transport them in very short time no matter what distance is required all around the globe. This can be done nowadays much faster than ever. We can extract the required data from very large amounts of data, found in a single location or combine from multiple locations through heterogenous interconnected networks. We may use the information provided in a very short period, so that we may easily find a solution and adapt depending on the circumstances.

Availability of resources, personalized products, social enterprise, environmental awareness, globalization of markets, all of these represent important factors in today's enterprises' management and economic development. The problem is, they represent also pressure factors in the decision-making process, considering the following aspects:

- Decision making is often a problem of selecting from several possible solutions, based on available information and optimization (efficiency) criteria.
- As now there are huge amounts of available data, this availability often increases the complexity of the problem, by taking into account too much information (irrelevant for the core of the decision).

- This complexity either may generate such models that their analysis and performance evaluation take too much time with respect to the problem-solving time horizon or may encourage model simplification (including finding false similarities) which results in incorrect models.

Either way, a correct decision making implies both an appropriate problem modelling and a correct selection of relevant information and data to be used (gathered and exchanged) in its solving.

The approach of this paper is thus oriented on:

- Defining a modelling approach for a system of systems structure, oriented on problem-solving (the intelligent cyber-enterprise).
- Defining a generic problem model, in order to support systems dynamics, reconfiguration and adaptation to the environment and context, with respect to information flow (the perception-reasoning-learning model).
- Identification of the role and place of perceptive interfaces in ensuring the appropriate gathering and flow of information (including human in the loop aspect).

Some aspects related to process identification in order to allow appropriate design of perceptive interfaces are included. Information technologies that are used in control systems and knowledge management represent the most important aspects out of all the tools that are available on the market. Also, because the degree of complexity has increased in different engineering domains, new models and paradigms appeared over the years. On the other hand, globalization and sustainable development require a new economic perspective that must include social and environmental impact in order to develop sustainable businesses and production systems [1]. In the present paper the authors analyze sensing and perception in relation to the intelligent cyber enterprise model and propose a new concept: perceptive system interface.

2. Related Work

New emerging concepts, such as internet of things (IoT) and cyber-physical systems (CPS) paradigm represent only a small part of the technological drivers of the next industrial revolution. In this section we address important concepts and paradigms related to the development of intelligent cyber enterprise model.

CPS are related to the study of complex systems as well as systems of systems focusing on physical and virtual components, interactions, process interconnections and information processing and addressing a wide range of temporal and spatial scales [2]. CPS are characterized by the National Science Foundation [3] as "engineered systems that are built from and depend upon the synergy of computational and physical components. Emerging CPS will be coordinated, distributed, and connected, and must be robust and responsive."

Cyber-physical systems must have the capability to be reconfigurable in order to integrate the human factor into system engineering and socio-technical systems. This new trend emphasizes the constant need of interoperability paradigm, seen both from the classical considerations as well as the new trends as we must sense and perceive the smart system taking into consideration all interoperating systems. This new shift can have an important consequence in the near future in the design process as well as the implementation phase. The adoption of cloud-based technologies will have an impact on the design and implementation on the next generation of sensing systems [4].

With the development of the internet of things and future internet paradigms, the number of systems that need to cooperate and interact in the future will increase both in number and complexity. Thus, we must perceive the purpose of every system in order to use the proper methodology in order to verify and validate each functionality. One solution can be the model-based cyber-physical systems [5].

Advances in internet-oriented paradigms have led to the development of Edge/Fog computing model aiming to optimize smart applications to expand their functionalities close to the geographic location of the IoT devices, rather than outsourcing computations to far away datacenters. A proposed architecture includes an edge/fog/cloud monitoring system and a capillary container orchestrator that

is able to handle highly dynamic IoT environments in which the edge node may be overloaded at runtime due to an increase in the workload [6].

Security is an important component of future internet systems. One important aspect in addressing security is trust. Edge, fog and cloud computing have to rely on resources and services under ownership of various entities [7]. A proposed trust management architecture that relies on the use of blockchain-based smart contracts (SCs) and specifically designed trustless smart oracles is presented in [7].

Correlated with these concepts, new enterprise models have emerged. Such models have been related to the factories of the future paradigm. Factories of the future aggregate the concepts and represent the future of an enterprise that is built on collaboration, connectivity at different organizational structures, machineries and human operators, in order to become part of interconnected networks of suppliers, transporters and customers.

Distributed manufacturing systems are an important component of factories of the future and require new scheduling optimization models. A proposed model is presented in [8] and is developed based on a discrete fruit fly optimization algorithm with three heuristic methods proposed to initialize.

Another important component of factories of the future is represented by material flow analysis. Results have been used with success in optimizing material flows and waste streams in production processes [9].

A sensing enterprise represents a new paradigm based on the digital innovation concept that deals with the adoption of sensing and future internet technologies. Sensing enterprises are mostly connected to virtual enterprises and their networks. Seen from the perspective of a smart system in general, sensing enterprises must have the capability to sense, model and interpret signals from the real world and thus to be able to adapt into a more agile configuration of the system [10]. In relation to such systems a product service systems conceptual framework is proposed in [11] in order to facilitate the development of interoperable product systems and service systems in accordance with the stakeholders needs.

Another important model associated with factories of the future is the cognitive manufacturing. Cognitive manufacturing is a proposed manufacturing organization method, based on perception and cognition that integrates IoT principles, Artificial intelligence and data analytics technologies. One of the main objectives of cognitive manufacturing implementations is to integrate process as well as enterprise wide data and information, as to achieve improvements in the use of equipment, processes reengineering based on decision-making models and data analytics models, enterprise integration of knowledge management models [12].

Cognitive manufacturing systems address the following:

- Improvement of product lifecycle management by providing an integrating enterprise and environment focused approach;
- Adaptive systems integration;
- Analyzing manufacturing data obtained from sensors: production management systems are generating huge amounts of manufacturing data;
- Using decision making models in correlation with business intelligence systems;
- Linking the overall decision-making procedures of the enterprise with knowledge management systems.

3. Intelligent Cyber Enterprise

In this section, the authors briefly describe the concept of intelligent cyber-enterprise (ICE), [13] seen as a socio-technical complex system. In relation to the implementation enterprise wide CPS systems the concept of ICE was proposed.

The connection between physical processes and their cybernetic representation is a very important aspect in order to modeling and design exactly the behavior of a process and thus being capable to integrate hardware-in-the-loop and human-in-the-loop components [5].

A cyber-enterprise represents an aggregation of physical objects, knowledge represented through process models (workflows), control algorithms, humans and information represented by adequate software tools and communication processes [14]. Thus, within an ICE, all these components must be capable to interact and cooperate in order to achieve intelligent enterprise goals, in terms of production, costs, time bound and resources limitations in order to satisfy all clients, suppliers and economic partners in a sustainable environment. Such a proper interaction must take into consideration the level of required intelligence, the amount of required data and how to extract information from heterogeneous flows within the enterprise, as well as by the emergent intelligent behavior of the enterprise in order to become adaptive, reactive or proactive, depending on the economic constraints and demands [15].

The ICE must be modeled in order to be capable to solve any specific problem, being capable to divide any problem at the lowest level of complexity. On the other hand, problems that must be solved are goals existing at the operational level that must include the minimum amount of enterprise resources.

From the CPS perspective, in order to achieve these results, we must include at least a control loop and at least one enterprise resource. The more complex a problem becomes there is a demand to include multiple control loops and specific resources. Control loops must operate in real time and must be based on targeted algorithms and rules, using well defined sets of data. The more complex a problem becomes there is a necessity to integrate heterogeneous information structures and heuristics. An intelligence-based model for the ICE is represented in Figure 1.
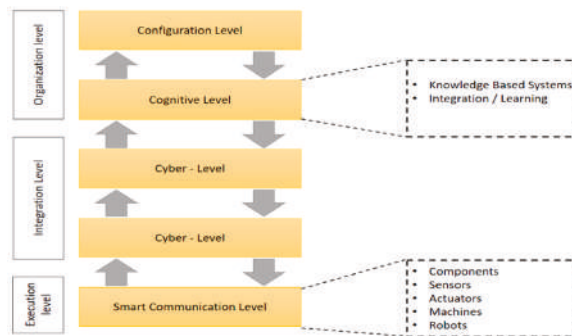
Figure 1. CPS-based enterprise model. Adapted from [16].

Every component of the ICE architecture must be capable to be independent according to different pre-defined specifications and must have the capacity to communicate, allowing the transfer of meta-data, information and knowledge in order to provide reliable context-oriented behavior [17].

Another aspect that must be taken into consideration is to connect subsystems in clusters, depending on the initial functional requirements. For examples, machines are a part of manufacturing cells that communicate based on protocols, standards and rules, all of them being embedded in control algorithms. Thus, functionalities can be added afterwards depending on the required specifications, initial or during the process implementation. Manufacturing cells are part of a manufacturing system, and thus are capable to fulfill any given problem and to develop various products [3]. The main characteristics of ICE include:

- *Perception*: this characteristic has the purpose to develop and integrate results from measurement science for sensing and perception. The role of perception is to better understand the system's complexity and thus to reduce the risks related to the adoption of new technologies.

- *Mobility of Systems*: this characteristic will develop specific test methods in order to determine the performance of each intelligent system.
- *Human-Machine Interaction*: this characteristic has the purpose of delivering specific test methods, protocols and information models in order to facilitate a more effective human-machine collaboration.
- *Agility of Industrial Systems*: this characteristic has the purpose of delivering agility performance metrics, data sets and information models in order to enable manufacturers to adapt and reconfigure system components.
- *Embodied AI and Data Generation for Manufacturing*: this characteristic has the purpose to deliver structural artificial intelligence and machine learning models and tools in order to improve the performance and autonomy of manufacturing applications.
- *Collaboration*: this characteristic has the purpose to deliver specific models in order to facilitate calibration, coordination, in order to mitigate the lack of advanced automation.
- *Safety Performance*: this characteristic will provide performance metrics, specific test methods and measurement tools in order to support the development of next-generation systems that will have the capability to integrate human behavior and to enable tactual-based safe human collaboration and manufacturing tasks.

Intelligent cyber enterprise systems design is focused on adopting new technologies in order to become agile, safe and productive, being also capable to interoperate with smart manufacturing applications and to better evolve.

3.1. Sensing in Correlation to the Enterprise Environment

In order to easily reconfigure, taking into consideration all existing constraints, a sensing system need different resources and conditions in order to be constantly under control. All the functions of a sensing system can be seen as a CPS. These systems can easily make the switch between physical world and thus merge to virtual world. By the help of sensor networks and actuating systems, the sensing system can monitor all the physical processes, collect data from all devices connected, interpret and expose data in different formats in order to make it widely available for every interested stakeholder and software applications to use it in order to better understand the real world.

The artificial intelligence concept can be seen from two different perspectives: both from the traditional approach but more important, from the machine learning perspective. Thus, a smart system will need to be capable of integrating generating mechanisms (physical and virtual) in order to extract the exact data that is needed. This might mean that simulation tools must be used in order to better under the behavior of the system and thus know how to instantly react based on its behaviors. Or, it might mean that data can be extracted from different web resources or multiple physical systems.

In order to develop a measurement mechanism in order to better understand a smart system, and thus sense and perceive the system performances in order to reduce the risk related to the adoption of new technologies, there is a need to integrate sensors and algorithms in order to create productive manufacturing environments.

If in the past we could identify manufacturing systems formed from simple sensors, actuators and controllers, now, there is an urgent need to integrate complex sensor networks in order to use them for the system perception. This is a complex task that makes it very difficult to implement it for academia, system integrators and even manufacturers in order to identify the right solution. The technical idea is to identify existing and emerging sensing and perception technologies and related software in order to understand the product and research solutions landscape.

Due to specific constraints related to flexibility and reusability, smart systems play a very important role in strengthening worldwide manufacturing competitiveness by stimulating responsiveness and innovation. To achieve this goal, smart systems must be capable to perceive and adapt depending on

the need of collaboration with other smart systems or humans. Thus, they must have the capability to learn, adapt and quickly integrate into the rest of the enterprise.

3.2. Perception and Cognition in Correlation to Enterprise Environment

The design of enterprise systems based on the ICE model includes the integration of perception, reasoning, learning and cognition models. These aspects become relevant in the development of perceptive system interfaces for the ICE. An important aspect of ICE is related to extending the enterprise sensing function to a perception function. The perception-reasoning-learning loop (PRL) is further addressed, from a functional point of view and with a special emphasis on the awareness concept.

The ICE functioning is based on problem solving. Usually, such problems are described in terms of system-specific production/functioning goals, with different levels of precision, from strategical (profit versus investments, sustainability, market coverage, utilities availability, durations for problem-solving, quality of processes and products, safety, etc.) to more resource oriented, as number of products/services delivered, cost/profit per product, a.s.o.

Every problem is solved by a sequence of activities, performed by specified resources, at given times (eventually triggered by specific events)—which will denominate as workflows, and which are pieces of knowledge.

Resource oriented problems are solved by deterministic, predictable, well-defined workflows; operational and strategic problems, implying collaboration between resources, are solved by ad-hoc combination of workflows, with respect to availability of resources and context. Such problems may be, usually, solved in different ways, implying very different resources, approaches (interconnections of resources), costs and time. Solving approaches are determined mainly by resource availability, but also by the information availability, interpretation and possibility of gathering (sensing/perception). Also, networking is determined by the possibility of information interoperability between resources.

Appropriate interfaces for information transmission and appropriate perception of external information become crucial for such problem-solving approach. As designing specific interfaces for any possible problem is impossible, it is important to focus on the interface adaptability in order to allow the maximum flexibility for problem solving.

It results that an appropriate complex problem model should underline:

- System goals—allowing the definition of specific parameters to be measured, transferred, contextualized and used as information for solution evaluation.
- Available system resources, for the time horizon of problem-solving—every resource allowing specification of functionality (workflows available—as knowledge stakeholders) input/output information and triggering events.
- Problem context—external information available and, respectively, necessary.

Such a model allows, once the appropriate solution is found, to design appropriate perceptive interfaces either between interconnected systems (including human operators) or with the environment.

Perception in an ICE environment may be referred as the active process that facilitates the interpretation of the environment by integrating information and "stimuli" from the sensory system. Two processing mechanisms can be relevant in the context of an enterprise environment:

- "bottom-up" or non-aware consisting mainly of passively listening to all sensorial channels);
- "top-down" or aware consisting mainly of anticipating and selecting certain stimuli, also referred as perception control).

Relevant aspects of human brain functions and processes that can be modeled in enterprise processes and systems, include:

- Forecasting of information that is expected in a certain situation;
- Focusing on specific information (neglecting other information);

- Interpolating data with respect to an existing pattern, when real information is presenting gaps (the phenomenon of "we see what we expect to see");
- Structuring large amounts of (sometimes possible irrelevant) information, in order to select, during the process of reasoning, the relevant one;
- Hierarchically organization of recurrent interconnected networks each providing specific functions;
- Connectivity patterns distributed among functionally specialized processes based on input-output connectivity specifications and local architecture;
- Dynamics of interfaces between functionally specialized processes;
- mechanisms allowing processing modules to incorporate adaptive changes enhancing processing system parameterization with respect to the adaptability factors of the network;
- Dynamically reconfigurable models which are incorporating functional clusters;
- Adaptive agents which connect depending on context and on the complexity of the goal;
- Interconnection process of several structural and functional modules in parallel-distributed configurations that are generating emergent behavior.

Three phases of perception can be identified, in association to the following gestalt principle: every person has a role in one's perception process, designating a three-stage sequence:

1. Hypothesis related to perception. This will guide the selection, organization and interpretation of the stimuli.
2. Acquisition of information from the sensing system.
3. Comparison of the hypothesis with the acquired sensory information.

Reasoning is considered the (brain) capability of solving problems. The main components of the reasoning process in an ICE environment are: the problem identification, the problem categorization and the identification of the solution.

The solution identification process of a specific enterprise problem can be associated with a reasoning mechanism. Action based on the proposed solution generates feedback that can be measured and interpreted in order to generate a new perception/reasoning cycle. An generic perception/reasoning cycle that can be integrated in problem solving mechanisms of enterprise systems can include the following:

- The categorization phase is the one that determines how the following phases are performed:
- Success criteria;
- Selection (of the solving patterns—if any);
- Filtering;
- Fusion (on relevant input information);
- Planning, estimation, validation: an internal loop whose execution depends on the category of the identified problem, but whose goal is to advance towards the solution by a stepwise decomposition of the problem;
- Feedback;
- Evaluation of success (achieving the desired goal or estimation of the distance towards it).

Learning in an ICE environment can be associated to saving and retrieving a solved problem using a knowledge model. A deep learning process can also be associated to the enterprise system environment. In this process knowledge, used and validated for a given number of times may be retrieved "reflexively" i.e., without a volitional act. Rule mining and self-learning are being used with success in various industrial cases. One such a case is related to the development of a mixed-integer mathematical model to represent the direct energy consumption of machines and indirect energy consumption on a shop floor [18].

3.3. Intelligent System Interfaces

An interface has various interpretations across technical literature: a connection, a composition of connections, a boundary, a linkage, an interaction between functional entities, a logical relation, a specification [19].

ISO/IEC standards (ISO/IEC 2382-1/1993) define interfaces as: "A shared boundary between two functional units, defined by various characteristics pertaining to the functions, physical signal exchanges, and other characteristics." [20].

System Interface design can be addressed taking into consideration the following aspects:

- Characteristics of the system for which the interface is developed.
- Characteristics of the connected systems or of the environment.
- Interaction type. The type of interface can be related to different classifications: simple or complex, human- machine (H2M) or machine-machine (M2M), physical or logical, hardware, software or hybrid, internal or external.
- Interface structure.
- Interface behavior is related to behavioral patterns designed for the interface. Interface behavior describes how interfaces are used. This can be related to data handling, protocol, connection initiation procedure, synchronization, data-driven or event-driven.
- Interface monitoring.
- Interface configuration.
- Interface error handling. Some aspects relevant for error handling include data integrity, confirmation of transmission, error correction and error recovery.
- System security in relation to the interface.
- Interface control document.

When addressing the necessity of developing intelligent system interfaces, the following characteristics are relevant:

Adaptation. The adaptation of a system to changes in the environment or in an interconnected system needs to be addressed as both a change in the system itself but also as a change in its interfaces. In this context adaptation can become a function of the interface.

Learning. Machine learning techniques can be used in association with interface components. Patterns in data exchanged through the interface can emerge as a result of machine learning techniques applied at interface level. Such patterns can be used to optimize the functions of an interface. An example can be related to the amount of time the interface is used.

Prediction. Prediction is not only associated with intelligent systems but can become a function of the interface. Components fulfilling this functional role should be able to generate predictive models related to the environment and/or behavior of the interface.

Optimization. An interface can be extended with components that can monitor the function and behavior. Aggregated data from such components can be used in interface optimization processes.

When developing intelligent interfaces, the following aspects need to be addressed:

- Interface structure in accordance with system structure and behavior.
- Interface behavior in relation to the input and output data.
- Interface functionality in correlation with inputs and in accordance with the needs of the system.

4. Design of Systems Interfaces for the Intelligent Cyber Enterprise

In the current section the authors propose a model for a generic perceptive systems interface as a part of an ICE architecture. In order to implement the perception function two facilitator components are proposed: a process/behavior identification component and a semantic routing component. The semantic routing component facilitates the transfer of data and information between

enterprise system components. The process identification component discovers behaviors associated to the function and current use of the interface (how the interface is used) in accordance with patterns identified in the data and event streams transferred through the interface. As the interface connects system components, the process identification system analyses data in order to determine a workflow or a process associated with data passing through the interface.

4.1. Intelligent Cyber Enterprise Perceptive Systems Interface

The main functional aspects of a perceptive interface are showcased in the context of an ICE system model, described in this section. The proposed interface relies on components that enable its adaptation through the discovery of behavioral models based on observations of transferred information and then by selecting the appropriate behavior from a behavior repository.

The ICE architecture enabling the usage of perceptive system interfaces is depicted in Figure 2 and has the following components:

Integration Layer—this layer is responsible to integrate data from all levels of the system and to provide the basis for the design and implementation of the cyber intelligent systems. This layer is also responsible with the integration of all nodes and layers intro a homogeneous structure that hosts the intelligent entity instances needed to manage this structure. In order to implement the integration layer, the system must have specific components in order to fulfill different roles:

Nodes—represent computation units that are responsible with hosting the intelligent entity instances. The entire lifecycle of a node is managed by the "lifecycle manager". Each node must expose its own semantic interface which appears in the system as an Intelligent Entity. Through this interface, every node can receive different commands in order to check the state of every instance or to generate new events.

Domain Knowledge Repository—this repository is responsible with the interpretation of the system's domain in an easy to understand machine format. The elements of the domain must be uniquely addressable so that metadata can easily target the domain's concepts. Also, the relation between all components of a system domain must be clearly stated for the domain knowledge repository to provide reasoning capabilities and thus, to be capable to determine the context in which the needed information is processed.

Semantic Service Repository—this repository is responsible with the storage of the semantic description of all the services that are exposed by the intelligent entities instances. To achieve this, each service must contain metadata that refers to the system's knowledge base. Thus, these services can be described based on existing standards (e.g., OWL-s, SAWDL, etc.).

Semantic Event Subscription Manager/Event Routing—this layer is responsible with the subscription of all the system's components to the semantic events that are generated by the intelligent entities. This layer must select the exact events based on specific contexts in order to generate an event generation rule such as subscription to the events generated by sensors. The information regarding subscriptions must be sent to the "semantic event router" which is responsible to the transfer of information between the event generator and the subscriber.

Intelligent Interface Instance Registry—this registry is responsible with the storage of all the information that is sent through the entire system and that is deployed in the intelligent instances, thus, providing a shortcut to the services that are exposed by the intelligent entities.

Intelligent Interface Repository—this repository is responsible with the management of each process creation and deployment or un-deployment of the intelligent interface. On the other hand, this repository is responsible with the management of every behavior of the Intelligent Interface.

The perceptive system interface is deployed at the boundary of each intelligent entity instance and requires the following components: semantic interface, enterprise system adapters, physical adapters, lifecycle manager and behavior selection/execution component.

The proposed interface (Figure 3) will allow the connection of both virtual and physical resources of an enterprise system. The use of semantic data models facilitates service and event-based interactions

between the system's components. The components of the interface: service interface, event sinks and event sources include a semantic description and the information transferred between the systems is semantically enhanced.
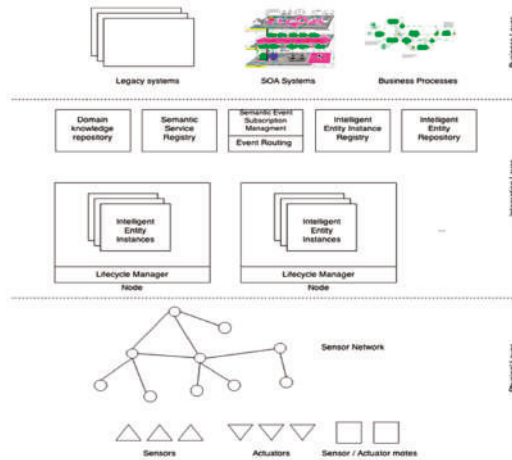

Figure 2. Generic ICE system interface and system model.
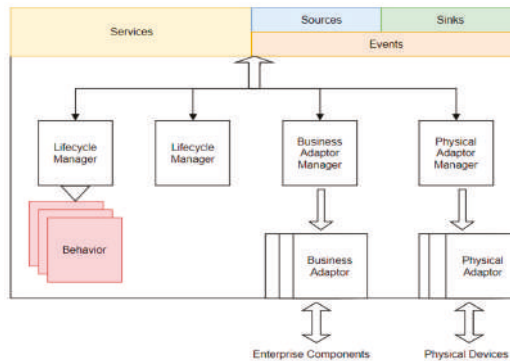
Figure 3. Block diagram of an intelligent entity instance.

Enterprise system adapters are components that facilitate the connection to services, business process execution engines or service orchestration engines. Behavior selection will allow the adaptation in accordance to the process requirements. The semantic augmentation of data will facilitate data interoperability between system components:

Physical adapters—adapters for the physical devices. These adapters are designed for data acquisition and integrate sensors, sensor networks and sensing systems. These adapters can also connect actuators, actuator networks and actuating systems providing input form control systems.

Behavior Selection/Execution System. Behaviors are associated with execution engines. The selection system will facilitate the selection of the appropriate behavior, from the behavior repository, in accordance with predefined criteria.

Lifecycle manager addresses the following operations: initializing, maintenance, management of the components of the current instance and selecting behaviors or adapters. Using a semantic interface, the lifecycle manager can expose a set of services including deploy components of the instance, routing to the appropriate event source.

4.2. Semantic Routing System

The ICE interfaces need to be associated with a scalable, semantically enhanced middleware solution that facilitates the integration of required resources in a distributed system. The middleware solution is characterized by the following elements [21]:

- Integration of semantics—this middleware solution manages the system's ontology that is used to describe each component of the system and their interactions. This ontology will furthermore help in accessing resources and to the creation of the applications that are built on top of the middleware in order to provide information regarding the state of the system;
- The solution must support communication at different levels within a distributed system (e.g., service-based or event-based);
- The solution must contain all the features that are implemented in the system and thus, being capable to enable the development of robust and scalable systems.

On the other hand, the systems that are developed based on the middleware solution must have a homogeneous distributed nature, so that, within each node of the system, the middleware can have the same components and will execute the same operations. Each node of the system is determined by specific applications that, together with the distributed nature of each system built using this middleware solution, represent the basis for the creation and implementation of the ICE.

Thus, the middleware solution components are further detailed:

Components that manage the resources of each local node include: node manager and node application manager.

Components that manage the semantic layer include: semantic manager, semantic service executor, automatic event manager, distributed RDF store and the P2P overlay manager.

The P2P overlay manager supervises the P2P overlay network containing the nodes of the system. Scalable distributed systems can be developed using structured P2P overlays based on distributed hash tables (DHT). P2P overlay stores key-value pairs in the nodes of the system. The system facilitates nodes to become part and leave the network without interfering with the system's operation. A uniform distribution of data across the network is desired in order not to generate congestions. The proposed component uses overlay to manage RDF triples and to transfer messages containing results of the query evaluation.

The semantic manager component facilitates the system components access to the system's ontology. The ontology referring functions of the semantic manager is implemented using Apache Jena. The semantic API exposes functions that allow easier creation of individuals from the classes of the ontology including application, node, event. The semantic service executor component allows system components to invoke remote semantic services.

The automatic event manager facilitates the event-based connection of the system components. New events will be accepted only if they are in accordance with the predefined requirements. The operation of the declarative event management system is associated with the automatic event manager component. The semantic manager component handles system's ontology new definitions of EventSink individuals. The component identifies the corresponding event sources by using semantic queries containing properties related to the event sink descriptors.

Therefore, the proposed middleware solution must have the capability to manage all the processes of most of the applications that are deployed in the "managed applications container" managed by the node application manager. Analyzing these applications, we have identified various operations that must be exposed, bundled into the following APIs:

- Semantic API—they represent basic operations of the system's ontology.
- Event/Quert API—they are responsible with the execution of SPARQL queries with RDFS reasoning of the distributed semantic store.
- Service API—they enable each application to call remote semantic services.

In order to achieve the best results, the middleware solution must contain a set of predefined semantic definitions that are responsible with providing a consistent implementation of communication and interoperability at different levels and thus, to support various features (e.g., remote application management) [22,23]. On the other hand, on top of these semantic definitions, each system must define its own semantic class.

4.3. Process Identification

The focus of this section is to detail the application of process discovery techniques on data streams in order to generate behaviors for the adaptive interface. Although most of the steps required to generate the relevant sequences of events are specific to each implementation, for some of the common ones, domain-independent solutions are investigated [12,24]:

Event identification—using the data transferred through the interface, which might be in continuous form, to create sequences of events relevant to the application domain. This is a domain-specific problem as the rules for extracting events from observations are dependent on the layout of the monitored environment. Internet-of-Things oriented systems can be used in relation to processing large streams of data coming from heterogeneous devices, such as employing semantically-rich representations.

Activity recognition—transforming sequences of "low-level" events into sequences of activities that will be represented in the resulting process model; this task is especially important given the granularity of the input data streams considered here. Solving this problem requires domain-dependent knowledge in the form of "activity models".

Process instance or process case identification—this task involves associating each collected event or detected activity to one of more process instances; it should be noted that, depending on the application, this pre-processing step can be applied before or after the activity recognition one. The solutions are mostly limited to domain-specific rules, for example, based on the resources (and/or agents) involved in each process instance.

A process mining system implementation can be developed around the previously described tasks of event recognition/collection, process instance identification, action recognition and process model discovery, following an informational flow similar to that depicted in Figure 4.
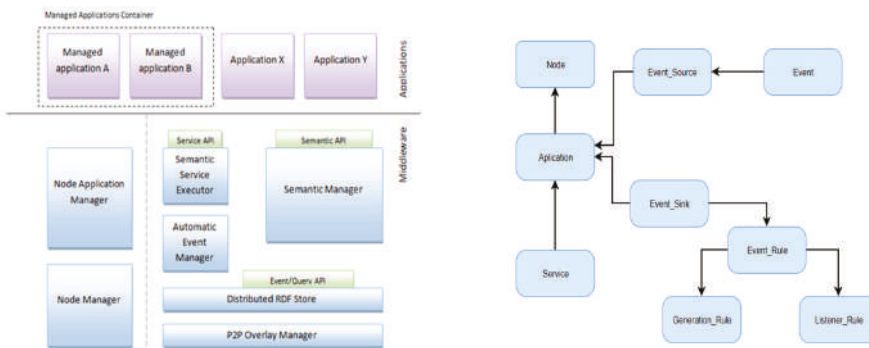
Figure 4. Semantic routing system and middleware ontology.

Taking into consideration several design concerns such as portability—making the system capable of handling several related application domain—and the minimization of the effort required the experts in providing the necessary system KB, a system architecture similar to the one presented in Figure 5 can be implemented—a set of major functional blocks, each corresponding to one of the previously mentioned processing tasks loosely arranged around a central data repository containing everything from raw collected to events to recognized activity sequences and process instances.

Figure 5. Process identification system model.

The required input from domain experts, for each processing tasks, is:

- Event identification—rules for mapping streams of observations to event sequences;
- Activity recognition—depending on the complexity of the monitored environment and the event sources, ranging from simple translation rules (for events generated by "virtual sensors" is fetched from the logs of existing workflow-management systems) to state-space action models;
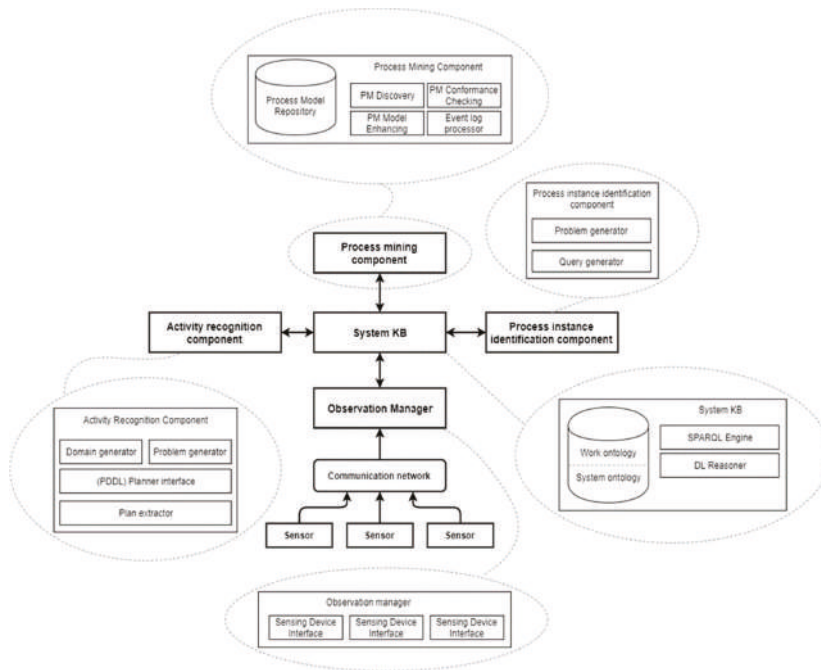- Process instance identification—the "observation manager" component handles the data acquisition and optionally the process of "semantically lifting" the observations based on a set of domain-specific transformation rules. The other components of the system each deal with one of the three information processing tasks, with the "process mining component" relying on the results generated by the other two. The "process instance identification component" can be implemented using either domain specific rules (such as a combination of keys that can uniquely identify a process case) or around a generic "event correlation" detection algorithm. The translation from the "event" to the "activity" abstraction level is performed by the "activity recognition" component. This processing step can be performed either before or after running the instance identification procedure. As mentioned in the previous section, the "high-level" activities that result in the sequences of observed events can be inferred using a schema that generates a pair consisting in a planning domain and problem, solvable using an off-the-shelf external planner.

The result of solving each planning problem will be an activity trace, from which the causal and independence relations between the activity instances can be derived. This information is subsequently used to build a workflow net. Unlike many process discovery methods that use an "event log" as input, newly proposed approaches based on Petri Net "unfoldings" accept (labelled) partial ordered sequences of events and as such offer better results from a smaller sample size in the case of processes with high concurrency (of course, having the downside that this information must be provided by an expert, or derived from another source).

5. Implementation of the Intelligent Cyber Enterprise System Interfaces and Event Routing System

5.1. Implementation of the Event Routing System

The proposed implementation involves the routing of data to a specific systems interface. The implementation of an event routing system facilitates the integration of new data sources such as sensors or system components without the reconfiguration of the communication network [25]. The data is routed to the appropriate source based on the semantic description. In order to perform the routing process, the following sequence of operations is proposed:

1. An application connected to a node generates a new instance of class EventSink though the semantic API.
2. The semantic RDF store is updated and the component notifies the automatic event manager (AEM) component.
3. The AEM receives the description of the sink and extracts the ListenerRule and the GenerationRule.
4. A list of event sources is obtained by the AEM calling the methods defined in Query API using the SPARQL query defined in the ListenerRule component. The AEM registers the query in as to obtain notifications when sources are available.
5. The process associated to the event routing is configured by the AEM based on the event sources register.

- The association event-node is identified based on the system ontology association of nodes with EventSource class individuals.
- Every identified node is associated by the AEM with sematic service exposes by the AEM component. The function of the semantic service is to manage the event generation process related to the requirements of the application associated to the identified EventSource. The description of the GenerationRule is correlated with the details of the new EventSink.
- The AEM creates events based on the predefined requirements of the application.

The implementation of the event routing system is based on a P2P overlay network that allows dynamic configuration of new nodes. Identified nodes are integrated in the system without interfering with the systems operation. Operations of the event routing process are distributed. The operations involved in the process of adding the node to the system are further described:

- The Node Manager component of node initializes the local components and the node manager defines the EventSources and semantic services.
- The AEM selects a sink for the event source form the applications that imposes a value change event generation. The node manager component sends the first event signaling a node start.
- Deployment events related to the new application are generated, events that can be used by the SI to update a detailed view of the system.
- The deployed application configures an EventSource. This registration will trigger a notification of the AEM component, that monitors the creation of new EventSources that correspond with the context requirements of the EventSink.
- The ARM receives the notification and configures a "value-change" event generation strategy for the registered EventSource.

5.2. Process Identification Based on Interface Traffic

By addressing the enterprise as a CPS, different enterprise systems can be interconnected and data collected form sensors can be aggregated. An industrial scenario was chosen involving the following event sources: position detection devices, RFID readers and contact sensors. The scenario involves human activities and the use of sensors that are present in an industrial environment but used for

different purposes and by different industrial systems. The scenario is related to the movement of parts in an industrial environment. The part is transported in a container, by a worker operating a vehicle. The contact sensor detects the opening of a gate. The RFID reader identifies the tags on the vehicle and the container. The vehicle travels through three areas: AREA 1 to AREA3 and a position detection device, detects the vehicle in the specific area.

The first step in extracting the process model involves the event identification tasks. Through the application of rules such as "An observation collected from RFID reader X of a tag Y implies that Y is currently in AREA 1", an event sequence such as the one in Table 1 can be interpreted in relation to a specific component.

Table 1. Case study observations from sensors.

Timestamp.	Device	Previous Value	Current Value
ts1	Contact_sensor_A	Closed	Open
ts2	RFID_Reader_1	-	5023f674ae4d2e
ts2	RFID_Reader_1	-	2537f674ae4d3b
ts4	Contact_sensor_B	Closed	Open
ts5	Positon_1	Area_1	Area_2
ts6	Position_1	Area_2	Area_3
ts7	Contact_2	Closed	Open
ts8	RFID_Reader_2	-	2537f674ae4d3b
ts9	RFID_Reader_2	-	5023f674ae4d2e

To showcase a possible solution for the activity recognition tasks using the state-based action model library mentioned previously, the movement of vehicles can be considered. The action models are designed in terms of the system's state space. Each action definition will specify the action's preconditions and effects as expressions based on state variables. As such, the action models will not directly refer event types, leading to the separation of the knowledge engineering tasks required in order to deploy the observation manager and activity recognition components.

As mentioned previously, the action recognition problem can be converted into a planning problem. This approach allows the reuse of existing tools developed in the field of automated planning such as PDDL editors and planners with various capabilities. For this approach, the development of a "template" planning domain and problem definitions is required. The template contains a type/object hierarchy, a set of predicates (for describing the state) and a set of action definitions.

Listing 1 details an action definition for "Transport". Procedures can be developed to infer the type hierarchy of the planning domain from the system's Knowledge Base.

Listing 1. Action definition for "Transport"

```
(:durative-action transport
:parameters (?v—Vehicle ?l1—Location ?l2—Location)
:duration (= ?duration 22)
:condition
(at start (at ?v ?l1) (next_to ?l1 ?l2))
:effect
(at end
(and
(at ?v ?l2)
(not (at ?v ?l1))
) ) )
```

A set of PDDL domain / problem files is created, based on the template defined for each instance of the action recognition problem. Using the information resulting from the event detection phase, the "init", "goals" and "actions" will be augmented (Listing 2).

Listing 2. Augmentation of "init", "goals" and "actions"

```
(:init
 …
at 5023f674ae4d2e (event3_pred)
at 5023f674ae4d2e (not (event3_pred))
)
(:goal and(
 …
(event3_constr_satisf)
))
(:action event3_constr
:precondition (and (event3_pred) (at_loc vehicle1 Area3))
:effect (and (event3_constr_satisf) (increase (total-cost) 2))
)
```

The schema used to constrain the trajectory, relies on the usage of timed initial literals, available since PDDL 2.2. In the proposed case, a timestamp and a "window" predicate is be enabled and the predicate is be added as a goal. This predicate is related to a new action, whose preconditions are related to an event. A conjunction of predicates is describing the change in the monitored environment's state represented by the event, which in this case corresponds to "vehicle1"'s arriving in "Area2". Similar constructs will be added for all the available events.

Solving the problem with these actions will allow an existing planner to instantiate planning operators (actions) capable of "explaining" (at a higher level of abstraction) the trajectory in the monitored environment's state space, described by the collected events. For the proposed scenario (Table 1), in the case of events at t6 and t7, satisfying the goal requires predicates "event6_constr_satisf" and "event7_constr_satisf" to be true. These can only be set by adding instances of actions "event6_constr" and "event7_constr" to the plan. However, in order to satisfy the preconditions of "event7_constr", a "transport" action instance for the areas "Zone2" and "Zone3" must be added before it.

Several post-processing steps must be applied on the resulting plan. Such steps include the removal of the special actions used to create the trajectory constraints and the detection of the concurrency relations between activities extracted.

The process instance identification step can be performed before or after the action recognition phase, based on the initial sequence of events or the results of the activity recognition stage (partially-ordered sequences of activities). This can be accomplished using either domain-specific rules or one of the semi-automated, iterative approaches mentioned in the previous section.

The results obtained after the action recognition and instance identification steps can be used to compile an event log, referring high-level activities instead of low-level events. The activities can be subsequently used as the input for the process model discovery step. Given the availability of additional information regarding the concurrent execution of some of the activities (derived from the set of action models used in the recognition step), the study focused on methods capable of using this knowledge. As such, a method based on building event structures followed by folding them into Petri Nets, was used to generate a process model. An example of a process model is described in Figure 6. Some aspects of the process model such as "AND" and "XOR" blocks need to be further analyzed and modeled with the aid of a higher level Petri Net formalism. The system implementation was used to validate the proposed approach in a case study built around a simulated logistic process.
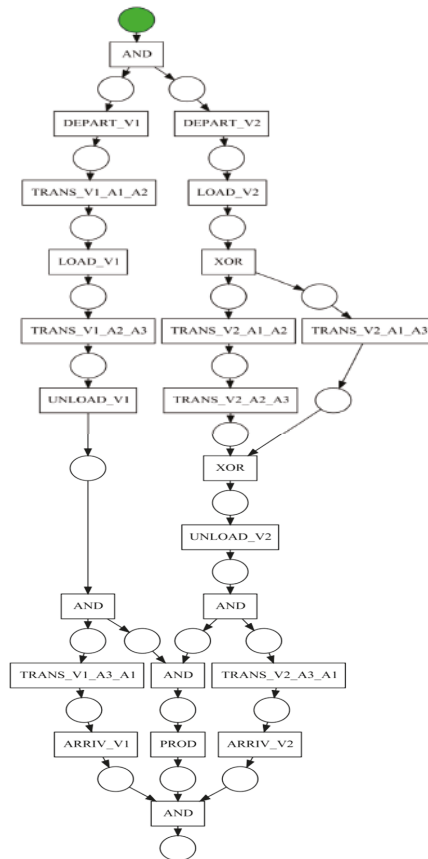
Figure 6. Process model of the proposed scenario.

6. Discussion

In this article, we have explored the possibility of developing a process identification model aimed towards identification of processes associated to data transferred through a system interface. Such processes usually occur in an enterprise environment and are beyond the scope of established process management systems [26]. The identified workflows or processes are associated to the interface as interface behaviors.

The proposed model enhances the capability of a system interfaces by attaching a perception function that facilitates the interpretation of the working environment and connected systems by integrating information and "stimuli" from the sensory system [27].

Advantages of the proposed system include:

- Automated analysis of data and information transferred through a system interface;
- Pattern recognition in data and information transferred through a system interface;
- Automated identification of behaviors associated to an interface;
- Facilitation of interface redesign in accordance with the data and information transferred between systems;
- Interface operation optimization in accordance with the data and information transferred between systems;

- Enhancing system adaptation capabilities by adding the adaptation capability to the system interface.

This represents a novel approach that can provide benefits in enterprise environments adopting IoT and CPS technologies, in which a large number of events collected from sensors or system components are transferred through system interfaces [28]. The resulting interface behavior model offer meaningful insights into the behavior of the system and allows for the use of novel process mining techniques for conformance checking.

In order to address the entire complexity of the considered domain, ontologies based on DL were used. Using this method, the tasks associated to processes can be associated to reasoning problems. The proposed model demonstrates how data acquired form sensors or other system components can be analyzed and used in determining interface behaviors.

Two main categories of activity recognition problems have been analyzed in regard to the proposed: model learning based and specification base. Due to user requirements level, portability and flexibility issues only specification-based methods have been further addressed. The authors followed the main lines of research into applying and extending existing process workflow management techniques in CPS and particularly in ICE environment. Due to the benefits for industrial automation systems, the authors focused on the applications of the developed techniques in process mining in a CPS environment.

Knowledge engineering tasks are still required to develop the planning domain used in the activity recognition step and the rules that enable the identification of individual instance traces in the stream of events. An important factor for the adoption of the proposed method are the events that relate directly to state changes in the observed system. Usually process discovery techniques are used in correlation to logs of various enterprise systems.

Although the discovered process model has the same structure as the process used to generate the dataset, some performance and scalability issues related to the activity recognition step were discovered. Planning time is a relevant factor associated to the proposed method and is directly related to the increase in the system resource consumption. An "object-based" decomposition method, can be implemented in order to increase the efficiency in solving the simple planning problems and identifying "macro-operators" from existing cases. These aspects can correlate to the original domain and lead to improvements in planning time, by creating "short-cuts" in the state space and transferring them to automated planners [13,18]. Such an approach allows the augmentation of enterprise systems and their components with system interfaces capable of analyzing patterns in the transferred data and associating the patterns to workflows or processes. This approach couples the systems interface to the process associated with the actual system [12].

The event routing system facilitates the implementation of the process identification model and routes relevant data to the appropriate system interface.

The advantage proposed by the process identification model is the ability to generate process models in the absence of a pre-modeled process or to offer a basis for comparison of the existing pre-modeled process to the actual execution. The proposed approach utilizes the data already handled by the interface augmenting the function of the interface and enabling the future autonomous adaptation to process chances [12].

Further research is conducted in the area of autonomous adaptation models for systems interfaces that will allow an automatic reconfiguration of an interface in accordance with the detected changes in interface behavior and based on the selection of appropriate interface behaviors form behavior repositories [29]. An extensive discussion regarding the available translation methods for different process modelling paradigms and the association to models of interface behaviors needs to be addressed in future research.

7. Conclusions

A new approach to future enterprise as a complex system is proposed: intelligent cyber enterprise. The authors address the enterprise model of the intelligent cyber enterprise, describing a key component,

the perceptive interface, thus proposing a method to solve the operational context adaptability problems in complex infrastructures. A behavioral model is developed, based on data and information acquired from the enterprise environment.

The proposed model integrates key functions such as processing, perception, communication, learning, pattern recognition and data mining. The proposed case studies address an enterprise environment consisting of intelligent systems-based components with a high degree of autonomy. The case studies illustrate the advantages of the proposed model in relation to complex systems behavior modeling, thus facilitating system adaptation to a dynamic working environment.

References

1. Jardim-Goncalves, R.; Grilo, A.; Agostinho, C.; Lampathaki, F.; Charalabidis, Y. Systematisation of Interoperability Body of Knowledge: The foundation for Enterprise Interoperability as a science. *Enterp. Inf. Syst.* **2013**, *7*, 7–32. [CrossRef]
2. Monostori, L.; Kádár, B.; Bauernhansl, T.; Kondoh, S.; Kumara, S.; Reinhart, G.; Sauer, O.; Schuh, G.; Sihn, W.; Ueda, K. Cyber-physical systems in manufacturing. *Cirp Ann.* **2016**, *65*, 621–641. [CrossRef]
3. *Cyber-Physical Systems (CPS), Program Solicitation NSF 19–553*; National Science Foundation: Alexandria, VA, USA, 2019.
4. Vladareanu, V.; Dumitrache, I.; Vladareanu, L.; Sacala, I.S.; Tont, G.; Moisescu, M.A. Versatile intelligent portable robot control platform based on cyber physical systems principles. *Stud. Inform. Control* **2015**, *24*, 409–418. [CrossRef]
5. Dumitrache, I.; Sacala, I.S.; Moisescu, M.A.; Caramihai, S. A conceptual framework for modeling and design of Cyber-Physical Systems. *Stud. Inform. Control* **2017**, *26*, 325–334. [CrossRef]
6. Taherizadeh, S.; Stankovski, V.; Grobelnik, M. A capillary computing architecture for dynamic internet of things: Orchestration of microservices from edge devices to fog and cloud providers. *Sensors* **2018**, *18*, 2938. [CrossRef]
7. Kochovski, P.; Gec, S.; Stankovski, V.; Bajec, M.; Drobintsev, P.D. Trust management in a blockchain based fog computing platform with trustless smart oracles. *Future Gener. Comput. Syst.* **2019**, *101*, 747–759. [CrossRef]
8. Zhang, X.; Liu, X.; Tang, S.; Królczyk, G.; Li, Z. Solving Scheduling Problem in a Distributed Manufacturing System Using a Discrete Fruit Fly Optimization Algorithm. *Energies* **2019**, *12*, 3260. [CrossRef]
9. Krolczyk, J.B.; Krolczyk, G.M.; Legutko, S.; Napiorkowski, J.; Hloch, S.; Foltys, J.; Tama, E. Material Flow Optimization—A Case Study in Automotive Industry. *Tehnicki vjesnik/Tech. Gaz.* **2015**, *22*, 1447–1456.
10. Panetto, H.; Zdravkovic, M.; Jardim-Goncalves, R.; Romero, D.; Cecil, J.; Mezgár, I. New perspectives for the future interoperable enterprise systems. *Comput. Ind.* **2016**, *79*, 47–63. [CrossRef]
11. Pirayesh, A.; Doumeingts, G.; Seregni, M.; Gusmeroli, S.; Westphal, I.; Gonzalez, L.; Hans, C.; Núñez Ariño, M.J.; Canepa Eugenio, A.; Laskurain, A. Conceptual Framework for Product Service Systems. *Systems* **2018**, *6*, 20. [CrossRef]
12. Repta, D.; Moisescu, M.A.; Sacala, I.S.; Dumitrache, I.; Stanescu, A.M. Towards the development of semantically enabled flexible process monitoring systems. *Int. J. Comput. Integr. Manuf.* **2017**, *30*, 96–108. [CrossRef]
13. Ponce-de-León, H.; Rodríguez, C.; Carmona, J.; Heljanko, K.; Haar, S. Unfolding-Based Process Discovery. In *Automated Technology for Verification and Analysis*; Springer International Publishing: Cham, Switzerland, 2015; pp. 31–47.
14. Wang, L.; Törngren, M.; Onori, M. Current status and advancement of cyber-physical systems in manufacturing. *J. Manuf. Syst.* **2015**, *37*, 517–527. [CrossRef]
15. Dumitrache, I.; Caramihai, S.I. Intelligent cyber-enterprise in the production context. *IFAC Proc. Vol.* **2014**, *47*, 821–826. [CrossRef]

16. Lee, J.; Bagheri, B.; Kao, H.A. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manuf. Lett.* **2015**, *3*, 18–23. [CrossRef]

17. Leitão, P.; Colombo, A.W.; Karnouskos, S. Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges. *Comput. Ind.* **2016**, *81*, 11–25. [CrossRef]

18. Zhang, L.; Li, Z.; Królczyk, G.; Wu, D.; Tang, Q. Mathematical Modeling and Multi-Attribute Rule Mining for Energy Efficient Job-Shop Scheduling. *J. Clean. Prod.* **2019**, *241*, 118289. [CrossRef]

19. Parslov, J.F.; Mortensen, N.H. Interface definitions in literature: A reality check. *Concurr. Eng.* **2015**. [CrossRef]

20. Standard, M.J. *ISO/IEC 2382-1:1993 Information Technology—Vocabulary*; Department of Standards Malaysia: Cyberjaya, Malaysia, 1993.

21. Sacala, I.S.; Moisescu, M.A.; Repta, D. Towards the development of the future internet based enterprise in the context of cyber-physical systems. In Proceedings of the 2013 19th International Conference on Control Systems and Computer Science, Bucharest, Romania, 29–31 May 2013; pp. 405–412.

22. Kannengiesser, U.; Neubauer, M.; Heininger, R. Subject-oriented BPM as the glue for integrating enterprise processes in smart factories. In *OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"*; Springer: Basel, Switzerland, 2015.

23. Seiger, R.; Huber, S.; Schlegel, T. Proteus: An integrated system for process execution in cyber-physical systems. In *Enterprise, Business-Process and Information Systems Modeling*; Springer: Basel, Switzerland, 2015.

24. Van Der Aalst, W.M.; Dustdar, S. Process mining put into context. *IEEE Internet Comput.* **2012**, *16*, 82–86. [CrossRef]

25. Jiang, Y.; Chen, C.P.; Duan, J. A new practice-driven approach to develop software in a cyber-physical system environment. *Enterp. Inf. Syst.* **2016**, *10*, 211–227. [CrossRef]

26. Dumitrache, I.; Caramihai, S.I.; Stanescu, A. From mass production to intelligent cyber-enterprise. In Proceedings of the 2013 19th International Conference on Control Systems and Computer Science, Bucharest, Romania, 29–31 May 2013; pp. 399–404.

27. Starzyk, J.A.; Guo, Y.; Zhu, Z. Dynamically reconfigurable neuron architecture for the implementation of self-organizing learning array. In Proceedings of the 18th International Parallel and Distributed Processing Symposium, 2004 Proceedings, Santa Fe, NM, USA, 26–30 April 2004; p. 143.

28. Davis, J.; Edgar, T.; Porter, J.; Bernaden, J.; Sarli, M. Smart manufacturing, manufacturing intelligence and demand-dynamic performance. *Comput. Chem. Eng.* **2012**, *47*, 145–156. [CrossRef]

29. Wang, S.; Wan, J.; Zhang, D.; Li, D.; Zhang, C. Towards smart factory for Industry 4.0: A self-organized multi-agent system with big data based feedback and coordination. *Comput. Netw.* **2016**, *101*, 158–168. [CrossRef]

3D Object Reconstruction from Imperfect Depth Data Using Extended YOLOv3 Network

Audrius Kulikajevas [1], Rytis Maskeliūnas [1], Robertas Damaševičius [2,3,*] and Edmond S. L. Ho [4]

[1] Department of Multimedia Engineering, Kaunas University of Technology, 51423 Kaunas, Lithuania; audrius.kulikajevas@ktu.edu (A.K.); rytis.maskeliunas@ktu.lt (R.M.)

[2] Department of Applied Informatics, Vytautas Magnus University, 44404 Kaunas, Lithuania

[3] Faculty of Applied Mathematics, Silesian University of Technology, 44-100 Gliwice, Poland

[4] Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne NE1 8ST, UK; e.ho@northumbria.ac.uk

* Correspondence: robertas.damasevicius@vdu.lt

Abstract: State-of-the-art intelligent versatile applications provoke the usage of full 3D, depth-based streams, especially in the scenarios of intelligent remote control and communications, where virtual and augmented reality will soon become outdated and are forecasted to be replaced by point cloud streams providing explorable 3D environments of communication and industrial data. One of the most novel approaches employed in modern object reconstruction methods is to use a priori knowledge of the objects that are being reconstructed. Our approach is different as we strive to reconstruct a 3D object within much more difficult scenarios of limited data availability. Data stream is often limited by insufficient depth camera coverage and, as a result, the objects are occluded and data is lost. Our proposed hybrid artificial neural network modifications have improved the reconstruction results by 8.53% which allows us for much more precise filling of occluded object sides and reduction of noise during the process. Furthermore, the addition of object segmentation masks and the individual object instance classification is a leap forward towards a general-purpose scene reconstruction as opposed to a single object reconstruction task due to the ability to mask out overlapping object instances and using only masked object area in the reconstruction process.

Keywords: 3D scanning; 3D shape reconstruction; RGB-D sensors; imperfect data; hybrid neural networks

1. Introduction

One of the pressing issues in computer vision is three-dimensional (3D) object reconstruction, due to it becoming a core technology in numerous high-end industrial applications such as smart manufacturing, industrial automation and Industry 4.0 [1]. Moreover, there exists a wide variety of applications that would benefit from real time computer vision systems that are capable of fully reconstructing scenes, with most notable examples being an interactive medium such as virtual reality (VR) games and simulations [2], augmented reality (AR) applications or even in newest booming technologies such as extended reality (XR) [3]. Further examples for applications of such systems could include gesture [4,5] and posture [6] applications, indoor mapping [7], obstacle detection [8] recreating environments in movies or even digital forensics [9] to allow for crime scene recreation, robotics [10], teleconferencing [11] with the use of holograms and more. Therefore, we can safely assert that there is definitely a need for affordable, commercially viable solutions capable of providing real-time reconstruction capabilities available to the average user with as little complexity and barrier of entry, in terms of both financial investments and knowledge about the field, as possible.

As we cannot expect an average user to have the access to professional filming sets, mounting arrays of laser scanners capable of scanning the entirety of the room, in addition to the computing

resources that would be required to stitch the data retrieved from multiple high-fidelity depth sensors, we need a solution that would meet or exceed the previous caveats. Therefore, we need a solution capable of working in real-time on a regular non-enthusiast grade workstation or even on a laptop. Furthermore, while we cannot expect the user to have a modern sensor array setup we can try to minimize the initial setup cost to a single depth sensor available in electronics stores or even in quite a few modern mid-tier and flagship phones. While solutions for scene reconstruction from a single depth sensor already exist, these solutions require incremental building per each frame [12,13]. This is done based on camera localization information and delta frames and in the scene reconstruction algorithms that make use of simultaneous localization and mapping (SLAM) [14]. To reliably fill all the holes in the areas that are occluded by other objects and even because of self-occlusion, we would have to scan the entirety of the object from all sides to have its full profile. Furthermore, incremental methods tend to underperform because of one principal flaw: changes in the scene can disrupt the mesh [15]. Making the applications in non-static real world scenes limited, where instead of the entirety of the view moving some objects can change their localization, or even suddenly pop-in or pop-out of the frame. Other proposed methods, such as space carving [16], would bypass some of the incremental building problems by performing what is essentially a subtractive reconstruction from multiple perspectives. However, these methods assume that you can accurately acquire the mask, which can be impossible in certain lighting conditions.

A majority of current algorithms for performing 3D object reconstruction have limitations: objects must be monitored from a large number of views; or views must follow a small baseline, thus the methods cannot function properly when provided only a small number or a single view. To solve these issues one of the most novel approaches employed for state-of-the-art reconstruction algorithms is to employ a priori knowledge of the objects that are being reconstructed [17,18]. These are generally relying on black-box models such as neural networks (NN). One of the most obvious advantages of using a priori information is for the algorithm to approximate the occluded object information, which we as humans are capable inferring quite easily. These methods have shown success in solving this task. For example, 3D Recurrent Reconstruction Neural Network (3D-R2N2) for multi-view reconstruction on the Sanford Online Products [19] and ShapeNet [20] datasets, has managed to achieve this task with fewer images available with competitive results [21], with the proposed improvement that uses densely connected structure as encoder and utilizing Chamfer Distance as loss function [22]. Additionally, Generative Adversarial Networks (GANs) can be used to generate 3D objects from multiple 2D views [23] or even from a single image [24]. GANs have also been shown to be able to predict former geometry of damaged objects [25]. Other authors have used feedforward NNs to detect valid matches between points in an image using different views with more than 98% accuracy [26]. Additionally it was shown that by adopting Bernstein Basis Function Networks (BBFNs) it is also possible to solve the task of reconstructing a 3D shape [27]. A trilateral convolutional neural network (Tri-CNN) that uses three dilated convolutions in 3D to extend the convolutional receptive field was applied on the ShapeNet and Big Data for Grasp Planning [28] data sets to obtain 3D reconstruction from a single depth image [29].

A majority of methods are using voxel based representations, e.g., PointOutNet [30] has shown the ability to predict and generate plausible 3D object shapes. This allows for the model to perform multiple predictions from a single input and using point cloud distribution modeling to refine the final results. Other approaches include: hierarchical surface predictions (HSPs) [31] for predicting high resolution voxel grids using convolutional neural networks (CNNs); discrete wavelet transform (DWT) and principal component analysis (PCA) can be used to get targeted object models, which can be used as an input to an artificial neural network (ANN) to recognize the 3D shape. Other authors have used geometric adversarial loss (GAL) in order to regularize single-view 3D object for object reconstruction using a global perspective by training the GAN to reconstruct multi-view valid 3D models [32]. RealPoint3D network composed of an encoder, a 2D-3D fusion module, and a decoder, accepts a single-object image and a nearest-shape retrieved from ShapeNet to generate fine-grained

point clouds [33]. Similarly, PGNet [34], a recurrent generative network, uses the original images and partial projection images for fine-grained 3D reconstruction. Finally, it was shown that using ANNs it is possible to produce a fully textured, appropriately proportioned 3D model from a single RGB [35] or RGB-D frame [36], however, this approach was limited to basic volume primitives (rectangular boxes and spheres) .

Even though the black-box methods have shown substantial improvements over existing state-of-art reconstruction algorithms such as incremental reconstruction, they can still be prone to severe mishaps due to poor illumination conditions, and object material interaction with light (mainly reflectivity). Furthermore, due to the fact that these methods rely on the visible light spectrum, they are incapable of working in dark environments. Therefore, they would not be suitable to be used in critical applications such as security.

Starting with the *Microsoft Kinect* released in 2010 [37] to *Intel Realsense* [38], the depth sensors are becoming the norm not only in the flagship mobile phones. As of late, stereoscopic depth is becoming available in newer budget phones with the introduction of multiple back facing cameras on a single device. For these reasons we have almost reached an era of the RGB-Depth (RGB-D) sensors being readily available. Therefore, focusing solely on the RGB cameras is missing the potential that the RGB-D cameras may provide for the object reconstruction tasks. For example, depth data stream from the Kinect camera has been used to generate topologically correct 3D mesh models [39].

Applying additional information provided by the RGB-D senor is the logical next step in the lifecycle of the object reconstruction algorithms as we believe they are less dependent on ambient conditions and could potentially be used in pitch black situations due to modern depth sensors using infrared cameras for object depth calculations on the hardware level. We concede that the depth sensors have their own limitations such as speckling due to surface properties [40,41] and distortions caused by infrared projections [42]. However, we believe that the addition of the depth sensor information in conjunction with readily available color data adds useful information. This information helps ANNs to better generalize input data and increase robustness against different lighting conditions. This includes pitch black environments as the depth information is sufficient to reconstruct the captured scene in most cases.

We present an improved hybrid ANN architecture for reconstructing polygonal meshes using only a single RGB-D frame, and employing a priori knowledge, which allows the neural network to be deployed on low-end RGB-D sensor devices with low frame rates.

2. Materials and Methods

2.1. Proposed Hybrid Neural Network Architecture

Our hybrid NN architecture (Figure 1) consists of two major branches: the preliminary input branch that is used for object instance classification and their mask extraction; secondary input branch, which uses the results of preliminary branch in conjunction with the inputs of preliminary branch to perform individual object reconstruction. However, unlike preliminary branch we do not use generalized branches for reconstruction, instead we have *n* of specialized branches for each of the object categories. This allows us to more easily train additional types of objects in the reconstruction branches without having to re-train for classification, in addition this allows to re-train any of the individual reconstruction branches without losing the existing gradients by performing the training on more models [43]. The modularity of the system also provides the advantage of reduced training times as each branch can specialize onto its own generalization task, which gives the ability to change the network configurations of the reconstruction branches by simplifying for easier objects or having more elaborate ANN structures for more complex objects.
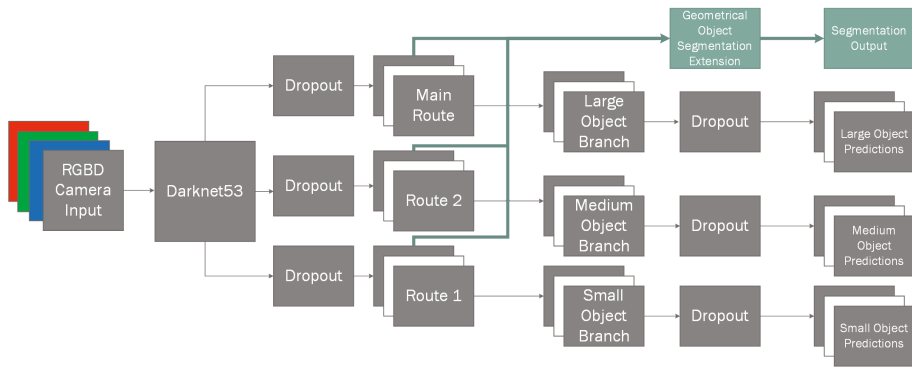
Figure 1. Our extended *YOLOv3* capable of extracting geometric object segmentation along with object bounding boxes.

2.2. Classification and Segmentation Algorithm

Our aim is to detect individual object instances in the scene in order to have a system that is usable in real-world environments. Therefore, we need a classifier that is capable of detecting more than a single object instance for given frame, for example, having two cups and a toy plane on a table would require us to rebuild both of the cups and the toy plane models, respectively. Fortunately, some research has already been performed in the area of individual object instance classification [44–46].

For this reason, to perform our classification task we use one of existing state-of-the-art classifiers as it has shown to produce some of the best results in classification tasks, i.e. *YOLOv3* [47], which we have adapted to our needs to output an additional geometric segmentation mask (Figure 1), while authors have mentioned to be unable to achieve object instance segmentation in their original paper. Additionally, we define the term geometric segmentation as extension to segmentation that allows to discriminate between nearby object instances. This is done by generating a heatmap glow that radiates from the origin of the object. While other more lightweight methods exist, such as *MobileNet* [48], in our paper we try to compare the classification results using three different methods: using only color information; using only depth information; using both color and depth information. Therefore, we have decided to use a slower, but more accurate algorithm to have the most representative results.

Just as the majority of the individual object instance classifying algorithms, *YOLOv3* uses what is know as anchors for object detection. These anchors are used as jumping off bounding boxes when classifying objects, for example, a motor vehicle has a very different profile from a basketball. While the basketball in most cases has close to 1:1 aspect ratio bounding box, meaning that their width is the same, or very close when the image is distorted, to its height, while a motor vehicle like an automobile for the most part has height that is lower than its width. For this reason, one anchor could specialize in detecting automobiles, while the other can specialize in detecting basketballs. Additional feature, albeit a less useful one due to the way our training and testing dataset is generated, is the specification of bounding box scales by the authors of *YOLOv3*. These size specializations group bounding boxes into three groups: small, medium and large. For example small objects may include kitchen utensils, medium objects may include people, large objects may include vehicles. However, these bounding box groups are not exclusionary for these objects unlike anchors as these can vary a lot based on the camera distance from the object. Therefore, as our dataset is completely uniformly generating object scales this grouping loses some of its usefulness.

In our work, we have experimented with three types of inputs into the ANN: color space, front-to-back object depth field and the combination of both. In the case of color space, we use 3 channel inputs for representation of *red*, *green*, *blue* colors; when using depth field, we use a single channel input containing only normalized depth field values and for the combination of both we use

RGBD channels in the same principle. Depth value normalization is performed by dividing each pixel z value using z_{max} of the frame thus landing the depth in range of $z = [0, 1]$. Our input layer is thereafter connected to *DarkNet53* network containing 53 convolutional layers as per specifications, which outputs three routes: *main route* used generally used for larger objects, *route 2* used for medium sized objects and, finally, *route 1* for smaller objects. Due to testing set being uniformly randomly generated, and containing the same object in potentially all size categories, we lose some of the flexibility that is provided by this setup and it impacts classification performance minimally, if removed. However, to stay true to the original algorithm and have an as unbiased result as possible, we have decided to keep all of the branches used in the source material. Additionally, these three routes provide good jumping off points for shortcuts to be used in our segmentation extension (Figure 2).
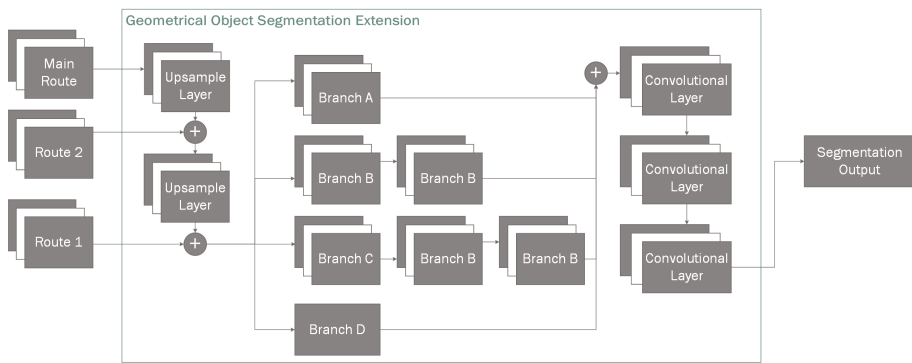

Figure 2. Our proposed geometrical object segmentation extension.

Due to each of the nearby routes being separated by the power-of-two scale, we use transposed convolutional layer [49] to upscale them gradually and then and merge them into desired final shape matrix. We construct our classless geometric segmentation mask by firstly upscaling the *main route* output and merging it with *route 2*, and the resulting layer is then upscaled again and merged with the final *DarkNet* output (*route 1*) which provides us a layer containing latent information of all previous layers that are each specified in learning different sized objects.

Next, we branch out our resulting hidden nodes into four different layers. Each layer contains slightly different network configuration, allowing them to essentially vote on their influence in the final result by extracting different latent feature-maps from the previous layers (Table 1). The first three branches (*A, B, C*) are convolutional branches containing one, two and three convolutional layers, respectively. However, for our final branch (*D*) instead of the convolutional layer, we use a max pool layer to extract the most prominent features. We have selected this parallel stacked approach, because we found it to be more efficient in extracting the object masks than linearly stacked layers when training the segmentation layers independently from the entirety of a model. This decoupling of the segmentation task from the classification task when training gives the additional benefit of allowing us to use transfer learning, which has shown to have very good practical results [50].

Next, we run our concatenated branches through convolutional layers to extract the most viable features and normalize their output in the range of (0, 1) giving us the final segmentation image. In our case the final segmentation output is 80×60 due to it being more than sufficient to extract approximate depth masks as we do not require pixel perfect segment representations. Finally, we use cascading flood-fill (Algorithm 1) to classify the masks pixels-wise. This is done because we found the generated binary masks to be impervious to false positives and false negatives, unlike classification using bounding boxes which can have three types of errors: false positives, false negatives and misclassification. This allows us to remove false positive bounding box detections when they do not intersect the origin of the mask. In our testing set, best cascade parameters were $\epsilon = 0.9$, $\theta = 0.01$.

Table 1. Geometric Segmentation architecture.

	Type	Filters	Size	Output
Main route	Transposed Convolution	1024	$2 \times 2 \times 2$	20×20
	Concatenate	-	-	20×20
	Convolution	256	1×1	20×20
Route 2	Transposed Convolution	256	$2 \times 2 \times 2$	40×40
	Concatenate	-	-	40×40
	Convolution	256	1×1	40×40
	Upscale	-	-	160×120
Branch A	Convolution	128	$1 \times 1/2$	80×60
Branch B	Convolution	32	1×1	160×120
	Convolution	128	$1 \times 1/2$	80×60
Branch C	Convolution	32	1×1	160×120
	Convolution	128	2×2	160×120
	Convolution	256	$3 \times 3/2$	80×60
Branch D	Max Pool	256	$3 \times 3/2$	80×60
	Concatenate	-	-	80×60
	Convolution	256	1×1	80×60
	Convolution	128	1×1	80×60
	Convolution	1	1×1	80×60
	Clip Values	-	-	80×60

Algorithm 1 Cascading flood-fill

1: **procedure** GET_SEED($box, mask, \epsilon$)   ▷ Seeds initial values.
2:    $cx, cy \leftarrow box$ ▷ Get center for *box*.
3:    $seed \leftarrow \varnothing$
4:    $seed \leftarrow find\_closest\_max(box, mask)$ ▷ Find closest max pixel within bounds.
5:    **if** $seed \neq \varnothing \wedge seed_{value} \geq \epsilon$ **then**
6:       $seed_{id} \leftarrow box_{id}$ ▷ Set seed *id* to box *id*
7:       **return** *seed*   ▷ Return seed if value greater than ϵ
8:    **end if**
9:    **return** $\varnothing$ ▷ No valid seed was found.
10: **end procedure**
11: **procedure** FILL_NEIGHBOURS($seed, \theta$) ▷ Recursively fill free neighbours with same or lower values.
12:    **for each** $n \in seed_{neighbours}$ **do**   ▷ For every neighboring mask pixel.
13:       **if** $n_{id} = \varnothing \wedge n_{value} \leq seed_{value} \wedge n_{value} > \theta$ **then**
14:          $n_{id} \leftarrow seed_{id}$ ▷ Set neighbor to same *id* as *seed*.
15:          $FILL\_NEIGHBOURS(n)$ ▷ Call recursively.
16:       **end if**
17:    **end for**
18: **end procedure**
19: $bounding\_boxes \leftarrow sort\_confidence(bounding\_boxes)$ ▷ Sort bounding boxes by confidence.
20: **for each** $box \in bounding\_boxes$ **do**   ▷ For each bounding box *b*
21:    $seed \leftarrow GET\_SEED(box, \epsilon)$
22:    **if** $seed \neq \varnothing$ **then**
23:       $FILL\_NEIGHBOURS(seed, \theta)$
24:    **end if**
25: **end for**

Additionally, we have also modified *YOLOv3* network for we had issues with the network being unable to train by consistently falling into local minima during gradient descent and getting perpetually stuck in them. To solve this issue we introduced periodic hyper parameters [51] during model learning. Specifically, we had changed the learning rate to alternate in specified range of $lr_{min} = 1e^{-6}, lr_{max} = 1e^{-4}$.

$$y(x) = \begin{cases} \frac{x}{w_1} \times (lr_{max} - lr_{min}) + lr_{min}, & if\, x < w_0 \\ \frac{e^{1+\pi \times \frac{\cos{(x-w_1)}mod(w_0+1)}{w_0}}}{\pi e^3} \times (lr_{max} - lr_{min}) + lr_{min}, & otherwise \end{cases} \tag{1}$$

This periodical learning rate (Equation (1)) has vastly improved our models ability to learn the underlying relationships of input date by alternating between low and high training rates, therefore jumping out of potential local minima that it might start orbiting around during stochastic gradient descent. Our function has two stages, the first stage that consists of two training iterations, where $w_1 = 2 \times s$, and the second stage of 4 iterations, where $w_0 = 4 \times s$ where s is the number of steps per batch. We selected the two state learning function because having high learning rates initially may cause the model to diverge. Therefore, during the first stage we linearly increase the learning rate. Once in the second stage we use the cosine function and the modulus operator for the model to alternate between two values. The shape of the alternating function also can have influence in model convergence as some models require to be in different extremum points for different amounts of times. Therefore, having a different dataset may require more fine-tuning of parameters of this equation for different slope shapes, while still maintaining the benefits of having alternating learning rates.

Additionally, as we are training the NN from scratch, we have noticed that our network, despite being able to find better convergence results due to periodical learning rate jumping out of local minima, had a high bias rate. A high bias rate is an indicator that our model is over-fitting on our data set. To solve this additional issue, we modified the *YOLOv3* network by adding additional dropout layers with the dropout rate of $P(x) = 0.5$ after each branch of *DarkNet53* and before each of the final layers predicting the bounding boxes.

Furthermore, we had issues of model overfitting to the training set, to solve this we additionally modified the neural network by adding two additional dropout layers. We trained our model 6 times, each with 50 iterations using mini-batch of size 8 for comparison, because after about 50 iterations the standard *YOLOv3* model starts to overfit and loose precision with our validation dataset. Therefore, for most objective comparison we trained our modified network for same number of epochs. Note that even though our method also starts to overfit, unlike the YOLOv3 network model, the accuracy of our modified model when overfitting remains roughly at the same value from which we can deduce that the changes make the model more stable.

Figure 3 shows the differences in loss function when trained using the RGB, RGB-D and Depth data as input. For the unmodified *YOLOv3* we are using $lr = 1e^{-5}$ as the midpoint between our minimum and maximum learning rates in the periodic learning rate function. As we can see from the graph, the loss function using static learning rate on the RGB and RGB-D datasets reaches a local minimum causing the model to slow down its ability to learn new features, unlike our periodic learning rate which seems to temporarily force the model to overshoot its target which sometimes causes it to fall into a better local minimum. This effect can be seen in the distinct peaks and valleys in the graphs. The outlier in these graphs are depth-only data points. While in both cases the loss function seems lower and has a better downwards trajectory in stochastic descent, however, we have noticed that despite seemingly lower loss when compared to RGB and RGB-D, the actual model accuracy is very unstable on epoch-per-epoch basis. We assert that this is the case due to depth alone providing very unstable data that's very hard to interpret. We make this assumption due to the fact that even when taken an expert to evaluate the depth maps alone, it is usually very hard to discern what type of object it is without knowing its texture; it is only possible to tell that there is in fact an object in the frame.

Finally, we can see that the RGB-D data is a clear winner when training in both cases, which means that depth data can indeed help in model generalization.
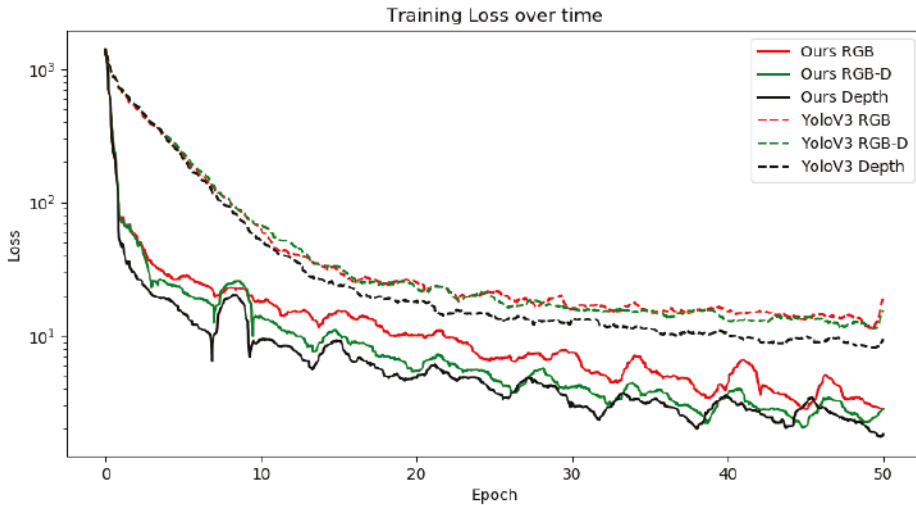

Figure 3. Training loss comparison between baseline *YOLOv3* and our modified version when using RGB, RGB-D and depth data as training. Due to the loss function being inherently noisy for each of the mini-batches, we have used Savitzky-Golay [52] digital filter to perform the smoothing of the overall graph.

2.3. Reconstruction Algorithm

The proposed algorithm for 3D object reconstruction consists of two subsystems: voxel cloud reconstruction and post-processing (Figure 4). In reconstruction step we take the outputs of the 3D classifier mask for the object and in conjunction with the original depth map which we feed into our reconstruction ANN (Figure 5) that performs the object reconstruction task for the given masked input frame. Unlike the classification algorithm we only use the underlying depth input from the classifier as it provides enough information for the specific object reconstruction. This is due to fact that we already know the class of the object, which is required for classification because different objects can have very similar depth representations. However, during reconstruction this is not an issue because our ANN is designed in such a way that each branch is responsible for reconstructing similar object representations.

Once the classifier-segmentation branch has finished its task, for each object instance the appropriately trained reconstruction branch is selected. In our case all the branches are highly specialized on a single type of object that it can reconstruct, which is why object classification is required. However, we believe that there is no roadblock to having more generic object reconstruction branches for example all similar objects may be grouped to a single reconstruction task. This could potentially allow some simplifications in the classification-segmentation as it would no longer be required to classify highly specific object instances thus reducing failure rate caused by object similarities. For example, a cup and a basket can be very similar objects and be misclassified. Additionally, the hybridization allows for fine tuning of the reconstruction branches without having to retrain the entire neural network model potentially losing already existing gradients via on-line training skewing the results towards new data posed. This in turn reduces re-training time if new data points are provided for a specific object as we no longer need to touch the established branches due to modularity.

Inside our reconstruction network branch (Figure 2) for given depth input we use convolutional layers to reduce the dimensionality of the input image during the encoding phase (see Table 2). For a

given input, we create a bottleneck convolution layer which extracts 96 features, afterwards we use a spatial 2D dropout [53] layer before each with $P(x) = 0.1$ to improve generalization. We use spatial dropout as it is shown to improve generalization during training as it reduces the effect of nearby pixels being strongly correlated within the feature maps. Afterwards, we add an additional inception [54] layer (Figure 6) which we will use as a residual block [55] followed by another spatial dropout. Afterwards, we add two additional bottleneck residual layers, each followed by additional dropouts. With final convolution giving us final 256 features with the resolution of 20×15. Our final encoder layer is connected using a fully-connected layer to a variational autoencoder [56] containing 2 latent dimensions, as variational autoencoders have shown great capabilities in generative tasks. Finally, the sampling layer is connected to full-connected layer which is then unpacked into a $4 \times 4 \times 4$ matrix. We use the transposed three-dimensional convolutional layers in order to perform up-sampling. This is done twice, giving us 4 feature maps in $32 \times 32 \times 32$ voxel space. Up to this point we have used Linear Rectified Units [57] (ReLUs) for our activation function, however, for our final 3D convolutional layer we use a softmax function in order to normalize its outputs where each voxel contains two neurons. One neuron indicating the confidence of it being toggled on, the other neuron showing the confidence of the neuron being off. This switches the task from a regression task to a classification task, allowing us to use categorical cross entropy to measure the loss between the predicted value and our ground truth.
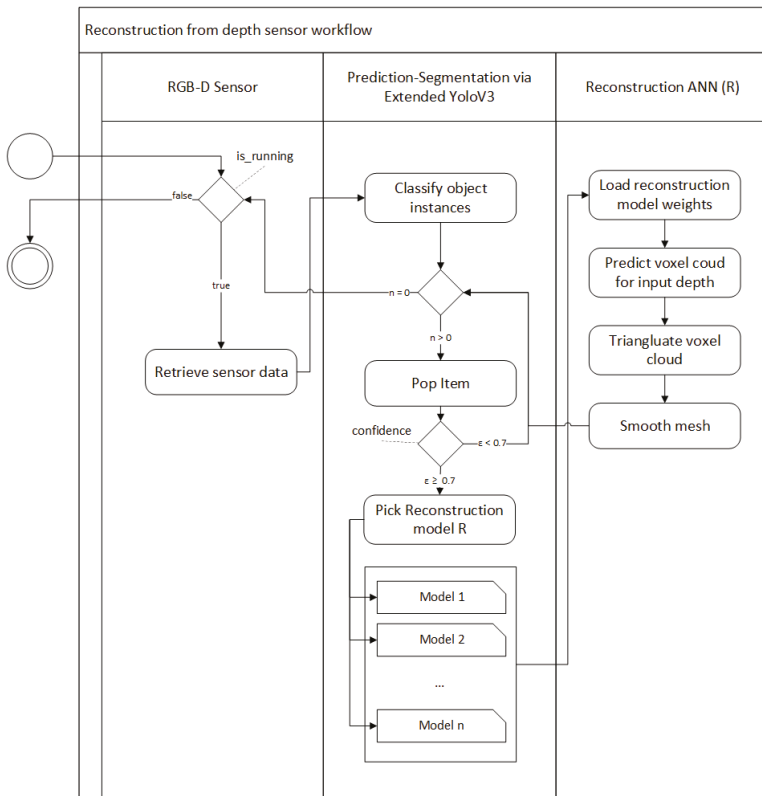

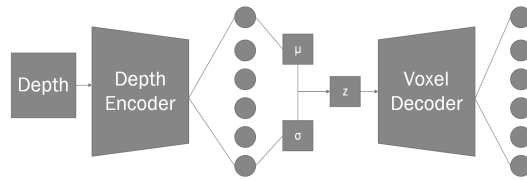**Figure 4.** Workflow of object reconstruction from sensor data.

Figure 5. Diagram of a single object reconstruction network architecture branch. For the given depth frame, the depth encoder creates a bottleneck, which is then directly connected to VAE node, the resulting sampler is connected into voxel decoder. The voxel decoder layer outputs a $32 \times 32 \times 32 \times 2$ matrix which can be explained as $x \times y \times z \times s$, where x, y, z components indicate position in 3D grid, and s component indicates voxel state encoded as one-hot.

Table 2. Architecture of the reconstruction neural network.

|	Type	Filters	Size	Output
	Input	-	-	320×240
Encoder	Convolution	96	$5 \times 5/2$	160×120
	Dropout 2D $P(x) = 0.1$	-		160×120
	Inception	(8, 4)	-	160×120
	Convolution	16	1×1	160×120
	Add	-	-	160×120
	Convolution	128	$5 \times 5/2$	80×60
	Dropout 2D $P(x) = 0.05$	-		80×60
	Inception	(8, 4)	-	80×60
	Inception	(16, 8)	-	80×60
	Convolution	32	1×1	80×60
	Add	-	-	80×60
	Convolution	128	$3 \times 3/2$	40×30
	Dropout 2D $P(x) = 0.025$	-		40×30
	Inception	(8, 4)	-	40×30
	Inception	(16, 8)	-	40×30
	Inception	(32, 16)	-	40×30
	Convolution	64	1×1	40×30
	Add	-	-	40×30
	Convolution	256	$3 \times 3/2$	20×20
VAE	Flatten	-	-	76 800
	Fully-Connected	-	-	512
	Mean	-	-	2
	Standard Deviation	-	-	2
	Sampling	-	-	2
Decoder	Fully-Connected	-	-	64
	Reshape	-	-	$4 \times 4 \times 4$
	Inception 3D	(32, 16)	-	$4 \times 4 \times 4$
	Inception 3D	(16, 8)	-	$4 \times 4 \times 4$
	Inception 3D	(8, 4)	-	$4 \times 4 \times 4$
	Convolution 3D	16	$1 \times 1 \times 1$	$4 \times 4 \times 4$
	Add	-	-	$4 \times 4 \times 4$
	Transposed Conv 3D	64	$3 \times 3 \times 3 \times 2$	$8 \times 8 \times 8$
	Inception 3D	(16, 8)	-	$8 \times 8 \times 8$
	Inception 3D	(8, 4)	-	$8 \times 8 \times 8$
	Convolution 3D	16	$1 \times 1 \times 1$	$8 \times 8 \times 8$
	Add	-	-	$8 \times 8 \times 8$
	Transposed Conv 3D	32	$3 \times 3 \times 3 \times 2$	$16 \times 16 \times 16$
	Inception 3D	(8, 4)	-	$16 \times 16 \times 16$
	Convolution 3D	16	$1 \times 1 \times 1$	$16 \times 16 \times 16$
	Add	-	-	$16 \times 16 \times 16$
	Transposed Conv 3D	4	$5 \times 5 \times 5 \times 2$	$32 \times 32 \times 32$
	Convolution 3D (Softmax)	2	$3 \times 3 \times 3$	$32 \times 32 \times 32$
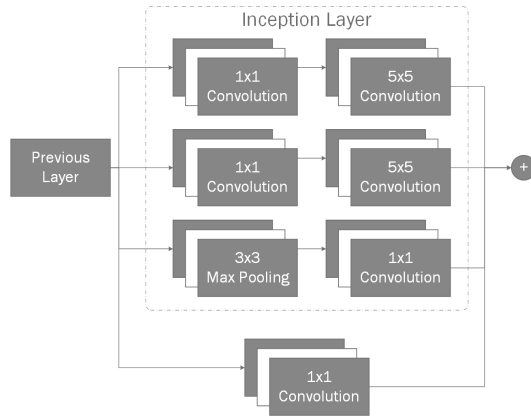
Figure 6. An example of the inception layer. An input layer is connected to three branches in parallel. If multiple inception layers are used inception layers are connected sequentially. Final inception layer outputs and 1×1 convolution are then connected using addition. The result is then used for subsequent layers.

2.4. Proposed Network vs. YOLOv3

Our approach is the hybridization of two ANN architectures: classification-segmentation branch and reconstruction branch (see Figure 7). The classification-segmentation branch as the name suggests performs object instance classification and segmentation. This information is then fed to the object reconstruction branches. Object reconstruction branch contains a fleet of specialized pre-trained autoencoder models where each of the auto-encoders can reconstruct the model's three-dimensional representation while being provided only a single depth frame. The initial classification-segmentation branch is our expanded interpretation of YOLOv3 which adds crucial output to already existing YOLOv3 network output, i.e., the object instance segmentation. This extension adds crucial information which is required for the reconstruction step by extracting the object instance mask that can be applied per each object on the initially captured depth.
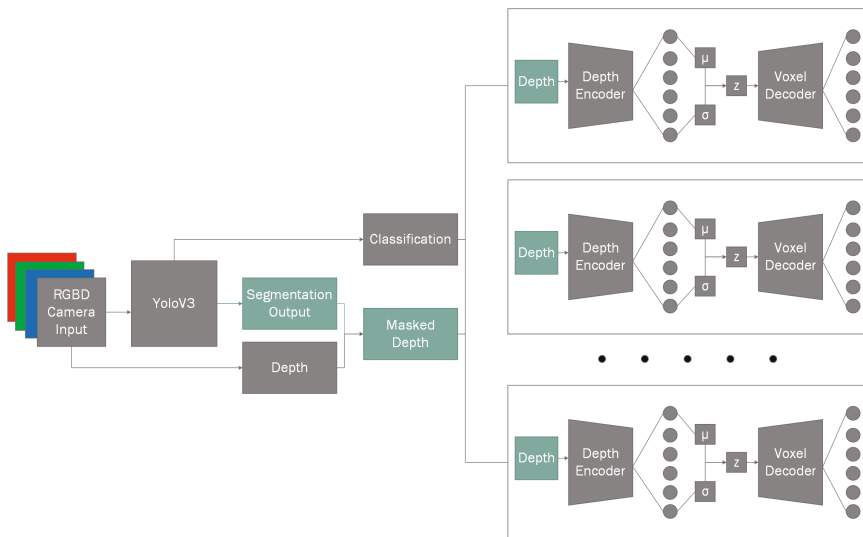

Figure 7. Full view of the proposed network model that extends the YOLOv3 network.

2.5. Dataset

As our method entails the requirement of a priori information for the captured object reconstruction, there is a need for a large well labeled element dataset. However, unlike for object recognition which has multiple datasets, e.g., *COCO* [58] dataset, *Pascal VOC* [59]; there seems to be a lack of any public datasets that provide RGB-D scene representation in addition to it's fully scanned point cloud information viable for our approach. While datasets like *ScanNet* [60] exist, they are missing finer object details due to focusing their scan on full room experience that we are trying to preserve. Therefore, our training data consists exclusively out of synthetically generated datasets, which use the *ShapeNetCore*, a subset of *ShapeNet* dataset that provides 3D object models spanning 55 categories (see an example of a coffee cup model in Figure 8). In addition, we use real-life data acquired by the *Intel Realsense ZR300* and *Intel Realsense D435i* (Intel Corp., Santa Clara, CA, USA) devices for visual validation as it is impossible to measure it objectively without having a 3D artist recreating a 1:1 replica of said objects, which is unfortunately unfeasible option. However, using real world samples as a validation set is not subject to training bias because they are never being use in the training process.

As mentioned, for the training of the black-box model we are using the *ShapeNetCore* dataset that we prepare using *Blender* [61] in order to create the appropriate datasets. Due to the fact that we are training a hybrid neural network, we need two separate training and testing sets, one for each task.

Figure 8. A coffee cup model from the *ShapeNetCore* dataset.

2.5.1. Classification Dataset

To create this subset of data we create random scenes by performing the following procedure. Firstly, we randomly decide how many objects we want to have in the scene in the range of $n_{objects} = [1; 10]$ and pick that many random objects from *ShapeNetCore* dataset to populate the scene. Before applying any external transformations we transform the object geometry so that all objects are of uniform scale and have the same pivot point. To perform the required transformations firstly we calculate the geometry extents. Once we know the object extents we can move all the objects on *Up* axis (in our case this is *z*) and scale down all vertices by the largest axis (Algorithm 2). This gives us a uniformly scaled normalized geometry that we can freely use.

Algorithm 2 Normalize geometry

1: **procedure** EXTENTS(*G*) ▷ Calculates extents for geometry *G*
2:     $min_x, min_y, min_z \leftarrow Infinity$ ▷ Initialize *min* vector
3:     $max_x, max_y, max_z \leftarrow -Infinity$ ▷ Initialize *max* vector
4:     **for each** $v \in G$ **do** ▷ For each vertex *v*
5:         $min_x \leftarrow min(v_x, min_x)$
6:         $min_y \leftarrow min(v_y, min_y)$
7:         $min_z \leftarrow min(v_z, min_z)$
8:         $max_x \leftarrow max(v_x, max_x)$
9:         $max_y \leftarrow max(v_y, max_y)$
10:         $max_z \leftarrow max(v_z, max_z)$
11:     **end for**
12:     **return** min, max
13: **end procedure**
14: $min, max \leftarrow EXTENTS(G)$
15: $bounds \leftarrow max - min$
16: $max\_bound \leftarrow 1/max(bounds_x, bounds_y, bounds_z)$
17: **for each** $v \in G$ **do** ▷ For each vertex *v*
18:     $v_x \leftarrow v_x/max\_bound$
19:     $v_y \leftarrow v_y/max\_bound$
20:     $v_z \leftarrow (v_z - min_z)/max\_bound$ ▷ Offset the vertex on *up* axis before normalizing bounds
21: **end for**

We place the selected objects with random transformation matrices in the scene, making sure sure that the objects would never overlap in space. To generate random local transformation matrix (*L*) (Equation (3)) we need three of it's components: Scale (*S*), Rotation (R_z) and with random value; use either capital or lower-case s in both places in the range of $s = [0.7, 2)$; Rotation (R_z), where rotation is random value in the range of $\theta = [0, 2\pi)$, we perform rotation only on *z* axis to ensure that randomly generated scenes are realistic and do not require artist intervention; Translation (*T*), where *x* and *y* values are non-intersecting values in the range of $r = [-5, 5]$ and $\alpha = [0, 2\pi)$ (Equation (2)).

$$\begin{cases} x = r \times \cos \alpha \\ y = r \times \sin \alpha \end{cases} \tag{2}$$

$$L = S \times R \times T = \begin{bmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ x & y & z & 1 \end{bmatrix} \tag{3}$$

Once the selection objects are placed we need to apply lighting in order to have real-life like environments. To do this, we use the Lambertian shading model and directional lights. We randomly generate $n_{lights} = [1; 4)$ lights in the scene. We pick a random light rotation, we ignore translation as it does not matter in directional lights; we generate a random color in the range of $Col_{RGB} = [0.7, 1]$, we selected the minimum bound of 0.7 to avoid unrealistic real-world lightning; and random intensity $I = [0.7, 1]$. This light acts as our key light. To avoid hard shadows being created, which wouldn't be the case unless using spotlight in real world, for each key light we create a backlight which is pointing the opposite direction of key light with half the intensity and identical color to the key light.

Once the scene setup is complete, we render the scene in three modes: *color*, *depth* and *mask*. Color mode gives us the scene representation from a regular light spectrum camera. As we are not putting any background objects into the scene the generated background is black. However, later on we use augmentation during training to switch the backgrounds to improve recall rates. Once the color

frame is extracted we extract the mask, in order to extract the mask we assign each object an incremental *ID* starting at 1, this allows us to differentiate between objects in the frame. Finally, we render the depth representation of the scene. Before rendering depth we place a plane on the ground that acts as our ground place, this allows for more realistic depth representations because the objects are no longer *floating* in space. The depth is rendered front-to-back, meaning the closer the object is to the camera the closer to zero depth value is, the front-to-back model was chosen because this is the same as *Intel Realsense* model.

Each of the scenes is rendered in *320 × 240* resolution $n = 25$ times by placing it in random locations (Algorithm 3) and pointing it at the center of the scene, where $r = 10$, $z_{min} = 4$, $z_{max} = 6$.

Algorithm 3 Camera location

1: $step\_size \leftarrow 2\pi/(1-n)$
2: **for** i < n **do**
3:     $\theta \leftarrow random(i, i+1)$ ▷ Random float in the range of [i, i+1]
4:     $x \leftarrow \cos(step\_size \times \theta) \times r$
5:     $y \leftarrow \sin(step\_size \times \theta) \times r$
6:     $z \leftarrow random(z_{min}, z_{max})$
7: **end for**

We save the perspectives as *OpenEXR* format [62] instead of traditional image formats instead of, for example, *PNG*, as *OpenEXR* file format is linear, allowing for retention of all depth range without any loss of information as it is not limited to 32 bits per pixel. The final *EXR* file has these channels in it *R*, *G*, *B* containing red, green and blue color information respectively; *id* channel contains the information about the mask for specific pixel; *Z* information containing the linear depth data.

Once we create the input image, we additionally label the data and extract the segmentation mask that will be used as output when training the artificial neural net. We perform this step after the scene is rendered in order to account for any kind of occlusion that may occur when objects are in front of each other causing them to overlap. We extract the object bounding boxes by finding the most top-left and bottom-right pixel of the mask. The binary mask is extracted based on the pixel square distance from the center of the bounding box. This means that the center pixels for the bounding box are completely white and the closer to the edges it is the darker it gets. We use non-flat segmentation to be able to extrapolate individual object instances in the mask when they overlap, and this is done by interpolating the pixel intensity from the bounding box edge to bounding box center. The mask is then scaled down to *80 × 60* resolution as it is generally sufficient and reduces the required resources.

2.5.2. Anchor Selection

The existing anchors that are being used with *COCO*, *Pascal VOC* and other datasets are not suitable for our dataset, rarely fitting into them. Therefore, we performed class data analysis and selected three most fitting anchors per classifier branch scale. As we can see from Figure 9, our classes generally tend to be biased towards 1:1 aspect ratio due to data set being randomly generated unlike in real world applications.

However, while the classes tend to be biased towards 1:1 for the most part, the assertion that all individual object instances would neatly fit into this aspect ratio would be incorrect as they still retain certain bias. According to previous Single Shot Detection (SSD) research [63], selecting inadequate base anchor boxes can negatively affect the training process and cause the network to overfit. Therefore, we chose to have 3 anchors per anchor size as this seems to sufficiently cover the entire bounding box scale spread by including tall, wide and rectangle objects. We select the anchor points using *K-Means* to split data into 9 distinct groups (Figure 10).

Once we have our cluster points for bounding box detections, we sort them in order to group into small, medium and large anchor sets. Giving us three different anchors, each having the most popular aspect ratios per that scale detection branch as it can be seen in Table 3.
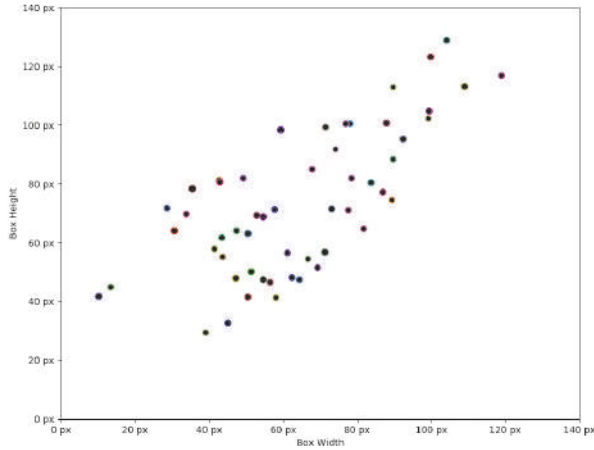

Figure 9. Each individual point denotes the mean object bounding box scale for each class type.
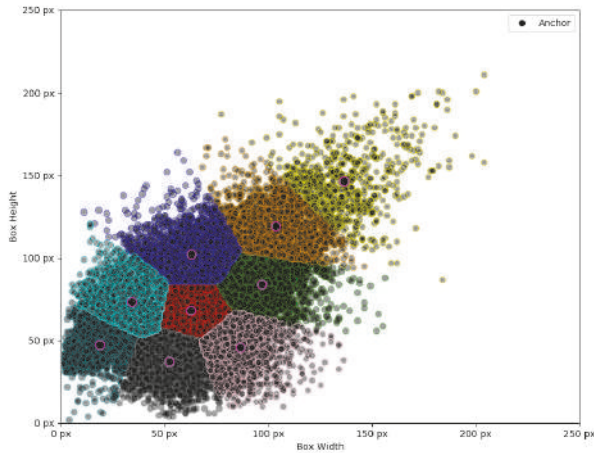

Figure 10. Selected anchors using *K-Means* clustering algorithm. Different colors denote distinct anchor groups responsible for detecting objects in the spread.

Table 3. Anchor scales in pixels calculated using the *K-Means* clustering method.

Anchor Type	Anchor 1	Anchor 2	Anchor 3
Small	18.83, 47.53	52.34, 37.53	34.13, 73.28
Medium	86.35, 46.02	62.74, 68.31	62.75, 102.19
Large	96.69, 84.20	103.66, 119.51	136.34, 146.64

The neural network architecture described in Section 2.2 was trained in three separate modes in order to infer how much the additional depth information improves the classification results.

These three modes consist of RGB, RGB-D and Depth training modes. Where RGB mode implies we train using only the color information that was generated from the dataset, the RGB-D mode uses both depth and color information and finally Depth mode trains the network using only depth information. We do not use any additional data augmentation when training in both RGB and RGB-D modes. We do however, add additional augmentation when training in the RGB-D mode. When training in the RGB-D mode there is a small chance that either RGB or Depth channel will not be included in the testing sample. We perform this augmentation because both RGB camera and Depth sensors may potentially have invalid frames. Therefore, we assert that both of these data points are equally necessary for the classification task, and that they must be generalized separately from each other and should provide equal contributions to the classification task. This is decided randomly when preparing the mini-batch to be sent to the artificial neural network for training. There is $\lambda = 0.1$ chance that the input specific data point will be picked for additional augmentation. If the data point is picked for augmentation then there is equal probability that either RGB or Depth Data will be erased from the input and replaced with zeros. We decided on this augmentation approach because both RGB and Depth frames using real sensors are prone to errors. For example, the RGB camera may fail in bad lighting or even be unavailable when the room is pitch black. Likewise, the depth frames are also prone to errors due to inconsistencies in generating depth map which causes the sensor to create speckling effect in the depth information, additionally cameras being too close to object may be completely unable to extract proper depth information. Therefore, we chose this augmentation approach as it allows for the sensors to work in tandem when both are available, but fill in the gaps, when one of them is failing to provide an accurate information.

2.5.3. Reconstruction Dataset

For the reconstruction training set, we use the same *ShapeNetCore* dataset to generate the corresponding depth images and ground truths for the individual objects voxel cloud. We used *Blender* to generate the training data. However, the generated input data is different. We assert that the object material does not influence the objects shape, therefore we no longer generate the color map unlike when generating classification data. Therefore, we only render the depth information for each object. We render individual objects by placing the cameras in such a way that the specific object would be visible from all angles from 45° to 90° at a distance from 1 to 1.5 m, excluding the bottom. As a result we have 48 perspectives for each of the object models. Once again we save the models as *OpenEXR* file in order to preserve the depth values in this lossless format. Finally, we generate the voxel-cloud representation [64]. Voxelization is performed by partitioning into the equally sized cells, where the cell size is selected based on the largest object dimension axis. Following the space partitioning, we repeat over each of the cells and compute whether the specific cell should be filled by ray-polygon intersection [65].

2.6. Evaluation

In order to evaluate the correctness of our results, we evaluate the results of the proposed algorithm, and additionally we evaluate both of the subsystems individually. To evaluate the classification accuracy, we use the *mAP* metrics to assess the quality of the classifier and it's output bounding boxes. When performing the classification accuracy evaluation, we evaluate all three train models: RGB, RGB-D and Depth. This allows us to determine the quality differences between the addition of depth information in the classification task.

For the reconstruction task we require the output voxel representation of the object to be as close to ground truth as possible. For that, we define our reconstruction quality as the *Intersection-over-Union* metric. Furthermore, we use the *Correctness*, *Completeness*, and *Quality* metrics during evaluation.

3. Results

3.1. Settings

Our experiments have been executed using two computers: (1) a workstation with *Intel i7-4790* CPU with *16 GB of RAM* which achieved 55.76 fps, and *nVidia 1070* graphics card with *8 GB GDDR5 VRAM*; and (2) a laptop computer using *nVidia 960M* graphics chip with *4GB GDDR5 VRAM*, *Intel i5-4210H* CPU and *12 GB of RAM*, which reached 11.503 fps. We consider that these machines should represent the target range of end user devices.

3.2. Quantitative Results

3.2.1. Object Instance Classification Results

In order to evaluate our model in all cases, we have used the mAP metric, which is a widely used method in order to evaluate mean average precision of the predicted bounding boxes with respect to their Intersection-over-Union (IoU), provided that the object classes match. As per suggested *COCO* evaluation we filter out bounding boxes which have an $IoU < 0.5$ in order to compare all of our trained model versions.

As Table 4 suggests, our iterative training approach in addition to dropout layer was substantially better in the object classification task as opposed to the originally suggested variant which would either plateau with too low of a learning rate or get stuck in a constant loop around the local minima due to the initial learning rate being too high. Therefore, we can assert that a periodic learning rate is a useful tool to improve model generalization and the speed at which the network can train by adding additional noise during training time in a form of sudden overshooting. Furthermore, we can see that the addition of depth information as input greatly increases the recall rate in both cases, while the depth information alone has similar recall rate in both cases. This suggests that the depth cameras can not only greatly benefit in the object classification task when used in conjunction with visible light spectrum cameras but it can be used as a fallback when no light source is available, albeit with lower precision.

Table 4. Mean precision values in respect to $IoU > 0.5$ for each of our trained models.

Network Type	mAP (%)
Our RGB-D	60.20%
YoloV3 RGB-D	55.75%
Our RGB	41.27%
YoloV3 RGB	37.96%
Our Depth	26.46%
YoloV3 Depth	20.87%

One of the glaring issues we noticed with the *ShapeNetCore* dataset during our experiments is that, while there are specified a total of 55 classes, a lot of those classes have major overlap in form and function which may dramatically affect the overall mAP value, such as classes that are categorized as distinct (e.g., *pistol* and *handgun*) could still be grouped into the same class as they share key characteristics which may not be viable to differentiate when using relatively low resolution images. Additionally, some groups of objects can be distinct in their use (e.g., *mug* and *ashcan* and *basket*) in many cases have no differentiable features and would require each individual scene to be hand crafted by an artist in order to provide visual queues about the objects in relation to the world, which should potentially allow for differentiation between very similar objects (Figure 11). However, this is currently beyond the scope of our paper.
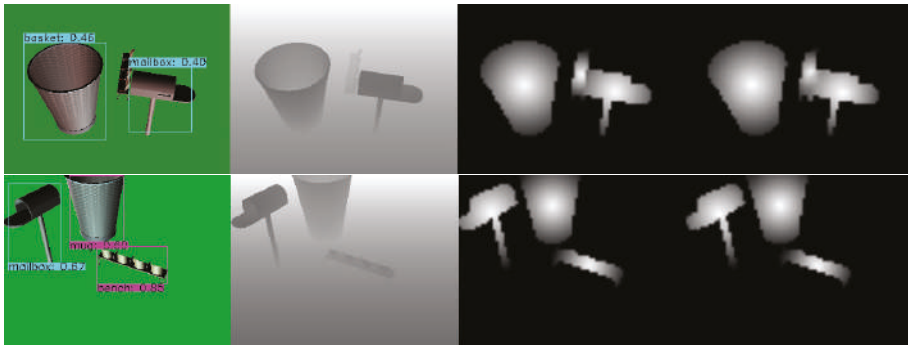
Figure 11. Prediction made with our extended *YOLOv3* network. Left to right: (1) Input color image with predicted object instances; (2) Input depth frame; (3) Upscaled to 320 × 240 ground truth mask; (4) Predicted mask upscaled to 320 × 240. Same object is being treated as two distinct classes due to lack of cues for the artificial neural network of what the specific object may be due to scenes being generated randomly.

3.2.2. Mask Prediction Results

As one of our main goals is to extract individual object instances from the depth map, we extended the *YOLOv3* network architecture to be able to predict object masks. In order to compare the predicted mask similarity with the ground truth we use the structural similarity index metric (SSIM) that measures perceived similarity between two images.

As we can see from Figure 12, in all cases our *YOLOv3* extension for object mask prediction is capable of extracting mask frame not only from the combined RGB-D frames but also from the RGB and Depth frames alone. This shows us that both color and depth information individually is generally enough for this task. However, both of these sensors may fail in different environments so the conjunction of both would most likely procure the most accurate results. Additionally, while in both method cases (static and periodical) the similarity is generally more than enough to extract accurate mask, using periodical approach provides a much lower standard deviation, hence better expected results. Additionally, the higher similarity also signals a tighter mask which may improve reconstruction quality due to reduction in bad data. While in our tests RGB has slight advantage over RGB-D when generating a mask, it is worth noting that Depth adds an additional dimension to the data which makes the dataset slightly harder when compared to RGB alone. This is due to RGB alone being able to drop the randomly generated background, unlike RGB-D which has a non-uniform background due to addition of ground plane. As we can see in Figure 13, our approach is applicable not only for synthetic but for real-world data too. This indicates that the network managed to generalize well and it's result can be used during reconstruction step.

3.3. Reconstruction Results

3.3.1. Quantitative Results

We can observe the achieved results for our proposed method in Figure 14 as they compare to previously achieved results in hybrid neural-network reconstruction [66]. As we can see the mean *IoU* metric value as compared to the results presented in [66] has significantly improved for some of the models, more importantly—even if the the improvement was minimal or if the results were slightly lower the error spread is lower. This indicates that the achieved results are much more stable. Additionally we can see that our reconstruction results are comparable to that of other state-of-art methods like *3D-R2N2* reporting 0.571 mean *IoU*.
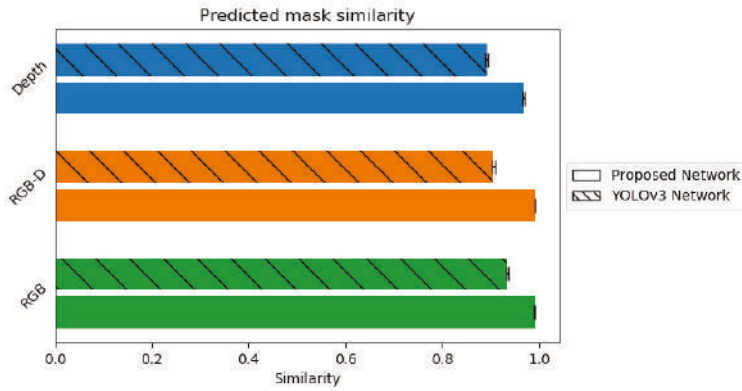
Figure 12. Similarity of created mask to the mask of ground truth. The hashed bar denotes the similarity of masks predicted by the *YOLOv3* network, the solid bar denotes the similarity of masks predicted for Depth, RGB-D and RGB frames by the network model proposed in this paper.

Figure 13. An example of real world object classification using the proposed network model: segmented and classified RGB frame (**left**), depth frame (**middle**), and predicted depth mask (**right**).
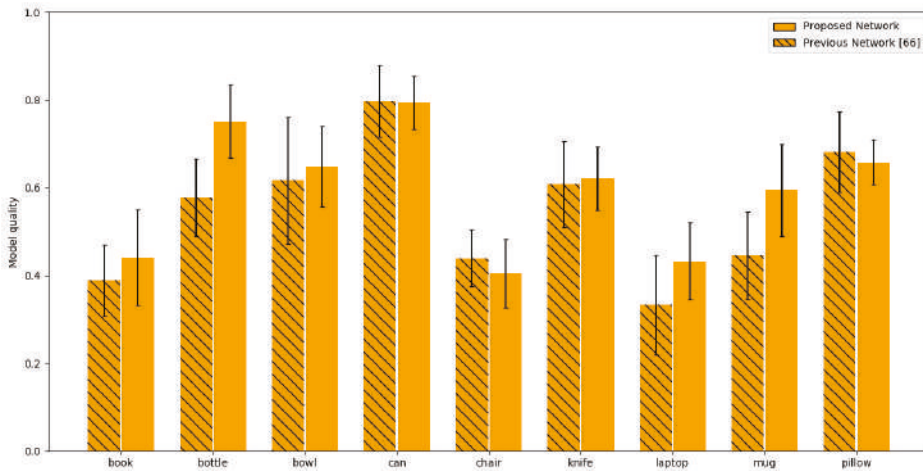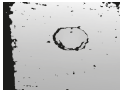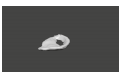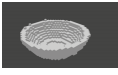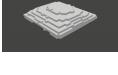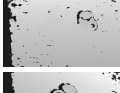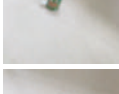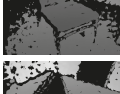

Figure 14. Comparison between the predicted object shape and ground truth using the IoU metric for different objects in the training set. The hashed bars denote the results achieved using the network proposed in [66]. The solid bars denote the results for the proposed network.

3.3.2. Visual Inspection and Result Validation

For every object that we have trained, we had collected real world examples using *Intel Realsense* device in order to compare how well synthetic results transfer into real world data. The results for the given dataset can be seen in the Table 5.

Table 5. Visual evaluation of object reconstruction. Table presents: RGB frame, original depth frame; reconstructed cloud of voxels; triangulated and smoothed surface created using predicted voxel cloud; and a corresponding similar object in the training set.

The reconstructed object shapes are generally recognizable. However, certain object angles cause the network to fail the task, for example, one of the bowls is missing half of it's voxels, while the other bowl may be considered a near perfect reconstruction. While the ANN has managed to reconstruct the *Book* and *Knife* datasets, it has generally only managed to reconstruct their base shape primitives which make the objects somewhat indistinguishable by experts without any prior information of what the objects may be. While the human bias may notice the minute structural differences between the knife handle and blade in terms of it's width, we still consider this a failed reconstruction. *Can* has managed to achieve great results in terms of reconstruction, while the pillow reconstruction could be considered near perfect. *Mug* in our training set is one of the trickiest objects as it contains a handle which should be reconstructed with a hole and additionally the mug cannot be fully filled in with voxels as in our case it is empty. While in all three cases the basic shape of the cup was maintained, there are some issues with two test cases. One of the test cases was missing a hole for the handle, while another is substantially distorted. However, the distortions may be explained by extremely noisy dataset. The *Chair* dataset allowed to reconstruct the shape of the chair although some of the details were missing. The *Laptop* and *Bottle* datasets are the hardest ones in terms of depth sensor capabilities. Depth sensor has issues in retrieving depth information for IR reflective surfaces causing it to distort the images fully. Such surfaces in our case are computer screen and a plastic bottle. However, the *laptop* data has surprisingly managed to account for this error in depth map, albeit containing some distortions.

3.3.3. Reconstruction of Multiple Objects

As a proof of concept, we have performed the experiments to reconstruct multiple objects in the scene (see an example of results in Figure 15). By extracting the individual object masks and performing an object reconstruction individually we have managed to reconstruct the specific objects in the scene. However, we are unable to predict the object's relative position, rotation and scale in relation to camera space. For this reason, we have had to specify the object transformation in relation to camera and other scene objects manually to perform final scene render.

Figure 15. An example of reconstruction of multiple objects in the scene: segmented and classified RGB frame (**left**), depth frame (**middle**), and predicted depth mask (**right**).

4. Discussion and Concluding Remarks

4.1. Discussion

One of the main advantages of our suggested hybrid NN based method is that unlike other non-hybrid approaches, it is relatively easy to include additional objects into the dataset, due to the fact that you can train network branches separately. Unlike other approaches, we do not need to re-train the model with all the previous data as we do not risk losing any of the existing gradients due to network being skewed to the new data points. The modularity of the approach allows us to train the network reconstruction nodes per each object category independently. Additionally, this modularity allows for variance of the model per object class, meaning we can adjust complexity of the ANN depending on the difficulty of the object that is being reconstructed. Furthermore, we believe that our approach is a step forward to generic object reconstruction as we are capable of extracting multiple

objects from the same scene thanks to masking during classification step, which allows to send only the relevant objects depth information into the reconstruction node.

While our approach is capable of extracting the individual object instances and reconstructing them, additional research is required for full scene reconstruction. This feat requires finding the camera space matrices, paving the way for application in Extended Reality systems. One of the standing issues with our current approach in terms of reconstruction is that our ground-truths are perspective-invariant. This makes training the network slightly harder, additionally it may somewhat reduce the quality of the results due to network somewhat trying to adjust to observation angle, therefore making the IoU metric values lower, despite visually being feasible. Solving the perspective invariance may also be a partial solution to the homography [67,68] problem as our reconstructed object would already be rotated with respect to the camera space.

Additionally, the improvements on the dataset may be obtained by creating and incorporating a real-world dataset along with synthetic data for the depth encoding step. Thus, we can potentially improved results when using real depth sensors. Additional improvements to the network architecture may also be found by changing the complexity of the model [69]; pruning dead neurons [70]; using neuro-evolutionary and neuro-genetic algorithms to find a much more fitting solution [71]; enhancing the learning of the artificial neural networks by using metaheuristic control mechanism [72]; or using multiple frames from a video feed instead of the current single frame solution as a large number of depth frames from a single view may reveal some hidden features and improve recall rate [73]. Using multiple frames would allow for exploration of what improvements may be achieved with the use of recurrent neural networks (RNN) for they have shown to be capable of predicting sequential data [74–76]. Finally, using the RGB frames combined with depth frames for reconstruction can potentially add some missing features from the depth due to inherent noisiness of the sensor, therefore improving the recall rate [77,78].

Finally, we have compared the complexity of the proposed network model with the YOLOv3 network as well as with other popular network architectures. The results presented in Table 6 how that the proposed network model is only sightly more complex than YOLOv3 in terms of the number of model parameters and operation, but outperforms other network architectures in terms of operations.

Table 6. Comparison of neural network complexity by the number of parameters, number of operations and model size.

Network Model	No. of Parameters	No. of Operations	Model Size (MB)
YOLOv3 [47]	61.81 M	294.86 M	946
Proposed (extended YOLOv3)	67.45 M	305.61 M	1010
AlexNet [79]	60 M	16.04 G	217
GoogleNet [80]	7 M	16.04 G	40
ResNet152 [81]	60 M	11.3 G	230
VGC16 [82]	138 M	154.7 G	512.24
NIN [83]	7.6 M	11.06 G	29
SimpleNet [84]	5.4 M	652 M	20

4.2. Concluding Remarks

Our proposed hybrid artificial neural network modifications have allowed to improve the reconstruction results with respect to theYOLOv3 network results by 8.53% which allows for much more precise filling of occluded object sides and the reduction of noise during the process. Additionally, the reconstruction results are a lot more stable when compared to previous results. Furthermore, the addition of object segmentation masks and the individual object instance classification is a leap forward towards a general purpose scene reconstruction as opposed to single object reconstruction task due to the ability to mask out overlapping object instances and use only masked object area in the reconstruction process. While further research is needed in order to retrieve object orientation and

position with respect to camera space, we believe our method allows for a much broader application in comparison to previous research due to its focus on single object reconstruction.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bitzidou, M.; Chrysostomou, D.; Gasteratos, A. Multi-camera 3D Object Reconstruction for Industrial Automation. In *IFIP Advances in Information and Communication Technology Advances in Production Management Systems. Competitive Manufacturing for Innovative Products and Services*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 526–533, doi:10.1007/978-3-642-40352-1_66. [CrossRef]

2. Fanini, B.; Pagano, A.; Ferdani, D. A Novel Immersive VR Game Model for Recontextualization in Virtual Environments: The uVRModel. *Multimodal Technol. Interact.* **2018**, *2*, 20, doi:10.3390/mti2020020. [CrossRef]

3. Fast-Berglund, Å.; Gong, L.; Li, D. Testing and validating Extended Reality (xR) technologies in manufacturing. *Procedia Manuf.* **2018**, *25*, 31–38, doi:10.1016/j.promfg.2018.06.054. [CrossRef]

4. Liao, B.; Li, J.; Ju, Z.; Ouyang, G. Hand Gesture Recognition with Generalized Hough Transform and DC-CNN Using Realsense. In Proceedings of the 2018 Eighth International Conference on Information Science and Technology (ICIST), Cordoba, Spain, 30 June–6 July 2018; pp. 84–90, doi:10.1109/icist.2018.8426125. [CrossRef]

5. Vaitkevičius, A.; Taroza, M.; Blažauskas, T.; Damaševičius, R.; Maskeliūnas, R.; Woźniak, M. Recognition of American Sign Language Gestures in a Virtual Reality Using Leap Motion. *Appl. Sci.* **2019**, *9*, 445, doi:10.3390/app9030445. [CrossRef]

6. Zhang, J.; Shum, H.P.H.; McCay, K.; Ho, E.S.L. Prior-less 3D Human Shape Reconstruction with an Earth Mover's Distance Informed CNN. In *Motion, Interaction and Games on-MIG19*; ACM Press: Newcastle upon Tyne, UK, 2019; pp. 1–2, doi:10.1145/3359566.3364694. [CrossRef]

7. Chen, C.; Yang, B.; Song, S.; Tian, M.; Li, J.; Dai, W.; Fang, L. Calibrate Multiple Consumer RGB-D Cameras for Low-Cost and Efficient 3D Indoor Mapping. *Remote Sens.* **2018**, *10*, 328, doi:10.3390/rs10020328. [CrossRef]

8. Połap, D.; Kęsik, K.; Książek, K.; Woźniak, M. Obstacle Detection as a Safety Alert in Augmented Reality Models by the Use of Deep Learning Techniques. *Sensors* **2017**, *17*, 2803, doi:10.3390/s17122803. [CrossRef] [PubMed]

9. Jusas, V.; Birvinskas, D.; Gahramanov, E. Methods and Tools of Digital Triage in Forensic Context: Survey and Future Directions. *Symmetry* **2017**, *9*, 49, doi:10.3390/sym9040049. [CrossRef]

10. Wang, L.; Li, R.; Shi, H.; Sun, J.; Zhao, L.; Seah, H.; Quah, C.; Tandianus, B. Multi-Channel Convolutional Neural Network Based 3D Object Detection for Indoor Robot Environmental Perception. *Sensors* **2019**, *19*, 893, doi:10.3390/s19040893. [CrossRef]

11. Lan, G.; Luo, Z.; Hao, Q. Development of a virtual reality teleconference system using distributed depth sensors. In Proceedings of the 2016 2nd IEEE International Conference on Computer and Communications (ICCC), Cordoba, Spain, 30 June–6 July 2018; pp. 975–978, doi:10.1109/CompComm.2016.7924850. [CrossRef]

12. Wald, J.; Tateno, K.; Sturm, J.; Navab, N.; Tombari, F. Real-Time Fully Incremental Scene Understanding on Mobile Platforms. *IEEE Robot. Autom. Lett.* **2018**, *3*, 3402–3409, doi:10.1109/LRA.2018.2852782. [CrossRef]

13. Daudelin, J.; Campbell, M. An Adaptable, Probabilistic, Next-Best View Algorithm for Reconstruction of Unknown 3-D Objects. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1540–1547, doi:10.1109/LRA.2017.2660769. [CrossRef]

14. Fuentes-Pacheco, J.; Ascencio, J.R.; Rendón-Mancha, J.M. Visual simultaneous localization and mapping: A survey. *Artif. Intell. Rev.* **2012**, *43*, 55–81, doi:10.1007/s10462-012-9365-8. [CrossRef]

15. Zollhöfer, M.; Stotko, P.; Görlitz, A.; Theobalt, C.; Niessner, M.; Klein, R.; Kolb, A. State of the Art on 3D Reconstruction with RGB-D Cameras. *Comput. Graph. Forum* **2018**, *37*, 625–652. [CrossRef]

16. Kutulakos, K.N.; Seitz, S.M. A theory of shape by space carving. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Corfu, Greece, 20–25 September 1999; Volume 1, pp. 307–314, doi:10.1109/ICCV.1999.791235. [CrossRef]

17. Li, C.; Zia, M.Z.; Tran, Q.; Yu, X.; Hager, G.D.; Chandraker, M. Deep Supervision with Shape Concepts for Occlusion-Aware 3D Object Parsing. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 388–397, doi:10.1109/CVPR.2017.49. [CrossRef]

18. Yang, B.; Rosa, S.; Markham, A.; Trigoni, N.; Wen, H. Dense 3D Object Reconstruction from a Single Depth View. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 2820–2834, doi:10.1109/tpami.2018.2868195. [CrossRef]

19. Song, H.O.; Xiang, Y.; Jegelka, S.; Savarese, S. Deep Metric Learning via Lifted Structured Feature Embedding. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016, doi:10.1109/cvpr.2016.434. [CrossRef]

20. Chang, A.X.; Funkhouser, T.A.; Guibas, L.J.; Hanrahan, P.; Huang, Q.X.; Li, Z.; Savarese, S.; Savva, M.; Song, S.; Su, H.; et al. ShapeNet: An Information-Rich 3D Model Repository. *arXiv* **2015**, arXiv:1512.03012.

21. Choy, C.B.; Xu, D.; Gwak, J.; Chen, K.; Savarese, S. 3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction. In *Lecture Notes in Computer Science, Proceedings of the European conference on computer vision, Amsterdam, The Netherlands, 816 October 2016*; SpringerLink: Berlin/Heidelberg, Germany, 2016; pp. 628–644. [CrossRef]

22. Ma, T.; Kuang, P.; Tian, W. An improved recurrent neural networks for 3d object reconstruction. *Appl. Intell.* **2019**, *50*, 905–923, doi:10.1007/s10489-019-01523-3. [CrossRef]

23. Dhondse, A.; Kulkarni, S.; Khadilkar, K.; Kane, I.; Chavan, S.; Barhate, R. Generative Adversarial Networks as an Advancement in 2D to 3D Reconstruction Techniques. *Data Manag. Anal. Innov. Adv. Intell. Syst. Comput.* **2019**, 343–364, doi:10.1007/978-981-13-9364-8_25. [CrossRef]

24. Turhan, C.G.; Bilge, H.S. Fused voxel autoencoder for single image to 3D object reconstruction. *Electron. Lett.* **2020**, *56*, 134–137. [CrossRef]

25. Hermoza, R.; Sipiran, I. 3D Reconstruction of Incomplete Archaeological Objects Using a Generative Adversarial Network. In *Proceedings of Computer Graphics International 2018*; Association for Computing Machinery: New York, NY, USA, 2018. [CrossRef]

26. Elaksher, A.F. 3D object reconstruction from multiple views using neural networks. *Appl. Geomat.* **2013**, *5*, 193–201, doi:10.1007/s12518-013-0110-z. [CrossRef]

27. Espinal, J.; Ornelas, M.; Puga, H.J.; Carpio, J.M.; Munoz, J.A. 3D Object Reconstruction Using Structured Light and Neural Networks. In Proceedings of the 2010 IEEE Electronics, Robotics and Automotive Mechanics Conference, Cuernavaca, Mexico, 28 September–1 October 2010; pp. 74–79, doi:10.1109/cerma.2010.19. [CrossRef]

28. Kappler, D.; Bohg, J.; Schaal, S. Leveraging big data for grasp planning. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 4304–4311, doi:10.1109/ICRA.2015.7139793. [CrossRef]

29. Rivera, P.; Añazco, E.V.; Choi, M.; Kim, T. Trilateral convolutional neural network for 3D shape reconstruction of objects from a single depth view. *IET Image Process.* **2019**, *13*, 2457–2466. [CrossRef]

30. Fan, H.; Su, H.; Guibas, L. A Point Set Generation Network for 3D Object Reconstruction from a Single Image. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2463–2471, doi:10.1109/cvpr.2017.264. [CrossRef]

31. Hane, C.; Tulsiani, S.; Malik, J. Hierarchical Surface Prediction for 3D Object Reconstruction. In Proceedings of the 2017 International Conference on 3D Vision (3DV), Qingdao, Canada, 10–12 October 2017; pp. 5–11, doi:10.1109/3dv.2017.00054. [CrossRef]

32. Jiang, L.; Shi, S.; Qi, X.; Jia, J. GAL: Geometric Adversarial Loss for Single-View 3D-Object Reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*; SpringerLink: Berlin/Heidelberg, Germany, 2018; pp. 820–834. [CrossRef]

33. Zhang, Y.; Liu, Z.; Liu, T.; Peng, B.; Li, X. RealPoint3D: An Efficient Generation Network for 3D Object Reconstruction from a Single Image. *IEEE Access* **2019**, *7*, 57539–57549. [CrossRef]

34. Zhang, Y.; Huo, K.; Liu, Z.; Zang, Y.; Liu, Y.; Li, X.; Zhang, Q.; Wang, C. PGNet: A Part-based Generative Network for 3D object reconstruction. *Knowl.-Based Syst.* **2020**, 105574, doi:10.1016/j.knosys.2020.105574. [CrossRef]

35. Payne, B.R.; Lay, J.F.; Hitz, M.A. Automatic 3D object reconstruction from a single image. In Proceedings of the 2014 ACM Southeast Regional Conference on-ACM SE 14, Kennesaw, GA, USA, 28–29 March 2014; pp. 1–5, doi:10.1145/2638404.2638495. [CrossRef]

36. Li, D.; Shao, T.; Wu, H.; Zhou, K. Shape Completion from a Single RGBD Image. *IEEE Trans. Vis. Comput. Gr.* **2017**, *23*, 1809–1822, doi:10.1109/TVCG.2016.2553102. [CrossRef] [PubMed]

37. Zhang, Z. Microsoft Kinect Sensor and Its Effect. *IEEE Multimed.* **2012**, *19*, 4–10, doi:10.1109/mmul.2012.24. [CrossRef]

38. Keselman, L.; Woodfill, J.I.; Grunnet-Jepsen, A.; Bhowmik, A. Intel(R) RealSense(TM) Stereoscopic Depth Cameras. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017; pp. 1267–1276, doi:10.1109/cvprw.2017.167. [CrossRef]

39. Zhang, C. CuFusion2: Accurate and Denoised Volumetric 3D Object Reconstruction Using Depth Cameras. *IEEE Access* **2019**, *7*, 49882–49893. [CrossRef]

40. Khoshelham, K.; Elberink, S.O. Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications. *Sensors* **2012**, *12*, 1437–1454, doi:10.3390/s120201437. [CrossRef] [PubMed]

41. Carfagni, M.; Furferi, R.; Governi, L.; Servi, M.; Uccheddu, F.; Volpe, Y. On the Performance of the Intel SR300 Depth Camera: Metrological and Critical Characterization. *IEEE Sens. J.* **2017**, *17*, 4508–4519, doi:10.1109/JSEN.2017.2703829. [CrossRef]

42. Hisatomi, K.; Kano, M.; Ikeya, K.; Katayama, M.; Mishina, T.; Iwadate, Y.; Aizawa, K. Depth Estimation Using an Infrared Dot Projector and an Infrared Color Stereo Camera. *IEEE Trans. Circuits Syst. Video Technol.* **2017**, *27*, 2086–2097, doi:10.1109/TCSVT.2016.2555678. [CrossRef]

43. Du, Y.; Fu, Y.; Wang, L. Representation Learning of Temporal Dynamics for Skeleton-Based Action Recognition. *IEEE Trans. Image Process.* **2016**, *25*, 3010–3022, doi:10.1109/TIP.2016.2552404. [CrossRef]

44. Lin, T.; Goyal, P.; Girshick, R.B.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.

45. Fu, C.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A.C. DSSD: Deconvolutional Single Shot Detector. *arXiv* **2017**, arXiv:1701.06659.

46. Ren, S.; He, K.; Girshick, R.B.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*; Neural Information Processing Systems Foundation, Inc.: La Jolla, CA, USA, 2015.

47. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.

48. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.

49. Seyfioğlu, M.S.; Özbayoğlu, A.M.; Gürbüz, S.Z. Deep convolutional autoencoder for radar-based classification of similar aided and unaided human activities. *IEEE Trans. Aerosp. Electron. Syst.* **2018**, *54*, 1709–1723, doi:10.1109/TAES.2018.2799758. [CrossRef]

50. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359, doi:10.1109/TKDE.2009.191. [CrossRef]

51. Loshchilov, I.; Hutter, F. SGDR: Stochastic Gradient Descent with Restarts. *arXiv* **2016**, arXiv:1608.03983.

52. Savitzky, A.; Golay, M.J.E. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Anal. Chem.* **1964**, *36*, 1627–1639, doi:10.1021/ac60214a047. [CrossRef]

53. Murugan, P.; Durairaj, S. Regularization and Optimization strategies in Deep Convolutional Neural Network. *arXiv* **2017**, arXiv:1712.04711.

54. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.E.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. *arXiv* **2014**, arXiv:1409.4842.

55. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2015**, arXiv:1512.03385.

56. Kim, Y.; Shin, J.; Park, H.; Paik, J. Real-Time Visual Tracking with Variational Structure Attention Network. *Sensors* **2019**, *19*, 4904, doi:10.3390/s19224904. [CrossRef]

57. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 27th International Conference on International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010; pp. 807–814.

58. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. *Microsoft COCO: Common Objects in Context*; Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., Eds.; Computer Vision–ECCV 2014; Springer: Cham, Switzerland, 2014; pp. 740–755.

59. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338, doi:10.1007/s11263-009-0275-4. [CrossRef]

60. Dai, A.; Chang, A.X.; Savva, M.; Halber, M.; Funkhouser, T.A.; Nießner, M. ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes. *arXiv* **2017**, 1702.04405.

61. Flaischlen, S.; Wehinger, G.D. Synthetic Packed-Bed Generation for CFD Simulations: Blender vs. STAR-CCM+. *ChemEngineering* **2019**, *3*, 52, doi:10.3390/chemengineering3020052. [CrossRef]

62. Kainz, F.; Bogart, R.R.; Hess, D.K. *The OpenEXR Image File Format*; ACM Press: New York, NY, USA, 2004.

63. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. *Lect. Not. Comput. Sci.* **2016**, 21–37, doi:10.1007/978-3-319-46448-0_2. [CrossRef]

64. Pantaleoni, J. VoxelPipe: A programmable pipeline for 3D voxelization. In Proceedings of the ACM SIGGRAPH Symposium on High Performance Graphics-HPG, Vancouver, BC, Canada, 5–7 August 2011; pp. 99–106, doi:10.1145/2018323.2018339. [CrossRef]

65. Baldwin, D.; Weber, M. Fast Ray-Triangle Intersections by Coordinate Transformation. *J. Comput. Gr. Tech. (JCGT)* **2016**, *5*, 39–49.

66. Kulikajevas, A.; Maskeliūnas, R.; Damaševičius, R.; Misra, S. Reconstruction of 3D Object Shape Using Hybrid Modular Neural Network Architecture Trained on 3D Models from ShapeNetCore Dataset. *Sensors* **2019**, *19*, 1553, doi:10.3390/s19071553. [CrossRef] [PubMed]

67. Cui, Z.; Jiang, K.; Wang, T. Unsupervised Moving Object Segmentation from Stationary or Moving Camera Based on Multi-frame Homography Constraints. *Sensors* **2019**, *19*, 4344, doi:10.3390/s19194344. [CrossRef]

68. Park, K.W.; Shim, Y.J.; Lee, M.j.; Ahn, H. Multi-Frame Based Homography Estimation for Video Stitching in Static Camera Environments. *Sensors* **2019**, *20*, 92, doi:10.3390/s20010092. [CrossRef]

69. Huang, G.B.; Saratchandran, P.; Sundararajan, N. A Generalized Growing and Pruning RBF (GGAP-RBF) Neural Network for Function Approximation. *IEEE Trans. Neural Netw.* **2005**, *16*, 57–67, doi:10.1109/tnn.2004.836241. [CrossRef]

70. Wang, J.; Xu, C.; Yang, X.; Zurada, J.M. A Novel Pruning Algorithm for Smoothing Feedforward Neural Networks Based on Group Lasso Method. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 2012–2024, doi:10.1109/TNNLS.2017.2748585. [CrossRef]

71. Arifovic, J.; Gençay, R. Using genetic algorithms to select architecture of a feedforward artificial neural network. *Phys. A Stat. Mech. Appl.* **2001**, *289*, 574–594, doi:10.1016/S0378-4371(00)00479-9. [CrossRef]

72. Połap, D.; Kęsik, K.; Woźniak, M.; Damaševičius, R. Parallel Technique for the Metaheuristic Algorithms Using Devoted Local Search and Manipulating the Solutions Space. *Appl. Sci.* **2018**, *8*, 293, doi:10.3390/app8020293. [CrossRef]

73. Izadi, S.; Kim, D.; Hilliges, O.; Molyneaux, D.; Newcombe, R.; Kohli, P.; Shotton, J.; Hodges, S.; Freeman, D.; Davison, A.; et al. Kinectfusion: real-time 3D reconstruction and interaction using a moving depth camera. In Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, Santa Barbara, CA, USA, 16–19 October 2011; pp. 559–568.

74. Wang, J.; Zhang, L.; Guo, Q.; Yi, Z. Recurrent Neural Networks With Auxiliary Memory Units. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 1652–1661, doi:10.1109/TNNLS.2017.2677968. [CrossRef] [PubMed]

75. Hawkins, J.; Boden, M. The applicability of recurrent neural networks for biological sequence analysis. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **2005**, *2*, 243–253, doi:10.1109/TCBB.2005.44. [CrossRef] [PubMed]

76. Wang, Y.; Liao, W.; Chang, Y. Gated Recurrent Unit Network-Based Short-Term Photovoltaic Forecasting. *Energies* **2018**, *11*, 2163, doi:10.3390/en11082163. [CrossRef]

77. Liu, Z.; Zhao, C.; Wu, X.; Chen, W. An Effective 3D Shape Descriptor for Object Recognition with RGB-D Sensors. *Sensors* **2017**, *17*, 451, doi:10.3390/s17030451. [CrossRef] [PubMed]

78. Hsu, G.J.; Liu, Y.; Peng, H.; Wu, P. RGB-D-Based Face Reconstruction and Recognition. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 2110–2118, doi:10.1109/TIFS.2014.2361028. [CrossRef]

79. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90, doi:10.1145/3065386. [CrossRef]

80. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015, doi:10.1109/cvpr.2015.7298594. [CrossRef]

81. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016, doi:10.1109/cvpr.2016.90. [CrossRef]

82. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.

83. Lin, M.; Chen, Q.; Yan, S. Network In Network. *arXiv* **2013**, arxiv:1312.4400.

84. Hasanpour, S.H.; Rouhani, M.; Fayyaz, M.; Sabokrou, M. Lets keep it simple, Using simple architectures to outperform deeper and more complex architectures. *arXiv* **2016**, arxiv:1608.06037.

Article

Exoskeleton Hand Control by Fractional Order Models

Mircea Ivanescu [1,*]**, Nirvana Popescu** [2]**, Decebal Popescu** [2]**, Asma Channa** [2]
and Marian Poboroniuc [3]

[1] Department of Mechatronics, University of Craiova, 200585 Craiova, Romania
[2] Department of Computer Science, University Politehnica of Bucharest, 060042 Bucharest, Romania;
    nirvana.popescu@cs.pub.ro (N.P.); decebal.popescu@cs.pub.ro (D.P.);
    asma.channa@admin.muet.edu.pk (A.C.)
[3] Department of Electrical Engineering, Technical University of Iasi, 700050 Iași, Romania; mpobor@tuiasi.ro
[*] Correspondence: ivanescu@robotics.ucv.ro

Abstract: This paper deals with the fractional order control for the complex systems, hand exoskeleton and sensors, that monitor and control the human behavior. The control laws based on physical significance variables, for fractional order models, with delays or without delays, are proposed and discussed. Lyapunov techniques and the methods that derive from Yakubovici-Kalman-Popov lemma are used and the frequency criterions that ensure asymptotic stability of the closed loop system are inferred. An observer control is proposed for the complex models, exoskeleton and sensors. The asymptotic stability of the system, exoskeleton hand-observer, is studied for sector control laws. Numerical simulations for an intelligent haptic robot-glove are presented. Several examples regarding these models, with delays or without delays, by using sector control laws or an observer control, are analyzed. The experimental platform is presented.

Keywords: exoskeleton hand; fractional order model; control

1. Introduction

The IHRG is an intelligent haptic robotic glove system for the rehabilitation of patients that have a diagnosis of a cerebrovascular accident. This system is created by a thin textile in order to have a comfortable environment for grasping exercises. An exoskeleton architecture ensures the mechanical compliance of human fingers. The driving and skin sensor system is designed to determine comfortable and stable grasping function. This paper analyzes the dynamics of an exoskeleton hand using fractional order operators and proposes control solutions.

The number of applications in the system modelling, where the fractional order calculus (FOC) is used, has increased significantly in the last few decades. Many authors proved that non-integer order integrals and derivatives are suitable for analyses of the properties of various materials. Recent achievements in the interpretation of FOC operators allowed to apply FOC for processes that are better described by fractional order models (FOM) rather than integer order models (IOM). The role of these models in soft matter physics and viscoelastic behavior, in the theory of complex materials, its quality to include effects with non-conservative forces and power-law phenomena suggest to describe the complexity of human dynamics using FOM operators [1].

The idea is supported by the evidence of s^β dynamics in muscles and joint tissues throughout human musculo-skeletal system [2,3]. Interaction and dependence between biological systems and associated mechanical components was analyzed in [4–6]. Fractional order models for metal polymer composite was discussed in [7]. In [8,9] viscoelastic properties for a large variety of biological entities were studied. A class of sensors based on fractional calculus was presented in [10]. Optimal techniques

using fractional calculus for sensor networks were discussed in [11]. A fractional model to capture muscular dynamics in the movement process was proposed in [12]. In [13–15] a class of neural, muscular, and vascular processes were studied to minimize sensor placement. Recently, there is a great deal of interest in so-called rehabilitation robotics, a branch of the areas of robotics and mechatronics that addresses the study of complex robotic systems aiming to restore human functions for those who suffer major trauma as a result of strokes and cerebrovascular accidents. Robotic therapy is a promising form of neurorehabilitation that can be delivered in more intensive regimens compared to conventional therapy [16]. The complexity of these systems associated with a specified class of sensors is well described by fractional order differential equations [17,18]. The methods for the analysis and design of fractional order operators can be found in [19]. Control and stability of FOM was investigated by techniques Lyapunov in [20,21]. In particular, the authors of [22,23] discuss the stability properties of solutions of nonlinear Caputo fractional differential equations. The exponential stability of nonlinear FOM using the Lyapunov method was analyzed in [24,25]. Other control problems for a class of FOMs with delay were rigorously investigated in [26,27]. Reference [28] proposed an observer for a class of linear and nonlinear FOM using Lyapunov methods. To our knowledge, this paper is the first paper to assess FOM for systems and sensors that monitor or control human behavior. The exoskeleton architecture, which ensures a mechanical compliance of human fingers, including the driving and sensor system, determines comfortable and stable grasping functions.

The dynamics of the whole system, exoskeleton hand (EXHAND), and the sensors can be accurately described by FOM operators. A class of 3D FOM bending sensors is analyzed. The control laws based on physical significance variables, for linear and delay FOM or IOM systems, are proposed and discussed. The sector control laws for linear FOM, with delays or without delays are studied. Lyapunov techniques and the methods that derive from Yakubovici-Kalman-Popov Lemma are used, and the frequency criteria that ensure asymptotic stability of the physical significance variable closed loop system are inferred. An observer control is proposed for the complex models, EXHANDs and sensors. The asymptotic stability of the whole system, the observer-system, is studied according to sector control laws. Frequency criteria and conditions for asymptotic stability are determined. Numerical simulations for the intelligent haptic robot-glove (IHRG) are presented. Several examples regarding the FOM or IOM systems, with delays or without delays, by using sector control laws or an observer control, are analyzed. The IHRG experimental platform is then discussed.

The paper is structured as follows: Section 2.1 discusses FOM sensors and FOM systems implemented in EXHAND; Section 2.2 presents the control systems; Section 3.1 analyzes IHRG numerical simulations; and Section 3.2 presents the IHRG platform. Section 4 provides concluding remarks and discussions.

2. Methods

2.1. Fractional Order Models

Notations:

1. The fractional order integral of order β is the Riemann-Liouville fractional integral:

$$I^\beta = \frac{1}{\Gamma(\beta)} \int\limits_0^t f(\theta)(t-\theta)^{\beta-1} d\theta$$

2. The Caputo derivative of order *β*, $0 < \beta < 1$ is:

$$D^\beta f(t) = \frac{1}{\Gamma(\beta-1)} \int\limits_0^t \dot{f}(\theta)(t-\theta)^{-\beta} d\theta$$

where β is the fractional order exponent and $\Gamma(\beta)$ is the gamma (Euler's) function.

(a) 3D curvature sensors described by FOM

Bending sensors represent a class of sensors with large applications in the control of complex systems. They convert changes in bend to an electrical parameter variation. Conventional bending sensors handle cases in which bending is produced in the 2D plane. The most common are the resistive sensors, described by IOM operators of order 0. For a special class of systems, such as the hyper-redundant robots [29] where bending is produced in a 3D space (Figure 1a), a special class of bending sensors defined by FOM operators (Figure 1b) is used.

(a)

(b)

(c)
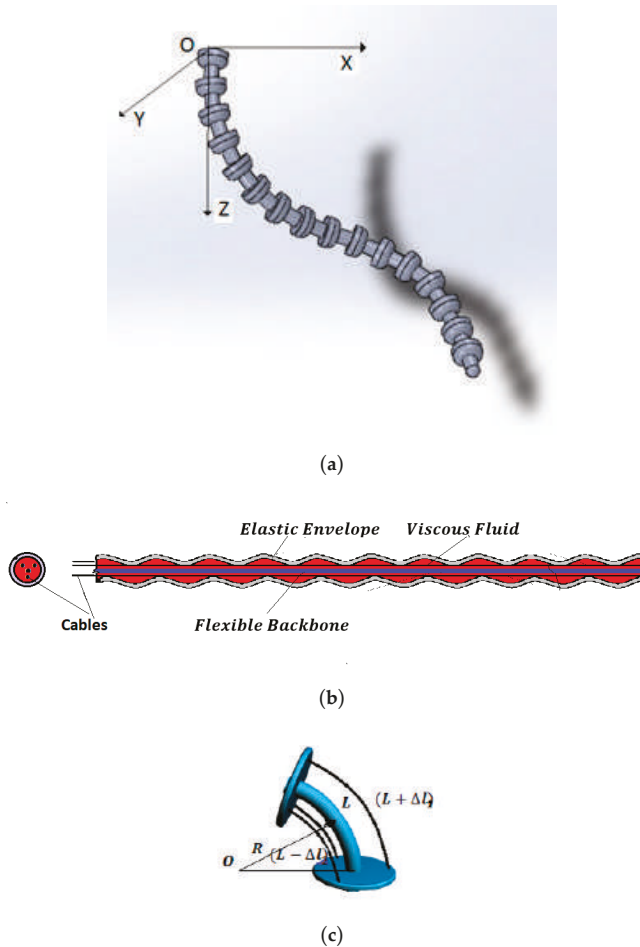
Figure 1. (**a**) 3D hyper-redundant robotic arm. (**b**) 3D FOM curvature sensor. (**c**) Measurement technique.

The architecture of this sensor consists of a main viscoelastic component determined by a long flexible backbone wrapped in a cylindrical elastic envelope. Three antagonist cables are implemented at the periphery of the system. In static behavior, curvature κ is obtained by the differential measurement of the cable lengths, Figure 1c [30]:

$$\kappa = F(\Delta L_1, \Delta L_2, \ \Delta L_3) \tag{1}$$

The dynamic behavior is inferred considering constant curvature along the length. Employing the same technique as that developed in [17] yields (Figure 2):

$$\ddot{\kappa}(t) = -c_{vs}b_s D^\beta \kappa(t) - k_s c_s \kappa(t) + k_M \mathcal{M}(t) \tag{2}$$

where c_{vs}, k_s are distributed viscous and elastic coefficient, assumed uniform distributed along the length, b_s, c_s are material parameters and $\mathcal{M}$ is the moment that determines the bending. The transfer function is derived from Equation (2) as:

$$H_s(s) = \frac{\kappa(s)}{\mathcal{M}(s)} = \frac{k_M}{s^2 + c_{vs}b_s s^\beta + k_s c_s} \tag{3}$$
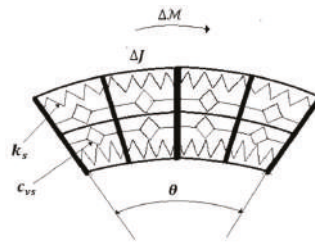
Figure 2. Technological equivalent model of the 3D curvature sensor.

That corresponds to an order 2 FOM operator.

(b) FOM systems

A large class of systems that monitors or controls the human behavior is well described by the FOM operators. Figure 3 shows the control system of an intelligent haptic robot-glove (IHRG) for the rehabilitation of patients that have a diagnosis of a cerebrovascular accident. The IHRG is a medical device that acts in parallel to a hand in order to compensate for lost functions [16]. The exoskeleton architecture that ensures the mechanical compliance of human fingers for the driving system determines comfortable and stable grasping functions.
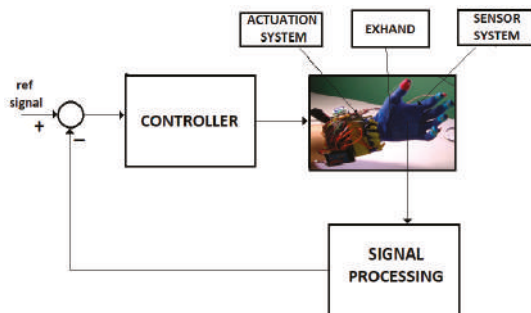
Figure 3. IHRG system.

The dynamics of the system (EXHAND) can be accurately described by FOM operators,

$$D^\beta z(t) = A_0 z(t) + f(z) + bu(t), \; t \in [0, \, T] \tag{4}$$

where z is the state vector $z = [z_1, z_2, \ldots, z_n]^T$ that defines the motion parameters, β is the fractional order exponent, and A_0, b are $(n \times n)$, $(n \times 1)$ constant matrices. In a FOM operator of EXHAND, the vector components are defined as

$$D^\beta z_1 = z_2, \ D^\beta z_2 = z_3, \ldots \tag{5}$$

The nonlinear term $f(z)$ is determined by the gravitational components and satisfies the inequality

$$\|f(z)\| < \eta \|z\| \tag{6}$$

The output of the system is generated by the bending sensors. Provided that the bending of the phalange musculoskeletal system is in 2-D plane, in this project, we used an Arduino Flex Resistive Sensor network. This sensor operates as a zeroth IOM operator,

$$y(t) = c^T z(t) \tag{7}$$

where c is a constant $(n \times 1)$ vector.

A new model can be inferred if the delay time constant, associated with the neuro-muscular system, the driving system and the processing time, is introduced,

$$D^\beta z(t) = A_0 z(t) + A_1 z(t - \tau) + f(z) + bu(t), \ t \in [0, \ T] \tag{8}$$

The initial conditions are defined by

$$z(t) = \varphi(t), \ \ t \in [-\tau, \ 0]$$

where the function φ is associated to initial states.

For Equations (4)–(8) we used the control system from Figure 4 with a FOM operator for the EXHAND and a IOM operator for the sensor system.
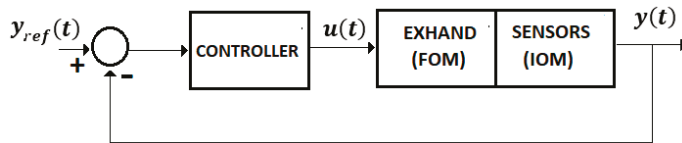
Figure 4. Control system.

2.2. Control Systems

Mathematical Preliminaries

Lemma [19]**.** *For any symmetric matrix $P \in R^{n \times n}$, the following inequality holds:*

$$\lambda_{\min(P)} I^* \ \leq P \ \leq \ \lambda_{\max(P)} I^* \tag{9}$$

where $\lambda_{min(P)}$, $\lambda_{max(P)}$ denote the minimum and maximum eigenvalue, respectively, of matrix P and I^* is the unit matrix.

Theorem 1 ([21,22,24])**.** *The system $D^\beta z(t) = A z(t)$, $0 < \beta < 1$, is asymptotically stable if*

$$\left| Arg(eig(A)) \right| > \beta \frac{\pi}{2} \tag{10}$$

Theorem 2 ([22–24]). *The system* $D^\beta z(t) = f(z(t))$, $z(t_0) = z_0$ *is asymptotically stable if there exists a continuously differentiable function* $V(t, z)$ *that satisfies*

$$\alpha_1\|z\|^2 \leq V(t, z(t)) \leq \alpha_2\|z\|^2 \tag{11}$$

$$D^\beta V(t, z(t)) \leq -\alpha_3\|z\|^2 \tag{12}$$

where α_1, α_2, α_3 *are positive constants,*

$$0 < \beta < 1.$$

2.2.1. Control for the EXHAND Without Delays

Consider the system from Figure 4 defined by Equations (4)–(7) without a delay time. Assume a control law.

$$u(t) = -k\left(y(t) - y_{ref}(t)\right) \tag{13}$$

where the control gain k verifies the condition

$$k\sigma \leq 1 \tag{14}$$

where σ is a positive constant (for simplicity, $y_{ref}(t) = 0$).

Remark 1. *For the EXHAND model with the state variables defined by Equation (5) and the output vector* $c = [c_1. c_2, c_3, c_4]^T$, *the control law (Equation (10)) becomes a* PD^β *law*

$$u(t) = -k\left(c_1 z_1 + c_2 D^\beta z_1 + c_3 z_3 + c_4 D^\beta z_3\right)$$

or

$$u(t) = -k_1 z_1 - D^\beta z_1 - k_3 z_3 - k_4 D^\beta z_3 \tag{15}$$

If c is selected as $c = [c_1. 0, c_3, 0]^T$, *the control becomes a PD law*

$$u(t) = -k_1 z_1 - k_3 \dot{z}_1 \tag{16}$$

Control System 1. The system (Equations (4)–(7)) with the controller defined by Equations (13) and (14) is asymptotically stable if:

The matrix $A^* = A - R$ is Hurwitz stable where $R = R^T > 0$.

$$Re\left(c^T(j\omega I - (A - R))^{-1} b\right) \geq -\sqrt{\sigma} \tag{17}$$

$$\varrho > \rho_{PR} + 2\lambda_{\max(P)}\eta \tag{18}$$

where $\varrho = \left\|\left(q + k\sqrt{\sigma}d\right)\left(q + k\sqrt{\sigma}d\right)^T\right\|$, $\rho_{PR} = 2PR$ and $Q = qq^T$, P are solutions of the Lyapunov equation [20,29].

Proof. Consider the Lyapunov function

$$V(z) = z^T P z \tag{19}$$

where $P = P^T > 0$. The first asymptotic stability conditions (Equation (8)) are verified for $\alpha_1 = \lambda_{\min (P)}$, $\alpha_2 = \lambda_{\max (P)}$, [21] (Theorem 1), where $\lambda_{\min (P)}$, $\lambda_{\max (P)}$ denote the minimum and maximum eigenvalues of P. The fractional derivative of Equation (14) will be [22,24],

$$D^\beta V(z) \le \left(D^\beta z^T\right) Pz + z^T P\left(D^\beta z\right) \tag{20}$$

By substituting Equation (4) into Equation (20), one derives

$$D^\beta V(z) \le z^T\left(A^T P + PA\right)z + 2z^T Pbu + 2z^T Pf \tag{21}$$

Employing the condition (a) yields

$$D^\beta V(z) \le z^T\left((A-R)^T P + P(A-R)\right)z + 2z^T Pbu + 2z^T PRz + 2z^T Pf \tag{22}$$

Considering Equations (13) and (14), this inequality becomes

$$D^\beta V(z) \le z^T\left((A-R)^T P + P(A-R)\right)z + 2z^T\left(Pb - \frac{1}{2}c\right)u - \sigma u^2 + 2z^T PRz + 2z^T Pf \tag{23}$$

By employing the condition (Equation (15)) and Yakubovici-Kalman-Popov (YKP) Lemma [31], results

$$z^T\left((A-R)^T P + P(A-R)\right)z = -qq^T \tag{24}$$

$$Pb - \frac{1}{2}c = \sqrt{\sigma}q \tag{25}$$

Now, considering the control law (Equation (13)), it follows that

$$D^\beta V(z) \le -z^T\left(q + k\sqrt{\sigma}d\right)\left(q + k\sqrt{\sigma}d\right)^T z + \rho_{PR} z^T z + 2\lambda_{\max(P)}\eta \tag{26}$$

or, by Equation (16),

$$D^\beta V(z) \le -\alpha_3 \|z\|^2 \tag{27}$$

where

$$\alpha_3 = \varrho - \rho_{PR.} \tag{28}$$

□

2.2.2. Control for the EXHAND with Delay

Control System 2. The system described by Equation (8) with the control law defined by Equation (13) is asymptotically stable if:

1. $A_0^* = (A_0 - R)$ is Hurwitz stable, where $R = R^T > 0$

$$Re\left(c^T\left(j\omega I - A_0^*\right)^{-1}b\right) \ge -\sqrt{\sigma} \tag{29}$$

$$\varrho - \left(2\eta\lambda_{\max(P_1)} + \frac{1}{2}\lambda_{\max(D)} + 2\lambda_{\max(P_1 R)} + \lambda_{\max(P_2)}\right) > 0 \tag{30}$$

$$\lambda_{\min(P_2)} - \frac{1}{2}\lambda_{\max(D)} > 0 \tag{31}$$

where $Q = qq^T$, P_1 are solutions of the Lyapunov equations and

$$\varrho = \left\|\left(q + k\sqrt{\sigma}c_1\right)\left(q + k\sqrt{\sigma}c_1\right)^T\right\|$$

$$D = \left(A_1^T P_1 + P_1 A_1\right)$$

Proof. Consider the following Lyapunov function:

$$V(z(t)) = I^{1-\beta}\left(z^T(t)\mathbf{P_1}z(t)\right) + \int_{t-\tau}^{t} z^T(\theta)\mathbf{P_2}z(\theta)d\theta \tag{32}$$

where $\mathbf{P_1}$, $\mathbf{P_2}$ are $(n \times n)$ are positive definite and symmetrical matrices, $P_1 > 0$, $P_2 > 0$, $P_1^T = P_1$, $P_2^T = P_2$. $V(z)$ satisfies the condition (Equation (11)) of Theorem 2.

$$V(z) \geq \lambda_{\min(P_1)}\left\|z(t)\right\|^2 \tag{33}$$

$$V(z) \leq \lambda_{\max(P_1)}\left\|z(t)\right\|^2 + \lambda_{\max(P_2)}\int_{t-\tau}^{t}\left\|z(\theta)\right\|^2 d\theta \leq M\left\|z(t)\right\|^2 \tag{34}$$

The derivative $D^\beta V(z)$ is computed from:

$$D^\beta V(z) = I^{1-\beta}\dot{V}(z) \tag{35}$$

where $I^{1-\beta}$ is the Riemann-Liouville fractional integral of the order $(1-\beta)$. The derivative $\dot{V}(z)$ is evaluated from Equation (32)

$$\dot{V}(z) = D^\beta\left(\left(z^T(t)\mathbf{P_1}z(t)\right) + \frac{d}{dt}\int_{t-\tau}^{t} z^T(\theta)\mathbf{P_2}z(\theta)d\theta\right) \tag{36}$$

which leads to the inequality

$$\dot{V}(z) \leq (D^\beta z^T(t))\mathbf{P_1}z(t) + z^T(t)\mathbf{P_1}\left(D^\beta z(t)\right) + z^T(t)\mathbf{P_2}z(t) - z^T(t-\tau)\mathbf{P_2}z(t-\tau) \tag{37}$$

By evaluating Equation (37) along of solutions of Equation (8) it turns out that

$$\dot{V}(z) \leq z^T(t)\left(A_0^{*T}P_1 + P_1 A_0^*\right)z(t) + z^T(t)\left(A_1^T P_1 + P_1 A_1\right)z(t-\tau) + \mathbf{2z^T(t)P_1 bu(t)} + \mathbf{2z^T Pf}$$
$$+ 2z^T(t)\mathbf{P_1}Rz(t)) + z^T(t)\mathbf{P_2}z(t) - z^T(t-\tau)\mathbf{P_2}z(t-\tau) \tag{38}$$

By applying the control law Equation (28), it yields

$$\dot{V}(z) \leq z^T(t)\left(A_0^{*T}P_1 + P_1 A_0^*\right)z(t) + 2z^T(t)(P_1 b - \tfrac{\varsigma}{2})u(t) - \sigma u^2(t) + 2z^T Pf$$
$$+ z^T(t)\left(A_1^T P_1 + P_1 A_1\right)z(t-\tau) + 2z^T(t)\mathbf{P_1}Rz(t) + z^T(t)\mathbf{P_2}z(t)$$
$$- z^T(t-\tau)\mathbf{P_2}z(t-\tau) \tag{39}$$

The following inequality will be used [23]

$$\left\|z^T(t)Dz(t-\tau)\right\| \leq \|z(t)\|\,\|D\|\,\|z(t-\tau)\| \leq \lambda_{\max(D)}\left(\frac{\|z(t)\|^2}{2} + \frac{\|z(t-\tau)\|^2}{2}\right) \tag{40}$$

Additionally, considering the YKP Lemma as in the previous Control System, yields

$$z^T\left(\left(A^* - R\right)^T P_1 + P_1(A^* - R)\right)z = -qq^T \tag{41}$$

$$P_1 b - \frac{1}{2}c = \sqrt{\sigma}q \tag{42}$$

Substituting this result into Equation (39), considering the inequalities of Equations (6) and (40), one derives that

$$\begin{aligned}
\dot{V}(z) \leq \ & -z^T(t)\left(q + k\sqrt{\sigma}d\right)\left(q + k\sqrt{\sigma}d\right)^T z(t) \\
& + \left(2\eta\lambda_{\max(P_1)} + \tfrac{1}{2}\lambda_{\max(D)} + 2\lambda_{\max(P_1 R)} + \lambda_{\max(P_2)}\right)\|z(t)\|^2 \\
& - \left(\lambda_{\min(P_2)} - \tfrac{1}{2}\lambda_{\max(D)}\right)\|z(t-\tau)\|^2
\end{aligned} \tag{43}$$

Employing Equations (30) and (31), yields

$$\dot{V}(z) \leq -\left(\varrho - \left(2\eta\lambda_{\max(P_1)} + \frac{1}{2}\lambda_{\max(D)} + 2\lambda_{\max(P_1 R)} + \lambda_{\max(P_2)}\right)\right)\|z(t)\|^2 \tag{44}$$

Denoted by

$$\alpha_3 = \varrho - \left(2\eta\lambda_{\max(P_1)} + \frac{1}{2}\lambda_{\max(D)} + 2\lambda_{\max(P_1 R)} + \lambda_{\max(P_2)}\right)$$

and from Equation (35) results

$$D^\beta V(z) \leq -\alpha_3 z(t)^2. \tag{45}$$

□

2.2.3. Control System with Observer for the EXHAND with Delay

Consider the linearized model of Equation (8) rewritten as

$$D^\beta z(t) = A_L z(t) + A_1 z(t-\tau) + bu(t), \ t \in [0, T] \tag{46}$$

$$y(t) = c^T z(t) \tag{47}$$

where the nonlinear term was approximated by

$$f(z) \cong \left.\frac{\partial f(z)}{\partial z}\right|_{z_0} \Delta z = \left.\frac{\partial f(z)}{\partial z}\right|_{z_0}(z_1 - z_0) = \left.\frac{\partial f(z)}{\partial z}\right|_{z_0=0} z_1 \tag{48}$$

Remark 2. *For the EXHAND model, the pair* (A_L, b) *is controllable and the pair* (C, A_L) *is observable.*

Consider the system defined by Equation (46). The following observer is proposed:

$$D^\beta \hat{z}(t) = A_L \hat{z}(t) + A_1 \hat{z}(t-\tau) + bu(t) + L_1(y_1(t) - \hat{y}_1(t)) + L_2(y_2(t-\tau) - \hat{y}_2(t-\tau)) \tag{49}$$

$$\hat{z}(t) = \hat{\varphi}(t), t \in [-\tau, 0] \tag{50}$$

$$\hat{y}(t) = [\hat{y}_1(t)\ \hat{y}_2(t)]^T, \quad \hat{y}_1(t) = c_1^T \hat{z}(t), \quad \hat{y}_2(t) = c_2^T \hat{z}(t-\tau) \tag{51}$$

where $\hat{z} \in R^n$ is the observer state, $\hat{y} \in R^2$ is the estimated output and L_1, L_2 are $(n \times 1)$ observability vectors. The observer error is

$$\Delta z(t) = z(t) - \hat{z}(t) \tag{52}$$

defined by the following equation:

$$D^\beta(\Delta z(t)) = \left(A_L - L_1 c_1^T\right)\Delta z(t) + \left(A_1 - L_2 c_2^T\right)\Delta z(t-\tau) \tag{53}$$

$$\Delta z(t) = \Delta\varphi, \quad t \in [-\tau, 0] \tag{54}$$

Consider the control law

$$u(t) = u_1(t) + u_2(t) = -k_1 c_1^T \hat{z}(t) - k_2 c_2^T \hat{z}(t - \tau) \tag{55}$$

The global state $(\hat{z}, z - \hat{z}) = (\hat{z}, \Delta z)$ is considered for the system "EXHAND-observer".

Control System 3. The whole system, "EXHAND-observer", Equations (46), (47), and (49)–(51) (Figure 5) with the control law (55), is asymptotically stable if
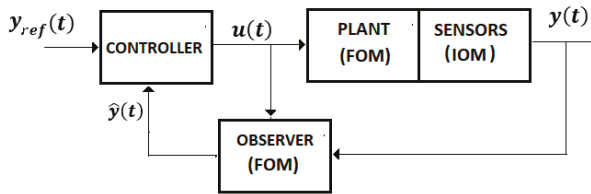
Figure 5. Control system with observer.

$A_L^* = (A_L - R)$ is Hurwitz stable where $R = R^T > 0$.

$$k_1 \sigma \leq 1, \quad \sigma > 0 \tag{56}$$

$$Re\left(c_1^T (j\omega I - A_L^*)^{-1} b\right) \geq -\sqrt{\sigma} \mathbf{q} \tag{57}$$

$$-\varrho + \lambda_{\max(D_1)} + \lambda_{\max(D_2)} + 2\lambda_{\max(P_1 R)} + \lambda_{\max(P_3)} < 0 \tag{58}$$

$$\lambda_{\max(E_1)} - \frac{1}{2}\lambda_{\max(D_1)} - \lambda_{\max(P_2)} > 0 \tag{59}$$

$$\lambda_{\max(P_2)} - \frac{1}{2}\lambda_{\max(E_2)} > 0 \tag{60}$$

$$\lambda_{\max(P_3)} - \lambda_{\max(D_2)} > 0 \tag{61}$$

where ϱ is defined by Equation (50) and

$$D_2 = (Lc_1^T)^T P_1 + P_1 Lc_1^T; \quad D_1 = (A_1^T P_1 + P_1 A_1) \tag{62}$$

$$E_1 = A_0^* - L_1 c_1^T; \quad E_2 = A_1 - L_2 c_2^T \tag{63}$$

Proof. Consider the Lyapunov function

$$V(\hat{z}, \Delta z) = I^{1-\beta}\left(\hat{z}^T(t) P_1 \hat{z}(t) + \frac{1}{2}\Delta z^T(t)\Delta z(t)\right) + \int\limits_{t-\tau}^{t}\left(\Delta z^T(\theta) P_2 \Delta z(\theta) + z^T(\theta) P_3 z(\theta)\right)d\theta \tag{64}$$

where P_1, P_2, P_3 are $(n \times n)$ are positive definite and symmetrical matrices. $V(z, \Delta z)$ satisfies the first condition (Equation (11)) of Theorem 2.

Applying the same procedures as in the previous control system, yields

$$
\begin{aligned}
\dot{V}(\hat{z}, \Delta z) \leq \quad &-\hat{z}^T(t)\left(\left(q + k_1\sqrt{\sigma}c_1\right)\left(q + k_1\sqrt{\sigma}c_1\right)^T \hat{z}(t)\right. \\
&+\left(\lambda_{\max(D_1)} + \lambda_{\max(D_2)} + 2\lambda_{\max(P_1 R)} + \lambda_{\max(P_3)}\right)\left\|\hat{z}(t)\right\|^2 - \left(\lambda_{\max(E_1)} - \frac{1}{2}\lambda_{\max(D_1)}\right. \\
&-\lambda_{\max(P_2)})\left\|\Delta z(t)\right\|^2 - \left(\lambda_{\max(P_2)} - \frac{1}{2}\lambda_{\max(E_2)}\right)\left\|\Delta z(t - \tau)\right\|^2 \\
&-\left(\lambda_{\max(P_3)} - \lambda_{\max(D_2)}\right)\left\|\hat{z}(t - \tau)\right\|^2
\end{aligned} \tag{65}
$$

By employing conditions (58)–(61) this inequality becomes

$$\dot{V}(\hat{z}, \Delta z) \leq -\left(\varrho - \lambda_{\max(D_1)} - \lambda_{\max(D_2)} - 2\lambda_{\max(P_1 R)} - \lambda_{\max(P_3)}\right)\left\|\hat{z}(t)\right\|^2 - (\lambda_{\max(E_1)} - \tfrac{1}{2}\lambda_{\max(D_1)}$$
$$-\lambda_{\max(P_2)})\left\|\Delta z(t)\right\|^2 \tag{66}$$

and using (53) yields

$$D^\beta V(\hat{z}, \Delta z) \leq -\alpha_3 \left\| \begin{array}{c} \hat{z}(t) \\ \Delta z(t) \end{array} \right\|^2. \tag{67}$$

□

Remark 3. *The asymptotical stability conditions of Control System 2, Control System 3 are independent by the time delay τ.*

3. Results

3.1. IHRG Control—Numerical Simulations

3.1.1. EXHAND with Sensors Without Delays

Consider the IHRG system of Figure 3. The exoskeleton drive system is a decoupled one, for each finger. The following parameters of the hand and exoskeleton mechanical architecture [16] will be used: the equivalent moment of inertia is $J = 0.005$ kg·m^2, the equivalent mass is $m = 0.015$ kg, the viscous and elastic coefficients of the equivalent Kelvin-Voigt model of the joint tissues throughout phalange musculoskeletal system and exoskeleton are [6,7] $c_v = 0.22$ Nm·s·rad^{-1} $c_e = 2.8$ Nm·rad^{-1}, respectively, and the damping coefficient is $c_d = 7.8$ Nm·s· rad^{-1}.

$$J\ddot{\theta}(t) = -c_v\,D^\beta\theta(t) - c_e\,\theta(t) - c_d\,\dot{\theta}(t) + mg\sin\theta + bu(t) \tag{68}$$

$$\theta(0) = \left[\frac{\pi}{3},\, 0\right] \tag{69}$$

where the nonlinear component verifies the inequality (Equation (6)) for $\eta = 0.2$. The sensor is considered as an IOM operator and the output is defined as

$$y(t) = c^T\theta(t) \tag{70}$$

The fractional order exponent is $\beta = \frac{1}{2}$. The FOM model (Equations (8) and (9)) is defined as

$$\theta_1 = \theta;\ D^{\frac{1}{2}}\theta_1 = \theta_2;\ D^{\frac{1}{2}}\theta_2 = \theta_3 = \dot{\theta};\ D^{\frac{1}{2}}\theta_3 = \theta_4;\ A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -0.6 & -0.1 & -7.8 & 0 \end{bmatrix}; b = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 4.5 \end{bmatrix}; c = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}.$$

The pairs are (A, b), (A, c) controllable, respectively observable. The IOM sensor output without delays is given by (7). A control law (Equation (13)) (for $y_{ref}(t)$) = 0) with $k = 200$ is applied. This control verifies the sector constraint (Equation (11)) with $\sigma = 5 \times 10^{-3}$. The matrix R was considered as, $R = \mathrm{diag}(3, 3, 3, 3)$ where $A_1 = A - R$ is Hurwitz stable. The vector $q = 0.05 \times [1111]^T$ and a matrix P were inferred with $\lambda_{\max(P)} = 0.725$. The polar plot of $c^T(j\omega I - A_1)^{-1}b$ is shown in Figure 6. The closed-loop system satisfies the frequential criterion (Equation (17)), condition (18) is verified for $\|Q\| = 14.5$, $\rho = 2.17$, MATLAB/SIMULINK and techniques based on the Mittag-Leffler functions are used for the simulation [1,2]. Figure 7 shows the trajectories of fractional order variables.
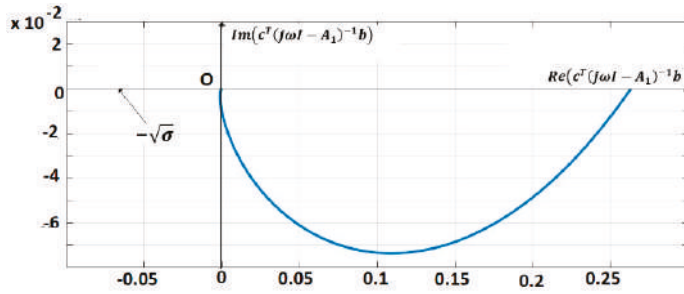
Figure 6. Polar plot of $c^T((j\omega I - A_1)^{-1}b$.
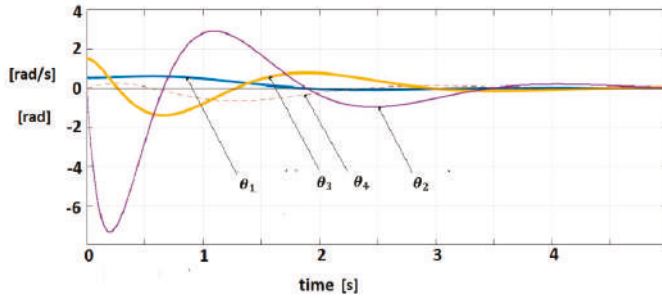
Figure 7. Fractional-order variable trajectories for FOM system with IOM sensor.

The steady state behavior can be analyzed by using by transfer function of Equations (65)–(67) (linearized model)

$$H_{EX}(s) = \frac{\theta(s)}{u(s)} = \frac{900}{s^2 + 44s^{\frac{1}{2}} + 1560s + 530} \tag{71}$$

The control law (Equation (13)) can be rewritten as a PD control

$$u_{PD} = -k_1 z_1 - k_3 D^{\frac{1}{2}}\left(D^{\frac{1}{2}} z_1\right) = -k_1 z_1 - k_3 \dot{z}_1 \tag{72}$$

and the controller transfer function will be

$$H_{C(PD)}(s) = k_1 + k_3\,s = 200(1+s) \tag{73}$$

For the PD$^{0.5}$ control law, we have

$$H_{C(PD^{0.5})} = k_1 + k_2 s^{0.5} = 200\left(1 + s^{0.5}\right) \tag{74}$$

The steady error can be inferred from Equations (71)–(73) as [21,32,33]

$$e_{s(PD)} = e_{s(PD^{0.5})} = 0.0042$$

The behavior of the linearized model (Equation (68)) for both control laws (Equations (73) and (74)) is studied. The trajectories of angular position θ for target signal $\theta_{targ} = \frac{\pi}{6}$ are shown in Figure 8.
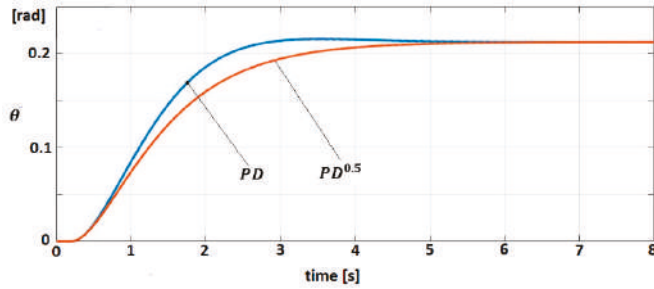
Figure 8. Trajectory $\theta(t)$ for PD$^{0.5}$ and PD

3.1.2. EXHAND with Delay

The sensor dynamics are

$$K_{s1}\, z_1(t) = y_1(t) \tag{75}$$

$$K_{s2}\, z_2(t - \tau) = y_2(t) \tag{76}$$

where $\tau \in [-0.1; 0]$. Substituting Equations (74) and (75) into (68) and using the control law (Equation (70)), yields

$$J\,\ddot{\theta}(t) = -c_v D^\beta \theta(t) - (c_e + K_{s1}k_1)\theta(t) - c_d\,\dot{\theta}(t) - K_{s2}k_2\,\dot{\theta}(t - \tau) + k_2 bc_2^T D^\beta \theta(t - \tau) + mg\sin\theta \tag{77}$$

with initial conditions

$$\theta(t) = \frac{\pi}{3}, \ \dot{\theta}(t) = -1, \ t \in [-0.1; 0] \tag{78}$$

The delay component of the dynamic model is defined by $\tau = 0.1$ s. The controller parameters are selected as $k_1 = 200$, $k_2 = 15$ that satisfy Equations (29)–(31) by employing the same parameters for q, R as in the previous example. The evolution of the fractional order variables is shown in Figure 9.
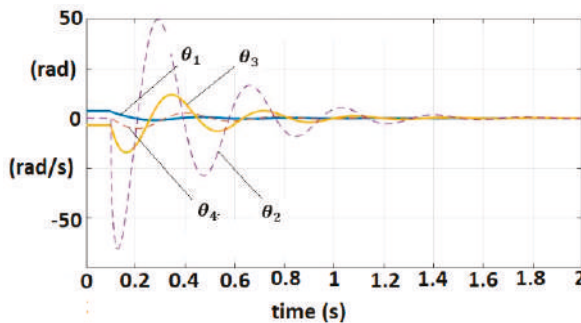
Figure 9. Fractional-order variable trajectories for FOM EXHAND with delay.

3.1.3. EXHAND with Delay and Observer

An observer (Equations (49)–(51)) with $L_1 = L_2 = [1.5\ 1.5\ 1.5\ 0]^T$ is associated to the linearized dynamic model. The matrix $R = diag(1\ 1\ 1\ 1)$ verifies the condition as $(A_L - R)$ to be Hurwitz matrix. For $q = [0.5\ 0.5\ 0.5\ 0.5]^T$, solution of P_1 is obtained with $\lambda_{\max(P_1)} = 0.0085$. A control law (55) with $k_1 = 20$, $k_2 = 8.5, \sigma = 0.05$ were selected. Equations (57)–(61) are easily verified. Figure 10 shows the trajectories of physical significance variables, position and velocity, for the system and observer.
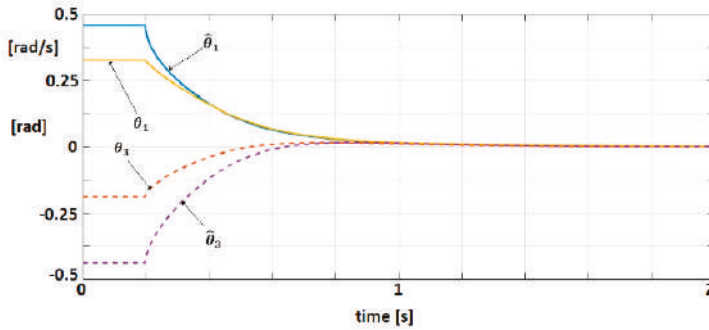
Figure 10. Trajectories of position and velocity for the system and observer.

3.2. IHRG Experimental Platform

The IHRG is an exoskeleton that supports the human hand and hand activities by using a control architecture for dexterous grasping and manipulation. IHRG is a medical device that acts in parallel to a hand in order to compensate for lost function. It is easy to use and can be a helpful tool in the home [16,34].

The mechanical architecture consists of articulated serial elements of which design covers functional and anatomic finger phalanges. The glove is created by a thin textile that represents an infrastructure suitable for actuation wires and sensors. A distributed actuation system is used for implementing the operations of the hand. An Arduino Flex Sensor network (with zeroth order sensors) is used to control the motions. An Arduino Mega 2560 hardware platform determines the movement of the glove's actuators for exercises like opening or closing of the fingers (Figures 11 and 12).

Figure 11. IHRG-human hand exoskeleton.
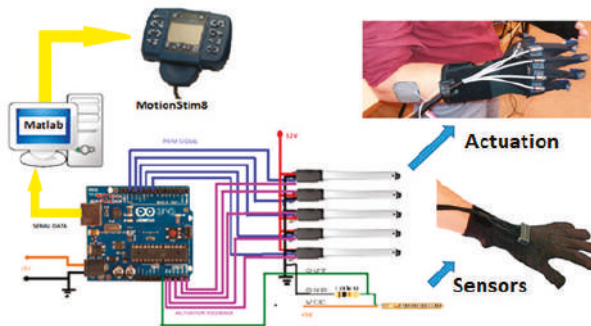
Figure 12. IHRG general architecture.

All the movements of the hand are controlled by the software of the hybrid IHRG system, which was developed in MATLAB and Simulink. The performance of each patient following the exercises program can be recorded by the same software. The control system of Control System 1 is implemented. In Figure 13 are shown the sensor signals during an open-close-open hand exercise.

(**a**)

(**b**)
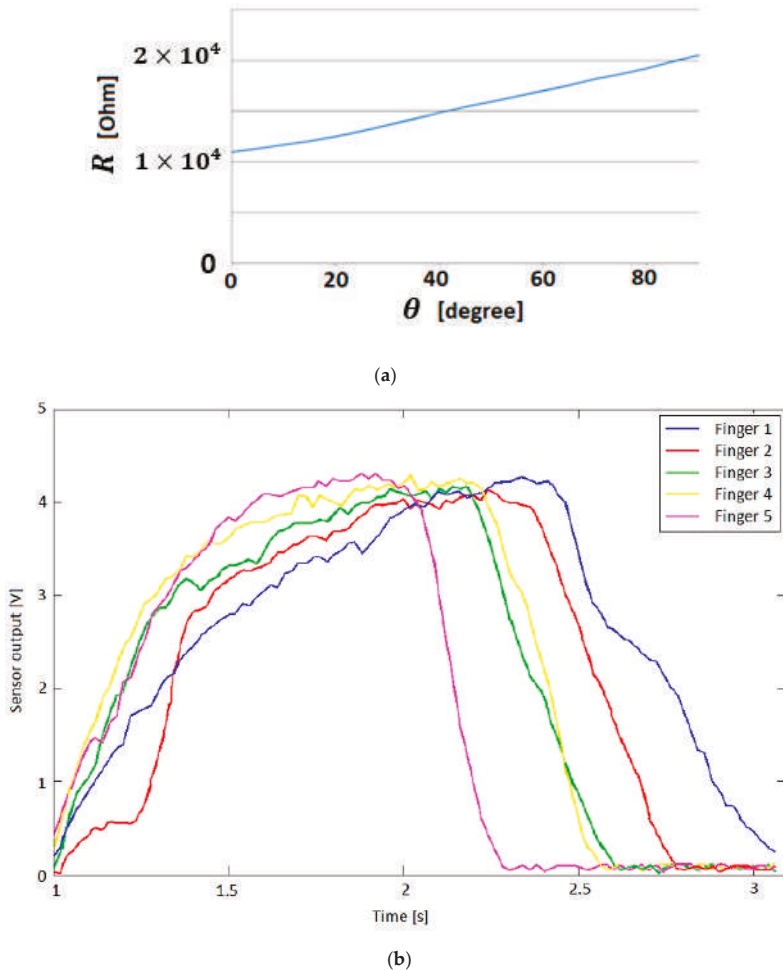
Figure 13. (**a**) Bending sensor characteristics; (**b**) Output values of the bending sensors.

4. Discussion

I. We designed, built, and tested an intelligent haptic robotic glove for the rehabilitation of the patients that have a diagnosis of a cerebrovascular accident. The glove is created by a thin textile in order to have a comfortable environment for the grasping exercises. This thin textile creates an infrastructure suitable for wire actuation and sensors. This exoskeleton architecture ensures the mechanical compliance of human fingers. The driving and skin sensor system is designed to determine comfortable and stable grasping function. The dynamics of the exoskeleton hand are modeled by fractional order operators. To our knowledge, this paper is the first paper in which the interaction between biological systems (human hand) and mechanical associated components (exoskeleton) is analyzed by fractional order models. These new models are used to develop a class of algorithms for

the control of the stable grasping function. The control systems are based on the physical significance variable control that are generated by sensor classes implemented in the system. These sensors are also modeled as operators with delays. The paper proposes control solutions and determines the criterions for controller parameter tuning for several classes of models. The observer techniques are also discussed and implemented. The quality and the stability of motion, are analyzed by Lyapunov methods and techniques that derive from Yakubovici-Kalman-Popov Lemma.

Despite of the model complexity, the control systems are very simple, and the controllers are easily implemented in an Arduino Mega 2560 hardware platform. There were many advantages for using this platform since this hardware board has ports for PWM signals that are useful to be sent to the actuators and ports for reading the signals coming from the bending sensors.

In order to help patients to follow an after-stoke recovery program, the system uses a set of predefined rehabilitation exercises like open the hand, close it, try to grab an object or simple wave. The system is very easy to use at home, with minimal training. The predefined rehabilitation set of exercises was created to be used.

II. The control systems discussed in the previous sections are focused on the control problems of the IHRG system, where the EXHAND model is described by FOM operators and the sensor system is based on zeroth order sensors. These control solutions can be also used for a larger class of complex systems as hyper-redundant systems, that use complex FOM sensors (Figure 14).
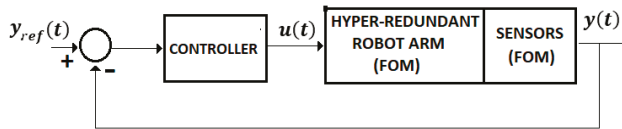
Figure 14. Control system with FOM sensors.

III. In addition, we consider that control systems discussed in the previous sections can be applied to a class of control problems associated to the persons with disabilities. Figure 15 presents a wheelchair control system for this class of persons. In this case, the human operator is represented by the persons with hemiparesis/hemiplegia, with motor restriction (arm or leg-emphasized hemiparesis) and with serious brain damage [35],
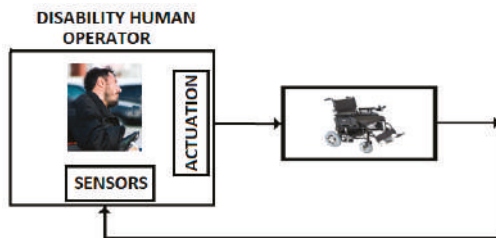
Figure 15. Control with disability human operator.

$$H_h(\mathrm{s}) = k_h \frac{e^{-\tau s}}{s^\beta}$$

The transfer function of this human operator has a model that corresponds to a time delay fractional order model with time constant τ and fractional order β. These parameters are determined by the characteristics of the damaged brain, viscoelastic properties of the atrophied muscles, and propagation time along the nervous terminals.

We consider that these models can be studied by using the techniques developed in this paper.

References

1. Chen, Y.; Sun, R.; Zhou, A. An overview of fractional order signal processing (FOSP) techniques. In Proceedings of the ASME 2007 International Design Engineering Technical Conferences, Computers and Information in Engineering Conference, Las Vegas, NV, USA, 4–7 September 2007; pp. 1205–1222.

2. Romanovas, M.; Klingbeil, L.; Traechtler, M.; Manoli, Y. Application of Fractional Sensor Fusion Algorithms for Inertial MEMS Sensing. *Math. Model. Anal.* **2009**, *14*, 199–209. [CrossRef]

3. Debnath, K. Recent applications of fractional calculus to science and engineering. *Int. J. Math. Sci.* **2003**, *54*, 3413–3442. [CrossRef]

4. Ortigueira, M.D. A coherent approach to non-integer order derivatives. *Signal Proces. Spec. Sec. Fract. Calc. Appl. Signal Syst.* **2006**, *86*, 2505–2515. [CrossRef]

5. Sierociuk, D.; Skovranek, T.; Macias, M.; Podlubny, I.; Petras, I.; Dzielinski, A.; Ziubinski, P. Diffusion process modeling by using fractional-order models. *Appl. Math. Comput.* **2015**, *257*, 2–11. [CrossRef]

6. Tzoumas, V.; Xue, Y.; Pequito, S.; Bogdan, P.; Pappas, G.J. Selecting Sensors in Biological Fractional-Order Systems. *IEEE Trans. Control Netw. Syst.* **2017**, *5*, 134–142. [CrossRef]

7. Caponetto, R.; Graziani, S.; Pappalardo, F.L.; Sapuppo, F. Experimental Characterization of Ionic Polymer Metal Composite as a Novel Fractional Order Element. In *Advances in Mathematical Physics*; Hindawi Publishing Corporation: London, UK, 2013; p. 10.

8. Kobayashi, Y.; Watanabe, H.; Hoshi, T.; Kawamura, K.; Fujie, M.G. Viscoelastic and nonlinear liver modeling for needle insertion simulation. In *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 41–67.

9. Wex, C.; Frolich, M.; Brandstadter, K.; Bruns, C.; Stoll, A. Experimental analysis of the mechanical behavior of the viscoelastic porcine pancreas and preliminary case study on the human pancreas. *J. Mech. Behav. Biomed. Mater.* **2015**, *41*, 199–207. [CrossRef] [PubMed]

10. Islam, T.; Shakil, S.; Lohani, B.; Mukhopadhyay, S.C. A Fractional Order Impedance Sensor for Measuring the Quality of Drinking Water. In Proceedings of the 12th International Conference on Sensor Technologies and Applications (SENSORCOMM 2018), Venice, Italy, 16–20 September 2018; pp. 25–31.

11. Dhaifallah, M.; Nassef, A.; Rezk, H.; Nisar, K.S. Optimal parameter design of fractional order control-based INC-MPPT for PV system. *Sol. Energy* **2018**, *159*, 650–664. [CrossRef]

12. Xue, Y.; Rodriguez, S.; Bogdan, P. A spatio-temporal fractal model for a cps approach to brain-machine-body interfaces. In Proceedings of the IEEE Design, Automation & Test in Europe Conference & Exhibition 2016, Dresden, Germany, 14–18 March 2016; pp. 642–647.

13. Xue, Y.; Pequito, S.; Coelho, J.M.; Bogdan, P.; Pappas, G.J. Minimum number of sensors to ensure observability of physiological systems: A case study. In Proceedings of the 54th Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, USA, 27–30 September 2016.

14. Xue, Y.; Bogdan, P. Constructing compact causal mathematical models for complex dynamics. In Proceedings of the ACM 8th International Conference on Cyber-Physical Systems, Pittsburgh, PA, USA, 18–20 April 2017; pp. 97–107.

15. Xue, Y.; Bogdan, P. Reliable multi-fractal characterization of weighted complex networks: Algorithms and implications. *Sci. Rep.* **2017**, *7*, 7487. [CrossRef] [PubMed]

16. Popescu, N.; Popescu, D.; Ivanescu, M.; Popescu, D.; Vladu, C.; Vladu, I. Force Observer-Based Control for a Rehabilitation Hand Exoskeleton System. In Proceedings of the Asian Control Conference (ASCC2013), Zurich, Switzerland, 23–26 June 2013; pp. 1–6.

17.   Agila, A.; Baleanu, D. A study of multi-degree of freedom fractional order damped oscillatory system. *Upb Sci Bul.* **2018**, *80*, 52–62.

18.   Vanoglio, F.; Bernocchi, P.; Mule, C.; Garofali, F.; Mora, C.; Taveggia, G.; Scalvini, S.; Luisa, A. Feasibility and efficacy of a robotic device for hand rehabilitation in hemiplegic stroke patients: A randomized pilot controlled study. *J. Clin. Rehabil.* **2016**, *31*, 351–360. [CrossRef] [PubMed]

19.   Podlubny, I. Fractional Differential Equations. In *Mathematics in Science and Engineering*; Academic Press: Millbrae, CA, USA, 1999; Volume 198.

20.   Diethelm, K. *The Analysis of Fractional Differential Equations*; Springer-Verlag: London, UK, 2004.

21.   Petras, I. *Fractional-Order Nonlinear Systems, Modeling, Analysis and Simulation*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2011.

22.   Argarwal, R.; Hristova, K.S.; O'Regan, D. Lyapunov functions and strict stability of Caputo fractional differential equations. *Adv. Differ. Equ.* **2015**, *346*, 2–20.

23.   Zhao, Y.; Wang, Y.; Liu, Z. Lyapunov Function Method for Linear Fractional Order Systems. In Proceedings of the 34th Chinese Control Conference, Hangzhou, China, 28–30 July 2015; pp. 1457–1463.

24.   Dadras, S.; Malek, H.; Chen, Y. A Note on the Lyapunov Stability of Fractional Order Nonlinear Systems. In Proceedings of the ASME 2017, Cleveland, OH, USA, 6–9 August 2017; pp. 123–129.

25.   Al-Saggaf, U.M.; Mehedi, I.M.; Mansouri, R.; Bettayeb, M. Rotary flexible joint control by fractional order controllers. *Int. J. Control Autom. Syst.* **2017**, *15*, 2561–2569. [CrossRef]

26.   Rhong, L.; Peng, X.; Zhang, B. A Reduced-Order Fault Detection Filter Design for Polytopic Uncertain Continuous-Time Markovian Jump Systems with Time-varying Delays. *Int. J. Control. Autom. Syst.* **2018**, *16*, 2021–2032. [CrossRef]

27.   Khimani, D.; Patil, M. High Performance Super-twisting Control for State Delay Systems. *Int. J. Control. Autom. Syst.* **2018**, *16*, 2063–2073. [CrossRef]

28.   Lee, Y.S.; Kim, D.S.; Kim, S.K. Disturbance Observer-Based Proportional-Type Position Tracking Controller for DC Motor. *Int. J. Control Autom. Syst.* **2018**, *16*, 2169–2176. [CrossRef]

29.   Gravagne, L.A.; Walker, I.D. Uniform Regulation of a Multi-Section Continuum Manipulators. In Proceedings of the 2002 IEEE International Conference on Rob and Automation, Washington, DC, USA, 11–15 May 2002; pp. 1519–1525.

30.   Ivanescu, M. A Decoupled Sliding Mode Control for a Continuum Arm. *Adv. Robot.* **2015**, *29*, 831–845. [CrossRef]

31.   Khalil, H. *Nonlinear Systems*; Prentice Hall: Upper Saddle River, NJ, USA, 2002.

32.   Heymann, N.; Podlubni, I. Physical interpretation of initial conditions for fractional differential with Riemann-Liouville fractional derivatives. *Rheol. Acta* **2014**, *7*, 45–63.

33.   Tejado, I.; Abdelhamid, D.; Vinagre, B.M. *Two Strategies for Fractional Sliding Mode Control of Integer Order Systems by System Augmentation: Application to a Servomotor*; IFAC-PapersOnLine: Laxemburg, Austria, 2017; Volume 50, pp. 8103–8108.

34.   Hartopanu, S.; Poboroniuc, M.; Serea, F.; Irimia, D.C.; Livint, G. New issues on FES and robotic glove device to improve the hand rehabilitation in stroke patients. In Proceedings of the 6th International Conference on Modern Power System, Cluj Napoca, Romania, 18–21 May 2015.

35.   Huang, J.; Chen, Y.; Li, H.; Shi, X. Fractional Order Modeling of Human Operator Behavior with Second Order Controlled Plant and Experiment Research. *IEEE/CAA J. Autom. Sin.* **2016**, *3*, 271–280.

Article

Detection of Anomalous Behavior in Modern Smartphones Using Software Sensor-Based Data

Victor Vlădăreanu [1], Valentin-Gabriel Voiculescu [2], Vlad-Alexandru Grosu [2], Luige Vlădăreanu [1,*], Ana-Maria Travediu [1], Hao Yan [3], Hongbo Wang [3] and Laura Ruse [4]

[1]   Institute of Solid Mechanics of the Romanian Academy, 010141 Bucharest, Romania; victor.vladareanu@imsar.ro (V.V.); ana.travediu@imsar.ro (A.-M.T.)
[2]   Faculty of Electronics, Telecommunications and Information Technology, University Politehnica of Bucharest, 061071 Bucharest, Romania; valentin.voiculescu@upb.ro (V.-G.V.); crisvlad74@gmail.com (V.-A.G.)
[3]   Parallel Robot and Mechatronic System Laboratory of Hebei Province, Yanshan University, Qinhuangdao 066004, China; yh@stumail.ysu.edu.cn (H.Y.); hongbo_w@ysu.edu.cn (H.W.)
[4]   Faculty of Automatic Control and Computers, University Politehnica of Bucharest, 060042 Bucharest, Romania; laura.ruse@cs.pub.ro
*   Correspondence: luige.vladareanu@imsar.ro

Abstract: This paper describes the steps involved in obtaining a set of relevant data sources and the accompanying method using software-based sensors to detect anomalous behavior in modern smartphones based on machine-learning classifiers. Three classes of models are investigated for classification: logistic regressions, shallow neural nets, and support vector machines. The paper details the design, implementation, and comparative evaluation of all three classes. If necessary, the approach could be extended to other computing devices, if appropriate changes were made to the software infrastructure, based upon mandatory capabilities of the underlying hardware.

Keywords: software sensor data; machine-learning classifier; smartphone security

1. Introduction

Among our gadgets, smartphones are our closest companions. They provide the primary access into the Internet and modern amenities, they hold our private data and are becoming one of the primary means of attack against the user, be it through power viruses (or other means to consume resources) or more ordinary malware menaces (calling or texting tolled numbers, install unwanted software, send the attacker private information about the device or its owner, spy on the owner using the camera or microphone, etc.).

For our research we picked an Android smartphone over an iPhone, primarily due to the open access into its software stack. This open source stack, provided by Google for their devices via Android Open Source Project (AOSP), includes a high-level Android framework, and an open source kernel. This level of openness into the source code of the stack allows access into both public and non-public API information, allowing for the purpose of our research access into the smartphone sensors. It also consists of the basis of the infrastructure of data collection architecture (used to obtain the raw data behind the initial dataset for this paper) described in [1] with an application for malware detection [2], using measurable events collected for a set of Android applications including both samples obtained from scientists in the field (Malgenome application set described by the authors of [3]) as well as more recent Internet sources (including [4,5] and Google Play for benign Android samples).

The purpose of this study was to assess if anomalous behavior could be detected through machine-learning classifiers based on input data sources from a variety of sensors within the device. We took into consideration that the smartphone phone itself can provide a large pool of data sources

about runtime behavior. Some of this data is accessible through public APIs and an additional set can become accessible by making suitable changes into the smartphone software stack. While typical developers do not have access to non-public APIs, such an approach could provide benefits for telecom or smartphone manufacturers which have access to the AOSP stack or its equivalent from the smartphone system-on-chip (SoC) manufacturers or similar providers. By adding such an application within their phone, smartphone manufacturers could increase the intrinsic value of their product.

Many recent papers are trying to tackle the problem of detecting anomalous behavior in modern smartphones using software sensors (measurable events). For modern smartphones approaches vary but usually involve a combination of static and dynamic behavioral extraction of interesting data pertaining to sensors of the target smartphone application.

The measurable sources of data used in static analysis fall into 4 categories according to a recent study of over 80 frameworks by Bakour et al. [6]: manifest-based, code-based, semantic-based, or application metadata-based. According to the same authors, using this data requires offline processing and is prone to weakness when obfuscation techniques are employed by the developers of malicious applications.

Recent frameworks use a combination of static [7–9] and dynamic [8,9] extraction of data from sensors before being processed through machine-learning techniques. SandDroid is a recent sandbox analysis project classifying Android applications on a set of features using both static and dynamic analysis [8].

Current anomaly-detection applications span a wide array of technical solutions, from traditional machine learning and statistics, to deep learning models for computer vision related tasks, with applications in virtually all fields of technology [10–12].

Gaussian Mixture Models (GMMs) are one of the traditional approaches to anomaly detection. Zong et al. [13] use a deep auto-encoder to generate a low-dimensional representation, which is further fed into a GMM for unsupervised anomaly detection in medical applications. Aljawarneh et al. [14] use a Gaussian dissimilarity measure for anomaly detection in Internet of Things (IoT) systems, while Li et al. [15] use GMM-based clustering for an application of early warning in civil aviation.

In viewing anomaly detection more as a classification problem, there are many options for the choice of classifier. Erfani et al. [16] use Support Vector Machines and Deep Belief Networks to mitigate the impact of high dimensionality in large-scale anomaly detection. Li et al. [17] use a modified Support Vector Machine (SVM) as an alternative to computer and information system security. The excellent surveys by Chalapathy [10] and Ye [11] give a detailed outlook of possible applications using deep learning and data mining solutions.

The literature involving detection of anomalies in modern smartphones usually focuses on centralized large-scale frameworks or collaborative efforts. Centralized frameworks are used for either static code analysis [7,8] or dynamic analysis using virtual machines on host systems [8], highly efficient in running Android apps in emulation. Collaborative efforts, such as Contagio [18] rely on a mix of online tools and offline community support. We considered the approach of running applications within an actual smartphone, as there are specific limitations (e.g., thermal throttling) or information (e.g., device identifiers) that do not appear in virtual machine simulation but make an impact on application behavior on real hardware (through metrics such as CPU load, network data frequency, and throughput of which we planned on measuring through our experiments). We used as starting points data acquired on actual smartphones, not via simulated/virtualized frameworks. In this way our environment also took into consideration the influence of additional constraint factors within the smartphone (such as thermal throttling, dynamic frequency changes and overall other types of computation throttling) which can modify an application resource requirements and through this means influence the collected raw data that are used as input for the machine-learning algorithms. Virtualized infrastructures do not simulate specific smartphone behavior such as thermal throttling, dynamic processor frequency changes etc.

Furthermore, the use of such methods, as described in the paper, is relatively limited in the context of smartphone technology. Most smartphone applications tend to veer towards classical anomaly detection and statistics, such as GMMs [13–15] or Hidden Markov Models (HMMs) [19], data mining and deterministic decision systems [20,21], or hybrid algorithms that incorporate learning classifiers (usually a version of SVMs) and belief networks [16,17,22,23]. An extensive literature review has found no comparative studies using the proposed methods on a challenging dataset, which is itself unique, as obtained through the methods described above, and very few instances of any one of the proposed methods being employed in anomaly detection on smartphones, despite their otherwise ubiquitous presence in machine-learning and anomaly-detection applications.

While all detections are aimed at malware applications, the detection itself is done through looking for patterns of anomalous behavior, as captured by the measurable sources of information (i.e., software sensors). The use of the proposed class of methods has to do with the way in which such potentially anomalous behavior could be learned, therefore it is intentionally modelled as a classification problem, rather than a standard anomaly-detection problem (for example through GMMs), due to the format of the data and the underlying assumptions of the learning model.

The choice of algorithm has very much to do with the available dataset. For the purposes of this paper, deep learning and some data mining solutions are not well suited to a small- to medium-sized dataset. In addition, GMMs are most frequently used when there is a large non-anomalous subset to train on, or when the dataset itself is imbalanced, and when it is thought that a classifier is unlikely to learn a coherent model for the positive (anomalous) samples, which is not the case here. As such, the paper investigates three types of classifiers for the machine-learning application: Logistic Regression, a shallow neural network for pattern recognition, and SVMs. The three are evaluated on several metrics, most important of which being the F1 score on the test set. The full details of the design, implementation, and evaluation of the learning algorithms are presented in their respective sections. Finally, the results show that all the three investigated algorithms perform reasonably well, with SVMs having a slight edge.

The remainder of the paper is divided as follows: Chapter 2 outlines the acquired data and the steps taken to curate the dataset, as well as giving a brief theoretical overview of the design of the classifier learning methods involved. Chapter 3 discusses the particulars of implementation and the obtained results for each model. Chapter 4 discusses the highlighted results and attempts to draw a conclusion based on the work reported. Opportunities and directions for future research are also discussed.

2. Materials and Methods

The general architecture, described in [1] includes measurable events as sources of data, extracted with a set of data collectors [1] into a server for further processing, as shown in Figure 1. The sources of data are made available through the Android framework or Linux user-space (via procfs virtual file system entries) and are provided originally by hardware sensors or software hooks added into the framework.

Synthesizing the information available in [1], Table 1 below shows the measurable events that were collected on a Nexus 4 smartphone running Android 4.2.2 JellyBean.

As said in [1] for the "SMS" event there are hooks for destination number and message content. For the "Call" event there is a hook for the destination number. For "WiFi", "Camera", "NFC", as well as "Bluetooth" we get information about sensor activity (turning on/off). "Sensors" offer information about what peripherals are registered as sensors in the Android framework terminology as well as real-time data from them. "Camera" indicates if the application wishes to access the camera peripheral or photo content within the phone.
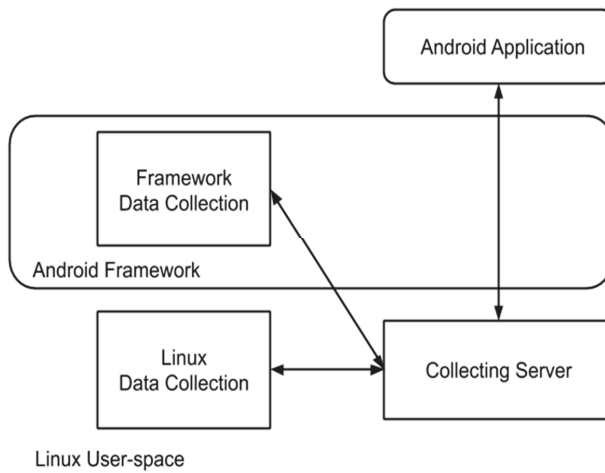
Figure 1. General data collection architecture.

Table 1. Measurable events and description.

Event	Intercepted Information
SMS	an SMS is sent, destination phone number, message content
Call	outgoing phone call takes place, destination phone number
WiFi	Wi-Fi state—enabled or not
Bluetooth	Bluetooth state—enabled or not
App Install	package installation/uninstallation and package name
Sensors	value information for registered sensors
Camera	Camera state—on or off
NFC	NFC state—on or off
Activity	current state of the Activity: Create, Start, Stop, Pause, Resume, Destroy
Runtime Crash	a runtime crash of the application is happening
ANR	Application Not Responding dialog box is being generated

Through the data acquisition procedure, the smartphone remained mostly stationary, in an artificially lit room with weak GPS reception. As such, the device was not able to offer pertinent GPS or other motion (accelerometer, gyroscope, magnetometer) or lighting (ambient light) information for our experiments. We chose not to include in our smartphone a functionality (normally missing from commercial devices) that artificially injects GPS data, considering that by itself any additional GPS data injection component would represent an example of atypical application (hard to install by regular users) and behavior (and also being potentially hazardous if requiring administrative privileges into the phone in order to operate). For our research we focused on the scenarios that for malware applications the abnormal behavior would manifest primarily through disguising themselves as top (or banking) applications (for the purpose of extracting confidential login data) or by sending phone information to toll numbers or various URLs.

Another reason not to include GPS information was that for the majority of applications in our pool this information was not required, the main part of the experiments not including applications focused on user movement information (whether it is from GPS or other types of movement based sensors).

"App Install" detects the installation of a new application within the smartphone and it offers the app name for it. "Activity" intercepts information regarding the state of the Android application: "Create", "Start", "Stop", "Pause", "Resume" and "Destroy". The "Runtime Crash" detects the similarly named event at runtime. The "ANR" intercepts "Application Not Responding" type events generated by the Android framework following an application entering a freezing/non-functioning

state. These hooks were implemented in Java and they offer the ability to acquire data for the data collection server upon request, for the purpose of detecting anomalous behavior.

Due to the user experience oriented programming in modern smartphone apps, each feature is accessible within a short number of Graphical User Interface (GUI) interactions (for example in email apps you have login, message lists, folders/categories, individual email widows and a limited set of clicking and scrolling to do in each such window), in our case exposing behavior coded within Android applications to our software sensors. Empirically, as a trade-off between time spent collecting data for each target application and the implication on overall collecting time spent on the entire application pool we studied and collected data for, we found that approximately 5 min was considered enough to stop acquiring new types of events, and thus new type of behavior, from the application.

The final application set comprises 361 Android applications; however not all of them were installed at once, but consecutively during the experiments. For each sample application, during the dataset acquisition the procedure involved side-loading (installing Android applications in apk format via the adb interface), starting the sample application and monitoring its behavior while interacting with it manually (in foreground/background) for a short period of time (approximately 5 min), then uninstalling it.

The initial data was collected through the framework described in [1,2] which relies on a data collection framework within the smartphone and the installation of an additional application that performs the real-time monitoring while running in background (called AMDS in [2]). The application pool was built by using both reputable applications (the ones that the phone came with, top Google Play applications, or similar stores for other markets such as SlideMe [4]) as well as known malware applications. The data acquisition process involved installing new applications one by one, manually interacting with each in application-specific manner (e.g., clicking buttons, swiping) while also toggling between foreground/background for the duration of the experiment (around 5 min for each app). After this step the target application was uninstalled, to prevent both reaching lack of storage space during the experiment as well as preventing some applications from influencing one another and the next target application from the list was installed and experimented with. During the training phase a data of monitoring each target application was stored locally in the phone, and during testing phase a different pool of applications was used, each application installed, monitored, local data stored updated, while the real-time monitoring application described at large in [2], was running in background, and if abnormal behavior was detected this application would be prevented from running, it would be flagged to the user who would make the final decision (to continue preventing the application from running or allow it to continue) on a case by case basis.

The raw data obtained from the experimentation is organized as a set of unique application names, a set of labels for each application (clean or malware) and 19 sets of feature data, one for each investigated feature. Inside the feature data, there are multiple values for the same {application name; feature} pair, due to multiple sensor interrogations for the same feature on the particular application. Table 2 shows an example of the raw data obtained from the sensors.

Table 2. Example of raw data obtained from software sensors.

Application Name	sent_bytes_wifi_fore	Application Name	sent_bytes_wifi_fore
air.CandyCatcher	23461	air.com.innmenu.free	7293
air.CandyCatcher	23617	air.com.innmenu.free	9316
air.CandyCatcher	23825	air.com.innmenu.free	12354
air.CandyCatcher	23825	air.com.innmenu.free	13367
air.CandyCatcher	23825	air.com.innmenu.free	19440
air.com.innmenu.free	7293	air.com.innmenu.free	21469

Table 3 shows the investigated features for which sensor triggers were programmed and a short description, as detailed in [1,2].

Table 3. Description of sensor data features.

No.	Feature Name	Description
1	SysCycles_subset	System-wise CPU cycles collected via performance counters
2	Cycles_subset	Application CPU cycles collected via performance counters
3	Load1_subset	CPU load for 1 min collected via procfs (cat /proc/loadavg) in user-space
4	Load5_subset	CPU load for 5 min collected via procfs (cat /proc/loadavg) in user-space
5	Load15_subset	CPU load for 15 min collected via procfs (cat /proc/loadavg) in user-space
6	Total_occ_mem_subset	Total memory information collected via procfs (cat /proc/meminfo) in user-space
7	Memory_subset	VmSize memory information collected via procfs (cat /proc/$pid/status | grep VmSize) in user-space
8	RSS_memory_subset	VmRSS memory information collected via procfs (cat /proc/$pid/status | grep VmRSS) in user-space
9	Threads_subset	Number of threads collected via procfs (cat /proc/$pid/status | grep Threads) in user-space
10	Recv_bytes_wifi_back_subset	Number of bytes, obtained using AOSP events and procfs stats, received via WiFi while the application is in background
11	Recv_packets_wifi_back_subset	Number of packets, obtained using AOSP events and procfs stats, received via WiFi while the application is in background
12	Sent_bytes_wifi_back_subset	Number of bytes, obtained using AOSP events and procfs stats, sent via WiFi while the application is in background
13	Sent_packets_wifi_back_subset	Number of packets, obtained using AOSP events and procfs stats, sent via WiFi while the application is in background
14	Recv_bytes_wifi_fore_subset	Number of bytes, obtained using AOSP events and procfs stats, received via WiFi while the application is in foreground
15	Recv_packets_wifi_fore_subset	Number of packets, obtained using AOSP events and procfs stats, received via WiFi while the application is in foreground
16	Sent_bytes_wifi_fore_subset	Number of bytes, obtained using AOSP events and procfs stats, sent via WiFi while the application is in foreground
17	Sent_packets_wifi_fore_subset	Number of packets, obtained using AOSP events and procfs stats, sent via WiFi while the application is in foreground
18	CPU_usage_user_float_subset	User-space CPU usage percentage
19	CPU_usage_system_float_subset	Total CPU usage percentage

We considered the possibility that while there is a usual correlation between the number of packets and bytes transferred via WiFi, in the case of applications disguising themselves as another application, there may be a difference in communication protocols between the fake and original application that could influence the correlation between number of packets and bytes transferred. Generally speaking, even if the two features were to be correlated, the presence of two correlated features among a total of 19 would not adversely affect the learning process to any significant extent. In this particular case, however, the possible lack of correlation could, potentially, in itself be a valuable learnable pattern for detecting behavior. Therefore, it was believed that including both features would result in new information being available to the model.

The measurable events pool provided by the infrastructure described in detail in [1] and used through the monitoring application described in [2] includes events collectable from Android (e.g., SMS, Bluetooth, Call etc.), events collectable from Linux user-space (e.g., via procfs, including CPU load, memory load etc.) and there were non-numerical metrics (e.g., app name, version) which were eliminated when we curated the data set. Features for which data was not available in all instances were then eliminated to preserve the integrity of the dataset, as a fully numeric, complete set of samples is required for the learning process.

As stated in [1,2] our collection infrastructure includes data collectors, a collection server, and an Android monitoring application. Data is acquired for both Android events (SMS, WiFi, and Bluetooth) as well as Linux user-space (CPU load, memory load, and network statistics) and kernel-space metrics

(performance counters). The collection server maintains the list of running Android applications (and correlation with corresponding process identification numbers) and based on application state manages monitoring via the data collectors. The Android application usually resides in background and allows high-level user interaction and control (including picking the infrastructure operating mode: idle, training, testing on a system-wide and per application level). Furthermore, part of the initial pre-processing of the dataset is that clearly anomalous readings are eliminated as data-points. This is partly achieved in the initial composition of the dataset, and partly done automatically when processing the numerical data.

The ground-truth target labels were manually set. Applications downloaded from reputable sources (top Google Applications) were marked as clean, while applications from the known malware sources (e.g., applications from the Malgenome set) were considered malware.

The dataset is first processed by parsing the feature data and identifying all the values associated with a unique application name on a particular feature set. These values are then averaged to obtain a single value for the {application name; feature} pair. The average value is obtained by running a first pass on the values and calculating their mean and variance, after which a new mean is computed, taking into consideration only the values that fall within a 95% confidence interval from the natural mean. In the extreme case that no values fall within this range, the median value of the set is selected instead. This contributes to the rejection of extreme sensor values, which are thought to most likely be the result of erroneous momentary interrogation.

Once all unique {application name; feature} values are parsed for a particular feature, the application names are checked against the existing list and only the pairs for which the application name already exists are kept. This is done to ensure the integrity of the dataset, in which all samples (i.e., applications) must have existing values for all features. The end result is a dataset with 361 samples for 19 features, plus the target labels, which are the inputs to the learning algorithms.

The features are all numeric, so there is no need to apply any type of coding for categorical variables. The target class labels are either clean (coded to 0) or malware (coded to 1).

The final dataset has its features normalized. The value for each sample within a feature is updated by subtracting the mean (μ_j) and dividing by the standard deviation (s_j) of that feature (j), as shown below.

$$x_{ij} := \frac{x_{ij} - \mu_j}{s_j} \tag{1}$$

This ensures that all input features are of similar range and centered on zero, which helps the speed and convergence of the learning process, as well as removing potential biases caused by the varying magnitudes. A cross-section of the dataset is exemplified in Table 4. Notice that the unique application names have been removed and each sample is now uniquely identifiable only implicitly through its row number. The actual data used for training is structured with samples as rows and features as columns, but is shown here transposed, for readability. It also includes an intercept term, which is a column of 1-s, not shown here. The sample and feature labels are not present in the actual dataset and are shown in the table only for convenience.

The dataset is then split into three sets, for training, cross-validation, and testing. The training set is used to learn the optimized parameters of the prediction model, the cross-validation set is used to tune its meta-parameters, and the testing set is used exclusively to rate the model performance on data it had not previously seen. The parameters and meta-parameters differ for each learning model and are explained in their respective sections.

Table 4. Example of dataset samples.

	App 1	App 2	App 3	App 4	App 5	App 6	App 7	App 8	App 9
Feat 1	2.1260	−0.2039	−0.9102	−1.0380	−0.7869	−0.4426	−0.5818	2.0124	0.3142
Feat 2	1.5776	0.2069	−0.8571	−0.8805	−0.7298	−0.8488	−0.6119	4.5156	−0.2921
Feat 3	1.3070	1.5609	−0.9932	−0.7135	0.8064	−1.4613	−0.6617	−0.0143	1.9401
Feat 4	2.0614	0.5280	−0.5014	−0.3219	0.4668	−0.7027	−0.2998	−0.3167	0.6111
Feat 5	4.8613	−0.0313	−0.4700	−0.3211	1.4666	−0.3339	−0.4885	−0.2823	0.1868
Feat 6	1.6025	−0.2129	−0.4023	0.7248	1.3224	−0.9621	−1.3607	1.0606	0.5359
Feat 7	1.6277	1.7521	−1.1560	−1.0896	0.8968	−1.1018	−1.1545	−0.7817	0.3637
Feat 8	1.3134	1.5573	−1.0430	−0.9086	−0.0139	−0.9776	−0.9513	−0.9219	0.0963
Feat 9	0.9160	2.1956	−1.2540	−1.1672	0.9956	−1.0804	−1.0804	−0.8200	0.3952
Feat 10	−0.1558	−0.1579	−0.1550	−0.1579	−0.0435	−0.1579	−0.1579	−0.0612	−0.1579
Feat 11	−0.1674	−0.1841	−0.1590	−0.1841	0.0255	−0.1841	−0.1841	0.0255	−0.1841
Feat 12	−0.2925	−0.3570	−0.3128	−0.3803	0.6953	−0.3803	−0.3803	0.5122	−0.3803
Feat 13	−0.2251	−0.2320	−0.2182	−0.2667	0.2182	−0.2667	−0.2667	0.3775	−0.2667
Feat 14	−0.1042	−0.1742	−0.2019	−0.1976	−0.1635	−0.2020	0.3298	−0.1763	−0.1719
Feat 15	−0.1062	−0.1707	−0.2224	−0.1955	−0.1546	−0.2245	0.2975	−0.1638	−0.1750
Feat 16	0.0372	−0.1001	−0.4477	−0.2964	0.1316	−0.4582	−0.0927	0.0206	0.1077
Feat 17	−0.0349	−0.2000	−0.3626	−0.2479	−0.1387	−0.3678	0.2105	−0.0466	−0.2137
Feat 18	−0.0548	−0.0547	−0.0202	−0.0548	−0.0548	0.0555	−0.0548	−0.0528	−0.0548
Feat 19	−0.0660	−0.0659	−0.0323	−0.0660	−0.0660	0.0497	−0.0660	0.0804	−0.0660

The split is done randomly each time the overall script starts, so the data is different for each complete run which learns and evaluates the three models, meaning the overall results differ slightly at each run. However, the same training, cross-validation, and test sets are used for learning and evaluation on each of the three learning models, within a complete run of the script, so that their performance can be properly compared.

The samples are split into 70% for the training set, 15% for the validation set and 15% for testing. This is done by generating a random index with the same length as the number of samples and then taking the appropriate percentage splits, using the index as the row number inside the dataset. In the context of 361 total samples, this works out to 253 samples for training, 54 samples for validation, and 54 samples for testing.

A True Positive (TP) is a sample correctly identified as malware, a True Negative (TN) is a sample correctly classified as clean, a False Positive (FP) is a clean sample misclassified as malware, and a False Negative (FN) is a malware sample misclassified as clean. A chart showing the number and rates of samples thus classified is called a Confusion Matrix. The results of each algorithm include such a chart for each sample set and for the overall set.

Two standard loss functions are used for the classification algorithms, Cross-Entropy Loss and Hinge Loss. Cross-entropy is the default loss function used for binary classification problems. It is intended for use with binary classification where the target values are in the set {0, 1}, as is the default coding in our dataset. It is used for the Logistic Regression and Pattern net learning algorithms [24]. Using Cross-Entropy Loss, the cost function to be minimized is shown below. $h_\theta(x^i)$ is the hypothesis outputted by the model (see also Equation (5)) and y^i is the ground-truth label for that sample.

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} [y^i(-\log h_\theta(x^i)) + (1 - y^i)(-\log(1 - h_\theta(x^i)))] + \frac{\lambda}{2m} \sum_{j=1}^{n} \theta_j^2 \tag{2}$$

where $h_\theta(x^i) = \frac{1}{1+e^{-\theta^T x^i}}$.

Hinge Loss is an alternative to cross-entropy for binary classification problems, primarily developed for use with SVM models. It is intended for use with binary classification where the target values are in the set {−1, 1}, which are set automatically by the learning algorithm. The Hinge Loss function encourages examples to have the correct sign, assigning more error when there is a difference in the sign between the actual and predicted class values [24]. The formula of the Hinge Loss function is shown below.

$$J(\theta) = C \sum_{i=1}^{m} \left[y^i cost_1\left(\theta^T x^i\right) + \left(1 - y^i\right) cost_0\left(\theta^T x^i\right) \right] + \frac{1}{2} \sum_{j=1}^{n} \theta_j^2 \tag{3}$$

where $cost(x) = \begin{cases} 0, & y * k(x) \geq 1 \\ 1 - y * k(x), & else \end{cases}$, $k(x)$ being the similarity (kernel) function [25–27].

The following metrics are used for evaluation, as shown in Table 5.

Table 5. Classifier metrics.

Number of True Positives	TP
Number of True Negatives	TN
Number of False Positives	FP
Number of False Negatives	FN
Accuracy	$ACC = \frac{TP+TN}{TP+TN+FP+FN}$
Precision (Positive Predictive Value—PPR)	$PPR = \frac{TP}{TP+FP}$
Recall (True Positive Rate—TPR)	$TPR = \frac{TP}{TP+FN}$
F1 Score	$F1 = 2 * \frac{PPR*TPR}{PPR+TPR} = 2 * \frac{precision*recall}{precision+recall}$

Accuracy is the rate of correctly classified instances, irrespective of class. Precision is the fraction of relevant instances among the retrieved instances, i.e., the number of correctly predicted positives out of the samples that were predicted positive. Recall is the fraction of total relevant instances retrieved, i.e., the number of correctly predicted positives out of the samples that actually were positive. The F1 score is the harmonic mean of precision and recall [28].

Table 6 below gives an overview of the parameters and criteria used for optimization at each level of the training process.

Table 6. Parameters, meta-parameters and criteria.

	Training		Cross-Validation		Testing
	Optimize	Evaluate	Choose/Test	Evaluate	Evaluate
Logistic Regression	Theta	Cross-Entropy	Lambda / Threshold	F1 score	F1 score
Support Vector Machine	Margin	Hinge	Kernel	F1 score	F1 score
Pattern net	Weights	Cross-Entropy	Weights	Cross-Entropy	F1 score

Logistic Regression is used to obtain a model for the classification and prediction of a binary outcome from a set of samples, in the presence of more than one explanatory variable. The procedure is quite similar to multiple linear regression, with the exception that the response variable is binomial [29,30].

The optimization problem is described as $\hat{Y} = \theta * X$, where X is the input data, namely the dependent variables, containing all data-points, plus an intercept term, which helps prevent overfitting. Y is the output data that the algorithm is trying to learn, and θ is the matrix of coefficients used to

estimate Y from X. The challenge is finding the best coefficients, which minimizes the error between the actual Y and the estimate. This is obtained from $\hat{Y} = \theta_{LR} * X$, or, in extended form:

$$
\begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \hat{y}_3 \\ \vdots \\ \hat{y}_n \end{bmatrix} = \begin{bmatrix} \theta_{11} & \cdots & \theta_{1(m+1)} \\ \vdots & \ddots & \vdots \\ \theta_{n1} & \cdots & \theta_{n(m+1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \\ int \end{bmatrix} \tag{4}
$$

Once the best available set of coefficients (θ) is found, a prediction for a given sample is done using the function:

$$
p(x) = \begin{cases} 0, & h_\theta(x) < th \\ 1, & h_\theta(x) \geq th \end{cases} \tag{5}
$$

where $h_\theta(x) = \frac{1}{1+e^{-\theta^T x}}$ and th is the prediction threshold.

The cost function, shown previously in Equation (2), is computed as a measure of the difference between estimated values and the ground truth for the input data. A visualization of the cost for a particular sample is shown in Figure 2, where y is the ground truth and $h_\theta(x)$ is the hypothesis outputted by our model [25].
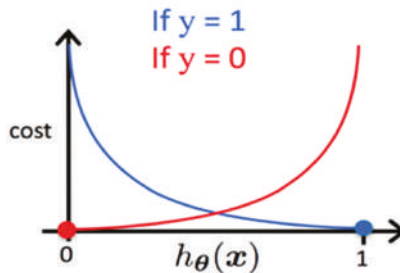
Figure 2. Logistic Regression cost function.

If there are too many features, the learned hypothesis may fit the training set very well, but fail to generalize to new examples, which is called overfitting [26]. Conversely, if there are not enough representative features or if the number of the training examples is not large enough to correctly create the model, it would underfit. Parameters such as alpha (α), the learning rate, or lambda (λ), the regularization parameter, must also be taken into account. The learning rate, α, should not be too large, because it could lead the algorithm to diverge, yet a value too small would cause slow learning and the cost function can get stuck in local minima [31]. Faster gradient descent can be obtained via an adaptive learning rate. Lambda (λ) is the regularization parameter and it determines how strongly a model is penalized for fitting too closely the learned features on the available data [25].

Gradient descent is an optimization algorithm where the potential solution is improved each iteration by moving along the feature gradient in the variable space. While it requires that the target function be differentiable and it is somewhat susceptible to local minima, gradient descent provides a stable and computationally inexpensive algorithm for function optimization. Figure 3 shows a visualization of the Gradient Descent algorithm, for two thetas [25]. This would correspond to a one variable learning problem, for which θ_1 is the coefficient of the independent variable x, and θ_0 is the intercept term. $J(\theta_0, \theta_1)$ is the associated cost for each tuple of the two theta parameters, shown as the landscape on which the gradient descent algorithm searches for optima.
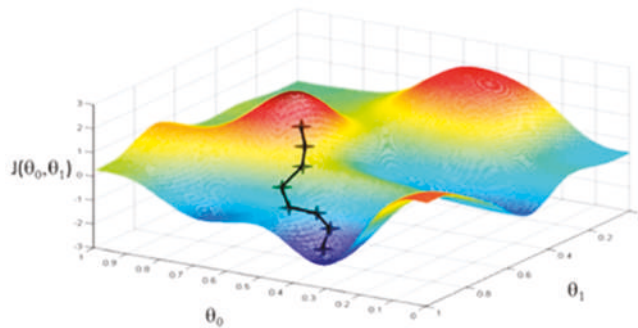
Figure 3. Gradient Descent algorithm.

SVMs, also called Large Margin Classifiers, are supervised learning models used for classification and regression. The input data is classified by optimizing the hyperplane equation so that the distance to the data representing the classes is maximum [32]. In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using kernels, implicitly mapping their inputs into high-dimensional feature spaces [25].

Having the training data, the problem is to determine the parameters of the hyperplane that best separates the data into classes. The cost function for SVMs was shown previously in Equation (3). The parameter C is important for the system because it is responsible for finding the minimum of the cost function, as well as providing regularization. If the value of C is too large then the SVM will change the decision boundary in an intent to integrate the outliers which produces overfitting, but if C is too small, there is a risk of high bias [25]. The similarity function, named kernel, is used by the SVM to draw the decision boundary. Kernels must be semi-positive and symmetrical. The paper investigates the use of three of the most common options: linear, polynomial, and Gaussian (Radial Basis Function), in order to find normally distributed features across the feature map, while also being able to find features that have unusual values [33]. Because of faster execution, an SVM with a Gaussian kernel might outperform Logistic Regression for a medium-sized dataset. SVM finds a solution hyperplane which is as far as possible for the two categories while Logistic Regression does not have this property. In addition, Logistic Regression is more sensitive to outliers than SVM, because the Logistic cost function diverges faster than Hinge Loss [33]. Figure 4 provides a visual representation of the difference between the two cost functions, using the notations of y and $\theta^T x$ discussed earlier in the paper [34].

Artificial neural network models are based on a layer of hidden features (i.e., neurons), which replace the specific features used in other models, and which control the classification. Neural networks have an input layer that matches the dimension of an input sample and an output layer which matches the target variable. The model is optimized by successively tuning the weights of these neurons, through a process called back-propagation. A standard neural network is exemplified in Figure 5 [35,36], where x^i are the features of the input layer, i.e., the dataset features shown previously, w_{ij} are the weights associated with the transition of from the input layer to the hidden layer, u_j are the artificial features of the hidden layer, the number of which is an important meta-parameter, to be chosen by the designer, w'_{jk} are the weights associated with the transition from the hidden layer to the output layer, and u'_k are the output features, or dependent variables, of which there is only one (y) in the case of binary classification, such as presented in this paper.
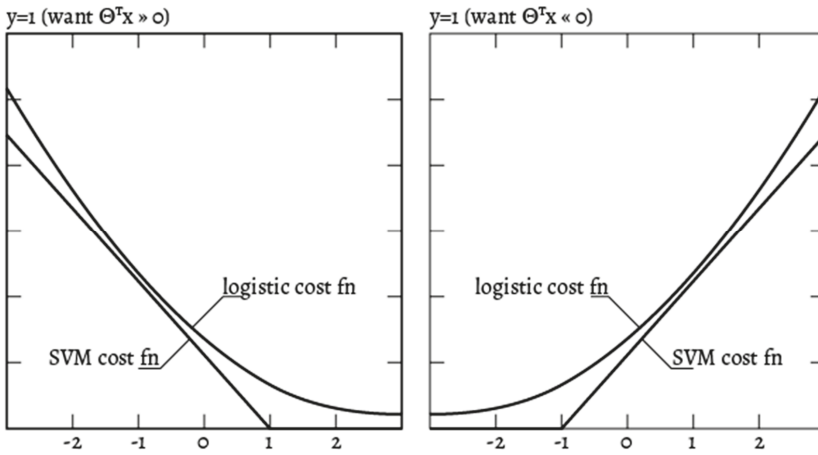
Figure 4. SVM and Logistic Regression cost functions.
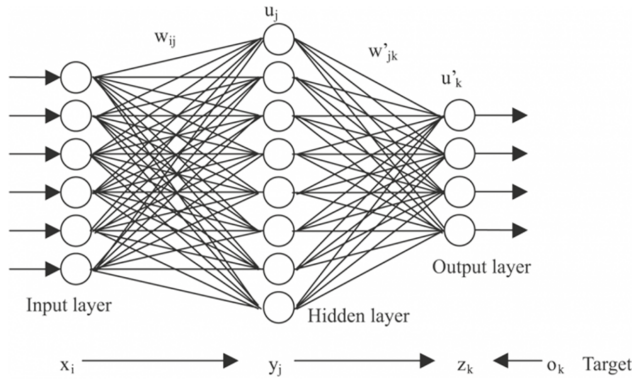
Figure 5. Artificial neural network with one hidden layer.

3. Results

3.1. Logistic Regression

The first learning algorithm investigated is Logistic Regression with polynomial features. The initial dataset is expanded by mapping the second- and third-degree polynomials of the original features, to allow the algorithm to better account for the nonlinearities in the dataset. This means that:

$$X := \begin{bmatrix} X & X^2 & X^3 \end{bmatrix} \tag{6}$$

for all training, cross-validation, and test sets. There are 57 resulting features, plus an intercept term (see also Equation (4)), used to model all other phenomena on which the target might be dependent, which also helps prevent overfitting. This comes out to 58 total features, training on 253 samples. The polynomial degree was chosen to provide a good equilibrium between accounting for nonlinearities and the risk of overfitting the dataset.

One training run of the algorithm takes place for up to 400 iterations, starting from an initial parameter vector (θ) of all zeroes. The decrease of the cost function in an example run using gradient descent is shown in Figure 6.
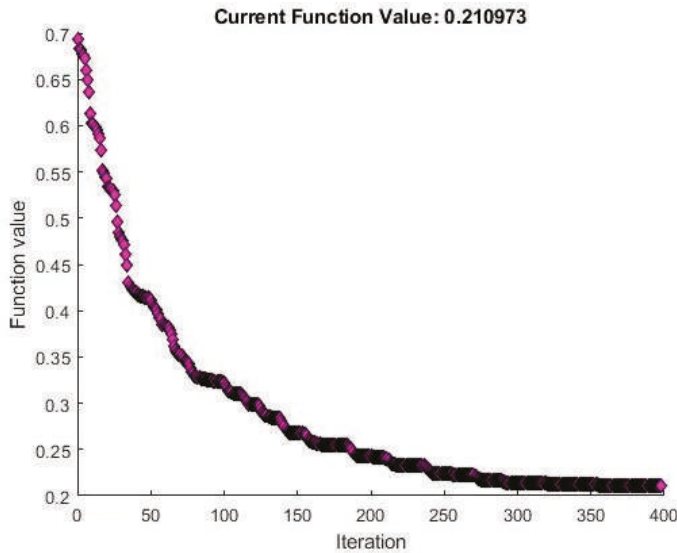
Figure 6. Logistic Regression Cost Function over algorithm iterations.

Cross-validation is done simultaneously for two meta-parameters, lambda (λ) and the prediction threshold (th). The ranges of values for the two meta-parameters are shown in Table 7.

Table 7. Ranges of values for lambda and threshold.

Lambda	0.1	0.3	1	3	10	25	50	100
Threshold	0.1	0.3	0.5	0.7	0.9			

There are therefore 40 combinations of possible {lambda; threshold} parameters. Each of these is used to perform a training run, for which the theta parameter vector is stored, and the F1 score of each resulting model is evaluated, with the best model being kept as the overall Logistic Regression model. Table 8 shows an example of the results obtained for a complete suite of training runs using Logistic Regression. For brevity, results are shown for each lambda, but with the threshold level set at the chosen best.

Table 8. Logistic Regression results on varying lambda.

λ	Training Set				Cross-Validation Set				Test Set			
	Acc	Prec	Rec	*F1*	Acc	Prec	Rec	*F1*	Acc	Prec	Rec	*F1*
0.1	0.91	0.92	0.87	0.89	0.87	0.83	0.87	0.85	0.81	0.80	0.86	0.83
0.3	0.91	0.91	0.88	0.89	0.89	0.87	0.87	0.87	0.83	0.83	0.86	0.84
1	0.90	0.90	0.88	0.89	0.89	0.87	0.87	0.87	0.83	0.83	0.86	0.84
3	0.90	0.90	0.87	0.88	0.89	0.87	0.87	0.87	0.83	0.83	0.86	0.84
10	0.89	0.89	0.87	0.88	0.91	0.88	0.91	0.89	0.83	0.85	0.82	0.84
25	0.89	0.89	0.85	0.87	0.89	0.87	0.87	0.87	0.81	0.82	0.82	0.82
50	0.88	0.88	0.84	0.86	0.89	0.87	0.87	0.87	0.80	0.79	0.82	0.81
100	0.88	0.88	0.84	0.86	0.87	0.83	0.87	0.85	0.80	0.79	0.82	0.81

An alternative approach was explored, whereby cross-validation was done separately, first for the lambda (λ) parameter, using the evaluation of the cost function. The best lambda (λ) and its respective theta parameter vector were chosen, after which the best prediction threshold was evaluated based on

the F1 score. However, this approach consistently obtained poorer results in terms of F1 score to the simultaneous cross-validation method mentioned earlier.

As can be seen from Table 8, the best F1 score on the validation set is 0.89, which leads to the choice of $\lambda = 10$ and $th = 0.5$. Depending on the initial dataset split, which is random, the results do have a slight variation, both in the actual best meta-parameters chosen, as well as the resulting best F1 score for the validation set. The results obtained in Table 8 are among the best recorded for Logistic Regression, after multiple runs, with the evaluation score on the validation set being about as high as can be expected. The final evaluation for the Logistic Regression algorithm is the F1 score of 0.84 recorded on the test set.

The Confusion Matrix for the chosen Logistic Regression model is shown in Figure 7, displaying the Confusion Matrices for each of the training, cross-validation, and test sets, as well as the overall Confusion Matrix, for the entire dataset.

Figure 7. Confusion Matrix for Logistic Regression.

3.2. Shallow Neural Network for Pattern Recognition

The following model is a shallow pattern recognition neural network (PatternNet), with a single hidden layer composed of 20 neurons. The indices for the training, cross-validation, and test sets were kept from the initial dataset split and provided to the pattern net, so that the same sets are used for training and testing, which allows a proper comparison of the results obtained by the models. Figure 8 shows the architecture of a PatternNet neural network, with w being the transition weights of the network and b being the weights of the bias, i.e., intercept, term, which is automatically added by the network upon initialization.
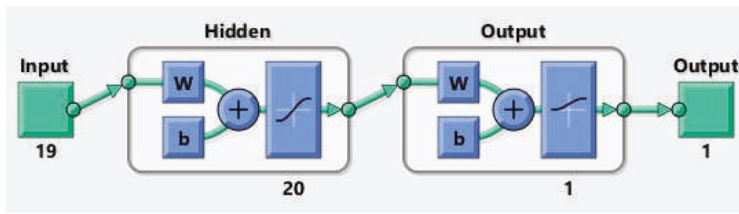
Figure 8. PatternNet architecture.

Training is performed using Scaled Conjugate Gradient, with a similar Cross-Entropy Loss function as the Logistic Regression. The point of cross-validation, in this case, is to supervise the performance of the network and prevent overfitting. Figure 9 shows the performance of the network on the three sets, with the set of weights chosen for best performance on the validation set, displaying the loss function value throughout the training iterations, i.e., epochs, for the training, cross-validation, and test sets.
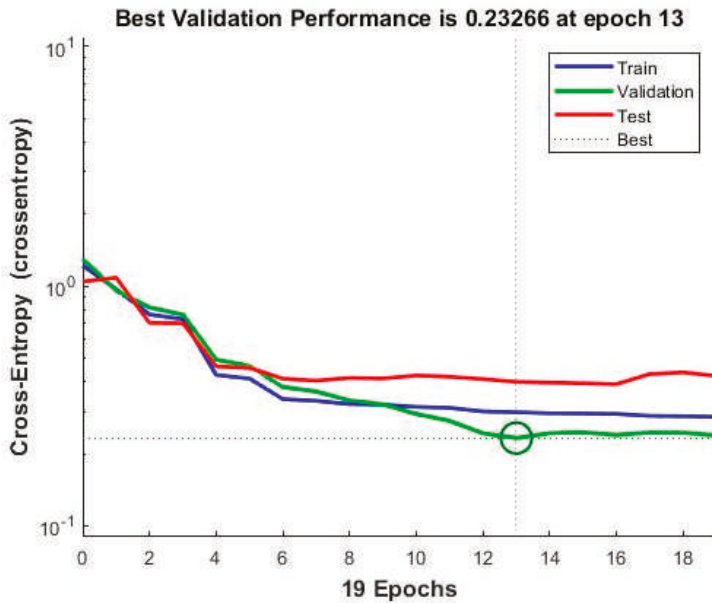
Figure 9. PatternNet performance.

The results of the PatternNet model are shown in Table 9.

Table 9. PatternNet results.

PatternNet	Accuracy	Precision	Recall	F1 Score
Training Set	0.88	0.88	0.84	0.86
Cross-Validation Set	0.85	0.88	0.81	0.84
Test Set	0.89	0.95	0.79	0.86

After multiple runs, the best F1 score obtained by a PatternNet model on this challenging dataset is 0.86. The associated Confusion Matrix is shown in Figure 10 for each of the training, cross-validation, and test sets, as well as the overall Confusion Matrix, for the entire dataset.
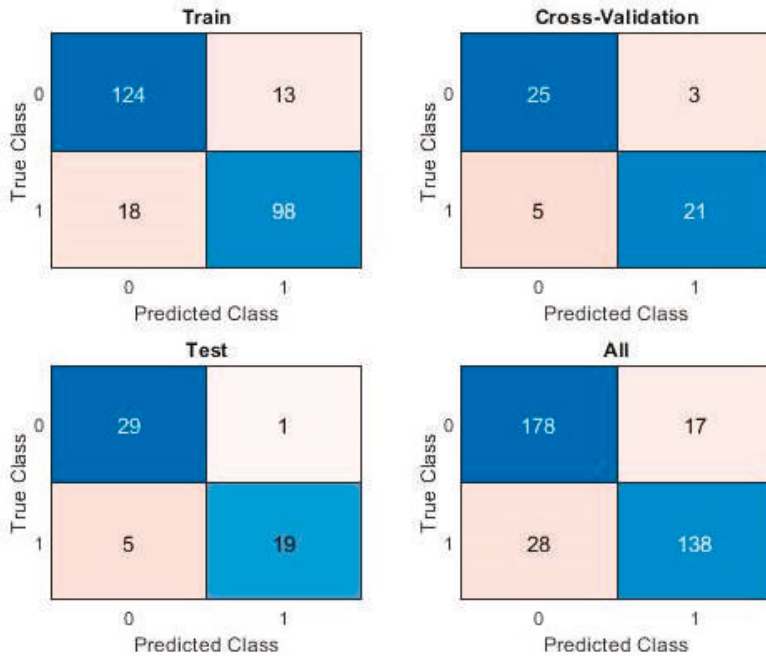
Figure 10. PatternNet Confusion Matrix.

3.3. Support Vector Machines

The SVM algorithm is trained using the same dataset split as for the other two models. Cross-validation is done on the choice of kernel, with the three possible hardcoded options of $K = \{'linear', 'gaussian', 'polynomial(3)'\}$. The maximum degree of the polynomial kernel was chosen to be comparable to the maximum degree of the feature mapping from Logistic Regression. An SVM model is trained for each of these kernel options and the best one is chosen, based on their respective F1 scores. The results are shown below in Table 10.

Table 10. SVM results on varying the kernel.

Kernel	Training Set				Cross-Validation Set				Test Set			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
Linear	0.88	0.87	0.87	0.87	0.89	0.90	0.83	0.86	0.89	0.93	0.87	0.90
Gaussian	0.98	0.98	0.97	0.98	0.80	0.93	0.57	0.70	0.83	1.00	0.70	0.82
Polynomial	1.00	1.00	1.00	1.00	0.81	0.78	0.78	0.78	0.85	0.92	0.80	0.86

The associated Confusion Matrix is shown in Figure 11 below, for each of the training, cross-validation, and test sets, as well as the overall Confusion Matrix, for the entire dataset.
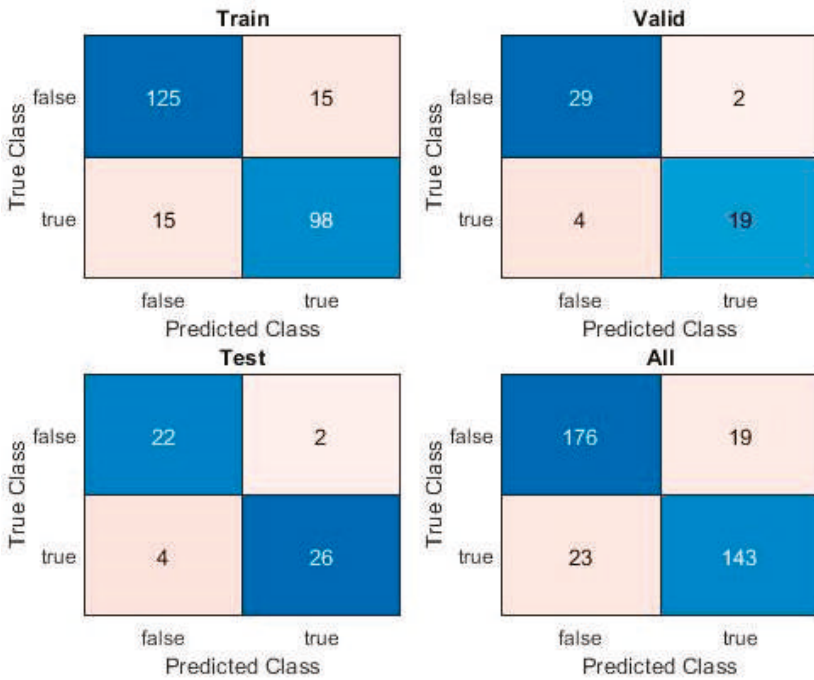
Figure 11. SVM Confusion Matrix.

4. Discussion

Table 11 provides an overall comparison of obtained metrics on the training, validation, and test sets, for each proposed model, using the meta-parameter values selected during validation. The results show that all of the three investigated algorithms perform reasonably well on a challenging dataset, with SVMs having an edge in performance on the test set, as well as maintaining a good equilibrium between the training, cross-validation, and test results.

Table 11. Overall comparison of model results.

	Logistic Regression	PatternNet	SVM
	λ=10	20 Neurons	Linear Kernel
Training Set			
Accuracy	0.89	0.88	0.88
Precision	0.89	0.88	0.87
Recall	0.87	0.84	0.87
F1 score	0.88	0.86	0.87
Cross-Validation Set			
Accuracy	0.91	0.85	0.89
Precision	0.88	0.88	0.90
Recall	0.91	0.81	0.83
F1 score	0.89	0.84	0.86
Test Set			
Accuracy	0.83	0.89	0.89
Precision	0.85	0.95	0.93
Recall	0.82	0.79	0.87
F1 score	0.84	0.86	0.90

A Cochran's Q test [37,38] was performed to evaluate the results, with the null hypothesis being that the three algorithms have similar performance in classification. The test obtains a *p*-value of $5.7268 * 10^{-10}$ and a Cochran Q value of 42.5614, for which the null hypothesis can be safely rejected at the $\alpha = 0.05$ significance level, providing proof that the SVM model significantly outperforms the others.

Furthermore, a comparative post-hoc analysis of the statistical significance of the results obtained through the three models was performed using three McNemar tests [39,40] for each of the three pairs of models. The tests are undertaken on pairs of models, since a McNemar Test can only be run on two models at a time. The first test (exact–conditional McNemar) compares the accuracies of the two models, while the second (Asymptotic McNemar) and third (Mid-*p*-value McNemar) tests assess whether one model classifies better than the other. The null hypothesis for the first test is that the two models have equal predictive accuracies. The null hypothesis for the second and third tests is that one of the models is significantly less accurate than the other. The comparison is not cost sensitive, i.e., it assigns the same penalty for different types of misclassification. Detailed descriptions of McNemar tests and the procedure used can be seen in [40,41]. Table 12 shows the results of the statistical significance tests, where h is the hypothesis test result ($h = 1$ indicates the rejection of the null hypothesis, while $h = 0$ indicates a failure to reject the null hypothesis), p is the *p*-value of the test, and e1 and e2 are the classification losses or misclassification rates of the respective two models (which are the same, irrespective of test).

Table 12. McNemar tests for significance of model performance.

	Exact–Conditional	Mid *p*-Value	Asymptotic
	Logistic Regression vs. PatternNet		
h	0	0	0
p	0.2295	0.1147	0.1103
e1		0.1108	
e2		0.1274	
	Logistic Regression vs. Support Vector Machine		
h	1	0	0
p	0.0488	0.9755	0.9761
e1		0.1108	
e2		0.0720	
	PatternNet vs. Support Vector Machine		
h	1	0	0
p	0.0041	0.9977	0.9976
e1		0.1274	
e2		0.0720	

The comparison between Logistic Regression and PatternNet is somewhat inconclusive. While the exact–conditional test cannot reject the null hypothesis, that the two methods have similar accuracies, neither can the other two tests reject the assumption that one test performs better than the other, although it should be noted that the *p*-value of the first test is twice as large as the values for the second and third tests. However, in the comparisons with the SVM model, both for Logistic Regression, as well as for PatternNet, the null hypothesis of the first test is rejected, meaning that the two methods have dissimilar accuracies. This is further collaborated by the second and third tests, where there is strong evidence that the null hypothesis cannot be rejected, meaning that the SVM model significantly outperforms both the Logistic Regression, as well as the PatternNet, models.

The obtained *p*-values of the exact–conditional McNemar tests are manually corrected using the Discrete Bonferroni–Holm Method for Exact McNemar tests [42–44]. As stated by Holm (1979) [42], "except in trivial non-interesting cases the sequentially rejective Bonferroni test has strictly larger probability of rejecting false hypotheses and thus it ought to replace the classical Bonferroni test at all

instants where the latter usually is applied". Table 13 shows the corrected *p*-values and the resulting decision on the significance of the test.

Table 13. Corrected exact McNemar tests for significance of model performance.

	Test 1	Test 2	Test 3
	LogReg vs. PatternNet	**LogReg vs. SVM**	**PatternNet vs. SVM**
Corrected *p*-value	0.2200	0.0800	0.0120
Significance	0	0	1

The corrected *p*-values are obtained as described in Westfall et al. [43] and implemented in [44]. For the test result to be considered significant, its corrected *p*-value should be below the $\alpha = 0.05$ threshold. The first test is evaluated as expected, since the initial series of McNemar tests showed that there is no statistical difference in the performance of Logistic Regression and PatternNet, by a very high *p*-value, in relation to the threshold. The second test is most interesting, as the hypothesis that the Logistic Regression model and the SVM model have significantly different performance is now rejected, upon strengthening the test. However, it should be noted that it is a borderline decision, and the test would have been accepted at a $\alpha = 0.1$ significance level. The third test reinforces the conclusion that the SVM model performs significantly better, this time compared to the PatternNet model, meeting the significance requirement by a wide margin.

However, all their performances are satisfactory and comparable on the initial metric used. The intention is that the dataset be expanded in the future, at which time further testing will be undertaken to evaluate the ability of each model to generalize, as well as the performance of various ensemble methods, possibly comprising two or more of these same models.

The literature review has yielded few comparable studies for the chosen methods on smartphone applications, without even taking into account the considerably different datasets, features, extraction techniques, etc. that other studies work with. Most papers discuss detection accuracy, with very few using the F1 score metric. The reported accuracy also varies considerably across methods and especially datasets. The final metric values, on the test set, for the linear kernel SVM, compare favorably with most papers in the literature review, for which similar metrics were available. The closest comparison is the application described in [2], which obtains slightly higher metrics on a previous version of the dataset.

While the final hard numbers of the F1 score metric and the statistical significance tests clearly favor the SVM implementation, the difference in performance is oftentimes a relatively small number of misclassified samples, which is one of the issues to keep in mind when working with a small number of data-points.

Therefore, one of the main directions for future research is expanding the dataset, both in terms of the number of data-points (samples), as well as the number of features, where applicable. This would be a great benefit to the ability to train and test the models, as well as allow the investigation of deeper and more complex model architectures, some of which were touched upon in the discussion on the state of the art.

Another interesting future option is ensemble learning, which leverages the positive results of having three good classifiers, whose outputs can then be composed into a final prediction. This is particularly appealing, given the comparatively small difference in accuracies of the three investigated models.

In addition, detection of anomalous behavior from the user's interaction pattern standpoints could be a promising topic for further research, starting from the framework already described.

The obvious future step for the application is that of actual hardware implementation on a working prototype, through which future data can also be more easily gathered. The dataset split procedure, which was discussed at length throughout the paper, should give the model a good ability to generalize to as yet unseen data.

References

1. Gheorghe, L.; Mogosanu, L.; Carabas, M.; Voiculescu, V.G.; Gibson, G. Infrastructure for Learning the Behaviour Of Malicious and Abnormal Applications. In *The International Scientific Conference eLearning and Software for Education*; Carol I National Defense University: Bucharest, Romania, 2015; Volume 1, p. 202.

2. Gheorghe, L.; Marin, B.; Gibson, G.; Mogosanu, L.; Deaconescu, R.; Voiculescu, V.-G. *Smart Malware Detection on Android*; SCN, Wiley: Hoboken, NJ, USA, 2015; Volume 8, pp. 4254–4272.

3. Zhou, Y.; Jiang, X. Dissecting Android Malware: Characterization and Evolution. *IEEE Symp. Secur. Privacy* **2012**, 95–109. [CrossRef]

4. Slideme. Available online: Slideme.org (accessed on 20 March 2020).

5. 42matters. Available online: https://42matters.com (accessed on 20 March 2020).

6. Bakour, K.; Ünver, H.M.; Ghanem, R. The Android malware detection systems between hope and reality. *Sn Appl. Sci.* **2019**, *1*, 1120. [CrossRef]

7. Li, C.; Millis, K.; Zhu, R.; Niu, D.; Zhang, H.; Kinawi, H. Android Malware Detection based on Factorization Machine. *arXiv* **2018**, arXiv:1805.11843v2. [CrossRef]

8. Botnet Research Team. SandDroid—An Automatic Android Application Analysis System. Available online: http://sanddroid.xjtu.edu.cn:8080/# (accessed on 20 March 2020).

9. Qamar, A.; Karim, A.; Chang, V. Mobile malware attacks: Review, taxonomy & future directions. *Future Gener. Comput. Syst.* **2019**, *97*, 887–909.

10. Chalapathy, R.; Chawla, S. Deep learning for anomaly detection: A survey. *Arxiv Prepr.* **2019**, arXiv:1901.03407.

11. Ye, Y.; Li, T.; Adjeroh, D.; Iyengar, S.S. A survey on malware detection using data mining techniques. *ACM Comput. Surv. (Csur)* **2017**, *50*, 1–40. [CrossRef]

12. Dumitrache, I.; Caramihai, S.I.; Moisescu, M.A.; Sacala, I.S.; Vladareanu, L.; Repta, D. A Perceptive Interface for Intelligent Cyber Enterprises. *Sensors* **2019**, *19*, 4422. [CrossRef] [PubMed]

13. Zong, B.; Song, Q.; Min, M.R.; Cheng, W.; Lumezanu, C.; Cho, D.; Chen, H. Deep Auto-Encoding Gaussian Mixture Model for Unsupervised Anomaly Detection. Available online: https://sites.cs.ucsb.edu/~bzong/doc/iclr18-dagmm.pdf (accessed on 20 March 2020).

14. Aljawarneh, S.A.; Vangipuram, R. GARUDA: Gaussian dissimilarity measure for feature representation and anomaly detection in Internet of things. *J. Supercomput.* **2018**, 1–38. [CrossRef]

15. Li, L.; Hansman, R.J.; Palacios, R.; Welsch, R. Anomaly detection via a Gaussian Mixture Model for flight operation and safety monitoring. *Transp. Res. Part C Emerg. Technol.* **2016**, *64*, 45–57. [CrossRef]

16. Erfani, S.M.; Rajasegarar, S.; Karunasekera, S.; Leckie, C. High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. *Pattern Recognit.* **2016**, *58*, 121–134. [CrossRef]

17. Li, K.L.; Huang, H.K.; Tian, S.F.; Xu, W. Improving one-class SVM for anomaly detection. In Proceedings of the 2003 IEEE International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 03EX693), Xi'an, China, 5 November 2003; Volume 5, pp. 3077–3081.

18. Contagio Mobile. Available online: http://contagiomobile.deependresearch.org/index.html (accessed on 20 March 2020).

19. Witayangkurn, A.; Horanont, T.; Sekimoto, Y.; Shibasaki, R. Anomalous event detection on large-scale gps data from mobile phones using hidden markov model and cloud platform. In Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication, Zurich, Switzerland, 8–12 September 2013; pp. 1219–1228.

20. Isohara, T.; Takemori, K.; Sasase, I. Anomaly detection on mobile phone based operational behavior. *Inf. Media Technol.* **2008**, *3*, 156–164. [CrossRef]

21. Sanz, B.; Santos, I.; Ugarte-Pedrero, X.; Laorden, C.; Nieves, J.; Bringas, P.G. Anomaly detection using string analysis for android malware detection. In *International Joint Conference SOCO'13-CISIS'13-ICEUTE'13*; Springer: Cham, Switzerland, 2014; pp. 469–478.

22. Majeed, K.; Jing, Y.; Novakovic, D.; Ouazzane, K. Behaviour based anomaly detection for smartphones using machine learning algorithm. In Proceedings of the International conference on Computer Science and Information Systems (ICSIS'2014), Dubai, UAE, 17–18 October 2014; pp. 67–73.

23. Suarez-Tangil, G.; Tapiador, J.E.; Peris-Lopez, P.; Pastrana, S. Power-aware anomaly detection in smartphones: An analysis of on-platform versus externalized operation. *Pervasive Mob. Comput.* **2015**, *18*, 137–151. [CrossRef]

24. Brownlee, J. Better Deep Learning: Train Faster, Reduce Overfitting, and Make Better Predictions. Machine Learning Mastery, Blog Excerpt. 2018. Available online: https://machinelearningmastery.com/how-to-choose-loss-functions-when-training-deep-learning-neural-networks/ (accessed on 20 March 2020).

25. Arora, S.; Hazan, E. COS 402 – Machine Learning and Artificial Intelligence Fall 2016 Lecture Notes. Available online: https://www.cs.princeton.edu/courses/archive/fall16/cos402/lectures/402-lec5.pdf (accessed on 20 March 2020).

26. Ng, A. CS229 Machine Learning Lecture notes. *Cs229 Lect. Notes* **2000**, *1*, 1–3. Available online: http://cs229.stanford.edu/notes/cs229-notes1.pdf (accessed on 20 March 2020).

27. Gandhi, R.X. Support vector machine—Introduction to machine learning algorithms. Towards Data Science. 2018. Available online: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47 (accessed on 20 March 2020).

28. Powers, D.M. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation. *J. Mach. Learn. Res.* **2011**, *2*, 37–63. [CrossRef]

29. Cramer, J.S. The Origins of Logistic Regression. 2002. Available online: https://papers.tinbergen.nl/02119.pdf (accessed on 20 March 2020).

30. Sperandei, S. Understanding logistic regression analysis. *Biochem. Med. Biochem. Med.* **2014**, *24*, 12–18. [CrossRef] [PubMed]

31. Ravaut, M.; Gorti, S.K. Faster Gradient Descent via an Adaptive Learning Rate. Available online: http://www.cs.toronto.edu/~{}satyag/assets/final_report.pdf (accessed on 20 March 2020).

32. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]

33. Ramo, K. *Hands-On Java Deep Learning for Computer Vision: Implement Machine Learning and Neural Network Methodologies to Perform Computer Vision-Related Tasks*; Packt Publishing Ltd.: Birmingham, UK, 2019.

34. freeCodeCamp Notes on Machine Learning–Support Vector Machine. Available online: https://guide.freecodecamp.org/machine-learning/support-vector-machine/ (accessed on 20 March 2020).

35. Melinte, O.; Vlădăreanu, L.; Gal, I.-A. NAO robot fuzzy obstacle avoidance in virtual environment. *Period. Eng. Nat. Sci.* **2019**, *7*, 318–323. [CrossRef]

36. Templeton, G. Artificial Neural Networks Are Changing the World. What Are They? Available online: https://www.extremetech.com/extreme/215170-artificial-neural-networks-are-changing-the-world-what-are-they (accessed on 20 March 2020).

37. de JP Marques, S. *Applied Statistics: Using SPSS, STATISTICA, and MATLAB*; Springer: Berlin/Heidelberg, Germany, 2003.

38. Jos (10584) COCHRAN Q TEST. MATLAB Central File Exchange. 2020. Available online: https://www.mathworks.com/matlabcentral/fileexchange/16753-cochran-q-test (accessed on 25 April 2020).

39. Agresti, A. *Categorical Data Analysis*; John Wiley & Sons: Hoboken, NJ, USA, 2003; Volume 482.

40. Fagerland, M.W.; Lydersen, S.; Laake, P. The McNemar test for binary matched-pairs data: Mid-p and asymptotic are better than exact conditional. *BMC Med Res. Methodol.* **2013**, *13*, 91. [CrossRef] [PubMed]

41. Mathworks, Matlab Documentation–Compare Predictive Accuracies of Two Classification Models (Testcholdout). Available online: https://www.mathworks.com/help/stats/testcholdout.html (accessed on 15 April 2020).

42. Holm, S. A simple sequentially rejective multiple test procedure. *Scand. J. Stat.* **1979**, *6*, 65–70.

43. Westfall, P.H.; Troendle, J.F.; Pennello, G. Multiple McNemar tests. *Biometrics* **2010**, *66*, 1185–1191. [CrossRef] [PubMed]

44. Bonferroni Holm Correction for Multiple Comparisons in Matlab. Available online: http://freesourcecode.net/matlabprojects/64176/bonferroni-holm-correction-for-multiple-comparisons-in-matlab (accessed on 28 April 2020).

Facial Expressions Recognition for Human–Robot Interaction Using Deep Convolutional Neural Networks with Rectified Adam Optimizer

Daniel Octavian Melinte and Luige Vladareanu *

Department of Robotics and Mechatronics, Romanian Academy Institute of Solid Mechanics,
010141 Bucharest, Romania; octavian.melinte@imsar.ro
* Correspondence: luige.vladareanu@vipro.edu.ro

Abstract: The interaction between humans and an NAO robot using deep convolutional neural networks (CNN) is presented in this paper based on an innovative end-to-end pipeline method that applies two optimized CNNs, one for face recognition (FR) and another one for the facial expression recognition (FER) in order to obtain real-time inference speed for the entire process. Two different models for FR are considered, one known to be very accurate, but has low inference speed (faster region-based convolutional neural network), and one that is not as accurate but has high inference speed (single shot detector convolutional neural network). For emotion recognition transfer learning and fine-tuning of three CNN models (VGG, Inception V3 and ResNet) has been used. The overall results show that single shot detector convolutional neural network (SSD CNN) and faster region-based convolutional neural network (Faster R-CNN) models for face detection share almost the same accuracy: 97.8% for Faster R-CNN on PASCAL visual object classes (PASCAL VOCs) evaluation metrics and 97.42% for SSD Inception. In terms of FER, ResNet obtained the highest training accuracy (90.14%), while the visual geometry group (VGG) network had 87% accuracy and Inception V3 reached 81%. The results show improvements over 10% when using two serialized CNN, instead of using only the FER CNN, while the recent optimization model, called rectified adaptive moment optimization (RAdam), lead to a better generalization and accuracy improvement of 3%-4% on each emotion recognition CNN.

Keywords: computer vision; deep learning; convolutional neural networks; advanced intelligent control; facial emotion recognition; face recognition; NAO robot

---

1. Introduction

Humans use their facial expressions to show their emotional states. In order to achieve an accurate communication between humans and robots, the robot needs to understand the facial expression of the person who it is interacting with. The aim of the paper is to develop an end-to-end pipeline for the interaction between a human and NAO robot using computer vision based on deep convolutional neural networks. The paper focuses on enhancing the performance of different types of convolutional neural networks (CNN), in terms of accuracy, generalization and inference speed, using several optimization methods (including the state-of-the-art rectified Adam), FER2013 database augmentation with images from other databases, asynchronous threading at inference time using the Neural Compute Stick 2 preprocessor, in order to develop a straightforward pipeline for emotion recognition on robot applications, mainly NAO robot. Thereby, we are using the pipeline to first localize and align one face from the input image using a CNN face detector for facial recognition as a preprocessing tool for emotion recognition. Then, the proposal from the face recognition (FR) CNN is used as input for the second CNN, which is responsible for the actual facial emotion recognition (FER CNN).

The performance of this serialized-CNN model, hereafter referred to as NAO-SCNN, depends on many factors such as the number of images in the training dataset, data augmentation, the CNN architecture, loss function, hyperparameters adjusting, transfer learning, fine-tuning, evaluation metrics, etc., which leads to a complex set of actions in order to develop an accurate and real-time pipeline. The interaction between human and robot can be regarded as a closed loop interaction: the images captured by the robot are first preprocessed by the FR CNN module. The output of this module is a set of aligned face proposals, which are transferred to FER CNN. This second module is responsible for emotion recognition. There are seven emotions that are considered: happiness, surprise, neutral, sadness, fear, angriness and disgust. Based on the emotion detected from the images/videos the robot will adjust its behavior according to a set of instructions carried out using medical feedback from psychologists, physicians, educators and therapists in order to reduce frustration and anxiety through communication with a humanoid robot. In case there is no change in the user emotion, the robot will perform another set of actions Since, at this time, the research is focused on the artificial intelligence solution for emotion recognition the set of actions for the robot interaction is basic (the robot displays the user emotion using video, audio and posture feedback) and the conclusions will regard this matter. Based on the results of this study future social studies regarding the behavior of the persons the robot is interacting with will be developed. The expression recognition has been carried out on NAO robot using one Viola–Jones based CNN trained on AffectNet in order to detect facial expressions in children using the NAO robot with a test accuracy of 44.88% [1]. Additionally, a dynamic Bayesian mixture model classifier has been used for FER human interaction with NAO achieving an overall accuracy of 85% on the Karolinska Directed Emotional Faces (KDEF) dataset [2].

In terms of facial emotion recognition, the recent studies focused on developing and achieving neural models with high accuracy using deep learning. For static images different approaches have been developed and involved: pretraining and fine-tuning, adding auxiliary blocks and layers, multitask networks, cascaded networks and generative adversarial networks. In [3] a multitasking network has been developed in order to predict the interpersonal relation between individuals. It aims for high level interpersonal relation traits, such as friendliness, warmth and dominance for faces that coexist in an image. For dynamic image sequences there are other techniques: frame aggregation, expression intensity network and deep spatiotemporal FER network. Zhao et al. proposed a novel peak-piloted deep network that uses a sample with peak expression (easy sample) to supervise the intermediate feature responses for a sample of non-peak expression (hard sample) of the same type and from the same subject. The expression evolving process from non-peak expression to peak expression can thus be implicitly embedded in the network to achieve the invariance to expression intensities [4]. Fine-tuning works for pretrained networks on datasets that include face or human images but the face information remains dominant. The inference will focus on detecting faces rather than detecting expressions. H. Ding et al. proposed a new FER model that deals with this drawback. The method implies two learning stages. In the first training stage the emotion net convolutional layers are unfrozen, while the face net layers are frozen and provide supervision for emotion net. The fully connected layers are trained in the second stage with expression information [5]. In [6] a two stage fine-tuning is presented. A CNN pretrained on an ImageNet database is fine-tuned on datasets relevant to facial expressions (FER2013), followed by a fine-tuning using EmotiW. To avoid variations introduced by personal attributes a novel identity-aware convolutional neural network (IACNN) is proposed in [7]. An expression-sensitive contrastive loss was also developed in order to measure the expression similarity. There are studies that research other aspects of emotion recognition, such as annotation errors and bias, which are inevitable among different facial expression datasets due to the subjectivity of annotating facial expressions. An inconsistent pseudo annotations to latent truth (IPA2LT) framework has been developed by Jiabei Zeng et al. to train a FER model from multiple inconsistently labeled datasets and large scale unlabeled data [8]. A novel transformations of image intensities to 3D spaces was designed to simplify the problem domain by removing image illumination variations [9]. In [10]

a novel deep CNN for automatically recognizing facial expressions is presented with state-of-the art results, while in [11] a new feature loss is developed in order to distinguish between similar features.

There are different types of databases for FER such as Japanese Female Facial Expressions (JAFFE) [12], Extended Cohn–Kanade (CK+) [13], FER 2013 [14], AffectNet [15], MMI (MMI, 2017) [16,17], AFEW [18] and Karolinska Directed Emotional Faces (KDEF) [19].

The CK+ is the most used dataset for emotion detection. The performance of CNN models trained on this dataset are greater than 95% due to the fact that the images are captured in a controlled environment (lab) and the emotions are overacted. The best performance was reached by Yang et al. with a performance of 97.3% [20]. The dataset contains 593 video frames from 123 subjects. The frames vary from 10 to 60 and display a translation from neutral to one of the six desired emotion: anger, contempt, disgust, fear, happiness, sadness and surprise. Not all the subjects provide a video sequence for each emotion. The images in the dataset are not divided into training, validation and testing so it is not possible to perform a standardized evaluation.

The Jaffe is a laboratory-controlled dataset with fewer video frames than CK+. The CNN models tested using this database have performances over 90%, with Hamester et al. reaching 95.8% [21]. The KDEF is also a laboratory-controlled database that focuses on the emotion recognition from five different camera angles. The FER2013 is the second widely used dataset after CK+. The difference is that FER 2013 images are taken from the web and are not laboratory-controlled, the expressions are not exaggerated, thus harder to recognize. There are 28,079 training images, 3589 validation images and 3589 test images that belong to seven classes. The performances using this dataset do not exceed 75.2%. This accuracy has been achieved by Pramerdorfer and Kampel [22], while other researches reached 71.2% using linear support vector machines [23], 73.73% by fusing aligned and non-aligned face information for automatic affect recognition in the wild [24], 70.66% using an end-to-end deep learning framework, based on the attentional convolutional network [25], 71.91% using three subnetworks with different depths [26] and 73.4 using a hybrid CNN–scale invariant feature transform aggregator [27].

The research carried out in this paper aims to improve the accuracy of FER 2013 based CNN models by adding laboratory controlled images from CK+, JaFFE and KDEF. Hopefully there is an emotion compilation that satisfies our requirements and is available on Kaggle [28]. This dataset is divided in 24,336 training, 6957 validation and 3479 test sets. The FER2013 images make up the biggest part of the dataset, around 90% for each emotion.

When it comes to face detection the backbone networks such as VGGNet [29], Inception by Google [30] or ResNet [31] play an important role. In [32] the DeepFace architecture, a nine-layer CNN with several locally connected layers, is proposed for face detection. The authors reported an accuracy of 97.35% on the labeled faces in the Wild dataset. The FaceNet model uses an Inception CNN as the backbone network in order to obtain an accuracy of 99.63% on the widely used labeled faces in the Wild dataset [33]. Another widely used FR network is the VGGFace, which uses a VGGNet base model trained on data collected from the Internet. The network is then fine-tuned with a triplet loss function similar to FaceNet and obtains an accuracy of 98.95% [34]. In 2017, SphereFace [35] used a 64-layer ResNet architecture and proposed an angular softmax loss that enables CNNs to learn angularly discriminative features with an accuracy of 99.42%.

In terms of the human–robot interaction, the research is concentrated on different levels and implies, among others, the grasping configurations of robot dexterous hands using Dezert–Smarandache theory (DSmT) decision-making algorithms [36], developing advanced intelligent control systems for the upper limb [37–41] or other artificial intelligence techniques such as neutrosophic logic [36,41,42], extenics control [43,44] and fuzzy dynamic modeling [45–47], applicable for the human–robot interaction with feedback through facial expressions recognition. The research in this paper is focused on developing an AI based computer vision system to achieve the communication human–NAO robot aiming the future researches to develop an end-to-end pipeline system with advanced intelligent control using feedback from the interaction between a human and NAO robot.

2. Methods

Direct training of deep networks on relatively small facial datasets is prone to over fitting. Public databases are no larger than 100k images. To compensate this, many studies used data preprocessing or augmentation and additional task-oriented data to pretrain their self-built networks from scratch or fine-tuned on well-known pretrained models.

In order to choose the appropriate solution for the human–NAO robot face detection two different types of object detectors have been taken into account: single shot detector (SSD) and regional proposal network (RPN). These two architectures have different outputs in practice. The SSD architecture is a small CNN network (around 3 million parameters) with good accuracy for devices with limited computation power. The inference time is very small compared to other large object detectors. The faster region-based convolutional neural network (Faster R-CNN), instead, has high accuracy, better results on small objects, but the inference time takes longer than SSD models and is not suitable for real time applications with low computational power.

The FR architecture consists of a base model (Inception for SSD architecture and ResNet + Inception for Faster R-CNN) that is extended to generate multiscale or regional proposals feature maps. These models were pretrained on different databases: the SSD+InceptionV2 was trained on the common objects in context (COCO) dataset, while the Faster R-CNN was pretrained on a much larger database, the Google OpenImage database.

2.1. Face Detection CNN

The human faces share a similar shape and texture, the representation learned from a representative proportion of faces can generalize well to detect the others, which are not used in the network training process. The performance of the trained model depends on many factors such as number of images in the training dataset, data augmentation, the CNN architecture, loss function, hyper-parameters adjusting, transfer learning, fine-tuning, evaluation metrics, etc., which leads to a complex set of actions in order to develop the entire pipeline.

The face detector was used to localize faces in images and to align them to normalized coordinates afterwards. The CNN architecture is made up of a backbone network, a localization CNN and the fully connected layers for classification. The backbone networks used for facial recognition were the Inception V2 for SSD architecture and ResNet-InceptionV2 for Faster R-CNN, respectively. In terms of the localization network there were two types of architecture: SSD and Faster R-CNN. The SSD architecture adds auxiliary CNNs after the backbone network, while the Faster-RCNN uses a regional proposal network (RPN) for proposing regions in the images, which were further sent to the CNN, which was also used as a backbone. The classification layers of both face detectors were at the top of the architecture and used the softmax loss function together with L2 normalization in order to adjust the localization of the face region and to classify the image.

2.1.1. Inception Based SSD Model

The SSD architecture uses an Inception V2 pretrained model as a backbone. The Inception model was designed for solving high variation in the location of information, thus is useful for localization and object detection when is added as base network for SSD and RPN architectures. Different types of kernels with multiple sizes (7×7, 5×5 and 3×3) were used. Larger kernels can look for the information that is distributed more globally, as the smaller one search in the information that is not as sparse. There is one important filter that is also used for this type of CNN, which is the 1×1 convolution for reducing or increasing the number of feature maps. The network is 22 layers deep when counting only layers with parameters (or 27 layers with pooling). The overall number of layers (independent building blocks) used for the construction of the network is about 100. The SSD architecture adds auxiliary structure to the network to produce multiscale feature maps and convolutional predictors for detection. At prediction time, the network generates scores for the presence of each object category in

each default box and produces adjustments to the box to better match the object shape. Additionally, the network combines predictions from multiple feature maps with different resolutions to naturally handle objects of various sizes [48].

2.1.2. Inception-ResNetV2 Based Faster R-CNN Model

The face detection and alignment is tested with a second object detector, the Faster R-CNN with Inception and ResNet as backbone CNN. Faster R-CNN convolutional networks use regional proposal method to detect the class and localization of the objects. The architecture is made up of two modules: the regional proposal network (RPN) and the detection network. The RPN has CNN architecture with anchors and multiple region proposals at each anchor location to output a set of bounding boxes proposals with a detection score. The detection network is a Fast R-CNN network that uses the region proposals from RPN to search for objects in those regions of interest (ROI). ROI pooling was performed and then resulted feature maps pass through CNN and two fully connected layers for classification (Softmax) and bounding box regression. The RPN and the detection network share a common set of convolutional layers in order to reduce computation [49].

As comparison, it is a common fact that the Faster R-CNN performs much better when it comes to detecting small object, but shares the same accuracy with the SSD network when detecting large objects. Due to the complex architecture of Faster R-CNN, the inference time is three times higher than SSD architecture.

For *face detection* the backbone of the Faster R-CNN is an Inception+ResNetV2 network that has the architecture similar to Inception V4. The overall diagram implies a five convolutions block in the first layers, five Inception-ResNetV2-typeA, ten Inception-ResNetV2-typeB and five Inception-ResNetV2-typeC. The type A and type B Inception-RResNet layers were followed by a reduction in feature size, while the type C Inception-ResNet layer was followed by average pooling. The average pooling was performed before fully connected (FC) layers and the dropout was chosen to be 0.8.

2.1.3. Pipeline for Deep Face Detection CNN

Transfer learning and fine-tuning has been used for training SSD and Faster R-CNN models pretrained on COCO and Open Image databases. The input for the fine-tuned training consisted of 2800 face images randomly selected from the Open ImageV4dataset images [50,51]. The dataset was split into three categories: 2000 images for training, 400 for validation and 400 for testing.

The hyperparameters for training each network are described below. The values were obtained under experiments and represent the best configuration in terms of accuracy, generalization and inference speed obtained after training each model. Due to the limitations of the graphical process unit used for training the number of operations was reduced for each training epoch, which implied a low number of images per batch, especially for Faster R-CNN model.

For fine-tuning the SSD architecture, the hyper-parameters were set as follows:

- Batch size = 16 images;
- Optimizer: momentum RMS optimizer;
- Initial learning rate: 6.0×10^{-5};
- Learning rate after 100k steps: 6.0×10^{-6};
- Momentum optimizer: 0.9;
- Number of epochs: 150k;
- Dropout: 0.5.

For fine-tuning the Faster R-CNN architecture, the hyperparameters were set as follows:

- Batch size = 1 images;
- Optimizer: momentum schedule learning;
- Initial learning rate: 6.0×10^{-5};

- Momentum optimizer: 0.9;
- Number of epochs: 150k;
- Dropout: 0.5.

2.2. Facial Emotion Recognition CNN

The database used for training FER CNN is a selection of uncontrolled images from FER2013 and laboratory controlled images from CK+, JaFFE and KDEF [28]. This dataset was divided in 24,336 training, 6957 validation and 3479 test sets. The labeling of the training and testing dataset were previously made by the authors of the database and, in addition, were verified by our team in order to avoid bias. A small amount of the images that did not meet our criteria in terms of class annotations and distinctiveness were dropped or moved to the corresponding class. In addition, data augmentation was used during training in order to increase the generalization of the model. A series of rotations, zooming, width and height shifting, shearing, horizontal flipping and filling were applied to the training dataset. The FER2013 images represents the majority of the dataset, around 90% for each emotion.

The facial expressions were divided in seven classes: angry, disgust, fear, happy, neutral, sad and surprise. Training classes distribution over the dataset was angry 10%, disgust 2%, fear 3%, happy 26%, neutral 35%, sad 11% and surprise 13%. The validation and test set followed the same distribution.

The CNN models used for FER were pretrained on the ImageNet database and could recognize objects from 1000 classes. We did not need a SSD or RPN architecture as the face localization was already achieved with face detection CNN. ImageNet did not provide a class related to face or humans but there were some other classes (e.g., t-shirt or bowtie) that helped the network to extract these kinds of features during the prelearning process. This is related to the fact that the network needs these features in order to classify emotions related to classes. Taking into account that only some bottleneck layers will be trained, we would use transfer learning and fine-tuning, as follows: in the first phase the fully connected layers of the pretrained model were replaced with two new randomly initialized FC layers that were able to classify the input images according to our dataset and classes. During this warm-up training all the convolutional layers were frozen allowing the gradient to back-propagate only through the new FC layers. In the second stage the last layers of the convolutional networks were unfrozen, where high-level representations were learned allowing the gradient to back-propagate through these layers but with a very small learning rate in order to allow small changes to the weights. The representation of the three CNN (VGG, ResNet and Inception V3) and the transfer learning approach is shown in Figure 1.
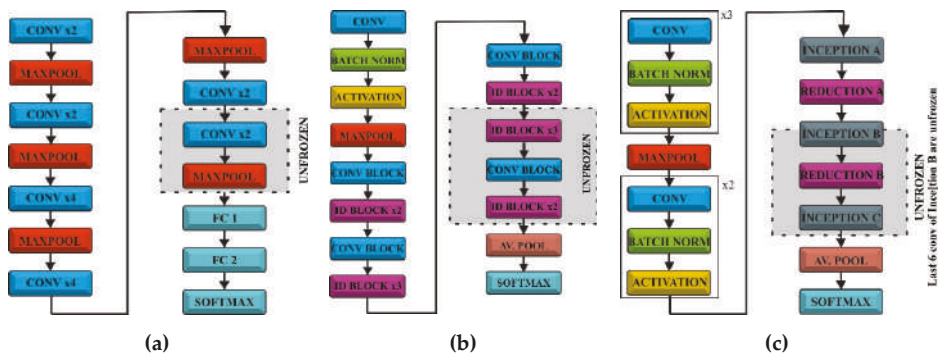
Figure 1. Architectures of the models used for facial emotion recognition (FER): (**a**) VGG; (**b**) ResNet50 and (**c**) Inception V3.

2.2.1. VGG16

In order to avoid over fitting several techniques were taken into account, such as shuffling data during training or adopting dropout. The images were shuffled during training to reduce variance and to make sure that the batches are representative to the overall dataset. On the other hand, using dropout in the fully connected layer reduced the risk of over fitting by improving the generalization. This type of regularization reduced the size of the network by "dropping" an amount of different neurons at each iteration. The CNN was made of five convolutional blocks (16 or 19 layers) each followed by feature map reduction using max-pooling. The bottleneck layer output was passed through an average pooling in order to flatten the feature map to a $1 \times 1 \times 4096$ and the representation was forwarded to the FC network. The FC had one dense layer of 512 neurons with rectified linear unit (ReLu) activation function and dropout of 0.7 and a Softmax classifier.

For fine-tuning the VGG model, the hyperparameter values were obtained under experiments and represent the best configuration in terms of accuracy, generalization and inference speed after training each model and were set as follows:

- Batch size = 32 images;
- Optimizer: momentum RMS optimizer;
- Initial learning rate: 1.0×10^{-4};
- Momentum optimizer: 0.9;
- Number of epochs: 50, for training the new FC layers;
- Number of epochs: 50, for training the last convolutional layers;
- Dropout: 0.5.

2.2.2. ResNet

ResNet is a deep convolutional network that used identity convolutional blocks in order to overcome the problem of vanishing gradients (Figure 2). The gradient may become extremely small as it is back-propagated through a deep network. The identity block use shortcut connections, which are an alternate path for the gradient to flow through, thus solving the problem of vanishing gradient. We would use ResNet with 50 convolutions, which were divided into five stages. Each stage had a convolutional and an identity block, while each block had three convolutions, with 1×1, 3×3 and 1×1 filters, where the 1×1 kernel (filter) was responsible for reducing and then increasing (restoring) dimensions. The parameter-free identity shortcuts are particularly important for the bottleneck architectures. If the identity shortcut is replaced with the projection, one can show that the time complexity and model size are doubled, as the shortcut is connected to the two high-dimensional ends. Thereby, identity shortcuts lead to more efficient models for the bottleneck designs [31].

Fine-tuning of ResNet architecture, presented at the bottom of Figure 2, implied the following hyperparameter configuration.

- Batch size = 16 images;
- Optimizer: rectified Adam;
- Initial learning rate: 1.0×10^{-4};
- Momentum optimizer: 0.9;
- Number of epochs: 50, for training the new FC layers;
- Number of epochs: 50, for training the last convolutional layers;
- Dropout: 0.5.

The values were obtained under experiments and represent the best configuration in terms of accuracy, generalization and inference speed after training each model.
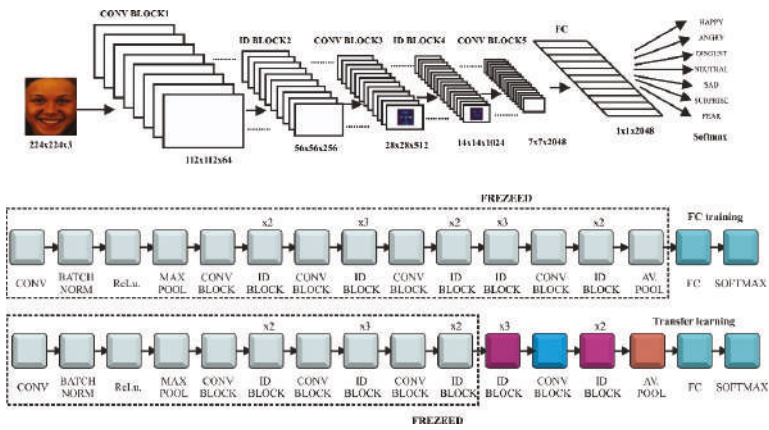
Figure 2. ResNet model using transfer learning and fine-tuning.

2.2.3. InceptionV3

Inception V3, presented in Figure 3, is a deep neural network with 42 layers, which reduced the representational bottlenecks. It was composed of five stem convolutional layers, three type A Inception blocks, followed by a type A reduction block, four type B Inception blocks and one reduction block, two type C Inception blocks followed by an average pooling layer and the fully connected network. In order to reduce the size of a deep neural network the factorization was taken into account. Different factorization modules were introduced in the convolutional layers to reduce the size of the model in order to avoid over fitting. Neural networks performed better when convolutions did not change the size of the input drastically, reducing the dimensions too much causing loss of information. One factorization implied splitting 5×5 convolutions to two 3×3 convolution (type A inception block). In addition, factorization of the $n \times n$ filter to a combination of $1 \times n$ and $n \times 1$ asymmetric convolutions (type B inception block) was found to dramatically reduce the computation cost. In practice, it was found that employing this factorization does not work well on early layers, but it gives very good results on medium grid-sizes [52]. The last factorization taken into account was the high dimensional representations by replacing two of the 3×3 convolution with asymmetric convolutions of 1×3 and 3×1.
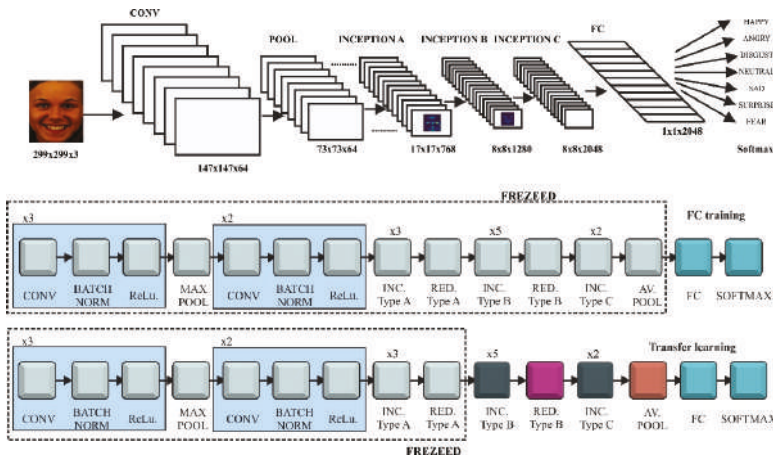
Figure 3. Inception model using transfer learning and fine-tuning.

For fine-tuning the Inception V3 model, the hyperparameters configuration is as follows:

- Batch size = 32 images;
- Optimizer: Adam;
- Initial learning rate: 1.0×10^{-4};
- Momentum optimizer: 0.9;
- Number of epochs: 50, for training the new FC layers;
- Number of epochs: 50, for training the last convolutional layers;
- Dropout: 0.5.

The values were obtained under experiments and represent the best configuration in terms of accuracy, generalization and inference speed after training each model.

2.3. Optimization Using Rectified Adam and Batch Normalization

There are several methods that accelerate deep learning model optimization by applying adaptive learning rate, such as the adaptive gradient algorithm (Adagrad), Adadelta, Adamax, root mean square propagation (RMSprop), adaptive moment optimization (Adam) or Nesterov adaptive moment optimization (Nadam). Rectified Adam is a state-of-the-art version of the Adam optimizer, developed by [53], which improves generalization and introduces a term to rectify the variance of the adaptive learning rate, by applying warm up with a low initial learning rate.

Computing the weights according to the Adam optimizer:

$$w_t = w_{t-1} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon} \tag{1}$$

The first moving momentum:

$$m_t = (1 - \beta_1) \sum_{i=0}^{t} \beta_1^{t-i} g_i \tag{2}$$

The second moving momentum:

$$v_t = (1 - \beta_2) \sum_{i=0}^{t} \beta_2^{t-i} g_i^2 \tag{3}$$

The bias correction of the momentums:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{4}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{5}$$

Adding the rectification term in Equation (1), the recent variant of Adam optimization, named rectified Adam (RAdam), has the form:

$$w_t = w_{t-1} - \eta r_t \frac{\hat{m}_t}{\sqrt{\hat{v}_t}} \tag{6}$$

where the step size, η, is an adjustable hyperparameter and rectification rate is:

$$r_t = \sqrt{\frac{(p_t - 4)(p_t - 2)p_\infty}{(p_\infty - 4)(p_\infty - 4)p_t}} \tag{7}$$

while $p_t = p_\infty - \frac{2t\beta_2^t}{1 - \beta_2^t}$ and $p_\infty = \frac{2}{(1 - \beta_2^t)} - 1$.

When the length of the approximated simple moving average is less or equal than 4, the variance of the adaptive learning rate is deactivated. Otherwise, the variance rectification term is calculated and parameters are updated with the adaptive learning rate.

After applying batch normalization to the activation function output of the convolutional layers, the normalized output will be:

$$x_i = \gamma \hat{w}_i + \beta \tag{8}$$

where γ and β are parameters used for scale and shift that are learned during training. Moreover, the weights normalization over a mini-batch is:

$$\hat{w}_i = \frac{w_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} \tag{9}$$

where the mini- batch average:

$$\mu_B = \frac{1}{m} \sum_{i=1}^{m} w_i \tag{10}$$

and the mini-batch variance:

$$\sigma_B^2 = \frac{1}{m} \sum_{i=1}^{m} (w_i - \mu_B)^2 \tag{11}$$

For FER our end-to-end human–robot interaction pipeline used convolutional neural network models that were trained using batch normalization after ReLu activation and RAdam optimizer. As we will later see in the paper the best results using RAdam have been obtained for the ResNet model.

2.4. Experiment Setup

In order to develop an end-to-end pipeline for the interaction between human and NAO robot using computer vision based on deep convolutional neural networks a preprocessing CNN for facial detection was added before the FER CNN.

The entire facial expression pipeline was implemented on the NAO robot and is presented in Figure 4. The system was divided in four parts: the NAO robot image caption (NAO camera), the face recognition model (FR CNN) and the facial emotion recognition model (FER CNN) and robot facial expression (output to the user/human). For image caption and output to the user we used the Naoqi library functions running on the robot, while the FR and FER models were uploaded to the robot and enabled when the emotion recognition was activated.
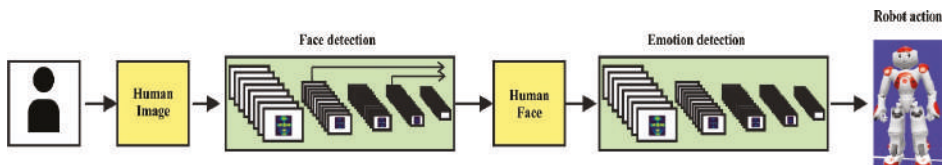
Figure 4. End-to-end emotion detection pipeline.

NAO is a humanoid robot developed by Aldebaran (Softbank Robotics), it has a height of 57 cm, weighs 4.3 kg and is widely used in the research and education due to its good performances, small size, affordable price and the wide range of sensors it is equipped with. Since the scripts can be developed in several programming languages and can be compiled both locally (on the robot) and remotely, NAO can be used in various applications. In order to achieve a human–robot interaction NAO's top front camera was used for taking pictures or videos of the person in front and NAO's voice module together with eye and ear LEDs for outputting robot emotion. NAO has two identical RGB video cameras located in the forehead, which provide a 640 × 480 resolution at 30 frames per second. The field of

view is 72.6°DFOV with a focus range between 30 cm and infinity [54]. Eye and foot LEDs are RGB full color, so it is possible to combine the three primary colors (red, green and blue) in order to obtain different colors. This feature will be used to associate an emotion to one color: happy is green, red is angry, blue is sad, disgust is yellow, neutral is black, surprise in white and fear is orange (Figure 5). The intensity of the color will be adjusted depending on the probability of emotion detection.

In terms of computer vision, NAO's embedded module is able to detect faces with a miss rate of 19.16% for frontal head positions, based on a face detection solution developed by OMRON. The NAO robot software (Choregraph) and libraries share a computer vision module for facial recognition named ALFaceDetection, which is able to recognize one particular face previously learned and stored in its limited flash memory. The learning and storing process of one face is tedious and involves several steps, in which, for example, NAO checks that the face is correctly exposed (no backlighting, no partial shadows) in three consecutive images. Compared to the NAO face detection module, the CNN based facial recognition, which represents the preprocessing module of the pipeline presented in the paper, was straightforward and could recognize one random face with an accuracy of 97.8% as it will be presented in the Section 3.1.1.

| disgust | sad | surprise | angry | neutral | fear | happy |

Figure 5. NAO color expression.

The training of the FR and FER models were performed on a GeForce GTX 1080 graphics processing unit (GPU) running on compute unified device architecture (CUDA) and CUDA deep neural network library (CUDNN) with the following specifications: memory 8 GB GDDR5X, processor 1.8 GHz and 2560 CUDA cores. The inference (detection) was running on the NAO robot, which was equipped with an Intel Atom Z530 1.6 GHz CPU and 1 GB RAM memory.

In order to increase the inference time we used a neural network preprocessor developed by Intel, called the Neural Compute Stick 2 (NCS 2), together with asynchronous threading, which boosts the NAO performances from 0.25 frames per second (FPS) to 4-6 FPS on our FR+FER pipeline. The inference speed when using only one CNN, reached 8-9 FPS with NCS 2.

3. Results

3.1. Facial Recognition

3.1.1. ResNet-Inception Based Faster R-CNN Model

The loss variation of Faster R-CNN is presented in Figure 6. It can be seen that the loss dropped very quickly. This happened because the complex architecture of R-CNN, which is a combination of ResNet and Inception models, increased the training accuracy. Another parameter that artificially boosted the accuracy of the network was the number of images in the batch. In our case there was only one image per batch due to the limitations of the GPU. Since the gradient descent was applied to every batch, not to the entire dataset, it was normal to have a rapid drop and small loss overall. After the rapid drop to a value of 0.4 in the first 5k steps, the loss decreased slowly until it started to converge at around 60k steps with the value of 0.25. The accuracy of Faster R-CNN model was 97.8% using the Pascal visual object class (VOC) evaluation metrics.
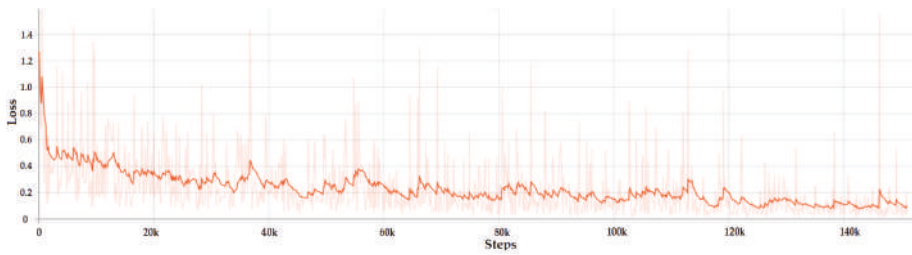
Figure 6. Total loss of the faster region-based convolutional neural network (Faster R-CNN) model.

3.1.2. Inception Based SSD Model

In contrast with Faster R-CNN the smoothened loss for SSD Inception has an exponential dropping and converged at around 120k steps. The loss variation can be observed in Figure 7, with the minimum value around 1.8. The precision of SSD Inception model was 97.42% using the Pascal VOC evaluation metrics.
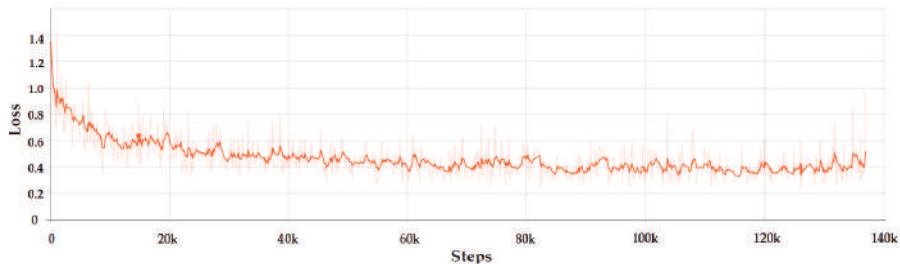
Figure 7. Total loss of the Inception based single shot detector (SSD) model.

3.2. Facial Emotion Recognition

3.2.1. VGG16

The fine-tuning of the FC layers and training of the last convolutional layers of the VGG model for FER are presented in Figure 8. The training (red line) and validation loss (blue line) during the "warm-up" drop after several epochs and, after completing the FC layers training, the train loss was 1.1 and validation loss was 0.8 (Figure 8a). The accuracy was 0.6 (60%) for training (purple line) and 0.7 (70%) for validation (gray line). Once the classification layers weight was initialized according to new output classes and learns some patterns from the new dataset, it was possible to skip to the next step. The last two convolutional layers and the max pooling of the VGG model were unfrozen in order to get feature maps that relate to emotion recognition. Initially, the training was set for 100 epochs but because the model started to over fit, the learning process was stopped after 50 epochs. When the training completed, the train loss (red line) dropped to 0.36 and the validation loss (blue line) started to become instable and converge around 0.5. The accuracy of training (purple line) increased to 0.87 (87%) while the loss (gray line) reached 0.84 (84%; Figure 8b).

3.2.2. ResNet

In Figure 9 the fine-tuning of the fully connected layers and last unfrozen layers of ResNet model is presented. The fully connected network was made of one dense layer of 1024 neurons activated by a ReLu function and one dense layer for classification activated by a Softmax function. The model that achieved the best results was compiled using a rectified Adam optimizer with a learning rate value set at 0.0001
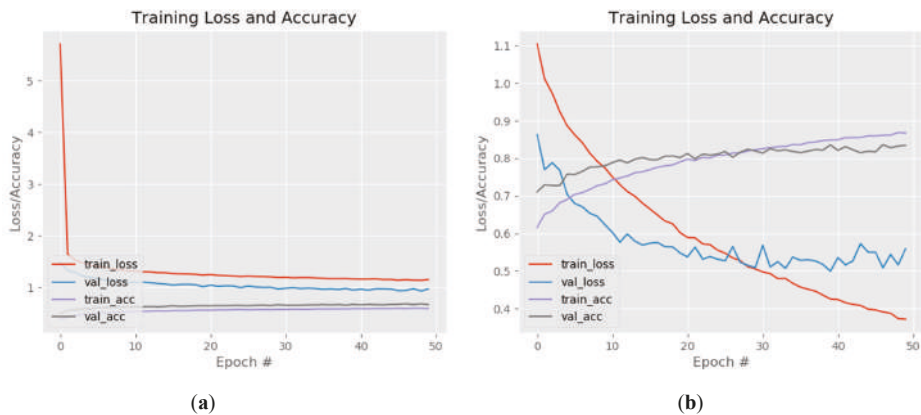
Figure 8. VGG training loss and accuracy: (**a**) fully connected (FC) layers warm-up and (**b**) training of last two convolutional layers.

The training of these two layers was made while keeping all the model layers frozen. The losses of the training and validation set converged around the value of 1.1 and the accuracy of both sets around 0.6 (Figure 9a). For the actual training of the entire model we tried different fine-tunings of the hyperparameters while unfreezing different top layers of the ResNet architecture. The setting that achieved the best results was: batch size: 16, optimizer rectified Adam, learning rate—0.0001 and unfreezing the layers from the 31st convolution (the three ID blocks before the last convolutional block). This model reached a training accuracy of 0.9014 (90.14%) when the learning process was stopped, after 50 epoch, because the validation loss started to increase and model was over fitting (Figure 9b). In other tests the training accuracy increased to 0.96 while over fitting the model but for our FER inference system we want the network to have a good generalization on different datasets. The ResNet gave the best results of all tested FER architectures, in terms of generalization and was used for the NAO-SCNN inference system pipeline.
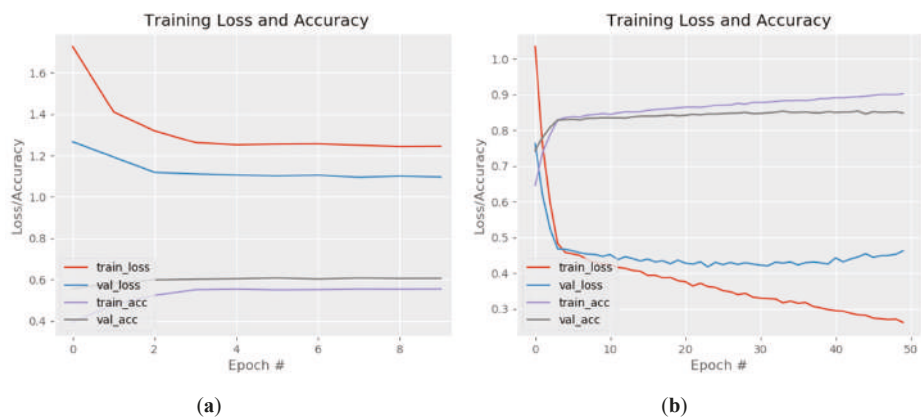
Figure 9. ResNet training loss and accuracy of the FC layers warm-up (**a**) and training of the last five convolutional layers (**b**).

3.2.3. InceptionV3

The results of fine-tuning the InceptionV3 network are presented in Figure 10. As with the VGG and ResNet models the training was divided in two steps: first the weights of the fully connected layers

were updated while freezing the rest of the network and then some of the layers of the Inception B block together with the Reduction B and Inception C block were unfrozen (Figure 10a). The fully connected network was made of one dense layer with 512 neurons activated by a ReLu function and one dense layer for classification activated by a Softmax function. The model was compiled using rectified Adam, Adam, RMSProp and stochastic gradient descent (SGD) optimizers with a learning rate value set at 0.0001. The training of the entire network was made by combining different hyperparameters values while unfreezing different top layers of the InceptionV3 architecture.

The best results were achieved for batch size: 32, Adam optimizer, learning rate—0.0001 and unfreezing the layers starting from the fifth convolution of the first Inception Block-type B. This model reached a training accuracy 0.81 (81%) and 0.78 (78%) when the training was stopped, after 50 epoch (Figure 10b), due to over fitting.

(**a**) (**b**)
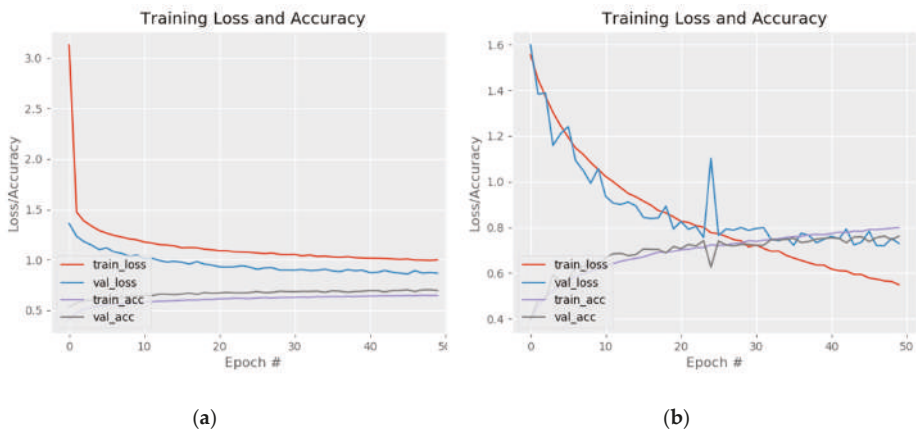
Figure 10. InceptionV3 training loss and accuracy for: (**a**) FC layers warm-up and (**b**) training of the last five convolutional layers.

The confusion matrix for FER predictions of the three architectures is presented in Figure 11. The network with the best overall results, from the confusion matrix perspective, was also ResNet (Figure 11b), followed by the VGG (Figure 11a) and InceptionV3 (Figure 11c). The first thing to notice was that the class with the highest score was the "happy" class regardless of the network model. There were two crucial factors that enabled such a good prediction: high variance and large dataset, with the first being the most important. Training samples play an important role for learning a complex future but the variance was determinant when it came to highlighting crucial features. This can be seen from the fact that the happy class is not the biggest in the training samples, being overtaken by the neutral class. Although it had the largest share of the dataset, the "neutral" class failed to generalize as well as others. The same thing when comparing angry, sad and surprise classes. The classes shared the same proportion from the dataset but the accuracy was different.

All of these observations mentioned above were influenced by emotion variance. The similarity of the emotion variation could also be observed in the confusion matrix. For example "fear" can be easily mistaken with "surprise" and "sad" with "neutral. An important number of images from "angry" and "sad" classes were classified as "neutral" images. This happens because these classes had low variance, in terms of mouth and eyebrows shape. The shape of mouth did not change significantly, while the displacement of the eyebrows was hard to be distinguished by the CNN model. It is interesting that the vice versa was not happening, probably because of the high share of training samples. Another important misclassification was the fear emotion classified as surprise. Due to the similarity of these emotions in terms of mouth shape and low changes in the eyebrows shape, fear was overly misclassified as surprise (19%). The misclassification of the classes could be improved by

increasing the number of images and by removing the neutral class in order to force the model to learn other distinct features. Bias played an important role, mainly when the expressions were difficult to be labeled, thereby affecting the models that were trained on them.
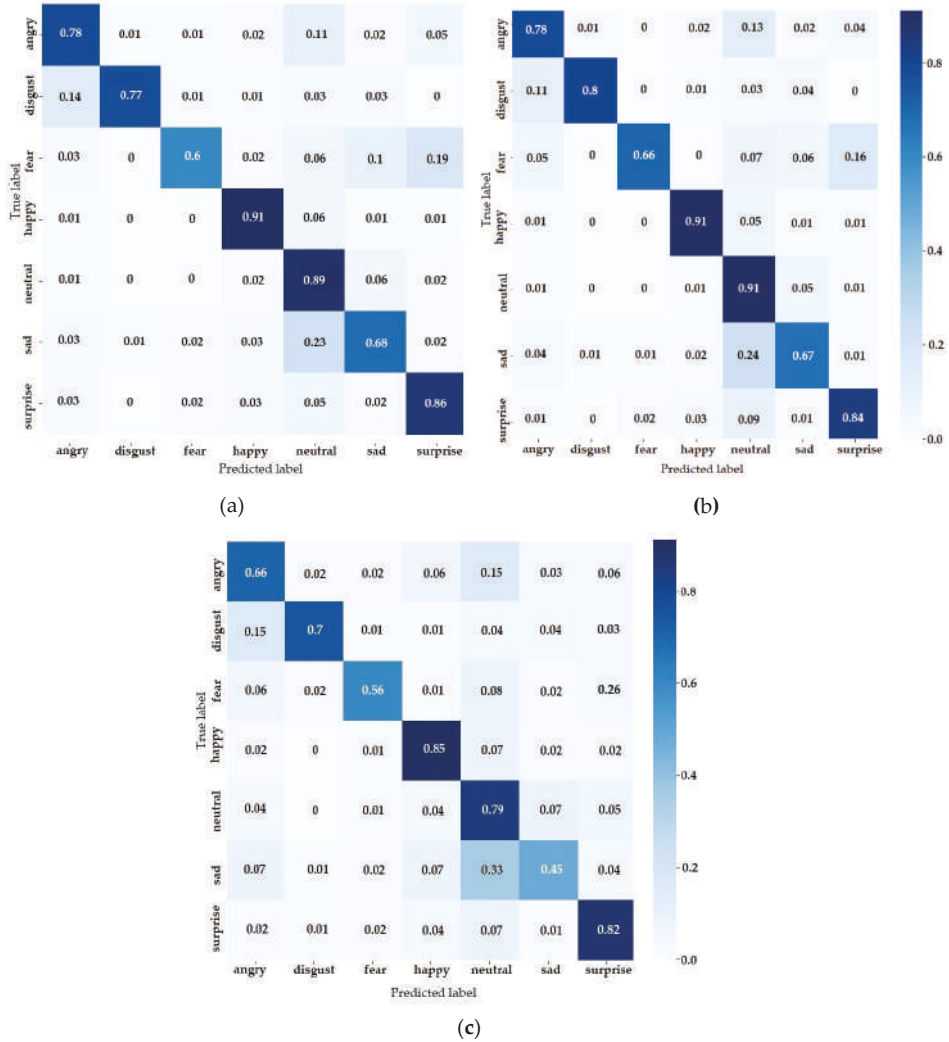
Figure 11. Confusion matrix for: (**a**) VGG; (**b**) ResNet and (**c**) InceptionV3.

In Table 1, the performances of state-of-the-art CNN models trained on FER2013 database are presented with respect to the CNN type, preprocessing method and optimizer. The NAO-SCNN models developed in this research achieved the highest accuracy, with the ResNet based architecture obtaining the best performances.

In Table 2, the performances of state-of-the-art CNN models used for emotion recognition on NAO robot are presented. The NAO-SCNN models achieved the highest accuracy, with the ResNet based architecture obtaining the best performances.

Table 1. Results for models trained on FER2013 database.

CNN Model	Accuracy [%]	CNN Type	Preprocessing	Optimizer
Pramerdorfer et al. [22]	75.2	Network Ensemble	-	SGD
Tang [23]	71.2	Loss layer	-	SGD
Kim et al. [24]	73.73	Network Ensemble	Intraface	SGD
Minae et al. [25]	70.02	Loss layer		Adam
Hua et al. [26]	71.91	Ensemble Network	Straightforward	Adam
Connie [27]	73.4	Multitask Network	-	Adam
NAO-SCNN (ResNet)	**90.14**	**Loss layer**	**CNN**	**RAdam**
NAO-SCNN (Inception)	81	Loss layer	CNN	Adam
NAO-SCNN (VGG)	87	Loss layer	CNN	Adam

Table 2. Results for models used on NAO robots.

CNN Model	Accuracy [%]
CNN Viola-Jones [1]	44.88
DBMM Classifier [2]	85
NAO-SCNN (ResNet)	90.14
NAO-SCNN (Inception)	81
NAO-SCNN (VGG)	87

In Figure 12 the manner the VGG model learns patterns for a given input image is presented. We followed the feature maps transformation of two outputs from convolutional blocks four and nine. After the convolution four our fine-tuned model focused on learning different edges like face, mouth, nose, eyes or teeth outline. This was the result of the convolutional filters learning how to identify lines, colors, circles or other basic information.
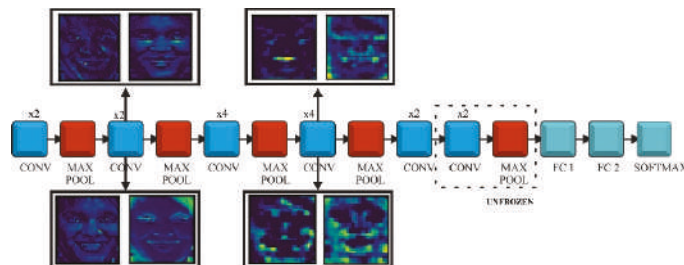
Figure 12. VGG feature maps at convolution 4 and convolution 9.

The feature maps after the second max pooling were concentrated to learn more complex representations from the picture. At this stage the convolutional filters were able to detect basic shapes. The filters from the deeper layers were able to distinguish between difficult shapes at the end of training and the representation of the feature maps became more abstract.

The feature maps for two layers of ResNet model are presented in Figure 13. The first set of images were captured after the second convolutional block. At this point, the feature maps had high resolution and focused on sparse areas of the face. The second set of images was taken from the second convolution of third ID block. It could be observed that at this point the filters were concentrated on particular areas, such as eyes, mouth, teeth, eyebrows and the representation of the feature maps was more abstract.

The feature maps for two layers of Inception V3 model are presented in Figure 14. The first set of images was captured after the first Inception type-A block. As VGG and ResNet, the convolutional filters were able to detect the shape of areas responsible for emotion detection such as eyes, mouth, eyebrows and also neighbor regions with relevant variance.
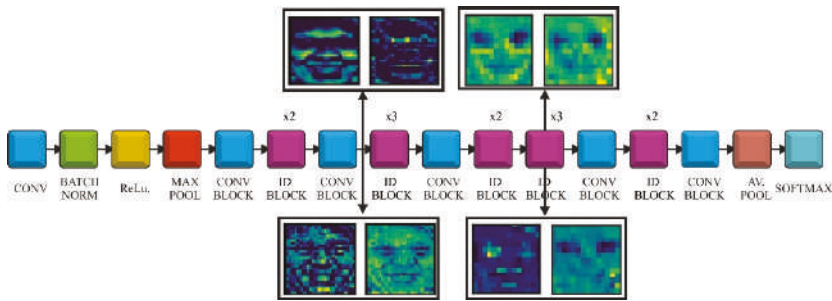
Figure 13. ResNet feature maps after the second convolution block and 8th ID block.
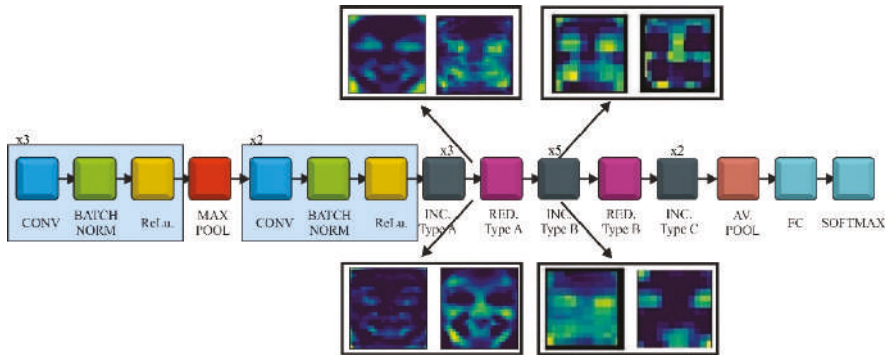
Figure 14. Inception V3 feature maps after the Inception block type A and Inception block type B.

The second set of images was taken after the first Inception block type B. At this point, the resolution was low as the filters concentrated on small regions of the input image and the feature maps had a more abstract representation.

4. Conclusions

The SSD and the Faster R-CNN models for face detection shared almost the same total loss and accuracy. The accuracy of the Faster R-CNN model was 97.8% based on the VOC metrics, while for the SSD Inception model it was 97.42%. Thus, taking into account that Faster R-CNN is a large network, in terms of total parameters, the SSD architecture was chosen for FR detection in order to keep a low inference speed.

The network with the best overall performances, in terms of accuracy and loss for FER detection was ResNet and the results were confirmed by the confusion matrix. This model obtained a training accuracy of 90.14%, while VGG had 87% accuracy and Inception V3 reached 81%. By adding 5–6% of laboratory controlled images from other databases the accuracy of FER2013 increased with more than 10%, of the highest score so far, using the ResNet model together with other optimizations. In order to develop a model that better generalizes FER it is important to mix controlled and uncontrolled images.

We also noticed that using the preprocessing module for FR before running the inference for emotion recognition enhances the FER confidence score for the same images. The results show improvements well over 10% when using two serialized CNN, instead of just using the FER CNN.

A recent optimization model, called rectified Adam, was applied for training the three FER CNN models, which led to a better generalization. RAdam introduces a term to rectify the variance of the adaptive learning rate, by applying warm up with a low initial learning rate. The tests showed a more

robust loss during training and accuracy improvement of 3%-4% on each FER CNN compared to other optimizers.

The FR and FER architectures were implemented on the NAO humanoid robot in order to switch from the current computer vision library running on the robot to a newer and more accurate serialized model based on deep convolutional neural networks.

In order to increase the inference time we used a neural network preprocessor developed by Intel, called Neural Compute Stick 2(NCS 2), together with asynchronous threading, which boosts the NAO performances from 0.25 to 4-6 FPS on our FR+FER pipeline. The inference speed when using only one CNN, reached 8-9 FPS with NCS2.

Future developments will involve the fusion of other inputs, such as audio models, infrared images, depth information from 3D face models, physiological data, etc. will provide additional information and further enhance the accuracy of the process.

The human robot interaction model obtained during this research will be further developed in order to be applied for medical purposes, mainly for improving communication and behavior of children with autism spectrum disorder. The research will be carried out using medical feedback from psychologists, physicians, educators and therapists to reduce frustration and anxiety through communication with a humanoid robot.

References

1. Lopez-Rincon, A. Emotion recognition using facial expressions in children using the NAO Robot. In Proceedings of the International Conference on Electronics, Communications and Computers (CONIELECOMP), Cholula, Mexico, 27 February–1 March 2019; IEEE: Piscataway, NJ, USA; pp. 146–153.

2. Faria, D.R.; Vieira, M.; Faria, F.C. Towards the development of affective facial expression recognition for human-robot interaction. In Proceedings of the 10th International Conference on PErvasive Technologies Related to Assistive Environments, Island of Rhodes, Greece, 21–23 June 2017; pp. 300–304.

3. Zhang, Z.; Luo, P.; Loy, C.C.; Tang, X. From facial expression recognition to interpersonal relation prediction. *Int. J. Comput. Vis.* **2018**, *126*, 550–569. [CrossRef]

4. Zhao, X.; Liang, X.; Liu, L.; Li, T.; Han, Y.; Vasconcelos, N.; Yan, S. Peak-piloted deep network for facial expression recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 425–442.

5. Ding, H.; Zhou, S.K.; Chellappa, R. Facenet2expnet: Regularizing a deep face recognition net for expression recognition. In Proceedings of the 12th IEEE International Conference on Automatic Face & Gesture Recognition, Washington, DC, USA, 30 May–3 June 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 118–126.
6. Ng, H.W.; Nguyen, V.D.; Vonikakis, V.; Winkler, S. Deep learning for emotion recognition on small datasets using transfer learning. In Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, Seattle, WA, USA, 9–13 November 2015; pp. 443–449.
7. Lu, G.; He, J.; Yan, J.; Li, H. Convolutional neural network for facial expression recognition. *J. Nanjing Univ. Posts Telecommun.* **2016**, *36*, 16–22.
8. Zeng, J.; Shan, S.; Chen, X. Facial expression recognition with inconsistently annotated datasets. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 222–237.
9. Levi, G.; Hassner, T. Emotion recognition in the wild via convolutional neural networks and mapped binary patterns. In Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, Seattle, WA, USA, 9–13 November 2015; pp. 503–510.
10. Mayya, V.; Pai, R.M.; Pai, M.M. Automatic facial expression recognition using DCNN. *Procedia Comput. Sci.* **2016**, *93*, 453–461. [CrossRef]
11. Masi, I.; Wu, Y.; Hassner, T.; Natarajan, P. Deep face recognition: A survey. In Proceedings of the 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Paraná, Brazil, 29 October–1 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 471–478.
12. Lucey, P.; Cohn, J.F.; Kanade, T.; Saragih, J.; Ambadar, Z.; Matthews, I. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, San Francisco, CA, USA, 13–18 June 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 94–101.
13. Lyons, M.; Akamatsu, S.; Kamachi, M.; Gyoba, J. Coding facial expressions with gabor wavelets. In Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition, Nara, Japan, 14–16 April 1998; IEEE: Piscataway, NJ, USA, 1998; pp. 200–205.
14. Goodfellow, I.J.; Erhan, D.; Carrier, P.L.; Courville, A.; Mirza, M.; Hamner, B.; Cukierski, W.; Tang, Y.; Thaler, D.; Lee, D.H.; et al. Challenges in representation learning: A report on three machine learning contests. In Proceedings of the International Conference on Neural Information Processing, Daegu, Korea, 3–7 November 2013; Springer: Berlin/Heidelberg, Germany, 2013; pp. 117–124.
15. Mollahosseini, A.; Hasani, B.; Mahoor, M.H. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Trans. Affect. Comput.* **2017**, *10*, 18–31. [CrossRef]
16. Pantic, M.; Valstar, M.; Rademaker, R.; Maat, L. Web-based database for facial expression analysis. In Proceedings of the IEEE international Conference on Multimedia and Expo, London, UK, 6–10 July 2005; IEEE: Piscataway, NJ, USA, 2005; p. 5.
17. Valstar, M.; Pantic, M. Induced disgust, happiness and surprise: An addition to the mmi facial expression database. In Proceedings of the 3rd Intern. Workshop on EMOTION (satellite of LREC): Corpora for Research on Emotion and Affect, Valetta, Malta, 30 May–3 June 2010; p. 65.
18. Dhall, A.; Goecke, R.; Lucey, S.; Gedeon, T. Collecting large, richly annotated facial-expression databases from movies. *IEEE Multimed.* **2012**, *1*, 34–41. [CrossRef]
19. Lundqvist, D.; Flykt, A.; Öhman, A. The Karolinska directed emotional faces (KDEF). *CD ROM Dep. Clin. Neurosci. Psychol. Sect. Karolinska Inst.* **1998**, *91*, 2.
20. Yang, H.; Ciftci, U.; Yin, L. Facial expression recognition by de-expression residue learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 2168–2177.
21. Hamester, D.; Barros, P.; Wermter, S. Face expression recognition with a 2-channel convolutional neural network. In Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN), Killarney, Ireland, 12–17 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–8.
22. Pramerdorfer, C.; Kampel, M. Facial expression recognition using convolutional neural networks: State of the art. *arXiv* **2016**, arXiv:1612.02903.
23. Tang, Y. Deep learning using linear support vector machines. *arXiv* **2013**, arXiv:1306.0239.

24. Kim, B.-K.; Dong, S.-Y.; Roh, J.; Kim, G.; Lee, S.-Y. Fusing aligned and non-aligned face information for automatic affect recognition in the wild: A deep learning approach. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 48–57.

25. Minaee, S.; Abdolrashidi, A. Deep-emotion: Facial expression recognition using attentional convolutional network. *arXiv* **2019**, arXiv:1902.01019.

26. Hua, W.; Dai, F.; Huang, L.; Xiong, J.; Gui, G. HERO: Human emotions recognition for realizing intelligent Internet of Things. *IEEE Access* **2019**, *7*, 24321–24332. [CrossRef]

27. Connie, T.; Al-Shabi, M.; Cheah, W.P.; Goh, M. Facial expression recognition using a hybrid CNN–SIFT aggregator. In Proceedings of the International Workshop on Multi-Disciplinary Trends in Artificial Intelligence, Gadong, Brunei, 20–22 November 2017; Springer: Cham, Switzerland, 2017; pp. 139–149.

28. Emotion-Compilation. Available online: https://www.kaggle.com/qnkhuat/emotion-compilation (accessed on 30 August 2019).

29. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.

30. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.

31. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.

32. Taigman, Y.; Yang, M.; Ranzato, M.A.; Wolf, L. Deepface: Closing the gap to human-level performance in face verification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1701–1708.

33. Schroff, F.; Kalenichenko, D.; Philbin, J. Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 815–823.

34. Parkhi, O.M.; Vedaldi, A.; Zisserman, A. Deep face recognition. *BMVC* **2015**, *1*, 6.

35. Liu, W.; Wen, Y.; Yu, Z.; Li, M.; Raj, B.; Song, L. Sphereface: Deep hypersphere embedding for face recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 212–220.

36. Gal, I.A.; Bucur, D.; Vladareanu, L. DSmT decision-making algorithms for finding grasping configurations of robot dexterous hands. *Symmetry* **2018**, *10*, 198. [CrossRef]

37. Yan, H.; Wang, H.; Vladareanu, L.; Lin, M.; Vladareanu, V.; Li, Y. Detection of Participation and Training Task Difficulty Applied to the Multi-Sensor Systems of Rehabilitation Robots. *Sensors* **2019**, *19*, 4681. [CrossRef]

38. Feng, Y.; Wang, H.; Vladareanu, L.; Chen, Z.; Jin, D. New Motion Intention Acquisition Method of Lower Limb Rehabilitation Robot Based on Static Torque Sensors. *Sensors* **2019**, *19*, 3439. [CrossRef] [PubMed]

39. Iliescu, M.; Vlădăreanu, L.; Radu, C.; Dugăeşescu, I.; Pandelea, M.; Marin, D. Research on upper limb biomechanical system. *Period. Eng. Nat. Sci.* **2019**, *7*, 267–274. [CrossRef]

40. Wang, H.; Zhang, D.; Lu, H.; Feng, Y.; Xu, P.; Mihai, R.V.; Vladareanu, L. Active training research of a lower limb rehabilitation robot based on constrained trajectory. In Proceedings of the IEEE International Conference on Advanced Mechatronic Systems (ICAMechS), Beijing, China, 22–24 August 2015; pp. 24–29.

41. Ali, M.; Smarandache, F.; Shabir, M.; Vladareanu, L. Generalization of Neutrosophic Rings and Neutrosophic Fields. *Neutrosophic Sets Syst.* **2014**, *5*, 9–14.

42. Smarandache, F.; Vlădăreanu, L. Applications of neutrosophic logic to robotics. In Proceedings of the IEEE International Conference on Granular Computing, Kaohsiung, Taiwan, 8–10 November 2011; pp. 607–612.

43. Vladareanu, V.; Schiopu, P.; Vladareanu, L. Theory and Application of Extension Hybrid Force-Position Control in Robotics. *Univ. Politeh. Buchar. Sci. Bull.-Ser. A-Appl. Math. Phys.* **2014**, *76*, 43–54.

44. Vladareanu, V.; Munteanu, R.I.; Mumtaz, A.; Smarandache, F.; Vladareanu, L. The optimization of intelligent control interfaces using Versatile Intelligent Portable Robot Platform. *Procedia Comput. Sci.* **2015**, *65*, 225–232. [CrossRef]

45. Vladareanu, L.; Tont, G.; Ion, I.; Velea, L.M.; Gal, A.; Melinte, O. Fuzzy dynamic modeling for walking modular robot control. In Proceedings of the 9th International Conference on Application of Electrical Engineering, Prague, Czech Republic, 16–19 May 2010; pp. 163–170.

46. Vladareanu, V.; Dumitrache, I.; Vladareanu, L.; Sacala, I.S.; Tonț, G.; Moisescu, M.A. Versatile Intelligent Portable Robot Platform applied to dynamic control of the walking robots. *Stud. Inform. Control* **2015**, *24*, 409–418. [CrossRef]
47. Vladareanu, L.; Tont, G.; Vladareanu, V.; Smarandache, F.; Capitanu, L. The navigation mobile robot systems using Bayesian approach through the virtual projection method. In Proceedings of the IEEE the 2012 International Conference on Advanced Mechatronic Systems, Tokyo, Japan, 18–21 September 2012; pp. 498–503.
48. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 21–37.
49. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*; Mit Press: Cambridge, MA, USA, 2015; pp. 91–99.
50. Kuznetsova, A.; Rom, H.; Alldrin, N.; Uijlings, J.; Krasin, I.; Pont-Tuset, J.; Kamali, S.; Popov, S.; Malloci, M.; Duerig, T.; et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv* **2018**, arXiv:1811.00982. [CrossRef]
51. Open Images Dataset V6. Available online: https://storage.googleapis.com/openimages/web/download_v4.html (accessed on 10 September 2019).
52. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2818–2826.
53. Liu, L.; Jiang, H.; He, P.; Chen, W.; Liu, X.; Gao, J.; Han, J. On the variance of the adaptive learning rate and beyond. *arXiv* **2019**, arXiv:1908.03265. Available online: http://doc.aldebaran.com/1-14/index.html (accessed on 20 January 2020).
54. NAO Software 1.14.5 Documentation. Available online: http://doc.aldebaran.com/1-14/index.html (accessed on 20 January 2020).

The Growth of Ga$_2$O$_3$ Nanowires on Silicon for Ultraviolet Photodetector

Badriyah Alhalaili [1,2], Ruxandra Vidu [2,3,*] and M. Saif Islam [2]

[1] Nanotechnology and Advanced Materials Program, Kuwait Institute for Scientific Research, Safat 13109, Kuwait; eng.kisr@hotmail.com
[2] Electrical and Computer Engineering, University of California at Davis, Davis, CA 95616, USA; sislam@ucdavis.edu
[3] The Faculty of Materials Science and Engineering, University of Politehnica of Bucharest, 060042 Bucharest, Romania
[*] Correspondence: rvidu@ucdavis.edu

Abstract: We investigated the effect of silver catalysts to enhance the growth of Ga$_2$O$_3$ nanowires. The growth of Ga$_2$O$_3$ nanowires on a P$^+$-Si (100) substrate was demonstrated by using a thermal oxidation technique at high temperatures (~1000 °C) in the presence of a thin silver film that serves as a catalyst layer. We present the results of morphological, compositional, and electrical characterization of the Ga$_2$O$_3$ nanowires, including the measurements on photoconductance and transient time. Our results show that highly oriented, dense and long Ga$_2$O$_3$ nanowires can be grown directly on the surface of silicon. The Ga$_2$O$_3$ nanowires, with their inherent n-type characteristics formed a pn heterojunction when grown on silicon. The heterojunction showed rectifying characteristics and excellent UV photoresponse.

Keywords: β-Ga$_2$O$_3$; nanowires; oxidation; silver catalyst; electrical conductivity; photodetector

1. Introduction

The development of wide band gap semiconductor technology has received considerable attention as basic materials that facilitate various ultraviolet (UV) applications in nanoscale electronics and optoelectronics [1] such as engine control, solar UV monitoring, astronomy, communications, or the detection of missiles. Recently, UV photodetectors (PDs) have received special attention, because the civil, military, environmental, and industrial markets need to improve UV instrumentation that operates at extremely harsh environments. Therefore, numerous studies have been proposed to fabricate UV photodetectors with specialized features to operate and survive in the UV region of the spectrum.

Semiconductor nanowires exhibit different and often improved material properties [2,3] compared to bulk or thin-film semiconductors. In recent years, gallium oxide (Ga$_2$O$_3$) became one of the most important materials that can operate in harsh conditions. With a band-gap of 4.8 eV, a high melting point of 1900 °C, excellent electrical conductivity, high figure of merit for high-frequency applications, and photoluminescence [4,5], it is an ideal candidate for visible-blind UV-light sensors, particularly for power electronics, solar-blind UV detectors, and devices for harsh environments [6,7]. New processes have been investigated to synthesize Ga$_2$O$_3$ nanowires (NWs) through a bottom-up approach, which include thermal oxidation [8,9], vapor-liquid-solid mechanism [10], pulsed laser deposition [11], sputtering [12], thermal evaporation [13–15], molecular beam epitaxy [16], laser ablation [17], arc-discharge [18], carbothermal reduction [19], microwave plasma [20], metalorganic chemical vapor deposition [21], and the hydrothermal method [22,23].

Due to the surface area, small nanowire diameter and high nanowire photoconductivity, high responsivity can be achieved in UV photodetectors. Additionally, one of the beneficial parameters of nanowires is their ability to enhance light absorption and confinement to increase photosensitivity [24]. The superiority of the growth of Ga_2O_3 nanowires, compared to thin film is the ability to increase the sensitivity in detection due to the higher surface-to-volume ratio, leading to more available surface states at the interface, and thus, exceptional interaction with analytes or physical states [25]. Although various reports have been obtained to grow Ga_2O_3 thin films on Si [26,27], there have been few reports on the growth of nanowires onto a silicon (Si) substrate [28], which will pave the way for future sensing devices and circuit technology integrations. The sensors obtained using this innovative approach will lead to new trends in design, control, and applications of real-time intelligent sensor system control by advanced intelligent control methods and techniques. The effect of Ag thin film as a catalyst to enhance the growth of Ga_2O_3 nanowire and crystalline thin film on quartz has been reported [29], but it has not been explored on the silicon surface. We also wanted to observe the contribution of silicon atoms in enhancing the conductivity of Ga_2O_3 nanowires via diffusion-enabled incorporation into the nanowires during the growth process.

In our previous work, the surface of the quartz was coated with a 5 nm Ag catalyst using a shadow mask intentionally to examine the effect of Ag nanoparticles (NPs)distribution. In this work, the entire silicon surface was coated with a 5 nm catalyst to enhance the growth of highly oriented nanowires that has not been shown before. Compared to other reported works [28], the length of the nanowires was much higher and highly oriented when Ag catalyst was used rather than the Au catalyst, where the nanowires were randomly oriented.

In this work, we proposed the growth of β-Ga_2O_3 nanowires on P^+-silicon substrate by thermal oxidation at 950 °C using an Ag catalyst. We studied the sensitivity of β-Ga_2O_3 nanowires for UV detection.

2. Materials and Methods

The UV photodetector was fabricated on (100) P^+-Si substrate doped with phosphorus. The substrate was 500 μm thick and had a resistivity between 0.001 and 0.005 Ω-cm. Before each experiment, the silicon substrate was cleaned for 5 min in acetone and then, in methanol for 5 min in an ultrasonic bath. Following the cleaning procedure, the wafer was rinsed with deionized water for 5 min. To obtain Ga_2O_3, 0.2 g of gallium [(Ga) (purity 99.999%)] was dripped onto cleaned quartz crucible. Silver was used as a catalyst to enhance the growth of gallium oxide NWs. An ultrathin layer of 5 nm Ag was sputtered on silicon. The silicon wafer was positioned with the Ag-coated surface to face the crucible quartz containing Ga. The distance between the substrate and the gallium pool was 10 mm. Then, the substrate was loaded into a quartz crucible which was placed into an OTF-1200X-50-SL horizontal alumina tube furnace made by MTI Corporation (Richmond, CA, USA). The oxidation was performed at 950 °C for 1 h in a 20 sccm nitrogen atmosphere.

Figure 1 illustrates the setup of the UV photodetector fabrication process. As the system cools down to room temperature, the samples were removed from the furnace, cleaved, and characterized by scanning electron microcopy (SEM), X-ray photoelectron spectroscopy (XPS), and high-resolution transmission electron microscopy (HRTEM), equipped with energy dispersive X-ray spectroscopy. The electrical contacts were patterned on top of the nanowires using shadow mask and then 1 nm Cr and 150 nm Au were sputtered using a Lesker sputtering system. Electrical characterization of the system was also carried out to assess the performance of the UV photodetector. For electrical measurements, a custom probe station attached to a Keithly 2400 series SMU instrument was used. For photocurrent measurements, UV illumination was from a Dymax Bluewave 75 UV lamp (280–320 nm) (Dymax Corporation, Torrington, CT, USA). A light intensity of 1.5 W/cm^2 was used.
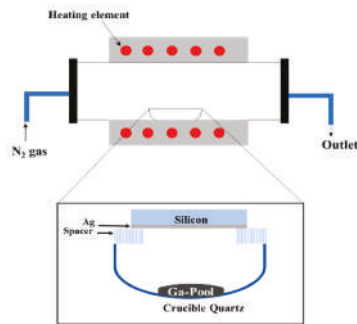
Figure 1. Schematic of the growth process of Ga_2O_3 NWs on Si substrate coated with 5 nm thin film of Ag and positioned downward to face liquid Ga pool in a quartz crucible. The distance between Ga pool and silicon substrate is about a ~10 mm gap.

3. Results and Discussion

3.1. Surface Morphology

Ga_2O_3 nanowires were grown on P^+-Si at 950 °C. As shown in Figure 2, the silver catalyst plays a major role in the growth mechanism. Using 5 nm Ag as a catalyst, a homogeneous coating and denser nanowires were achieved due to the low contact angle. A low contact angle reflects the extension of wetting, i.e., the liquid advances on the surface and homogeneously wets the surface. To control the wetting contact angle, deposition or incorporation of elements and molecules onto the surface is a standard procedure. We believe that Ag has the role to improve wettability, which will enhance the homogeneous appearance of Ga_2O_3 nuclei that could lead to dense nanowires. The contact angle of Ga on a silver film is 30° [30], and on a silicon substrate, it is 73.9° [31], leading to better wetting of Ga on Ag surface and uniform growth of Ga_2O_3 nanowires (Figure 3).
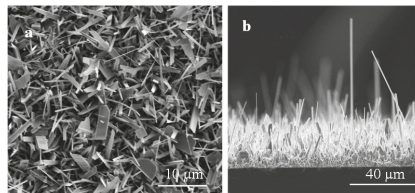
Figure 2. SEM images of Ga_2O_3 nanowires growth on Si at 950 °C (**a**) Top view and (**b**) Side view of Ga_2O_3 nanowires growth on Si. Denser and longer growth of nanowires were attained.
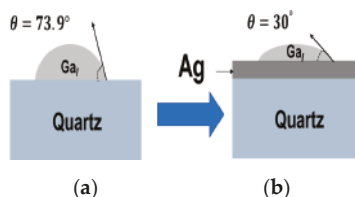
Figure 3. Contact angle of liquid Ga droplet on different surfaces. (**a**) Silicon. (**b**) 5 nm silver thin film. Areas coated with 5 nm Ag show uniform and high-dense growth of Ga_2O_3 nanowires.

Various research strategies were conducted in the past, mainly to enhance the nanowires' growth on the target substrate [10,32–34]. In contrast, these techniques to grow Ga_2O_3 nanowires have shown lateral growth, overlapping nanowires, less dense and weak adhesion to the substrate. None of the

previous techniques were able to produce a conformal growth process of Ga_2O_3 nanowires on the substrate surface.

The results obtained with the use of 5 nm Ag catalyst showed a remarkable improvement in the lengths and the density of the nanowires, most of them perpendicular to the surface. Even though the lengths of these nanowires were increased, their diameters were decreased. The diameters of the nanowires were in the range of 70–90 nm at the tip and 120–160 nm at the bottom. The average length of these nanowires was in the range of about 30–70 μm.

3.2. X-ray Photoelectron Spectroscopy (XPS)

To analyze the elemental composition of Ga_2O_3 nanowires, XPS was performed on a PHI 5800 model.

Figure 4 shows XPS spectra of Ga_2O_3 nanowires on Si. The XPS spectrum shows the chemical composition of the particles at the surface of β-Ga_2O_3 nanowires on Si in the presence of Ag. The binding energies of Ga2p3, O1s, and Ag3d (with two peaks) and Si2p are 1119.1 eV, 532 eV, 369.07 eV and 379.66 eV and 105.18 eV, respectively. The peaks of Ga and O for Ga_2O_3 and Ag are in agreement with the handbook of XPS spectra [35,36]. XPS analysis of the β-Ga_2O_3 nanowires on Si and the presence of Ag catalyst showed a positive shift due to the effect of the electronegativity difference [37]. In addition, this shift could be attained in Ag3d, as the size of Ag nanoparticles highly decreased [38].

(a)

(b)

(c)

(d)
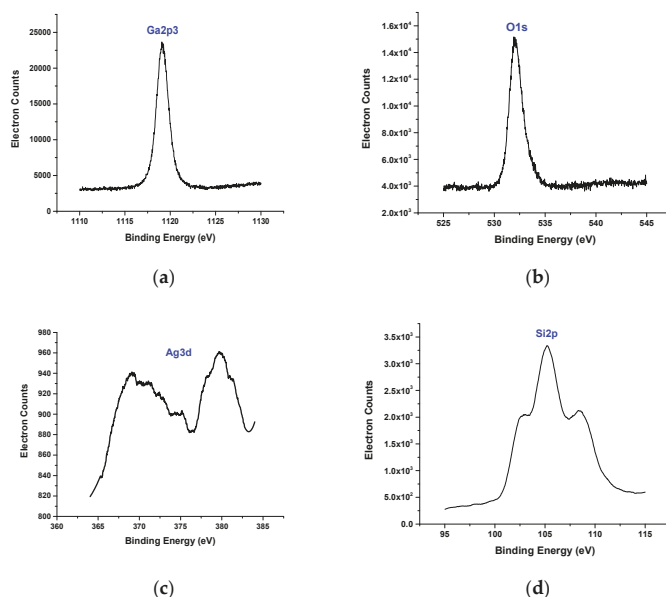
Figure 4. XPS of the β-Ga_2O_3 nanowires was obtained at 950 °C and in the presence of an Ag catalyst. Different peaks were detected by XPS. (**a**) Ga. (**b**) O. (**c**) Ag. (**d**) Si. The peaks of Ag and Ga have positive slight shifts due to the difference in electronegativity and work function.

3.3. High-Resolution Transmission Electron Microscopy (HRTEM)/Energy-Dispersive Spectroscopy (EDS)

An energy-dispersive spectroscopy (EDS) profile analysis was performed on β-Ga_2O_3 nanowires grown on Si (Figure 5). Interestingly, none of the Ag nanoparticles were clearly observed on the surface of the nanowires. However, a very small amount in atomic percentage of Ag was detected by HRTEM equipped with EDS. Because no Ag was observed on the nanowire surface, a very small amount of Ag might be embedded into the Ga_2O_3 nanowires. These remaining Ag nanoparticles could be trapped inside the nanowires after all Ag was consumed and evaporated.
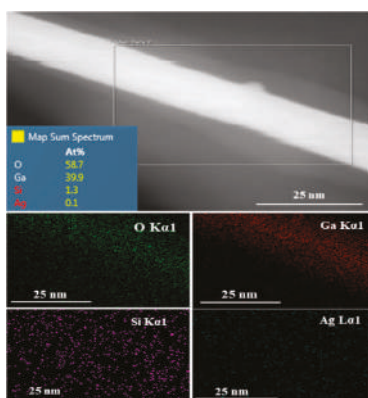
Figure 5. HRTEM image and the corresponding EDS mapping of Ga, O, Si and Ag of Ga_2O_3 NWs on P-doped (100) silicon substrate coated with 5 nm Ag.

Because silicon atoms can interact with silver at a high temperature (i.e., the oxidation temperature of 950 °C) the background impurity of silicon was measured in Ga_2O_3 nanowires. At high temperature and a few atomic percentages of Si, the Si-Ag phase diagram [39] shows that Si can interact with Ag. Silicon is one of the major impurities that strongly correlates to n-type conductivity [40]. If silicon were to be incorporated into Ga_2O_3 nanowires during oxidation, it could increase the n-type conductivity of nanowires. In addition, since Si has a strong effect on the dissolution of the large Ag NPs [41], there will be more Ag atoms available for diffusion on the Si surface, which could result in a denser growth of nanowires.

3.4. Growth Mechanism of β-Ga_2O_3 Nanowires

The contribution of a silver catalyst to the growth enhancement of β-Ga_2O_3 nanowires on Si showed a growth reaction rate strongly influenced by the oxidation temperature and follows the Arrhenius law [42]. Oxygen diffusivity and solubility are important parameters that distinguish Ag as an effective catalyst for Ga_2O_3 nanowire growth.

Diffusion is a result of the kinetic properties of atoms. In this case, diffusion appears to be due to the high capability of Ag to absorb oxygen, and it is greatly influenced by the variation of temperature. Different studies were focused on the oxygen diffusivity (D) in gallium [43] and silver [44]. Table 1 summarizes the major diffusivity coefficient of oxygen into solid Ag, liquid silver, and liquid gallium. The diffusion coefficient of oxygen in silver has a high tendency to absorb oxygen, and hence, boost nanowire growth.

Table 1. Summary of reported diffusivity coefficient and activation energy of oxygen in silver and gallium.

Metal	Diffusion Coefficient (D) (cm^2/s)	Activation Energy (E_A) (eV/K)	T (°C)	Year	Ref.
Ag_s	1.79×10^{-3}	0.58	127–977	2016	[44]
	4.90×10^{-3}	0.56		1972	[45]
	3.66×10^{-3}	0.48	740–915	1990	[46]
	4.98×10^{-3}	0.63		1991	[47]
Ag_l	20.1×10^{-4}	0.91	980–1130	1971	[9]
	4.9×10^{-3}	0.12	763–937	1972	[45]
Ga_l	4.1×10^{-3}	8.9×10^{-5}	750–950	1981	[43]
	2.27×10^{-3}	8.33×10^{-5}	750–1000	1972	[48]

The solubility of oxygen is another factor that has essential perspective to speed up the growth of Ga_2O_3 nanowires. The activation energy of oxygen solubility in silver was 0.01192 eV/K at a temperature range of 763–937 °C [45]; however, in gallium, it was 2.38×10^{-4} eV/K at a temperature range of 750–1000 °C [48]. Oxygen solubility in silver exhibited a higher solubility than Ga. Further studies are needed to measure the Ag-Ga-O thermodynamics at higher temperatures.

Taking these results into consideration, the growth mechanism of nanowires can be explained as follows. First, at higher temperatures, the liquid gallium can form gallium oxide in the presence of oxygen. Then, the oxide is further reduced by liquid metallic gallium and forms a gas phase of gallium suboxide (Ga_2O), as shown in Equation (1) [29,49] as follows:

$$Ga_2O_{3(s)} + 4Ga_{(l)} \rightarrow 3Ga_2O_{(g)} \uparrow \tag{1}$$

The Ga_2O gas phase is transported to the cooler regions and decomposes to liquid gallium and Ga_2O_3 [50,51], leading to a vapor-liquid-solid (VLS) growth mechanism. At high temperatures (T > 950 °C) denser Ga_2O_3 grows as nanowires. It has been shown that the presence of Ga atoms can easily etch the surface of silica substrate around 950 °C, as shown in Equation (2) [52].

$$SiO_2 + 4Ga \rightarrow 2Ga_2O \uparrow + Si \tag{2}$$

In addition, the phase diagram of Si-Ag shows that a liquid phase exists in this system at high temperatures (T > 800 °C) at a small percentage of Si [39]. The contribution of small concentrations of silicon can detach and stimulate the melting point of Ag surface atoms [41]. Despite the fact that carrier doping in β-Ga_2O_3 is a difficult task, some impurity doping using Sn or Si has been shown to achieve electrical conduction [40,53–55]. In this growth mechanism, silicon has been detected by EDS (Figure 5), unintentionally improving the background conductivity of the nanowires. The presence of oxygen atoms segregated on the surface of Ag catalyst will react with Ga. This increases the flux of O atoms and Ga segregation at Ag-Si interface, leading to the formation of an equilibrium mixture of Ag-Si-Ga-O that becomes a solid phase source for Ga_2O_3 nucleation (Figure 6).
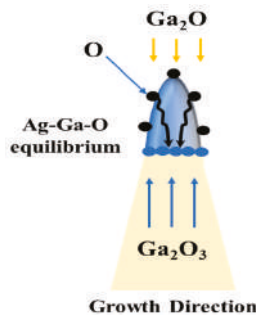
Figure 6. The growth mechanism of Ga_2O_3 NW on a silicon substrate coated with 5 nm Ag as a catalyst. The equilibrium liquid mixture of Ag-Ga-O at higher temperature (>900 °C) leads to the enhancement of the growth mechanism and increases the density of Ga_2O_3 NWs.

3.5. Electrical Characterization

3.5.1. I-V Characterization

The β-Ga_2O_3/P^+-Si PN heterojunction (Figure 7) was fabricated to determine the electronic properties of β-Ga_2O_3 nanowires. The choice of testing the P^+Si substrate is due to availability of low-cost materials for electronics and to observe how silicon from the substrate can influence the conductivity of Ga_2O_3. Impact of silicon doping in Ga_2O_3 during the growth processes were reported

in references [40,53–55] for the cases of thin films and bulk materials and we wanted to investigate if migration of silicon atoms from the substrate can have a similar effect. In addition, the formation of n-Ga$_2$O$_3$ nanowires on the surface of highly doped silicon substrates has not been reported so far. The results lead to the development of a simple growth technique for large-scale production of a highly sensitive and stable structure. In previous works, the growth of Ga$_2$O$_3$ was obtained due to the presence of an Au catalyst (instead of Ag) on the surface of the Si$_2$/Si template [28].
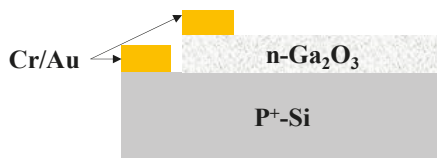
Figure 7. Schematic diagram of Au/β-Ga$_2$O$_3$/Silicon photoconductor. The distance between the gold probes is 0.8 mm.

The current-voltage (I-V) characteristics were measured in dark conditions and under UV illumination at different voltages 10 and 50 V. Photocarriers, which were excited by UV illumination, were from a Dymax Bluewave 75 UV lamp (280–320 nm) (Dymax Corporation, Torrington, CT, USA) (Figure 8). The photoconductivity mechanism of the β-Ga$_2$O$_3$ NWs is credited to a surface oxygen adsorption and desorption process [56], which is highly influenced by the presence of silver as a catalyst, leading to improve oxygen detection and hence the electrical properties of the β-Ga$_2$O$_3$ nanowires.

(**a**)

(**b**)
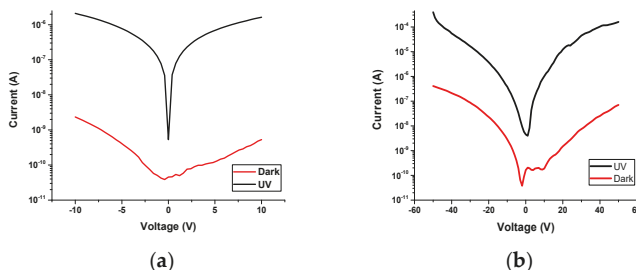
Figure 8. Semi-logarithmic plots of current density of dark and photocurrent characteristics of Ga$_2$O$_3$ NWs grown on silicon substrate at 950 °C with an Ag catalyst at 10 V, (**a**) 10 V, (**b**) 50 V.

The ratio of photo-to-dark current at 10 V was 3066.11 which is higher than other reported studies [57,58]. The reduction in performance could be attributed to the presence of Ag NPs, which were detected by XPS, although they are difficult to see in the scanning and transmission microscopy (SEM) images. The hot carriers of Ag NPs could increase the self-heating effects [59]. This issue is one of the major challenges that is still under investigation to improve the thermal conductivity of Ga$_2$O$_3$. It is well known that Ga$_2$O$_3$ generates self-heating effects that cause degradation of the carriers mobility [60], leading to reduced performance of Ga$_2$O$_3$ at high voltage.

Even the addition of Ag catalyst could cause a drawback, as it can enhance the sensitivity of the photodetector. The effect of the catalytic Ag nanoparticles can be explained as follows. First, Ag nanoparticles have a significant contribution in improving the conductivity of Ga$_2$O$_3$ nanowires, leading to better sensing performance. Secondly, Ag nanoparticles have the ability to greatly enhance the adsorption and desorption of O$_2$ on their surface due to the highly conductive behavior of Ag metal [61]. Consequently, the number of electrons drawn to O$_2$ increases greatly. Third, Ag nanoparticles play the role of electron mediators that allow electrons to migrate from the surface of Ga$_2$O$_3$ nanowires to the O$_2$ through the defect states of Ga$_2$O$_3$. As a result, the bulk defects of Ga$_2$O$_3$ may act as a secondary factor in the sensing mechanism in addition to the surface defects [4].

Consequently, Ag NPs significantly reduce the density of electrons of Ga_2O_3 and improve electrical conductivity, leading to better selectivity and sensitivity.

3.5.2. Transient Time

The transient response of the photodetector was measured by turning on and off a UV light source with wavelength range from 280 to 450 nm (Figure 9). Under UV illumination, the oxygen adsorption and desorption processes are attained to improve the photoconductivity response by increasing the carrier mobility. In contrast, when UV illumination is switched off, the excess electrons and holes recombine rapidly. Ga_2O_3 on silicon with Ag catalyst showed a rapid transient response due to the enhanced carrier transport. The rise was 0.8 s and fall time was 1.5 s. Due to the enhanced carrier transport process, fast rise and decay of the photocurrent were obtained.
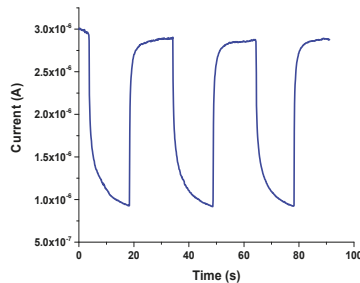
Figure 9. Transient response of the UV photodetector fabricated with Ag catalyst based on Au/β-Ga_2O_3/Silicon photojunction at 10 V.

3.5.3. Detection Mechanism

In dark current measurements, Ag NPs cause a localized Schottky junction and deplete the carriers at the interface of β-Ga_2O_3 nanowires. Therefore, there is a large depletion width at the interface between Ag NPs and β-Ga_2O_3 nanowires, leading to a decrease in the dark current of the UV photodetector.

UV detection mechanism is determined based on the contribution of two different parts, namely, Ag nanoparticles catalyst and P^+-silicon. Under UV illumination, when the photon energy is larger than the bandgap of Ga_2O_3, carriers (electron-hole pairs) are generated [$hv \rightarrow e^- + h^+$]. These enhanced photo-generated carriers by the large electric field increase the carrier density in β-Ga_2O_3 nanowires and improve the photocurrent response. The energy band diagrams of the AgNPs/ β-Ga_2O_3 /p-Si p-n junction is shown in Figure 10a. The band offsets values are estimated using the electron affinity 4.05 eV [62], 4.00 eV [63], and band gap 1.12 eV, 4.9 eV for p-Si, and β-Ga_2O_3, respectively. The work function (φ_{Ga2O3}) and electron affinity (χ_{Ga2O3}) of β-Ga_2O_3 are 4.11 eV and 4.00 eV [63], respectively. This is lower than the work function of Ag (4.26 eV), leading to the formation of a Schottky barrier which prevents the electrons transport from Ag NPs side to Ga_2O_3. In addition, Ag NPs on the surface of Ga_2O_3 is highly influenced with UV light below 320 nm due to the interband transitions, exciting the transition of highly energetic hot electrons from the 4d and 5-sp bands [64–66]. These hot electrons surmount the small height of Schottky barrier and lead to local band bending downward on the Ga_2O_3 side to enable the electron transfer to the conduction band of Ga_2O_3 nanowires.

Regarding the silicon contribution, when the applied voltage is positive on Ga_2O_3, the movement of holes can easily be achieved; hence, photocurrent response is increased. However, if the voltage is negative, the holes are constrained and cannot jump the hill to the side of p-Si. Consequently, the presence of more electrons can increase oxygen molecules absorption and ionization [$O_2 + e^- \rightarrow O_2^-$ [ad]] [67,68]. However, the holes drift to the surface, accumulate, recombine with adsorbed ionized oxygen and form free oxygen molecules from the surface [O_2^- [ad] + $h^+ \rightarrow O_2$]. The remaining

electrons become the majority carriers that contribute to an increase in the photocurrent by generation and recombination until reaching an equilibrium phase.

Nanowires offer a great opportunity to form a higher density of exposed surface states due to the dangling bonds at the surface of nanowires. These trap states of oxygen generated at the surface of Ga_2O_3 nanowires have a large impact on device performance [2]. The detector can be easily and fully integrated on a chip with proper metal contacts similar to the graphene-based detectors [69]. Due to the large surface to volume ratio of nanowires and the existence of Ag NPs, the surface of NWs with trapped oxygen becomes highly sensitive.

(**a**) (**b**)
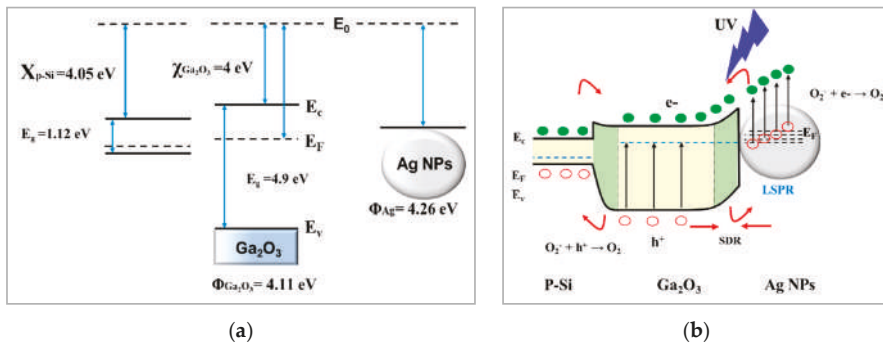
Figure 10. Energy band diagram of Ag NPs and Ga_2O_3 NWs pn P^+-Si. (**a**) at the interface before contact. (**b**) Under UV illumination, the interband transition in Ag NPs enhances the photosensitivity of the UV detection, and more photo-generated holes of Ga_2O_3 NWs migrate to the surface by band bending.

4. Conclusions

Highly oriented, dense, and long β-Ga_2O_3 nanowires were grown on P^+-Si (100) substrate in the presence of a 5 nm thin film of Ag catalyst and oxidation treatment at high temperature (1000 °C). Silver was shown to have a great impact to expedite the growth of Ga_2O_3 nanowires and retain their physical and chemical properties. The morphological, compositional, and electrical properties were explored. The growth mechanism of nanowires on the silicon substrate was discussed. During the growth process, Ga_2O_3 nanowires are highly influenced by silicon as unintentional impurities that increase the n-type doping. The photoresponse under UV irradiation was excellent. The ratio of photo-to-dark current (I_{photo}/I_{dark}) was measured to be around 3.07×10^3 at 10 V. The high photosensitivity could be attributed to the higher electron density in Ga_2O_3 nanowires with Ag NPs. The carrier transport process was shown to have a fast response. The energy band gap and carrier dynamics at the interfaces were discussed. This synthesis can be optimized for sensing, electronics, and photonic applications.

References

1. González-Posada, R.S.F.; den Hertog, M.; Monroy, E. Room-Temperature Photodetection Dynamics of Single GaN Nanowires. *Nano Lett.* **2012**, *12*, 172–176. [CrossRef]
2. Weng, W.Y.; Hsueh, T.J.; Chang, S.J.; Huang, G.J.; Hsueh, H.T. A beta-Ga_2O_3 Solar-Blind Photodetector Prepared by Furnace Oxidization of GaN Thin Film. *IEEE Sens. J.* **2011**, *11*, 999–1003. [CrossRef]

3. Mazeina, L.; Perkins, F.K.; Bermudez, V.M.; Arnold, S.P.; Prokes, S.M. Functionalized Ga_2O_3 Nanowires as Active Material in Room Temperature Capacitance-Based Gas Sensors. *Langmuir* **2010**, *26*, 13722–13726. [CrossRef]

4. Lin, H.J.; Baltrus, J.P.; Gao, H.; Ding, Y.; Nam, C.Y.; Ohodnicki, P.; Gao, P.X. Perovskite Nanoparticle-Sensitized Ga_2O_3 Nanorod Arrays for CO Detection at High Temperature. *ACS Appl. Mater. Interface.* **2016**, *8*, 8880–8887. [CrossRef]

5. Mao, A.K.H.; Gao, J.; Chopdekar, R.; Takamura, Y.; Chowdhury, S.; Islam, M.S. An Investigation of Electrical and Dielectric Parameters of Sol-Gel Process Enabled beta-Ga_2O_3 as a Gate Dielectric Material. *IEEE Trans. Elect. Devices* **2017**, *64*, 2047–2053.

6. Pearton, S.J.; Yang, J.; Carey, P.; Ren, F.; Kim, J.; Tadjer, M.J.; Mastro, M.A. A review of Ga_2O_3 materials, processing, and devices. *Appl. Phys. Rev.* **2018**, *5*, 011301. [CrossRef]

7. Kaya, A. β-Ga_2O_3 films grown via oxidation of GaAs substrates and their device demonstrations. In Proceedings of the Wide Bandgap Power Devices and Applications II SPIE, San Diego, CA, USA, 7–8 August 2017.

8. Patil-Chaudhari, D.; Ombaba, M.; Oh, J.Y.; Mao, H.; Montgomery, K.; Lange, A.; Mahajan, S.; Woodall, J.M.; Islam, M.S. Solar Blind Photodetectors Enabled by Nanotextured β-Ga_2O_3 Films Grown via Oxidation of GaAs Substrates. *IEEE Photon. J.* **2017**, *9*, 1–7. [CrossRef]

9. Otsuka, S.; Katayama, I.; Kozuka, Z. Measurements of Diffusivity of Oxygen in Liquid Silver by Potentiostatic Methods Employing Solid Electrolyte. *Trans. Jpn. Inst. Met.* **1971**, *12*, 442–447. [CrossRef]

10. Nguyen, T.D.; Kim, E.T.; Dao, K.A. Ag nanoparticle catalyst based on Ga_2O_3/GaAs semiconductor nanowire growth by VLS method. *J. Mater. Sci. Mater. Electron.* **2015**, *26*, 8747–8752. [CrossRef]

11. Guo, D.; Wu, Z.; Li, P.; Wang, Q.; Lei, M.; Lie, L.; Tang, W. Magnetic anisotropy and deep ultraviolet photoresponse characteristics in Ga_2O_3: Cr vermicular nanowire thin film nanostructure. *RSC Adv.* **2015**, *5*, 12894–12898. [CrossRef]

12. Lee, S.Y.; Choi, K.H.; Kang, H.C. Growth mechanism of In-doped beta- Ga_2O_3 nanowires deposited by radio frequency powder sputtering. *Mater. Lett.* **2016**, *176*, 213–218. [CrossRef]

13. Choi, K.H.; Cho, K.K.; Kim, K.W.; Cho, G.B.; Ahn, H.J.; Nam, T.H. Catalytic Growth and Structural Characterization of Semiconducting beta-Ga_2O_3 Nanowires. *J. Nanosci. Nanotechnol.* **2009**, *9*, 3728–3733. [CrossRef] [PubMed]

14. Park, S.; Sun, G.J.; Lee, C. UV-assisted room temperature-gas sensing of Ga_2O_3-core/ZnO-shell nanowires. *J. Ceram. Process. Res.* **2015**, *16*, 367–371.

15. Jang, Y.G.; Kim, W.S.; Kim, D.H.; Hong, S.H. Fabrication of Ga_2O_3/SnO_2 core-shell nanowires and their ethanol gas sensing properties. *J. Mater. Res.* **2011**, *26*, 2322–2327. [CrossRef]

16. Ghose, S.; Rahman, M.S.; Arias, A.; Rojas-Ramirez, J.S.; Caro, M.; Nedev, N.; Droopad, R. Structural and optical properties of beta-Ga_2O_3 thin films grown by plasma-assisted molecular beam epitaxy. *J. Vac. Sci. Technol. B* **2016**, *34*. [CrossRef]

17. Feng, Q.; Li, F.; Dai, B.; Jia, Z.; Xie, W.; Xu, T.; Lu, X.; Tao, X.; Zhang, J.; Hao, Y. The properties of gallium oxide thin film grown by pulsed laser deposition. *Appl. Surf. Sci.* **2015**, *359*, 847–852. [CrossRef]

18. Han, W.Q.; Kohler-Redlich, P.; Ernst, F.; Ruhle, M. Growth and microstructure of Ga_2O_3 nanorods. *Solid State Commun.* **2000**, *115*, 527–529. [CrossRef]

19. Cao, C.B.; Chen, Z.; An, X.Q.; Zhu, H.S. Growth and field emission properties of cactus-like gallium oxide nanostructures. *J. Phys. Chem. C* **2008**, *112*, 95–98. [CrossRef]

20. Sharma, S.; Sunkara, M.K. Direct synthesis of gallium oxide tubes, nanowires, and nanopaintbrushes. *J. Am. Chem. Soc.* **2002**, *124*, 12288–12293. [CrossRef]

21. Pallister, P.J.; Buttera, S.C.; Barry, S.T. Self-seeding gallium oxide nanowire growth by pulsed chemical vapor deposition. *Phys. Status Solidi Appl. Mater. Sci.* **2015**, *212*, 1514–1518. [CrossRef]

22. Zhao, Y.Y.; Frost, R.L.; Yang, J.; Martens, W.N. Size and morphology control of gallium oxide hydroxide GaO(OH), nano- to micro-sized particles by soft-chemistry route without surfactant. *J. Phys. Chem. C* **2008**, *112*, 3568–3579. [CrossRef]

23. Reddy, L.S.; Ko, Y.H.; Yu, J.S. Hydrothermal Synthesis and Photocatalytic Property of beta-Ga_2O_3 Nanorods. *Nanoscale Res. Lett.* **2015**, *10*, 364. [CrossRef] [PubMed]

24. Dai, X.; Zhang, S.; Wang, Z.; Adamo, G.; Liu, H.; Huang, Y.; Couteau, C.; Soci, C. GaAs/AlGaAs Nanowire Photodetector. *Nano Lett.* **2014**, *14*, 2688–2693. [CrossRef]

25. Alhalaili, B.; Mao, H.; Islam, M.S. Ga$_2$O$_3$ Nanowire Synthesis and Device Applications. In *Novel Nanomaterials—Synthesis and Applications*; IntechOpen Limited: London, UK, 2017; Volume 2.

26. Ogita, M.; Higo, K.; Nakanishi, Y.; Hatanaka, Y. Ga$_2$O$_3$ thin film for oxygen sensor at high temperature. *Appl. Surf. Sci.* **2001**, *175–176*, 721–725. [CrossRef]

27. Kim, H.W.; Kim, N.H.; Lee, C. Growth of Ga$_2$O$_3$ thin films on Si (100) substrates using a trimethylgallium and oxygen mixture. *J. Mater. Sci.* **2004**, *39*, 3461–3463. [CrossRef]

28. Wu, Y.L.; Chang, S.-J.; Weng, W.-Y.; Liu, C.; Tsai, T.Y.; Hsu, C.-L.; Chen, K.C. Ga$_2$O$_3$ Nanowire Photodetector Prepared on SiO$_2$/Si Template. *IEEE Sens. J.* **2013**, *13*, 2368–2373. [CrossRef]

29. Alhalaili, B.; Bunk, R.; Vidu, R.; Islam, M.S. Dynamics Contributions to the Growth Mechanism of Ga$_2$O$_3$ Thin Film and NWs Enabled by Ag Catalyst. *Nanomaterials* **2019**, *9*, 1272. [CrossRef]

30. Glickman, E.; Levenshtein, M.; Budic, L.; Eliaz, N. Interaction of liquid and solid gallium with thin silver films: Synchronized spreading and penetration. *Acta Mater.* **2011**, *59*, 914–926. [CrossRef]

31. Detz, H.; Kriz, M.; MacFarland, D.; Lancaster, S.; Zederbauer, T.; Capriotti, M.; Andrews, A.M.; Schrenk, W.; Strasser, G. Nucleation of Ga droplets on Si and SiOx surfaces. *Nanotechnology* **2015**, *26*. [CrossRef]

32. Mao, H.; Alhalaili, B.; Kaya, A.; Dryden, D.M.; Woodall, J.M.; Islam, M.S. Oxidation of GaAs substrates to enable β-Ga$_2$O$_3$ films for sensors and optoelectronic devices (SPIE Optical Engineering + Applications). In Proceedings of the Wide Bandgap Power Devices and Applications II, San Diego, CA, USA, 7–8 August 2017.

33. Song, P.Y.; Wu, Z.Y.; Shen, X.Y.; Kang, J.Y.; Fang, Z.L.; Zhang, T.Y. Self-consistent growth of single-crystalline ((2)over-bar01) beta-Ga$_2$O$_3$ nanowires using a flexible GaN seed nanocrystal. *Crystengcomm* **2017**, *19*, 625–631. [CrossRef]

34. Chun, H.J.; Choi, Y.S.; Bae, S.Y.; Seo, H.W.; Hong, S.J.; Park, J.; Yang, H. Controlled structure of gallium oxide nanowires. *J. Phys. Chem. B* **2003**, *107*, 9042–9046. [CrossRef]

35. Crist, V. *Handbook of Monochromatic XPS Spectra: The Elements of Native Oxides*; Wiley-VCH: Wenheim, Germany, 2000.

36. Logofatu, C.; Negrila, C.C.; Ghita, R.V.; Ungureanu, F.; Cotirlan, C.; Manea, C.G.A.S.; Lazarescu, M.F. Study of SiO$_2$/Si Interface by Surface Techniques. *Cryst. Silicon Prop. Uses* **2011**, *23–42*. [CrossRef]

37. Dong, C.Y.; Shang, D.S.; Shi, L.; Sun, J.; Shen, B.G.; Zhuge, F.; Li, R.-W.; Chen, W. Roles of silver oxide in the bipolar resistance switching devices with silver electrode. *Appl. Phys. Lett.* **2011**, *98*. [CrossRef]

38. Salido, I.L. Electronic and Geometric Properties of Silver and Gold Nanoparticles. Ph.D. Thesis, University of Konstanz, Konstanz, Germany, 2007.

39. Chevalier, P.Y. Thermodynamic Evaluation of the Ag-Si System. *Thermochim. Acta* **1988**, *130*, 33–41. [CrossRef]

40. Varley, J.B.; Weber, J.R.; Janotti, A.; van de Walle, C.G. Oxygen vacancies and donor impurities in beta-Ga$_2$O$_3$. *Appl. Phys. Lett.* **2010**, *97*. [CrossRef]

41. Gould, A.L.; Kadkhodazadeh, S.; Wagner, J.B.; Catlow, C.R.A.; Logsdail, A.J.; di Vece, M. Understanding the Thermal Stability of Silver Nanoparticles Embedded in a-Si. *J. Phys. Chem. C* **2015**, *119*, 23767–23773. [CrossRef]

42. Zinkevich, M.; Aldinger, F. Thermodynamic assessment of the gallium-oxygen system. *J. Am. Ceram. Soc.* **2004**, *87*, 683–691. [CrossRef]

43. Klinedinst, K.A.; Stevenson, D.A. Oxygen Diffusion in Liquid Gallium and Indium. *J. Electrochem. Soc.* **1973**, *120*, 304–308. [CrossRef]

44. Zhou, Z.Y.; Ma, Y.M.; Han, Q.F.; Liu, Y.L. Solubility, permeation, and capturing of impurity oxygen in Au/Ag: A comparative investigation from first-principles. *Comput. Mater. Sci.* **2016**, *114*, 79–85. [CrossRef]

45. Ramanarayanan, T.A.; Rapp, R.A. The Diffusivity and Solubility of Oxygen in Liquid Tin and Solid Silver and the Diffusivity of Oxygen in Solid Nickel. *Metall. Trans.* **1972**, *3*, 3239–3246. [CrossRef]

46. Park, J.H. Measuring Oxygen Diffusivity and Solubility in Solid Silver with a Gas-Tight Electrochemical-Cell. *Mater. Lett.* **1990**, *9*, 313–316. [CrossRef]

47. Kontoulis, I.; Steele, B.C.H. Determination of Oxygen Diffusion in Solid Ag by an Electrochemical Technique. *Solid State Ion.* **1991**, *47*, 317–324. [CrossRef]

48. Heshmatpour, B.; Stevenson, D.A. An Electrochemical Study of the Solubility and Diffusivity of Oxygen in the Respective Liquid-Metals Indium, Gallium, Antimony and Bismuth. *J. Electroanal. Chem.* **1981**, *130*, 47–55. [CrossRef]

49. Girija, K.; Thirumalairajan, S.; Mastelaro, V.R.; Mangalaraj, D. Catalyst free vapor–solid deposition of morphologically different β-Ga$_2$O$_3$ nanostructure thin films for selective CO gas sensors at low temperature. *Anal. Method* **2016**, 3224–3235. [CrossRef]

50. Butt, D.P.; Park, Y.; Taylor, T.N. Thermal vaporization and deposition of gallium oxide in hydrogen. *J. Nucl. Mater.* **1999**, *264*, 71–77. [CrossRef]

51. Kumar, S.; Singh, R. Nanofunctional gallium oxide (Ga$_2$O$_3$) nanowires/nanostructures and their applications in nanodevices. *Phys. Status Solidi Rapid Res. Lett.* **2013**, *7*, 781–792. [CrossRef]

52. Xu, C.; Chung, S.; Kim, M.; Kim, D.E.; Chon, B.; Hong, S.; Joo, T. Doping of Si into GaN nanowires and optical properties of resulting composites. *J. Nanosci. Nanotechnol.* **2005**, *5*, 530–535. [CrossRef]

53. Walukiewicz, W. Intrinsic limitations to the doping of wide-gap semiconductors. *Phys. B* **2001**, *302*, 123–134. [CrossRef]

54. Matsuzaki, K.; Yanagi, H.; Kamiyab, T. Field-induced current modulation in epitaxial film of deep-ultraviolet transparent oxide semiconductor Ga$_2$O$_3$. *Appl. Phys. Lett.* **2006**, *88*. [CrossRef]

55. Villora, E.G.; Shimamura, K.; Yoshikawa, Y.; Ujiie, T.; Aoki, K. Electrical conductivity and carrier concentration control in beta-Ga$_2$O$_3$ by Si doping. *Appl. Phys. Lett.* **2008**, *92*. [CrossRef]

56. Shao, D.L.; Yu, M.P.; Sun, H.T.; Hu, T.; Lian, J.; Sawyer, S. High responsivity, fast ultraviolet photodetector fabricated from ZnO nanoparticle-graphene core-shell structures. *Nanoscale* **2013**, *5*, 3664–3667. [CrossRef]

57. Oh, S.; Mastro, M.A.; Tadjer, M.J.; Kim, J. Solar-Blind Metal-Semiconductor-Metal Photodetectors Based on an Exfoliated beta- Ga$_2$O$_3$ Micro-Flake. *ECS J. Solid State Sci. Technol.* **2017**, *6*, Q79–Q83. [CrossRef]

58. Guo, D.Y.; Wu, Z.; An, Y.H.; Guo, X.; Chu, X.L.; Sun, C.L.; Li, L.; Li, P.G.; Tang, W.H. Oxygen vacancy tuned Ohmic-Schottky conversion for enhanced performance in β-Ga$_2$O$_3$ solar-blind ultraviolet photodetectors. *Appl. Phys. Lett.* **2014**, *105*, 023507. [CrossRef]

59. Manjavacas, A.; Liu, J.; Kulkarni, V.; Nordlander, P. Plasmon-Induced Hot Carriers in Metallic Nanoparticles. *ACS Nano* **2014**, *8*, 7630–7638. [CrossRef]

60. Oh, J.; Ma, J.; Yoo, G. Simulation study of reduced self-heating in β-Ga$_2$O$_3$ MOSFET on a nano-crystalline diamond substrate. *Results Phys.* **2019**, *13*, 102151. [CrossRef]

61. Kleyn, A.W.; Butler, D.A.; Raukema, A. Dynamics of the interaction of O$_2$ with silver surfaces. *Surf. Sci.* **1996**, *363*, 29–41. [CrossRef]

62. Hou, Y.N.; Mei, Z.X.; Liang, H.L.; Ye, D.Q.; Liang, S.; Gu, C.Z.; Du, X.L. Comparative study of n-MgZnO/p-Si ultraviolet-B photodetector performance with different device structures. *Appl. Phys. Lett.* **2011**, *98*, 263501. [CrossRef]

63. Mohamed, M.; Irmscher, K.; Janowitz, C.; Galazka, Z.; Manzke, R.; Fornari, R. Schottky barrier height of Au on the transparent semiconducting oxide beta-Ga$_2$O$_3$. *Appl. Phys. Lett.* **2012**, *101*. [CrossRef]

64. Aslam, U.; Rao, V.G.; Chavez, S.; Linic, S. Catalytic conversion of solar to chemical energy on plasmonic metal nanostructures. *Nat. Catal.* **2018**, *1*, 656–665. [CrossRef]

65. Emilio, M.G.; Alarcon, I.; Klas, I.U. *Silver Nanoparticle Applications: In the Fabrication and Design of Medical and Biosensing Devices*; Springer Berlin Heidelberg: New York, NY, USA, 2015.

66. Arora, K.; Kumar, V.; Kumar, M. Silver plasmonic density tuned polarity switching and anomalous behaviour of high performance self-powered β-gallium oxide solar blind photodetector. *arXiv* **2018**, arXiv:1809.10724. Available online: https://arxiv.org/pdf/1809.10724 (accessed on 25 October 2019).

67. Soci, C.; Zhang, A.; Xiang, B.; Dayeh, S.A.; Aplin, D.P.; Park, J.; Bao, X.Y.; Lo, Y.H.; Wang, D. ZnO nanowire UV photodetectors with high internal gain. *Nano Lett.* **2007**, *7*, 1003–1009. [CrossRef] [PubMed]

68. Prades, J.D.; Hernandez-Ramirez, F.; Jimenez-Diaz, R.; Manzanares, M.; Andreu, T.; Cirera, A.; Romano-Rodriguez, A.; Morante, J.R. The effects of electron-hole separation on the photoconductivity of individual metal oxide nanowires. *Nanotechnology* **2008**, *19*. [CrossRef] [PubMed]

69. Ding, Y.; Guan, X.; Zhu, X.; Hu, H.; Bozhevolnyi, S.I.; Oxenloewe, L.K.; Jin, K.; Mortensen, N.A.; Xiao, S. Effective electro-optic modulation in low-loss graphene-plasmonic slot waveguides. *Nanoscale* **2017**, *9*. [CrossRef] [PubMed]

Article

The Design and Experimental Development of Air Scanning Using a Sniffer Quadcopter

Endrowednes Kuantama [1], Radu Tarca [2,*], Simona Dzitac [3], Ioan Dzitac [4,5], Tiberiu Vesselenyi [2] and Ioan Tarca [6]

1. Department of Electrical Engineering, Pelita Harapan University, Tangerang 15811, Indonesia
2. Mechatronics Department, University of Oradea, 1 Universitatii St., Oradea 410087, Romania
3. Energy Engineering Department, University of Oradea, 1 Universitatii St., Oradea 410087, Romania
4. Department of Mathematics, Computer Science, Aurel Vlaicu University of Arad, St. Elena Dragoi, Arad 310330, Romania
5. R & D Center: "Cercetare Dezvoltare Agora", Agora University of Oradea, St. Piata Tineretului, Oradea 410087, Romania
6. Mechanical Engineering and Automotive Department, University of Oradea, 1 Universitatii St., Oradea 410087, Romania
* Correspondence: rtarca@uoradea.ro; Tel.: +40-259-408-249

Abstract: This study presents a detailed analysis of an air monitoring development system using quadcopters. The data collecting method is based on gas dispersion investigation to pinpoint the gas source location and determine the gas concentration level. Due to its flexibility and low cost, a quadcopter was integrated with air monitoring sensors to collect the required data. The analysis started with the sensor placement on the quadcopter and their correlation with the generated vortex. The reliability and response time of the sensor used determine the duration of the data collection process. The dynamic nature of the environment makes the technique of air monitoring of topmost concern. The pattern method has been adapted to the data collection process in which area scanning was marked using a point of interest or grid point. The experiments were done by manipulating a carbon monoxide (CO) source, with data readings being made in two ways: point source with eight sampling points arranged in a square pattern, and non-point source with 24 sampling points in a grid pattern. The quadcopter collected data while in a hover state with 10 s sampling times at each point. The analysis of variance method (ANOVA) was also used as the statistical algorithm to analyze the vector of gas dispersion. In order to tackle the uncertainty of wind, a bivariate Gaussian kernel analysis was used to get an estimation of the gas source area. The result showed that the grid pattern measurement was useful in obtaining more accurate data of the gas source location and the gas concentration. The vortex field generated by the propeller was used to speed up the accumulation of the gas particles to the sensor. The dynamic nature of the wind caused the gas flow vector to change constantly. Thus, more sampling points were preferred, to improve the accuracy of the gas source location prediction.

Keywords: quadcopter; drone; pollutant; carbon; air monitoring; kernel

---

1. Introduction

Several types of air, water, and soil pollutants are impossible to avoid, being encountered in almost all countries. Some of them represent real threats, rising risks to human health and environment degradation, such as (for the case of big cities) air pollution generated by stationary sources (e.g., factories, power plants), by mobile sources (e.g., cars, buses), and also natural sources (e.g., windblown dust, wildfires). Pollutants can be classified into two types: pollutants having known sources, and

pollutants having unknown sources. Each of them have specific approaches related to measurement methods and source identification. Most developed countries have developed law regulations to organize regular air quality supervision for primary urban pollutants [1–3]. The regulations could be put into practice only by monitoring regularly and accurately the concentration of the pollutant. Usually, stationary monitoring stations with network systems located in critical areas for data collection and processing are used [4–6]. This type of monitoring is limited by the sampling location and data accessibility [7]. With the advance in technology, this old paradigm gets surpassed by the utilization of sensor technology which is superior in respect to the low-cost material, simplicity and affordability, and also the portability of the air pollution monitoring systems [8,9]. The technology above can be integrated into an air scanning system by using an unmanned aerial vehicle (UAV) characterized by good maneuverability, controllable altitude, and location. [10,11]. Although the recent developments in UAV technology are promising, there are studies that show that in particular areas of applications (for example environmental restoration monitoring) there are still many issues to be addressed [12]. A comprehensive review of using autonomous vehicles in environmental monitoring for pollution source localization is presented in [13]. A design and manufacturing protocol for the low cost and low weight quadcopter platform prototype for the purpose of environmental monitoring and research in order to assess ecological devastation of the natural environment is presented in [14].

Different recently published studies show the interest of researchers in the applications of quadcopters used in different fields of environmental monitoring like sampling of different substances, monitoring soil contamination, and response to natural disasters. Lally, H.T. et. al. [15] addresses the development of water sampling by using drones specially equipped with water-sampling devices. The issues of biological and physico-chemical sampling are described as well as some solutions for these issues. The authors show that it is envisaged that drone-assisted water sampling will act as a pivotal supporting tool if the cost benefit analysis of the application gives positive results. In [16] the author's goal was to introduce and test a method able to predict copper accumulation points, using aerial photos taken by drones and micro-rill network modeling. In this case the drone collected photogrammetric data, which was compared with the results obtained by computer modeling. According to the results of the study, the authors were able to predict zones of copper accumulation at a plot scale. Other important applications of drones are presented in [17] where is stated that UAV's can have a crucial role in the case of natural disaster response and humanitarian relief aid. The key areas of intervention in this case are: aerial monitoring of post-natural disaster damage, natural disaster logistics and cargo delivery, and post-natural disaster aerial assessment. An application which is close to our paper's subject is presented in [18] where gas concentrations resulting from an underground coal fire (carbon dioxide emissions) were measured using aerial monitoring with drones. The authors state that it is estimated that these fires generate as much as 3% of the world's annual carbon dioxide emissions and that drone collected gas concentration data provides a safe alternative for evaluating the rank of burning coal deposits.

A portable air scanning system was developed using a quadcopter equipped with an air scanning sensor to perform air quality measurement, thus called a 'sniffer' quadcopter. The development began with analysis of the correlation between propeller's air trajectories and the sensors placement, determination of an appropriate flying pattern to optimize the air measurement, and investigation of ways to minimize wind effects in the measurement process. A computational method was used to ascertain the sensor's placement on the quadcopter, and the result was proven in the field test. The low-cost and portable gas sensors MQ-2 and MQ-135 were used to measure carbon concentration in the air. To develop this system, the sensor placement in the quadcopter is critical. Using computational fluid dynamic (CFD) simulation, the vortex field generated by the propeller was analyzed to determine the best place for sensor mounting. With an appropriate mounting place, the response time and the accuracy of data collected by the sensor can be increased [19]. Even with low-cost instruments, the data accuracy and detection range can be as good as conventional monitoring [20]. Two types of flight patterns for air measurement were used to detect the direction of gas dispersion and discover

the gas source location and gas concentration level. The present system was designed to perform flight pattern measurement methods which consist of point source measurement and non-point source measurement. A path planning with eight sampling points around the gas source was used to obtain the gas concentration at a known point source, for example, an industrial emission or chimney [21]. The sample points formed a square with the gas source in the middle. Thus, more accurate data can be obtained despite the dynamic environmental change. On the other hand, for a non-point or unknown sources such as forest fires or pipe gas leaks, a grid pattern with 24 sampling points was used to detect the gas source location based on the gas dispersion measurement and analysis. To optimize the measurement results, wind effect was considered. Gas source location and gas concentration level can only be estimated with statistical methods. In the case of fires, an extensive study had been made to detect sources with UAVs equipped with thermal detection capabilities [22]. Gas concentration detection (for example CO concentration) can give additional information besides the thermal data. The method described in this paper is much more cost effective and can be used by smaller communities (at the level of small cities and towns) to check for fire sources that can be a threat for the population or economy. The results from both methods were evaluated using analysis of variance (ANOVA) to obtain the gas concentration at the source, and using a Gaussian dispersion model to analyze the gas dispersion. In the Gaussian dispersion model, parameters such as wind speed and direction, source term, etc., were obtained by monitoring data to acquire positioned trajectories with bivariate input environmental data [23,24]. In a small scale of gas measurement (<100 m), a sparse Gaussian kernel method was used as a statistical evaluation on a two-dimensional spatial model of a grid pattern to deal with the specific properties of gas dispersion, including the turbulent features of the wind [25,26]. Spatial integration is made by convolving sensor readings and modeling the information data of the point measurements with a Gaussian kernel method. The grid pattern size and data retrieval time depend on the sensors' sensitivity. This research is limited with the usage of low-cost sensors and a narrow pattern, but these don't affect our goal which is to prove that the Gaussian kernel method is suitable to analyze gas dispersion vectors and detect gas source locations.

The sniffer quadcopter was designed to work automatically according to the command input in the pre-flight setting, one of which is a GPS coordinate. In automatic mode, the quadcopter will fly to the target point to perform the measurement with the pre-programmed flight pattern. In this research, the measurement target is carbon monoxide (CO) because it was easily found and/or made and was measured with low-cost and portable gas sensors MQ-2 and MQ-135. The tests were performed on an aerial zone with a maximum of 24 sampling points, each measuring 1 m^2. On each point, the sampling time was 10 s and the data was collected whilst maintaining the quadcopter in hover mode. The aerial zone was intentionally small so the research could be focused on the algorithm for gas source determination and gas concentration level, besides optimizing the sensor placement on quadcopter. The main purpose of our paper is to study the possibility of using low-cost aerial monitoring system and pollution source detection, which can be available to smaller communities (towns, NGOs, environmental protection associations) in order to detect and prevent threats to the environment.

2. Sniffer Quadcopter Design and Analysis

Prior to a quadcopter design, it is imperative to know the application of the drone itself, thus one can estimate the required lifting force in accordance to the total weight of the quadcopter. The physical factors of quadcopter design related to lifting force are propeller diameter, propeller's angle of attack, quadcopter size, and rotor angular velocity. Greater lifting force and high-speed flight are not synonymous with larger propellers or high-speed rotors because they can cancel out the advantages of using a quadcopter. For example, an oversized propeller produces substantial air flows which can cause turbulence and flight instability.

To optimize the sniffer quadcopter design, a detailed analysis of the vortex field generated by the propeller's angular velocity and its effect on sensor placement on the quadcopter was performed. This

research used 13-inch propellers, a 380 rpm/v rotor, and 18.5-volt lithium polymer (LiPo) batteries. With this specification, it can be deduced that the maximum no-load rotor speed is 7030 rpm. To get a more accurate calculation of the lifting force, the computational fluid dynamic (CFD) module of the SolidWorks software was used. Table 1 shows detailed specifications of the quadcopter. The frame size was 460 mm, the quadcopter weight was 1800 g, and the sensor weight was 100 g. Quadcopter flight time depends on the capacity of the LiPo batteries used and the maximum carriable load depends on the total lifting force. The magnitude of the force analyzed with the CFD method and the correlation between thrust, frame size, and level of stability will be explained in detail in the propeller vortex field section.

Table 1. Detail specifications of sniffer quadcopter.

Sniffer Quadcopter	Specifications
Frame material	Carbon fiber
Frame size	460 mm × 460 mm × 140 mm
	(width × length × height)
Propeller size	13 inches
Motor (rpm/V) Flight time	380 ± 40 min
Max. carry load	2460 gr
	(including quadcopter weight)

Due to short flight duration (<40 min), several settings must be initialized to permit the quadcopter to perform optimally, such as the target point, in the form of GPS coordinate, and quadcopter flight behavior, in the form of altitude, speed, and measurement pattern. The settings can also be made on the ground station system through the designed mission planner system which can communicate with the sniffer quadcopter using 433 MHz telemetry [27,28]. The architecture of the aerial platform system is presented in Figure 1. The ground station using an open source web application platform was also used to monitor all measurement processes. With the autonomous flight pattern, the quadcopter will run the sequential process according to the pre-flight command list. For intentional interruption of the process, an emergency system was designed so the user's command can be delivered via the ground station control which will force the quadcopter either to make a landing or to fly back to the home coordinates. On board the sniffer quadcopter system one can find two controllers: the flight controller and sniffer microcontroller. The flight controller serves to maintain the stability of the maneuvers, together with an orientation sensor and a tracking pattern based on the input coordinate [29]. The sniffer system serves to perform air scanning and save the data on a memory card. Both controllers are always communicating with each other to determine when to carry out the data retrieval process.

2.1. Vortex Field Analysis

Data retrieval was performed while keeping the quadcopter in the hover state at each analyzed point. It was noticed that data were profoundly affected by the air trajectories of the propeller's vortex field and also by the wind. The vortex method was applied [30,31] to analyze the aerodynamic behavior of the aircraft. The authors' previous studies [32,33] explain the quadcopter's propeller design and flight stability analysis. As mentioned previously, the maximum no-load rotor speed was 7030 rpm. When connected with a 13-inch propeller, the maximum rotor speed was 4080 rpm. To obtain the total thrust, a CFD analysis was performed using the SolidWorks software. The computationally analyzed parameter was the propeller rotation with a set up rotation area, with 0.1% turbulence intensity with 0.002 m turbulence length, at a thermodynamic pressure of 101,325 Pa and a temperature of 293.2 K. Each propeller had a rotation region, in which a diagonal pair of propellers rotated clockwise (CW) and the other diagonal pair rotated counterclockwise (CCW) with a velocity of 0–4080 rpm. The simulation generated the values of the vertical force and the total air velocity in all propeller rotation areas. With a velocity of 4080 rpm, each rotor can produce a total thrust of 24 N. Besides that, the vortex field was

also modeled using the same software with the same parameters. Figure 2 below shows the design of the sniffer quadcopter.
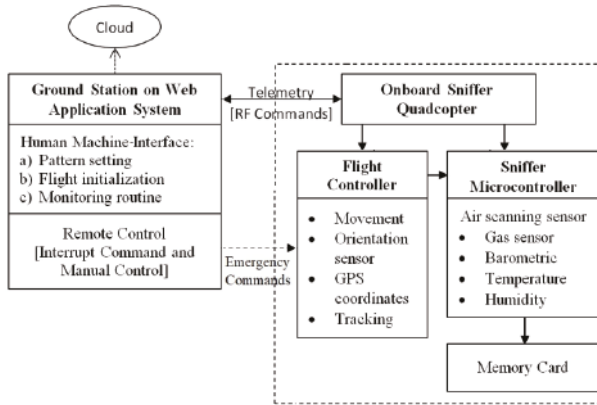
Figure 1. Sniffer quadcopter system architecture.
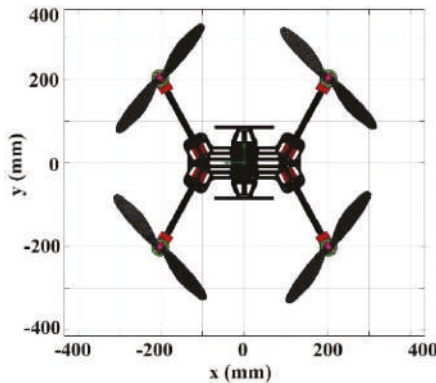
Figure 2. The sniffer quadcopter design.

In this study, the generated vortex field and its correlation with sensor position and the effect of the environmental change have been analyzed. The vortex field was modeled using computational fluid dynamics (CFD). The results showed that the air trajectories generated by each propeller rotation with maximum velocity didn't affect each other because the air velocities produced were the same. Figure 3 presents the analysis of the vortex field using CFD. The propellers having a maximum angular velocity (Ω_{max}) of 4080 rpm generate an air velocity of 6 ms^{-1} (v_{air}) oriented downwards (along the z-axis); the air velocity measured between propellers ($-30 \leq x \leq 30$) mm and ($-30 \leq y \leq 30$) mm was 0.5 ms^{-1}. For this study, we used a sniffer quadcopter with a total load of 1900 g keeping it in hover state at 80% of the maximum speed, which corresponds to 3264 rpm. Consequently, the angular velocity generated an air velocity of 4.8 ms^{-1} on the propeller, but the vortex fields generated by the propellers did not affect each other.
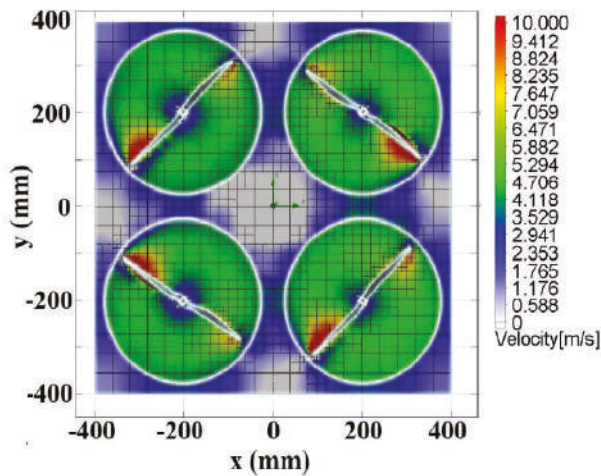
Figure 3. Air velocity dispersion.

This means that the vortices generated during the hover state with no wind effects didn't lead to turbulence on the quadcopter frame, which proved that the selected propeller diameter, quadcopter size, and rotor speed were appropriate. In the presence of wind effects, the quadcopter control system will work to maintain stability.

2.2. Correlation Between Vortex Field and Sensor Position

The transport of the monitored gas towards the gas sensor is a critical process to obtain accurate data which can be easily affected by the disturbances generated by the propeller's vortex. Sensor position, gas distribution, and wind resistance are factors which influence the measurement results [34,35]. The downside of using a low-cost gas sensor is the low response time. To determine the sensors' placement, one must consider the sensors' response time whilst still maintaining the quadcopter stability. The use of an extended pole to place the sensors outside of propeller vortex field is not advisable because it can affect quadcopter stability. Even though it might not matter much with a small-scale sensor, an extended pole attached with a heavier sensor will surely be effect by the wind. Thus, the design and computational analysis for the gas scanning sensors' placement on the quadcopter's frame were only tested for two points, i.e., point A and point B.

Point A corresponds to the placement of the sensors at the bottom of the main frame; thus, the gas flow is not influenced by the propeller's vortex. Point B corresponds to the situation in which the sensors are mounted on the front side of the frame so that the propeller's vortex blows the gas directly on the sensors. Both positions were analyzed using CFD prior to the field tests. The results of the simulation process for the case of the propeller's maximum angular velocity (4080 rpm) are presented in Figure 4. It can be noticed that the maximum air velocity occurs in each propeller rotation area while the minimum corresponds to the center of the frame. The placement of the air scanning sensor relies heavily on its type and response time. The following coordinates describe the point A's position: $(-50 \leq y \leq 50)$ mm; $(-130 \leq x \leq 130)$ mm, and $(z > -50)$ mm, where air speed is $\left(v_{air} \approx 0 \text{ ms}^{-1}\right)$ as seen in Figure 4c.
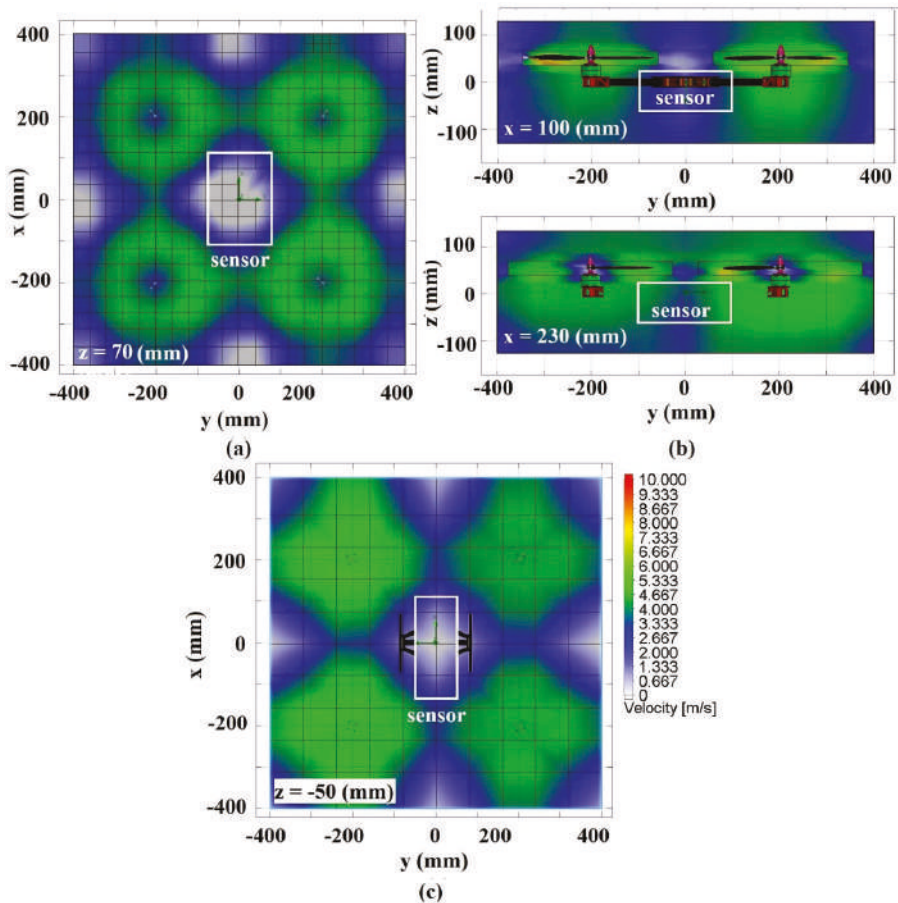
Figure 4. Air velocity on quadcopter in various perspectives: (**a**) Point A—top view; (**b**) Point B—front view; (**c**) Point A—bottom view.

It is evident that at point A, each sensor is surrounded by the propeller's vortices, with air trajectories going downward to produce thrust. For the analyzed gas to reach the sensor while the quadcopter is in the hover state, it is essential that the gas velocity be higher than the air velocity generated by the propeller. The vector of gas velocity towards the sensor, and the magnitude of total resultant velocity are very crucial to determine whether the gas can reach the sensor or not. Another way is to place the sensor on the top of the frame ($z > 70$) mm or higher than the propeller as seen in Figure 4a. Point B corresponds to the placement of the sensor on the front of the frame ($x > 100$) mm and takes advantage of the air trajectories generated by the propeller, as shown in Figure 4b. In this position, the gas around propeller is suctioned out by the propeller and passed through the sensor before going downward. This placement facilitates the gas to reach the sensor and is suitable for sensors with low response time.

Figure 5 shows the comparison between three sensor placements in a field test with durations of 160 s and a stable gas source position. The aim of this test was not to compare the reading of gas concentration, but rather to see the sensors' response time in different mounting positions. Analysis of point A showed that the gas concentration decreases as the speed of the propeller increases, and the gas concentration increases as the speed of the propeller decreases. This result was linear with

the CFD analysis results. This means the sensors were obstructed by the propeller's air trajectories and thus unable to properly measure the gas concentration. On the other hand, analysis of point B showed that the sensors' reading was a bit affected by propeller speed change, but the sensor could work well when the propeller speed was stable. This was due to the propeller's air flow 'directing' the gas towards the sensor. Lastly, the analysis was conducted on sensors only. The result showed the most stable measurement because it didn't get affected by the propeller's air trajectory. In conclusion, taking advantage of the propeller's air trajectory to 'direct' the gas toward the sensors resulted in valid data reading, on the condition that the measurement was done while the quadcopter was in a hover state to ensure stable propeller speed in every measurement.
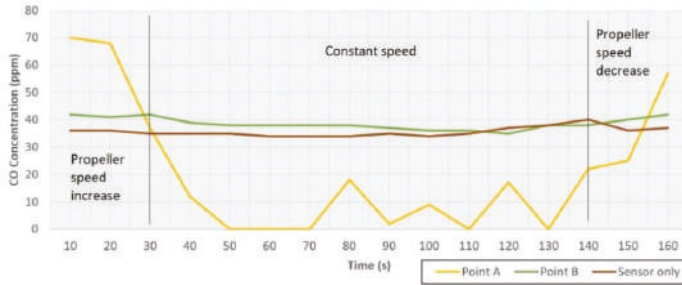
Figure 5. Comparison of sensor positions.

3. Grid Pattern Analysis

The grid pattern and wind algorithm were integrated into the gas measurement process which was dynamically distributed [36]. Measurements can be performed for the case of either point source or non-point source. The point-source measurement is used for gas emissions with known locations, for example, chimney exhaust gases in industrial districts. However, as the gas dispersion depends on the wind direction an eight-point (P1–P8) square pattern is used to cover all wind blowing directions, as illustrated in Figure 6a. On the other hand, non-point source measurement is used to locate the gas source based on the particle density and wind direction. Data acquisition for this method allows the user to observe the gas dispersion gradient toward the closest point to the source (which has the highest particle concentration), exemplified in Figure 6b. In the field test, the grid pattern with 24 sample points (S1–S24) placed in a 4 × 6 matrix was able to be extrapolated using the Gaussian kernel method. The size of the cell depends on the sensitivity of the gas sensor.

(**a**)　　　　　　　　　　　　　　　　　　(**b**)
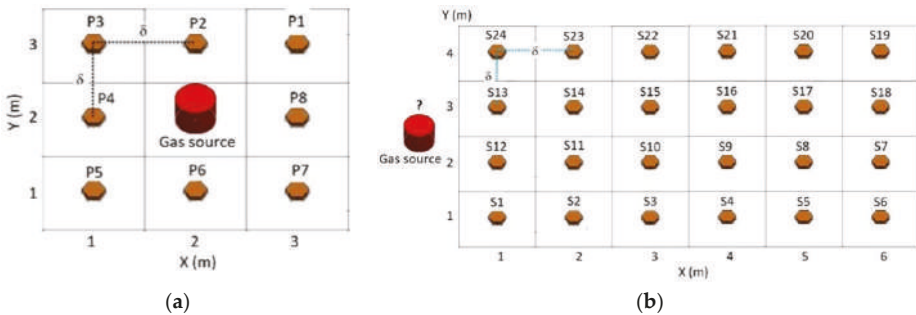
Figure 6. Pattern model with (**a**) point source, (**b**) non-point source.

The quadcopter's flying sequence and sample points took place according to the grid numbers shown in Figure 6. The methodology consisted of collecting sample data of gas concentration $\left(G_s^{(i)}\right)$ on each cell k, at the location $x^{(i)}$ where the symbol i represents the number of the measurement sample.

As we mentioned above, sometimes position error occurred while collecting data. Thus, position adjustments (ΔG) were needed to ascertain that the measurements were made in the center of each grid cell (G_c). The post-processing data errors relative to positioning errors on each cell can be minimized by data recovering process using Equation (1). The symbol k represents the number of the grid cell.

$$\Delta G_i^{(k)} = \left| G_c^{(k)} - G_s^{(i)} \right|,\tag{1}$$

where, for the k cell, in which a number of $m^{(k)}$ measurements were made, $G_c^{(k)}$ value can be estimated with

$$G_c^{(k)} = \frac{\sum_{i=0}^{m^{(k)}} G_s^{(k)(i)}}{m^{(k)}}\tag{2}$$

The differentiation of gas concentration between the center of the grid cell and the value acquired by the sensor was used to locate the gas source. If the relationship $\left(\Delta G^{(k)} > \Delta G^{(k-1)}\right)$ is true, it means that the flight pattern moves toward the gas source, and conversely $\left(\Delta G^{(k)} < \Delta G^{(k-1)}\right)$ means that the flight pattern moves away from the gas source. This post-processing of gas concentration variation in each cell helps in understanding the correlation between flight position and gas dispersion. In order to analyze the gas dispersion behavior, both for point source and non-point source models, an adaptive threshold with binary sample was used, as seen in Equation (3) [37].

$$\overline{P}^{(k)} = \overline{S}^{(k)} = \begin{cases} 1 & \rightarrow \left(\Delta G_t^{(k)} > \Delta G_t^{(k-1)}\right) \\ 0 & \rightarrow \left(\Delta G_t^{(k)} \leq \Delta G_t^{(k-1)}\right) \end{cases}\tag{3}$$

For both methods, sample acquisition was done with the same iteration (t) The first value of gas concentration acquired has been used as the reference for measurements. The $\left(\overline{P}^{(k)} = \overline{S}^{(k)} = 1\right)$ indicates an increase in gas concentration, whereas $\left(\overline{P}^{(k)} = \overline{S}^{(k)} = 0\right)$ indicates a decrease.

3.1. Gas Dispersion

In order to analyze the gas dispersion measured by the quadcopter, a statistic method has been used to generate a two-dimensional gas distribution map, using the DM + V kernel algorithm presented in [38,39]. This algorithm treats the gas distribution model as a density estimation problem which can be solved using convolution with a two-dimensional Gaussian kernel. The kernel's shape regulates the amount of extrapolation. When the wind is not blowing, the kernel's shape is a circle, thereby: $\sigma_x = \sigma_y = \sigma_0$.

The vector of gas dispersion with spatial extrapolation was analyzed using the Gaussian weighting function (N) which represented the importance of the gas reading value obtained for each cell. The first step of the algorithm is the weight calculus $f_i^{(k)}(\sigma_0)$, which, intuitively represents the information content of a single measurement, i, of the sensor inside a net's cell. The weight is calculated by the mean of a Gaussian kernel (N) evaluation applied to the distance between measurement location $x^{(i)}$ and the center point $x^{(k)}$ of the k cell.

$$w_i^{(k)}(\sigma_0) = N\left(\left|x^{(k)} - x^{(i)}\right|, \sigma_0\right)\tag{4}$$

Starting from equation (4), the following values are integrated and placed in a temporary grid map: weights $W^{(k)}(\sigma_0)$, weighted sensor readings $G^{(k)}(\sigma_0)$, and weighted variance contributions $V^{(k)}(\sigma_0)$, as follows:

$$W^{(k)}(\sigma_0) = \sum_{i=1}^{n} w_i{}^{(k)}(\sigma_0) \tag{5}$$

$$G^{(k)}(\sigma_0) = \sum_{i=1}^{n} w_i{}^{(k)}(\sigma_0) G_s^{(i)} \tag{6}$$

$$V^{(k)}(\sigma_0) = \sum_{i=1}^{n} w_i{}^{(k)}(\sigma_0) \tau^{(i)} \tag{7}$$

where $\tau^{(i)} = \left(\Delta G_i^{(k)}\right)^2$ is the variance contribution of reading i.

From integrated weight map $W^{(k)}(\sigma_0)$ a confidence map $\bar{x}^{(k)}(\sigma_0)$ can be obtained showing the degree of trust with which for one cell the readings is considered to be in the vicinity of the respective grid cell's center and is expressed as shown in the Equation (8).

$$w_i{}^{(k)}(\sigma_0) = N\left(\left|x^{(k)} - x^{(i)}\right|, \sigma_0\right) \tag{8}$$

In normal dispersion, the confidence value is within the interval (0–1) which can be affected by the trajectory of the quadcopter, the size of the grid cell, the width of the kernel (σ_0), and the scaling parameter (σ_r).

Normalizing the integrated weighted sensor readings $G^{(k)}(\sigma_0)$ with the integrated weights $W^{(k)}(\sigma_0)$, then applying the confidence value and adding with the best guess for the cells with a low confidence (i.e., for cells for which we do not have sufficient information from nearby readings, indicated by a low value of $\bar{x}^{(k)}$) results in the map estimation of the mean distribution $g^{(k)}(\sigma_0)$:

$$g^{(k)}(\sigma_0) = \bar{x}^{(k)} \frac{G^{(k)}(\sigma_0)}{W^{(k)}(\sigma_0)} + \left\{1 - \bar{x}^{(k)}\right\} G_0 \tag{9}$$

As the best guess of the mean concentration G_0 we use the average over all sensor readings.

In the same way, the corresponding variance map $v^{(k)}(\sigma_0)$ results from normalizing the weighted variance contributions $V^{(k)}(\sigma_0)$ with the integrated weights $W^{(k)}(\sigma_0)$ then multiplying with the confidence value and adding with a best estimate for the cells with a low confidence:

$$v^{(k)}(\sigma_0) = \bar{x}^{(k)} \frac{V^{(k)}(\sigma_0)}{W^{(k)}(\sigma_0)} + \left\{1 - \bar{x}^{(k)}\right\} v_{tot} \tag{10}$$

The estimate v_{tot} of the distribution variance in regions far from measurement points is computed as the average over all variance contributions.

The mean value of gas concentration from each cell was used to make a predictive model in ANOVA. The spatial structure of the dispersion variance provided information on the gas dispersion vector and on the highest gas concentration which surely is located near the source.

3.2. Correlation Between Wind and Gas Dispersion

The gas dispersion is in linear correlation with the wind dynamic movement vector. Knowledge about the wind vector is helpful in locating the gas source. The extrapolation of gas measurement using bivariate Gaussian kernel provides information about the wind vector [10,40]. Two possible models were considered, i.e., an idle state in which wind velocity is zero, and a windy state, correlated with wind direction. An example of a detailed model is presented in Figure 7.
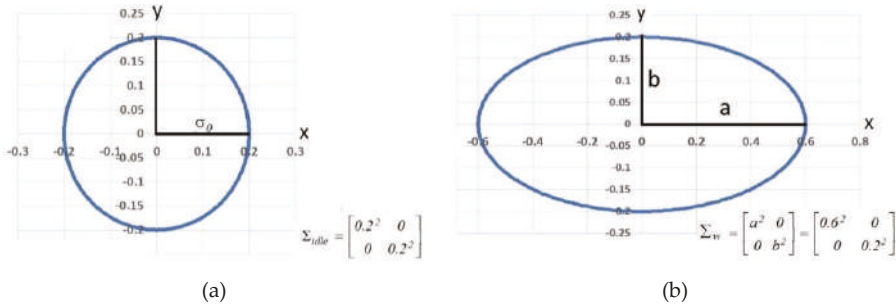
Figure 7. Wind direction model: (**a**) Idle state and (**b**) model considering wind velocity and direction.

The idle state with zero wind velocity is obtained from a normal dispersion having symmetrical kernel width (σ_0) along the *x*-, *y*-, and *z*-axis on grid cell; a diagonal matrix with variance data (Σ) represents this state as seen in Equation (11).

$$\Sigma_{idle} = \begin{bmatrix} \sigma_0^2 & 0 \\ 0 & \sigma_0^2 \end{bmatrix} \tag{11}$$

Wind velocity creates gas dispersion in the form of an ellipse with linear dependency. The wind vector changes the ellipse position according to the amount of change in the rotation matrix $R(\alpha)$. Rotation matrix is an orthogonal matrix in which $R(\alpha)^{-1} = R(\alpha)^T$ rotate bivariate Gaussian kernels around the *x*- and *y*-axis. Angle alteration (α) in the horizontal position (*x*- and *y*-axis) determines a two-dimensional wind vector which can be calculated using Equation (12). Data was enough to determine the pollutant source using the grid cell.

$$\sum_R = \begin{bmatrix} \sigma_x^2 & \alpha\sigma_x\sigma_y \\ \alpha\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix} = R(\alpha)\Sigma_w R(\alpha)^T \tag{12}$$

$$R(\alpha) = \begin{bmatrix} \cos\alpha & (-\sin\alpha) \\ \sin\alpha & \cos\alpha \end{bmatrix} \tag{13}$$

$$\sum_w = \begin{bmatrix} a^2 & 0 \\ 0 & b^2 \end{bmatrix} = \begin{bmatrix} \left(\sigma_0 + \gamma|\vec{v}|\right)^2 & 0 \\ 0 & \frac{\sigma_0}{\sqrt{1+\frac{\gamma|\vec{v}|}{\sigma_0}}}^2 \end{bmatrix} \tag{14}$$

The gas dispersion along with the wind velocity and vector $\gamma(\vec{v})$ were obtained from the wind sensor's measurements located on the gas dispersion contour; *x*-axis values were proportional to the wind velocity, while *y*-axis values decreased with the wind velocity. The variable γ is the stretching parameter which depends on many environmental variables. The bivariate Gaussian kernel was rotated according to the wind vector.

4. Environmental Monitoring

The experiment for measuring carbon concentration on each sample point was done in an open field for both the point source and the non-point source case. The pollutant source consisted of burning coals placed in a burner with a height of 0.5 m. The 24 grid cells covered an area of 6 m (width) × 4 m (length). Each cell was 1 m^2 in size, and the distance between measurement points was also 1 m. This distance depends on the sensor sensitivity; the more sensitive the sensor is, the larger the grid cell size may be, and a wider aerial zone requires a bigger gas source. Gas concentration measurement was done with the quadcopter in the hover state at the center of each cell during the 10 s sampling time.

Each measurement consisted of 50 readings, used to calculate the mean value for that measurement. The setup of the experiment is shown in Figure 8.

Figure 8. Sniffer quadcopter field test.

Data analysis of the gas dispersion and the point source was done manually by moving the quadcopter in a pattern using a remote control (RC), at a fixed altitude of 1 m from the ground. When the quadcopter reached the center of the cell, it remained in hover mode, and an interrupt control system was sent through the RC to collect data. Continuous flight patterns were completed in the order of the data acquisition sequence shown in Figure 6. This driving method has been used for the aim of manual correction during the field test and also to minimize errors; thus, a more accurate gas dispersion post-processing algorithm has been achieved.

The test for sensor positioning was done based on the CFD result, as presented in Figure 4. With the gas sensor mounted on the middle-bottom frame, unstable data reading and sometimes even zero value readings resulted. On the other hand, by having the gas sensor placed on the front of the quadcopter, the airflow generated by the propeller always passed the gas beyond the sensor. In hover mode, the quadcopter always adjusts its position to a stable state and produces equal air velocity on each of the rotors. Each quadcopter's propeller rotation generates a vortex that draws the air from above and directs it downwards, to the CO sensors. When sensors were placed in the front of the quadcopter, the gas sensor responded well. The recorded data was analyzed for two flight patterns, as follows:

4.1. Point Source

In the first stage of the experiment, the sniffer quadcopter was used to determine the level of CO concentration in the surrounding atmosphere. The quadcopter flew in a square or circle pattern to read the CO concentration in the center of each of the eight sample points. As much as 8 sample points were measured during the test, and the total flight time of quadcopter was 130 s without landing and take-off time. Every time a position error occurred, an adjustment was made using Equations (4) and (8) under the condition that the point sample for one grid cell must be greater than one sample; thus, the variant and average value near each cell's center can be obtained. Figure 9 shows the value of CO concentration measured for each cell. All data were distributed normally and the hypothesis from Equation (4) was applied to determine the vector of gas dispersion.
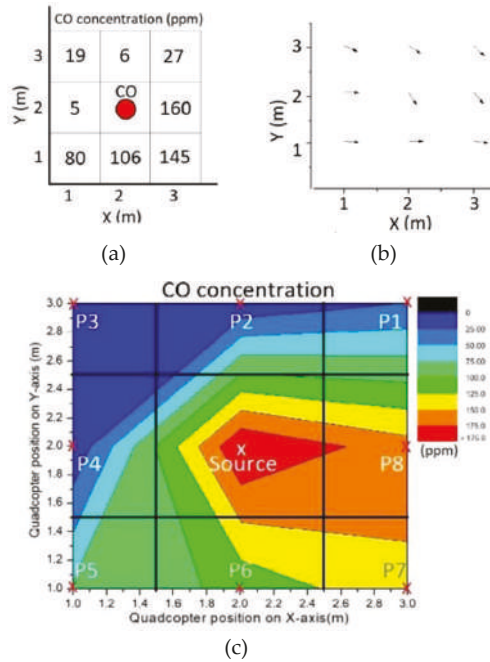
Figure 9. Point source experiment results: (**a**) CO concentrations; (**b**) CO vectors; (**c**) Gas dispersion.

The analysis result for the gas dispersion vector with respect to the gas source position $\{(X, Y) = 2\}$ showed that the vector was $\{(X, Y) > 2\}$. The carbon reading was modeled using a 2D contour in order to see the vector more clearly. The CO reading correlated with location was formulated in a matrix form shown in Equation (15).

$$
G_i^k = \begin{bmatrix} 19 & 6 & 27 \\ 5 & (>160) & 160 \\ 80 & 106 & 145 \end{bmatrix}
\tag{15}
$$

The gas dispersion contour along the x and y axes showed that the dispersion started from the highest to the lowest concentration, more precisely from the cell's center to the P1 point. In the point source method, the wind model of gas dispersion could not be seen clearly.

ANOVA analysis was used to compare cells two by two in all possible combinations, to get the estimation of CO concentration in the source. The analysis yielded the mean value $(\Delta G = 6.89)$, the standard deviation $(\sigma_0 = 0.36)$. After that, based on the gas dispersion vector, the irrelevant values were eliminated. To calculate the gas source concentration, the values from the first column $(X = 1)$ and the first row $(Y = 1)$ were eliminated. The sample points (P6, P7, P8) were used in the calculation since they were in accordance with the vector. The gas concentrations in each cell were summed up with the standard deviation value, and the results were averaged to get the CO source value, found to be 200.29 ppm. In comparison with real measurements of the gas source with sensors only, as much as 8.85% error was detected for CO concentration.

4.2. Non-Point Source

The non-point source experiment was done by collecting 24 sample points of CO concentration during 360 s of flight time. Data acquisition was done in the same manner used for the point source method. Two types of MOX sensors were used, i.e., MQ-135 and MQ-2 which were placed together on the quadcopter's front frame. Both sensors were used simultaneously to ascertain the validity of the

data and to get better analysis over the gas dispersion. In the field test, some burning coals were used as the CO source, placed on the position (X, Y) = (5, 2.5). Data of CO concentration were saved in the matrix form as presented in Equations (16) and (17). The CO concentration of the source was predicted, and the location of the source was analyzed through gas dispersion using the variant data of each cell.

$$S_{MQ135} = \begin{bmatrix} 19 & 19 & 19 & 35 & 82 & 40 \\ 24 & 44 & 44 & 67 & 110 & 28 \\ 49 & 18 & 16 & 18 & 33 & 28 \\ 16 & 17 & 25 & 19 & 20 & 23 \end{bmatrix} \tag{16}$$

$$S_{MQ2} = \begin{bmatrix} 37 & 30 & 27 & 39 & 65 & 45 \\ 34 & 67 & 71 & 70 & 98 & 34 \\ 54 & 33 & 29 & 42 & 38 & 18 \\ 29 & 25 & 28 & 13 & 24 & 34 \end{bmatrix} \tag{17}$$

All 24 sample points in the matrix were computed, yielding a 2D gas concentration contour for each of the sensors, presented in Figure 10.
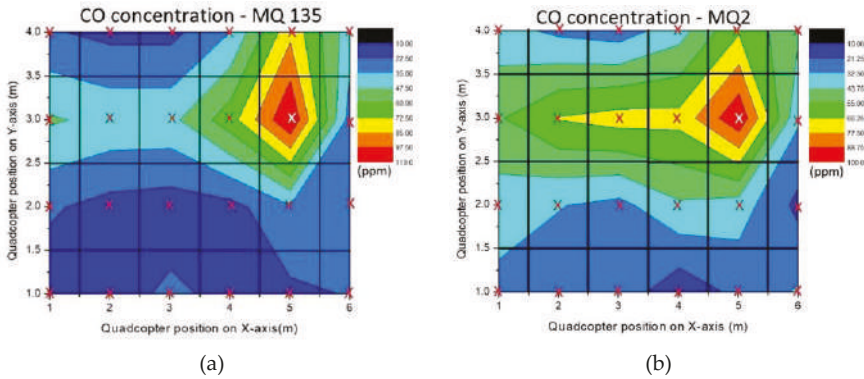
Figure 10. Non-point source experiment results in a 2D contour: (**a**) for MQ135 sensor; (**b**) for MQ2 sensor.

Based on the highest value of CO, the gas tended to drift toward the S13 cell or ($X < 5$) in the horizontal direction and toward the S20 cell or ($Y > 3$) in the vertical direction. Tentatively, it can be concluded that the gas source was around the red zone on the 2D contour. Further analysis using Equation (10) was done to discover the connection between gas dispersion and wind effect. A gentle breeze of 5 m s^{-1} blew around the field test area. Variant data of each sample was distributed normally to obtain standard deviation in form of kernel width for wind speed. Uncertainty of wind was conditioned as non-constant wind flow and represented by the (γ) parameter. Its value was estimated at 0.4 m based on the highest concentration stretch point and affected the stretch kernel shape. The calculation resulted in $\sigma_o(MQ135) = 0.247$ and $\sigma_o(MQ2) = 0.236$ with rotation vector $R_1(\alpha) = 0$ and $R_2(\alpha) = \frac{\pi}{2}$, thus a wind vector with range value a = 2.24–2.25 m and b = 0.08 m was obtained. From the analysis of wind direction and gas dispersion, two possible positions for CO source resulted, which were at the coordinates ($5 \leq x \leq 6.12$), ($2.92 \leq y \leq 3.08$) and ($4.92 \leq x \leq 5.0$), ($1.88 \leq y \leq 3$). Compared with the real position of the gas sensor (X, Y) = ($5, 2.5$), it can be concluded that only one out of the two possibilities is the source of pollutant area; bivariate Gaussian kernel analysis was used to assess the source location and minimize the error of CO reading.

5. Conclusions and Future Work

Air monitoring using a sniffer quadcopter with a flight pattern was designed to measure CO concentration in each cell. The post-processing analysis was used to determine the source location and its CO concentration. The results from CFD and the field test showed that the sensor placed in front of the frame $(x > 100)$ mm of the quadcopter was able to utilize the air trajectories generated by the propeller to direct the gas straight to the sensor. The best result was achieved when the data acquisition was made in hover mode to ensure constant airflow. Data acquisition was made using two methods, i.e., point source and non-point source. The point source method with a known location of the gas source was done using eight sample points forming a square pattern, having the source in the middle, which also is useful in facing the unpredictable wind effect.

The concentration of CO in the source was quantified using post-analysis by the means of the ANOVA method which was ran on eight samples during 130 s of flight time. Compared to the real data, the analysis showed as much as 8.85% error.

The other method was the non-point source, used to pinpoint the location of the gas source and also its concentration. This method adapts a grid pattern with 24 cells to collect data of CO with two types of gas sensors used simultaneously to ascertain data validity. The gas dispersion analysis results showed that the gas dispersion vector had changed twice, thus indicating two possible positions for gas source location. The gas dispersion vector has been analyzed using both the measurement position and CO concentration matrices. The readings of both sensors showed the same gas dispersion pattern, indicating the highest value of CO was in the S17 cell. The differences in the accuracy of data reading were affected only by the sensitivity of the MOX sensors. The correlation between gas dispersion and wind behavior must be known to overcome the possibility of result misinterpretation due to wind influence. Logically, the gas source must be in the vicinity of the cell having the highest CO concentration. A bivariate Gaussian kernel has been used to locate this cell's position. The gas source was calculated with the same method used in the case of the point source; thus, the weight cell in the form of standard deviation was obtained with the value of CO between 118.06–133.24 ppm while the actual value was 125 ppm. Overall, the field tests were done by manipulating the gas source; the quadcopter's altitude maintained at 1 m from the ground to collect data which then were calculated accordingly to acquire the gas source location. For the case of a small amount of burning coal, the experiment was possible only at low altitude with manual control of the quadcopter flight. The sensitivity of the sensors must pass the reliability test before being placed on the quadcopter because it affects the size of the cells. Finally, the analyses using normal dispersion and ANOVA were essential to obtain the gas concentration and gas source position.

This study using sniffer quadcopter has been limited to carbon monoxide measurements. The measurement method and gas source location detection method still have room for improvement. In the future, other pollutant compounds will be investigated and different gas sensors such as optical sensors will be used in comparison with the tested sensors. There are various application of mapping and measurement using a sniffer quadcopter, such as gas pipe leakage measurement, early warning systems for volcanic-prone areas, water pollution mapping with the pattern method, etc. From the perspective of flight pattern, with improvements in sensitivity and accuracy of reading, a larger scale grid pattern can be designed to save time on the data collection process. Quadcopter capability to withstand wind effects or heatwaves from the gas source can also be more developed. The higher goals are to utilize sniffer quadcopter as unmanned aerial security patrols to cope with environmental issues and monitor dangerous zones. In future experiments, we intend to test the data collection from the sensor at larger scales in quasi-real situations and also to use multiple drones which can communicate with each other and better map the field of interest.

References

1. World Health Organization. *Risk Assessment of Selected Pollutants, Air Quality Guidelines: Global Update 2005*; World Health Organization: Geneva, Switzerland, 2006; pp. 217–415.
2. European Economic Area. *Air Quality in Europe*; European Environment Agency: Copenhagen, Danmark, 2017; Volume 13, pp. 1–74.
3. Fenger, J. Urban Air Quality. *Atmos. Environ.* **1999**, *33*, 4877–4900. [CrossRef]
4. Kanaroglou, P.S. Establishing an air pollution monitoring network for intra-urban population exposure assessment: A location-allocation approach. *Atmos. Environ.* **2005**, *39*, 2399–2409. [CrossRef]
5. Ras, M.R.; Marce, R.M.; Borrull, F. Volatile organic compounds in air at urban and industrial areas in the Tarragona region by thermal desorption and gas chromatography-mass spectrometry. *Environ. Monit. Assess. J.* **2010**, *161*, 389–402. [CrossRef] [PubMed]
6. Chow, J.C. Measurement methods to determine compliance with ambient air quality standards for suspended particles. *J. Air Waste Manag. Assoc.* **1995**, *45*, 320–382. [CrossRef] [PubMed]
7. Snyder, E.G. The changing paradigm of air pollution monitoring. *Environ. Sci. Technol.* **2013**, *47*, 11369–11377. [CrossRef] [PubMed]
8. Dunbabin, M.; Marques, L. Robots for environmental monitoring: Significant advancements and applications. *IEEE Robot. Autom. Mag.* **2012**, *19*, 24–39. [CrossRef]
9. Zhang, J. Seeing the forest from drones: Testing the potential of lightweight drones as a tool for long-term forest monitoring. *Biol. Conserv. J.* **2016**, *198*, 60–69. [CrossRef]
10. Neumann, P.P.; Asadi, S.; Lilienthal, A.J. Autonomous gas-sensitive microdrone: Wind vector estimation and gas dispersion mapping. *IEEE Robot. Autom. Mag.* **2012**, *19*, 50–61. [CrossRef]
11. Juan, R. A mini-UAV based sensory system for measuring environmental variables in greenhouses. *Sens. J.* **2015**, *15*, 3334–3350.
12. Buters, T.M.; Bateman, P.W.; Robinson, T.; Belton, D.; Dixon, K.W.; Cross, A.T. Methodological Ambiguity and Inconsistency Constrain Unmanned Aerial Vehicles as A Silver Bullet for Monitoring Ecological Restoration. *Remote. Sens.* **2019**, *11*, 1180. [CrossRef]
13. Bayat, B.; Crasta, N.; Crespi, A.; Pascoal, A.M.; Ijspeert, A. Environmental monitoring using autonomous vehicles: A survey of recent searching techniques. *Curr. Opin. Biotechnol.* **2017**, *45*, 76–84. [CrossRef] [PubMed]
14. Anweiler, S.; Piwowarski, D. Multicopter platform prototype for environmental monitoring. *J. Clean. Prod.* **2017**, *155*, 204–211. [CrossRef]
15. Lally, H.T.; O'Connor, I.; Jensen, O.P.; Graham, C.T. Can drones be used to conduct water sampling in aquatic environments? A review. *Sci. Total Environ.* **2019**, *670*, 569–575. [CrossRef] [PubMed]
16. Capolupo, A.; Pindozzi, S.; Okello, C.; Fiorentino, N.; Boccia, L. Photogrammetry for environmental monitoring: The use of drones and hydrological models for detection of soil contaminated by copper. *Sci. Total Environ.* **2015**, *514*, 298–306. [CrossRef] [PubMed]

17. Estrada, M.A.R.; Ndoma, A. The uses of unmanned aerial vehicles—UAV's (or drones) in social logistic: Natural disasters response and humanitarian relief aid, ICTE in Transportation and Logistics 2018 (ICTE 2018). *Procedia Comput. Sci.* **2019**, *149*, 375–383. [CrossRef]

18. Dunnington, L.; Nakagawa, M. Fast and safe gas detection from underground coal fire by drone fly over. *Environ. Pollut.* **2017**, *29*, 139–145. [CrossRef] [PubMed]

19. Villa, T.F. Development and validation of a UAV based system for air pollution measurements. *Sens. J.* **2016**, *16*, 2202. [CrossRef] [PubMed]

20. Manuel, A.; Michel, G. Review of small commercial sensors for indicative monitoring of ambient. *Chem. Eng. Trans. J.* **2012**, *30*, 169–174.

21. Kristiansen, R.; Oland, E.; Narayanachar, D. Operational concepts in UAV formation monitoring of industrial emissions. In Proceedings of the IEEE 3rd International Conference on Cognitive Infocommunications, Kosice, Slovakia, 2–5 December 2012; pp. 339–344.

22. Ambrosia, V.G.; Wegener, S.S.; Sullivan, D.V.; Buechel, S.W.; Dunagan, S.E.; Brass, J.A.; Stoneburner, J.; Schoenung, S.M. Demonstrating UAV-Acquired Real-Time Thermal Data over Fires. *Photogramm. Eng. Remote Sens.* **2003**, *69*, 391–402. [CrossRef]

23. Zhi, Z. Data-driven hazardous gas dispersion modeling using the integrating of particle filtering and error propagation detection. *Int. J. Environ. Res. Public Health* **2018**, *15*, 1640. [CrossRef]

24. Zhao, Y. Gaussian processes for flow modeling and prediction of positioned trajectories evaluated with sports data. In Proceedings of the 19th International Conference on Information Fusion, Heidelberg, Germany, 5–8 July 2016; pp. 1461–1468.

25. Reggente, M.; Lilienthal, A.J. Statistical evaluation of the kernel DM+V/W algorithm for building gas distribution maps in uncontrolled environments. *Procedia Chem.* **2009**, *1*, 481–484. [CrossRef]

26. Brock, O.; Trinkle, J.; Ramos, F. Gas distribution modeling using sparse Gaussian process mixture models. *Robot. Sci. Syst.* **2009**, *4*, 336–344.

27. Ostrowski, W.; Hanus, K. Budget UAV systems for the prospection of small and medium scale archaeological sites. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *41*, 971–976. [CrossRef]

28. Gandor, F.; Rehak, M.; Skaloud, J. Photogrammetric mission planner for RPAS. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2015**, *40*, 61–65. [CrossRef]

29. Yıldırım, Ş.; Çabuk, N.; Bakırcıoğlu, V. Design and Trajectory Control of Universal Drone System. *Measurement* **2019**. [CrossRef]

30. Murua, J.; Palacios, R.; Graham, J.M.R. Applications of the unsteady vortex-lattice method in aircraft aeroelasticity and flight dynamics. *Prog. Aerosp. Sci.* **2012**, *55*, 46–72. [CrossRef]

31. Colmenares, J.D.; Lopez, O.D.; Preidikman, S. Computational study of a transverse rotor aircraft in hover using the unsteady vortex lattice method. *Math. Probl. Eng.* **2015**, *2015*, 1–9. [CrossRef]

32. Kuantama, E. Quadcopter propeller design and performance analysis. In *New Advances in Mechanisms, Mechanical Transmissions and Robotics. Mechanisms and Machine Science*; Springer: Cham, Switzerland, 2016; Volume 46, pp. 269–277.

33. Kuantama, E. Flight Stability Analysis of a Symmetrically Structured Quadcopter Based on Thrust Data Logger Information. *Symmetry* **2018**, *10*, 291. [CrossRef]

34. Neumann, P.P. Monitoring of CCS areas using Micro Unmanned Aerial (MUAVs). *Energy Procedia-Elsevier J.* **2013**, *37*, 4182–4190. [CrossRef]

35. Alvarado, M. Towards the development of a low-cost airborne sensing system to monitor dust particles after blasting at open-pit mine sites. *Sens. J.* **2015**, *155*, 19667–19687. [CrossRef]

36. Lilienthal, A.J.; Cuckett, T. Building gas concentration grid maps with a mobile robot. *Robot. Auton. Syst.* **2004**, *48*, 3–16. [CrossRef]

37. Li, J. Odour source localization using a mobile robot in outdoor airflow environments with a particle filter algorithm. *Auton. Robot.* **2011**, *30*, 281–292. [CrossRef]

38. Lilienthal, A.J.; Reggente, M.; Trincavelli, M. A statistical approach to gas dispersion modelling with mobile robots—The Kernel DM+V Algorithm. In Proceedings of the International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 570–576.

39. Neumann, P. An artificial potential field-based sampling strategy for a gas sensitive micro-drone. In Proceedings of the IROS-Workshop Robotics for Environmental Monitoring, San Francisco, CA, USA, 25–30 September 2011.
40. Reggente, M.; Lilienthal, A.J. The 3D-Kernel DM+V/W algorithm: Using wind information in three-dimensional gas dispersion modelling with a mobile robot. In Proceedings of the IEEE Sensors, Kona, HI, USA, 1–4 November 2010; pp. 999–1004.

Article

Novel PDMS-Based Sensor System for MPWM Measurements of Picoliter Volumes in Microfluidic Devices

Mihăiţă Nicolae Ardeleanu [1,2], Ileana Nicoleta Popescu [1,*], Iulian Nicolae Udroiu [3], Emil Mihai Diaconu [3], Simona Mihai [4], Emil Lungu [5], Badriyah Alhalaili [6] and Ruxandra Vidu [7,8,*]

1   Faculty of Materials Engineering and Mechanics, Valahia University of Targoviste, 13 Aleea Sinaia Street, Targoviste, 130004 Romania; mihai.ardeleanu@valahia.ro

2   S.C. Celteh Mezotronic S.R.L., Calea Câmpulung Street, No. 6A, Targoviste, 130092, Romania

3   Faculty of Electrical Engineering, Electronics and Information Technology, Valahia University of Targoviste, Targoviste, 130004, Romania; iulian.udroiu@valahia.ro (I.N.U.); emil.diaconu@valahia.ro (E.M.D.)

4   The Scientific and Technological Multidisciplinary Research Institute (ICSTM-UVT), Valahia University of Targoviste, Targoviste, 130004, Romania; simona.mihai@valahia.ro

5   Faculty of Sciences and Arts, Department of Mathematics, Valahia University of Targoviste, Targoviste, 130004, Romania; emil.lungu@valahia.ro

6   Nanotechnology and Advanced Materials Program, Kuwait Institute for Scientific Research, P.O. Box 24885, Safat 13109, Kuwait; eng.kisr@hotmail.com

7   Department of Electrical and Computer Engineering, University of California Davis, Davis, CA 95616 USA

8   Faculty of Materials Science and Engineering, University Politehnica of Bucharest, Bucharest 060042, Romania

*   Correspondence: ileana.nicoleta.popescu@valahia.ro or pinicoleta24@yahoo.com (I.N.P.); rvidu@ucdavis.edu (R.V.)

Abstract: In order for automatic microinjection to serve biomedical and genetic research, we have designed and manufactured a PDMS-based sensor with a circular section channel using the microwire molding technique. For the very precise control of microfluidic transport, we developed a microfluidic pulse width modulation system (MPWM) for automatic microinjections at a picoliter level. By adding a computer-aided detection and tracking of fluid-specific elements in the microfluidic circuit, the PDMS microchannel sensor became the basic element in the automatic control of the microinjection sensor. With the PDMS microinjection sensor, we precise measured microfluidic volumes under visual detection, assisted by very precise computer equipment (with precision below 1 μm) based on image processing. The calibration of the MPWM system was performed to increase the reproducibility of the results and to detect and measure microfluidic volumes. The novel PDMS-based sensor system for MPWM measurements of microfluidic volumes contributes to the advancement of intelligent control methods and techniques, which could lead to new developments in the design, control, and in applications of real-time intelligent sensor system control.

Keywords: microinjection sensor; PDMS capillary; micro-wire; picolitre volume measurement; MPWM microfluidic transport

1. Introduction

The increasing demands for a complete and real-time diagnosis of a larger population and the improvement of the quality of life in general has led scientists from all over the world to study and develop multidisciplinary science and technologies [1–6]. Among them, microfluidics are based on

life sciences and medical technologies of the future, which will revolutionize biology and medical diagnostics [4,5] through manipulation of very small volumes of fluids (from nano to pico and femtoliters) [1,6] and miniaturized devices [7].

To integrate one or more laboratory analyses and synthesis on a single chip of just millimeters in size, researchers and specialists have used a miniaturized device called lab-on-a-chip (LOC), which allows a quick response and diagnosis at low energy consumption and cost, using a very small volume of samples [3,8–10]. LOC devices, including microfluidic ones, have many advantages such as reducing the use of chemicals (reduced sample and reagent usage) that can be rare, expensive and polluting, reducing waste production, miniaturization, integration, portability, and automation [4,7,8,11]. In addition, the LOC offers a precise analysis of a large numbers of individual molecules and cells, flexibility of device design, high experimental control, and reduces measurement times [4,5,12]. Another advantage of the microfluidic cell culture device is the ability to incorporate analytical biosensors into the culture platform, thus combining, in a non-invasive way, living cells and sensors for detecting cellular physiological parameters and analyzing external stimuli in situ [12,13].

Nowadays, microfluidic devices can be used for biological and medical analysis, detection, control and manipulation of biological samples and cell biology research such as: analysis of the unpurified blood samples, analysis of complex mixtures and molecules (especial DNA and proteins), DNA sequencing, single cell manipulation, electrophoretic separations, drug screening, screens for protein crystallization conditions, cell culture studies and reproductive cell selection [2,9,12,14–18]. Also, with the help of microdevices, we can carry out environmental analysis, food analysis and control, detection and screening of residues and explosives [19,20]. Microfluidic chips provide unrivaled control over droplets and jets, which have advanced all natural sciences [21]. The development of new types of bioassay for monitoring patient response to therapy is another application [7]. LOC is a complex microsystem that includes a network of mechanical, electronic and fluid functions microchannels. These microchannels serve as pipes, valves, sensors, electronics and other structures [7,8]. LOC, bio-microelectromechanical systems (bioMEMS) and micro-total-analysis-systems (μTAS) are technologies that include both microfluidics and detection capabilities [10,22]. In microfluidic devices, the chemical and physical microenvironment can be easily controlled by using on-chip valves that allow the release of fluids containing target molecules and substances with precise timing [23]. The droplet-based microfluidic LOC platform has significant advantages for high-throughput, continuous flow and ultra-low volume studies of biological and chemical experiments [10].

For microfluidic design and fabrication it is very important to select the appropriate material that will be used and also the size and the geometry (i.e., cross-section) of the channel, and the surface characteristics of the microchannel to be made in accordance with the specific applications of the chip must be taken into account and they are briefly reviewed in the following section.

1.1. Microfluidic Device Materials

Microfluidic devices can be fabricated using several materials: (a) inorganic materials such as: silicon and glass; (b) polymeric materials such as polymethylmethacrylate (PMMA), polydimethylsiloxane (PDMS) and a relative novel one, the hydrogel; and (c) Paper-based for microfluidic chips [22,24,25]. Glass is transparent, but because it is amorphous, vertical side are more difficult to etch than Si. The production of microfluidic devices made of glass or silicon is generally expensive and time-consuming [2]. Paper is a very inexpensive material and easy to work with. It is compatible with biological samples and can be chemically treated to bind to molecules or protein, but the main disadvantage when using paper–based microfluidic devices is the difficulty in detecting channels on the chip [22]. The hydrogel is a colloid made of polymeric chains of molecules spread in water. Hydrogel is very malleable, and various feature designs and sizes can be molded onto it. Furthermore, the hydrogel is commercially available, non-toxic to the cells and cost effective. One common polymer used to make hydrogel is sodium polyacrylate. Their application for microfluidic devices depends on their suitability for biological experiments. For instance, hydrogel could serve as a matrix material of

new classes of solar cells and photoreactors with embedded microfluidic networks and also it is used for tissue engineering due to its high permeability and biocompatibility [25]. The most used material (both for academic and industrial applications) for prototyping and fabrication LOC device is PDMS, a mineral-organic polymer in the siloxane family. PDMS is an excellent material for the fabrication of microchannel systems/master mold because of its specific characteristics that will be discussed in detail in the next section, selection of materials for experiments [12,16]. The PDMS has some drawbacks, such as diffusivity and swelling in organic solvents [26]. The chemical structure of PDMS consist in repeating units of -OSi(CH$_3$)$_2$- groups, which leads to a hydrophobic surface ($\theta_a^{H_2O} = 108°$) [27], which is another major disadvantage of this material. There are some solutions to solve this problem, e.g., the hydrophobicity can be solved by changing the surface properties of PDMS using specific surface treatments or modification [26].

1.2. Techniques for Manufacturing Microfluidic Devices

Generally, microfluidic devices consist of different components, such as reservoirs, chambers, and microchannels [22]. Microchannels with varied geometries and surface properties were produced and studied over time by many researchers. There are many techniques for making microfluidic systems, all of which are consistent with their use in different fields of application [2,11,28–37]. The methods for fabrication of microfluidic devices include prototyping techniques, such as: hot embossing [29,38,39], injection molding [40] soft lithography (photolithography followed by etching and bonding) [2] rapid prototyping [2] and replica molding [16,41]. Other methods used are direct fabrication techniques like laser photoablation or laser micromachining [42] photolithography/ optical lithography [43] or photolithography followed by etching and bonding) [44] and x-ray lithography [37, 45]. These techniques have advantages and disadvantages also that may or may not be used effectively in the manufacture of microchip devices. For instance, according with Qi et al. [39], the hot embossing technique has the following advantages: cost-effective, accurate and fast replication of microstructures and mass production. This technique has some disadvantages such as: (i) the restriction of applying only to thermoplastics material and (ii) the difficulty to fabricate complex 3D structures. With the micro-injection molding method [40], it is easy to manufacture complex polymer geometry, fine features and 3D geometries. This technique allows mass production, is highly automated and has a short cycle time. But has also disadvantages, i.e., the difficulty of forming large undercut geometries, high cost mold, and is applied only to thermoplastic materials, as in the case of the hot embossing technique. Another type of prototyping technique capable of fabricating 3D geometries is soft lithography, which is a cost-effective method and high resolution features (down to a few nm) can be obtained. One of the major disadvantages is the vulnerability to defects [2]. Rapid prototyping is capable of producing prototypes of almost any geometrical complexity in relatively short time [41]. The advantages of this technique compared to the methods that used the chromed mask in the photolithographic step are the reduction of time (i.e., hours instead of weeks) and reduction of costs (20–100 times more expensive than the chromatic mask that replaces the transparency. The disadvantage is the lower resolution of transparency (>20 μm) compared to the chromed mask (approx. 500 nm). The low transparency resolution leads to two walls with rough edges [2].

Replica molding has the role to form the microchannels in PDMS by generating a negative replica of a master (made by metal/hard material) in PDMS through casting of pre-polymer against the master [2]. Despite rapid and large format production, laser photo-ablation has multiple treatment sessions, and is applied for a limited number of materials [42].

Ideal for microscale features is the conventional photolithography/optical lithography but usually it requires a flat surface to start with and chemical post-treatment steps [43]. Finally, x-ray lithography allows high resolution for nano-patterns fabrication, absorption without artifacts and is capable of producing straight, smooth, walls, but imposes difficulties in the manufacturing process, consuming time and high costs [45].

1.2.1. Geometry and Quality of Microchannels Surface

From the point of view of the geometry of the channels, the soft lithography technique can only make rectangular channels, which is a disadvantage in replicating the circular cross-sections of blood vessels, as well as for replicating cardiovascular flow conditions [30]. Rectangular microchannels limit cell growth on the bottom of the channel, exposing the cells to a non-physiological geometry [44]. Cells growing on the bottom of the channel have different shear stresses, which directly influence the alignment, elongation, differentiation and expression of the genes [44]. Microfabrication by photolithography method [32] is manufactured in the form of rectangular sections of channels. The rectangular channel has several disadvantages to generate drops [44,46], the fluid phase wets the upper and lower walls of the rectangular channel at a low flow rate of fluid in a continuous phase, which causes the attachments of cells to the channel walls, or even destroyed due to them: (i) surface shear force [36] and also due to (ii) capillary instability [10,47]. By increasing the flow of a continuous phase, the wetting surface of the disperse phase might be eliminated [47].

Using channels of circular section (or cylindrical geometry) instead of rectangular ones, the velocity profile can be generated evenly in the direction of the cross section, formation of stagnation areas at the corners is eliminated and a stable equilibrium position is established in the center of the channel, thus generating uniform, well controllable and monodisperse drops at a wide range of flow rates [44,47–49]. Another advantage of circular microchannels is that it can increase the efficiency of light transmission inside on-chip waveguides for light-sensing and light actuation methods implemented in lab-on-a-chip devices [44,50].

The surface chemistry of the microchannels is another important aspect to consider when making microchips. For example, due to its high hydrophobicity, PDMS absorbs certain organic solvents and hydrophobic analytes, causing contamination (fouling) of the inner surface channels [26]. Hydrophobic materials are difficult to fill (wet) with aqueous solutions and easily nucleate air bubbles (which makes it difficult to remove air bubbles from the channels), and consequently affect the quality of the material [2]. For this reason, to remove hydrophobicity it is necessary for PDMS to be treated with oxygen plasma to create hydrophilic PDMS surfaces by oxidation. After oxygen plasma treatment, it is necessary to maintain surfaces in contact with water or polar organic solvents [26] so as to avoid contact with the air. The best solution for surface modification is the treatment of surfaces with silanes or polyelectrolyte multilayers [27]. However, the direct contact with air results in surface rearrangements and the PDMS surface becomes hydrophobic in about 30 min [26,27].

1.2.2. The Microwire-Molding Technique

Recognizing the benefits of circular microchannels, several research groups have been motivated to search for new techniques for making circular channels [44]. In this context, to improve the manufacturing techniques and to solve some of the problems or disadvantages that arise in the fabrication of microchannels, especially in the circular channels, researches developed novel, simple and fast fabrication methods. Based on pouring of liquid PDMS pre-polymer on microwires, this method is called suggestive microwire-molding [51–55]. The advantages of the microwire molding process are: rapidly and simply fabricate the circular-channels with perfect circular cross-sections in bulk PDMS and also the possibility to make different and/or complex topological shapes from straight channels to helical or curving micro-channel fabrication [51–55].

These benefits result from two main properties of PDMS: low surface energy and reversible swelling by solvents [27,52–55]. Thus, Verma et al. [55] developed for the first time a simple and cost effectiveness methods for generating straight (1D), cross-connectors (2D) and more complex (3D orientation structure) microchannels inside cross-linked PDMS blocks. They used nylon wires of varying diameters (50–250 µm) as a template to generate channels and to ease the nylon removal. Chloroform and triethylamine solvents were used for swelling the cross-linked PDMS Sylgard 184 Type, (Dow Corning, Wiesbaden, Germany) [51–55]. For PDMS mixtures, pre-polymer were used: curing agent for 10:1 (w/w) [51–53,55] and 9:1 (v/v) for [54] (Table 1).

Table 1. Materials and process parameters used in the microwire molding process (PDMS, wires material and solvents type).

(Micro)wires Type	Microwires Diameters	Solvents and Proportions		Others Materials for Supporting Substrate	Ref.
Nylon	50–250 μm	Chloroform	100%	Rigid cylindrical rod of 100-500 μm Silicone oils for Channels Filling	[55]
		Chloroform + Triethylamine	70/30 (v/v)		
316 L Stainless steel	24 and 40 μm	Ethanol / Hexane	100%	PMMA ancillary supporting frame and, PTFE tubes GLASS slide as substrate	[53]
NiCr, Enameled Cu, Nylon	150, 350, 400 μm	Chloroform and Diisopropylamine	70/30 (v/v)	Deionized water, water and oil based colors, mineral (15 mPa.s) and silicone oils (10^3 mPa.s) for droplet formation	[52]
Stainless steel; Nylon	50, 80, 100 and 150 μm; 74, 105, 148 and 181 μm	Alcohol	100%	Particle suspensions added with 8 wt% PVP	[51]
Soldering (H-712 type) Sn and Pb: 60/40	300 μm	-		Slide glass for bonding PDMS. For coating the inner walls of channels PDMS: solutions diluted with serial % of Sylgard 184	[54]

In the same way, by embedding the microwires through molding process of PDMS mixture and then removal of microwires by swelling PDMS in different types of solvents, Jia et.al. [53], also made microchannels with different topological shapes (1D, 2D and 3D structures), in economic, flexible and convenient way in comparison with photolithography and conventional soft lithography [53]. To realize these channels, besides the casting and treatment parameters, of major importance are the preparations of microwires, substrates or supports (Table S1, Supplementary Materials).

For instance, at fabrication of straight channels [51–55], it is a very good stretching out the nylon/stainless steel/copper etc. microwires between rigid supports, also cleaning, drying of glass substrate/wires or degassing of PDMS mixture are required. For 2D micro-channel fabrication [53,55] it is necessary templates fabrication (2D mesh channels), formed by mechanical (hot) pressing/micro contacting and microspotting.

Thus more complex 3D structures [52–55], can be made by different techniques, such as (i) forming of a helix around a rigid cylindrical rod ($\Phi = 100$–500 μm), heated (100 °C) to fix the helical form permanently [55], casting the PDMS mixture on 2-D nylon mesh template and curing mixture at 90 °C for 1h, or (ii) rolling up by winding around a rigid rod, casting PDMS mixture at 100 °C for 35 min [52], followed by microwire removal in different solvents mixture [52,53,55], or removal the microwire by heating and suction with vacuum pump [54]. By the technique with heating and suction removing the wires pump [54], a roughness inner surface it is obtained. Due to possible remaining of solder material after the wire removal process, additional coating treatments is necessary (i.e., wall coating of channels using PDMS solutions diluted with curing agent, in different proportions) and also corona discharge treatment to increase the hydrophilicity (for improving the fluidic flow and cell adhesion); The final step, the sterilization device by filling with 70% ethanol and washing with deionized water, make the device suitable for biological application [54].

1.3. The Purpose Statements

For the general purpose of generating different experiments focused on the biological environment, we present in this paper the design and fabrication by microwire molding a PDMS-based sensor with circular cross-section microchannel with a good surface quality, for the measurement of microfluidic (nanoliter to picoliter) volumes. To control the microfluidic transport at the nano-, pico- and femto-liter level, we used Microfluidic Pulse Width Modulation (MPWM) [56] system. We designed and fabricated the novel sensor especially to calibrate a microinjection MPWM system that will be used for very high precision intra-cellular insertions. The calibration of MPWM system it is necessary for reproducing with high precision and the detection and measurements of microfluidic volumes.

2. Materials and Methods for PDMS –Based Sensor Fabrication

2.1. Selection of Materials

Based on literature data [2,12,26,27,38,44,57] we selected a polydimethylsiloxane (PDMS) polymer (Sylgard 184 type) for the fabrication of a circular-cross section channel microfluidic device due to its remarkable characteristics such as: (i) excellent optical transparency, down to 280 nm [57] (ii) flexibility (i.e., it is a ductile material); (iii) elasticity (its elasticity can be "tuned" using cross-linking agents); (iv) biocompatibility (v) capacity to seals materials like glass, polystyrene and PMMA [57]; (vi) high thermal stability up to T = 300 °C [38]; (vii) permeability to gases (is more permeable to CO_2 than to O_2 or N_2) [12] and also to water vapor; (viii) cost-effective production at microscale and (ix) does not require clean room environment [57]. To get a circular cross section channel we selected metallic microwires (enameled copper and steel) due the smooth surface and high flexibility of them. For creating different size of channels, we selected the following wire diameters: 63 μm and 120 μm for the enameled copper and 190 μm for the steel wire. The low surface energy and reversible swelling by solvents of the PDMS [26,27] allow the microwire removal with proper solvents. In this case, of our experiments, chloroform (with 1.39 swelling ratio) [27] were used for swelling the crosslinked PDMS. We made

ultrasonic cleaning of the mold and wires in isopropyl alcohol (commercially used for degreasing and cleaning electronic components). The ultrasonic cleaning was performed with a USC-TH type ultrasonic cleaner (VWR International, LLC) at 27 °C for 30 min.

2.2. Fabrication Methods of PDMS Sensor

For fabrication of PDMS based sensor it is necessary to generate straight microchannel inside PDMS block. The main steps include: (i) Preparation of mold support and microwires, (ii) Microwire fixing and alignment, (iii) PDMS Casting, Degassing and Curing, (iv) Demolding and Microwire removal, (v) Needles insertion and sealing by needles connection to the polyethylene tubes. Preparation of molds and devices before casting: The cylindrical polymer molds were punched diametral opus, with needles (300 μm in diameter). For fixing and aligning the wires there were custom made a custom stretching device with many working positions with screw (Figure 1) respectively with magnets for attachment of the wires were made. Every position is composed of two diametric disposed elements. The stretching was performed using a microscopic system by which a linear landmark was aligned with the stretched wire. There are two stretching devices: the first one was equipped with screws and locking nuts and the second one used permanent magnets for wires fixation. At the M4 × 1 screw, we used the nut & lock-nut system to lock the convenient position to ensure the linearity of the wire. The linearity was tested with a standard edge by parallelization with the stretched wire and microscopic observations. The permanent magnets have been chosen so that by application they will secure the firm lock of the pre-tensioned wire manually. The linearity was proved as in the case with a screw.
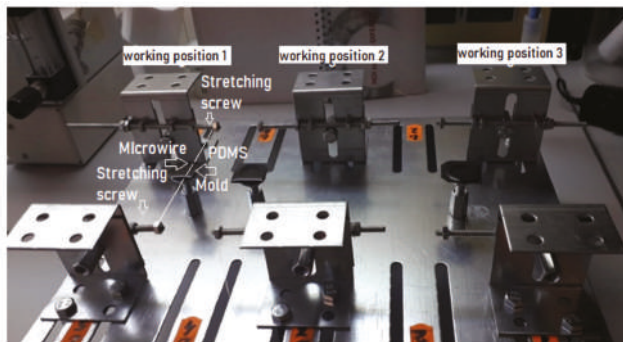
Figure 1. Customized stretching device with three working positions and the casted PDMS mixture inside of polymeric molds.

The degassed of PDMS mixture (removing of air bubbles) was made using a low level vacuum. After casting of mixture, the casted chip made of PDMS has a height of 2.5 mm, which allows a quick natural degassing, no more than 3 h. However, if a bubble remains stuck in the PDMS table, it is mechanically removed with a needle tip. All molded chips did not show any bubbles left after casting. The degassed PDMS mixture (components A to B = 10/1 w/w) was poured in the polymer molds (Figure 1). The PDMS mixture was cured at 80 °C for 2 h and then it was cooled slowly in a Venticell type heating oven (MMM Group, Planegg, Germany), with natural air circulation,. After degassing and curing, the demoulding step consist into gentle pulled out from the PDMS block of the microwires by swelling PDMS in chloroform, leading to the formation of microchannels in the PDMS block.

The characterization of obtained microchannels using microwire-molding technique were done from microstructural point of view using different types of microscopes: (i) 1000 × 8 LED Zoom Digital USB Handheld Microscope PC Endoscope Camera Test TE306 (AmScope, Irvine, CA USA) and (ii) 40× – 800× Inverted Tissue Culture Trinocular Microscope with different video camera (custom and AmScope FMA type, Irvine, CA USA).

3. Volumes Detection Method

3.1. The Principle of Volume Detection Using PDMS-Based Sensor

By applying a constant pressure in microchannel, the flow of the liquid is defined by the microchannel geometry due to the common laws of the laminar flow [58]. The general characteristics of the liquids (incompressibility, fill and copy the shape of the vessel) [1,2], allow that the volumetric measurement of a fluid flow along the channel (with constant circular section) to be reduced to the measurement of the distance traveled between two given impulses. In this experimental work, impulses are given by a system called microfluidic pulse width modulation (MPWM). Based on a concept from the electrical engineering field, the MPWM system controls the transport of fluids through the microfluidic channels. Ardeleanu et al. [56] used MPWM to measure volumes/flow rates for future application in cell/particle trapping and release. Ainla et al. [59] used pulse width modulation (PWM) for microfluidic diluter and Unger et al. used PWM for dynamic duty cycle over time for liquid flows [60]. The experiments are performed with very small volumes of liquid using microchannels obtained by microwire molding [51–55] using wires of 63, 120 and 190 μm in diameter.

For precise flow control, many methods including optical volumes and flow rates measurements were developed [56,58]. An optical volumetric measurement technique has been developed to enhance the measurement in the picoliters region. Usually, a microscope with 500x magnification is used when working with picoliter volumes. The visualization of the capillary requires a very good clarity to determine the distinctive front of the transported liquid. For this reason, we used a microchannel with 120 μm in diameter, and a good magnification to have a clear image of the boundary between two immiscible fluids like air-liquid and/or liquid$_1$-liquid$_2$. The computerized generator of absolute coordinates needs to have a clear edge for internal accurate measurements [56,58]. Thus, in this volume determination method, the image of this boundary is very important. This generator is in fact a complex microscopic system that calculates automatically the current absolute coordinates of the mouse cursor (pointer) within the microscopic field of view. Summarizing all the above considerations, the principle of volumes detection and measurement mainly involve three elements: 1) a cylindrical capillary; 2) a fluid boundary as detectable tracking element; 3) an absolute coordinate's generator in microscopic field of view. The capillary must be fabricated based on PDMS molding techniques [51–55] and it must have finally the imposed cylindrical geometry with the precise dimensions determined using microscopic tools [56,58].

The PDMS chip element (Figure 2a) sustains the spatiality of the microchannel (capillary) creating the mechanical premise for the microinjector connector and inverse microscopic visualization (Figure 2d). The tracked fluid boundaries are air-liquid (Figure 2b) and the visual volume detection is obtained by computing two successive boundary positions (Figure 2c). The liquid is a commercial red color ink.

The PDMS-based sensor chip is connected to a microinjector system (Figure 3a). This microinjector system is a MPWM equipment type that will be used to circulate the working fluid using the pulse modulation principle (Figure 3b). The generation of fluid volumes is quantified by controlling the closing and opening times of the valve. When the valve is opened (ON position), the pressure difference between the two chambers connected by the valve will allow a precise volume of fluid to flow. When the valve is closed (OFF position), the fluid circulation is blocked. The time when a sequence of closed/open states occurs is called a period. The ratio between ON time and period is called duty cycle. The duty cycle is the basic parameter of the volume generation from the experiment, keeping the pressures and the MPWM signal frequency constant. Figure 3c shows the schematic Digital Tracking Coordinates System (DTCS) of Celteh Company (Targoviste, 130092, Romania), which allows the determination of absolute coordinates in the videomicroscope active screen. The DTCS or also called bi-dimensional micrometric terminal blocks (BMS) [61], is presented in detail in the paper of by Ardeleanu et al. [61].
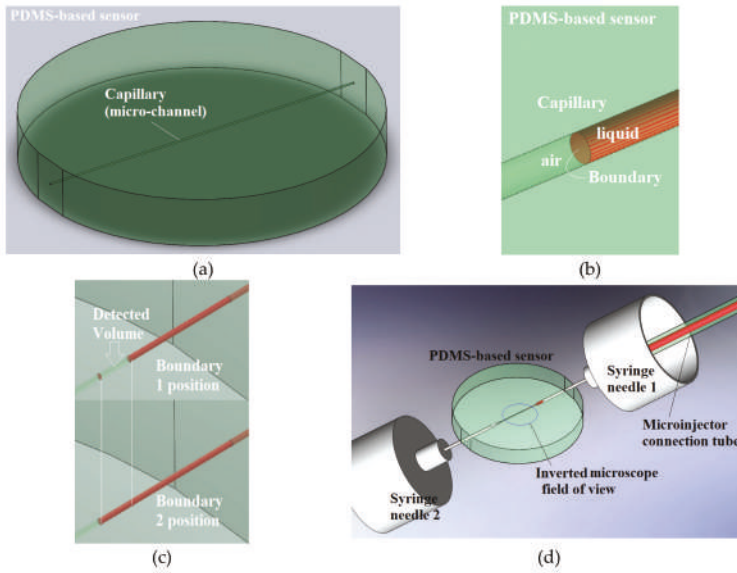
Figure 2. PDMS-based sensor: (**a**) Sensor-chip and (**b**) Boundary air-liquid; (**c**) Illustration of two distinct positions of successive boundaries accounted in the measurement of the microfluidic volume; (**d**) Inverted microscope field of view for PDMS-based sensor.
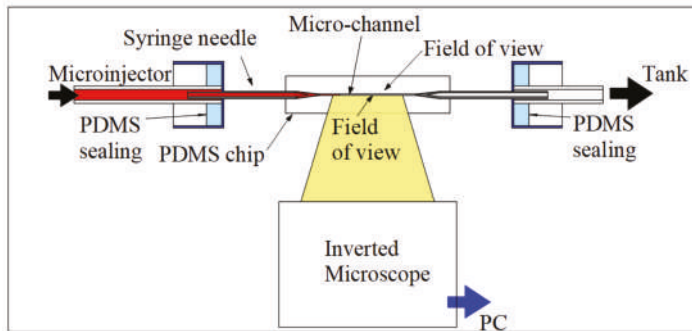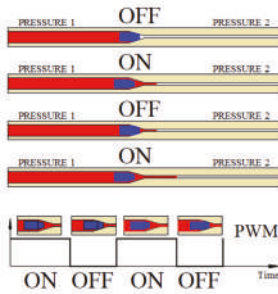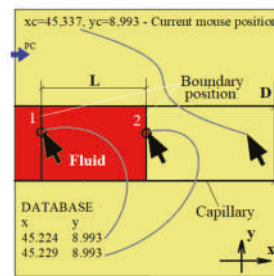
Figure 3. Experimental set up and measurements: (**a**) Schematic representation of the experimental set up, (**b**) MPWM principle, (**c**) The measurement performed by taking into account the advancement of the liquid boundary with the length (L) (the cross-sectional area of the channel is constant).

The mouse pointer is the object to which DTCS interactively assigns absolute coordinates. With the "double-click" command, the current position of the mouse pointer is registered by the DTCS in a database with a certain order number. The recorded coordinates are also visible on the interface, the operator being able to use them for the purpose of determinations and calculations.

The minimum transported volume by microinjector is in the femtoliters range. In the field of view of the inverted microscope, the air-liquid boundary will be detected using DTCS (Figure 3a). Figure 3b shows the fluid advance from the initial moment at different times, t, represented by closing/opening (on/off) of the chamber with valves, controlled by the MPWM signal. The fluid advance after each MPWM signal is thus controlled and precisely determined by the advancement of very small volumes of liquid from the microfluidic level. A volume determination involves two distinct positions of the fluid boundary at two distinct moments (Figure 3b). The absolute coordinate generator has the ability to determine with high precision of less than 1 μm the current mouse cursor position. A simple click allows the position data to be recorded, in absolute coordinates of x and y. A single (pL) volume measurement involves a succession of two distinct boundary positions and two distinct position data records. The researcher uses these data sets to determine the length L of injected fluid volume by one MPWM impulse that generates the fluid movement, as shown in Figure 3b. The liquid volume determination consists in using the Equation (1):

$$V = L \cdot \frac{\pi \cdot D^2}{4} \tag{1}$$

For every injected volume in capillary, the database will be populated with distinct coordinates of the position of the fluid boundary. The capillary is aligned with one of the two coordinates, which is x-axis in this case. The *x* becomes the main coordinate for calculating the length L of an injected volume, calculated by subtracting two successive values from the database, i.e., x_{k-1} and x_k.

3.2. Method of Experimental Measurements of Picoliter Volumes

The measurement of the movement of the boundaries of liquid during fluid movements was done by two methods: the first one is the current coordinate mouse registration "mouse head tracking method" (MHT) and the second one is the "pointing method" (P). In the MHT method, the current coordinates of the mouse pointer are displayed on the interface at the associated positions on the screen. The tip of the mouse is a visible landmark that offers precision due to its small size relative to the tracked object. Figure 4 illustrates the MHT method, which was also used for MPWM calibration.

In Figure 4a,b, the mouse pointer coordinates are shown as pointed on the interface with x_m and y_m. These x_m and y_m values change interactively with the change of the position of the mouse, which allows the tracking of elements of interest on the screen of the video-microscope.

After the calibration was performed with the MHT method, the second method, P, was used to test the fluid volume generation. For a series of successive steps of injected volume with picoliter amounts, we use the P method presented in Figure 5. We dimensioned the Pointer Width (diameter) (PW) bigger than the determined Boundary Width (BW) (Figure 5a).

In this way, the operator will position the point symmetric related to the liquid boundary in the same horizontal line (Figure 5a). The precision of measurements is determined by the operator's scoring precision (human factor) (Figure 5b).

For calibration, we used a magnification of 800× for visualization, because the volumes of fluid passing through microchannels were of the order of 25 - 50 pL, i.e., 1 MPWM step, and this condition assumes that the distance between two successive boundaries was very small. The MHT method is appropriate in this case, because the mouse size on the working screen is quite large in relation to the thickness of the boundary and allows a good positioning of the mouse tip relative to the center of the boundary, which is visually approximated by the human operator.

(a)

(b)

(c)
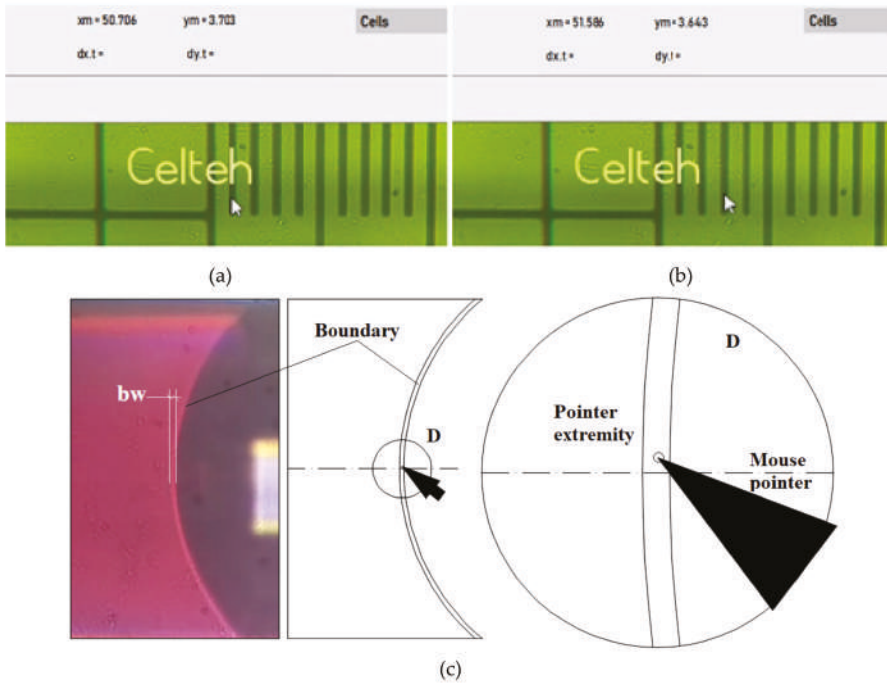
Figure 4. Representation of the MHT method: (**a**) microscope image capture of the first position of the mouse pointer, (**b**) microscope image capture of the second position of the mouse pointer, (**c**) schematic representation of the MHT method, applied to the liquid boundary measurement on a PDMS-based sensor

(a)

(b)
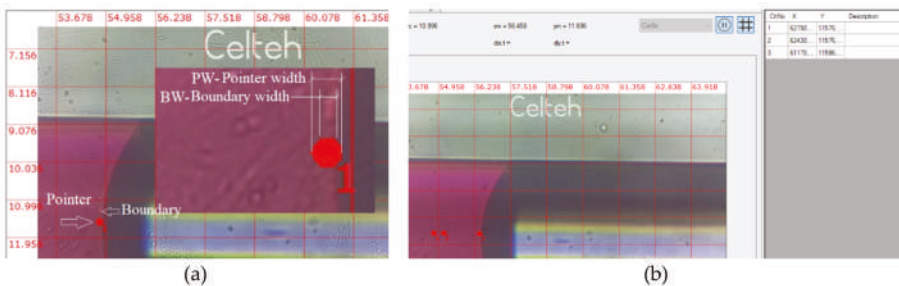
Figure 5. Microscope image captures of (**a**) method P for visual volumes detection (the insert shows the magnified pointer point and boundary); (**b**) Successive pointing positions.

For measurements, the amounts of fluid passing through microchannels are large, usually tens of MPWM steps, which means that the volumes are of the order of hundreds and thousands of pL. To include two successive boundaries on the screen, a smaller magnification (i.e., >500×) is required. In this case, the thickness of boundary is defined with fewer pixels, which leads to an increase in the uncertainty of the position of the mouse tip in the center of boundary. Therefore, a point in the shape of a red circle (marker), with a diameter slightly larger than the boundary thickness has been proposed. This marker can be identified very well if it has been positioned by the operator in the center of boundary. The tip of the mouse becomes inaccurate as a reference to boundaries, therefore the

geometry of the reference object is increased, i.e., a circular point with a diameter slightly larger than the thickness of boundaries. In this way, the thickness of the boundary is positioned relative to the point, becoming very visible the symmetry of crossing the boundary through the middle of the point (red circle).

To measure, the operator uses the resulted coordinates from the interface and processes these data using Equations (2) and (3). The Equation (2) is the theoretical base for Equation (3), when we took in consideration the calibration factor $u_{calibration} = 44$, which results from Figure 6.
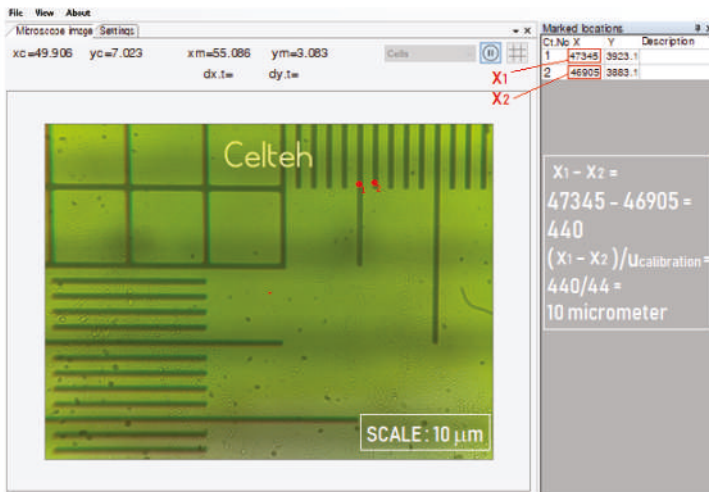
Figure 6. Validation of measurement precision of the distance between two successive gradation lines on the calibration scale representing 10 μm, taking into account the value of the calibration factor of 44.

After each MPWM executed step (impulse), a new boundary position is formed in the microscope field of view. The operator must point every new position boundary that generates automatically the X and Y absolute coordinates. The microchannel is aligned with the x-axis and so it will be used to calculate only the x-coordinate from the DTCS interface table. The volume depends on the x_k and x_{k-1} coordinates, which are the only variables in Equation (2) and, taking into account the following constants: D- microchannel diameter and $u_{calibration}$ – calibration DTCS factor. The volume measured at "Point k" is given by Equation (2):

$$V_k = \frac{\pi \cdot D^2}{4} \frac{x_k - x_{k-1}}{u_{calibration}} 10^{-3} \; [\text{pL}] \qquad (2)$$

where: V_k is the volume measured at "Point k" [pL], x_{k-1} is the absolute coordinate at "Point (k-1)" [a.u.]; $_{xk}$ is the absolute coordinate at "Point k" [a.u.]; D is the microchannel diameter [μm]; $u_{calibration}$ is the calibration. Equation (2) is obtained from the volume calculation of a cylinder with the surface of a circle of diameter D and the length deduced from the absolute coordinates x_{k-1} and x_k.

In our case, for a channel with $D = 120$ μm and $u_{calibration} = 44$, the Equation (2) becomes:

$$V_k = 0.257039 \cdot (x_k - x_{k-1}) \; [\text{pL}] \qquad (3)$$

The displacement of the liquid through the channel appears when a pressure difference is applied between the two chambers bounded by the liquid-air front. The channel has a very small section compared to the rest of the liquid path (the internal tubes of the MPWM system), which will produce a hydraulic resistance behavior. The pressures used must be appropriate to the geometry of the

microchannel. The applied pressures used in these experiments were in the range of hundreds of millibars.

The volume determination will be performed automatically in the near future, through the image processing boundary recognition. At the moment, the measurement process is manual, the human factor being "the trigger" of coordinate recording, which leads to the determination of the length (L) and volume calculation. In order to generate picoliter volumes, the MPWM system must be calibrated at this working level.

4. Calibration of MPWM for Picolitre Volumes

The MPWM device is a complex technical system [56] that was built specifically for microfluidic transportation and measurements. An on/off valve is actuated to allow an energetic transfer of fluid between a pressurized and a depressurized space. For calibration, the pressures, frequency of the command signal and duty cycle are combined to quantify the fluidic volumes that pass through the on/off valve. Each volume generated by the MPWM system represents the quantity and the force that has an impact on the fluidic environment. In bioprocess microfluidics, extracting an adherent cell from a Petri dish requires a force applied for a short period of time, which means for MPWM a great negative pressure, high frequency and medium to high duty cycle. To transport a cell after extraction to a remote microfluidic room, speed and precision is required, this means for MPWM high pressure, high frequency and small to medium duty cycle. In this way, the MPWM signal can be tailored on specific application. For this experimental work, the final goal is to obtain a very high resolution for the 120 μm channel and a very high precision of the transported fluidic volumes. To detect the smallest volumes as possible, we used the PDMS-sensor along with the MPWM system.

On this occasion, we also tested the working capacity of the MPWM device, dimensioning the volumes generated by the MPWM system. For this purpose, we used small pressure levels, medium frequencies, and small duty cycles factor. The MPWM system presents a high repeatability of the generated volumes. The PDMS-based sensor allows to precise measuring the volumes generated by the MPWM device and storing the information along with the associated values of the input parameters. The correlations between causes (pressures, frequencies and duty cycles) and effects (generated volumes) will be the calibration process and will be stored as a map. We used in the calibration process a frequency of 100 Hz, which is one previously used for cell extraction and membrane cell elasticity testing [62].

Tables S2 and S3 (Supplementary Materials) present experimental data for the calibration in the following conditions: a constant pressure of 256 mbar, frequency of 100 Hz and duty cycle range [13.5%, 16.8%]. These tables include the *x*-coordinates measured with the DTCS system and the calculated using Equation (2) and corresponding to the determined volumes. These data allow the MPWM user to determine the calibration function. This function allows us to calculate the appropriate duty cycle parameter for the transport of certain required volumes (imposed).

For data processing, the function of approximating the correlation between the generated volume size and the duty cycle was sought. For this, the average value of the volume generated by the MPWM system was calculated for each duty cycle tested. For the average values resulting from each filling factor tested, the calibration MPWM function was obtained as shown in Figure 7.
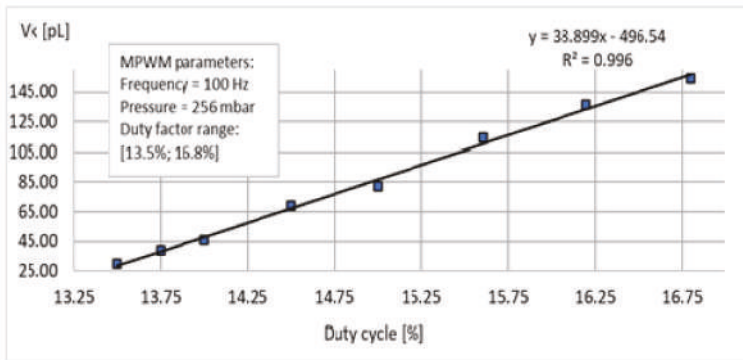
Figure 7. Chart of the MPWM experimental data calibration.

The use of this function must also take into account the deviation of measurements corresponding to each average value, deviation of measurements determined by reporting the average value at fitted values (those obtained by applying linear regression). By processing the MPWM calibration data, for each duty cycle, a series of average values of the volumes generated were obtained, which are presented in Table 2. In addition, the deviations corresponding to each average value are shown. Considering the average value as a reference, the relative deviation was calculated as the maximum fluctuation relative to the average value. The calibration function (Figure 7) is a mathematical tool for the human operator to choose the optimum MPWM parameters into a given application (transportation small/tiny volumes).

Table 2. A series of average values of the volumes generated by processing the MPWM calibration data, for each duty cycle.

Duty Cycle(%)	Average Volume(pL)	Fitted Value(pL)	AverageDev. of Meas. (%)
16.80	154.13	156.96	1.81
16.20	136.42	133.62	2.10
15.60	114.17	110.28	3.53
15.00	82.35	86.95	5.28
14.50	69.13	67.50	2.88
14.00	45.85	48.05	4.57
13.75	38.59	38.32	2.7
13.50	29.68	28.60	3.96

4.1. Experimental Results

4.1.1. Microscopic Analyses of Microchannels

Figure 8 presents the images of the circular cross-sectional area of channels obtained after 120 μm copper wires removal. A smooth surface of microchannel was observed, which allows an uniform and well controllable flow of the fluid.

The microwire-molding techniques allow fabricating rounded channel shapes in PDMS. These fabricated channels have a circular cross section (Figure 8), that allow a very good visualization in the analyzed field, due to the efficiency of light transmission inside on-chip waveguides for light-sensing and light actuation methods.
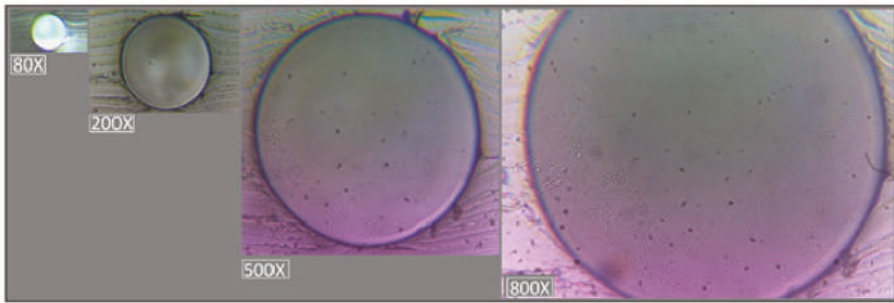
Figure 8. The circular cross-sectional area of microchannels produced by microwire molding technique, after 120 μm copper removal, at different magnifications (80 ×, 200 ×, 500 ×, 800 ×).

By extracting the wire under the conditions of swelling of PDMS, the longitudinal micro-irregularities have been created that do not affect the laminar flow of the fluid, but for the precise determination of the maximum error affecting the calculation of the fluid volume, the variation of the channel section due to the roughness can be taken into account. The surface irregularities affect in a small proportion (maximum 3.8%) only the area of the channel section at the diameter level (Figure 8).

The experiments were performed by researchers from Valahia University of Targoviste in collaboration with Celteh Mezotronic. An automatic microinjection system was adapted to existing MPWM equipment, by integrating a sensor based on image processing of fluid flow through the microchannel realized with the technique known as microwire-molding. The image processing software was developed by Ardeleanu in collaboration with Company Celteh and is an innovation in the field of microscopic mechanical displacement determination with submicrometric precision.

The validation of the idea that the sensor allows the measurement of microfluidic volumes is the subject of this work.

The precision of the equipment for determining the absolute dimension is below 1 μm. The display in the interface (Figure 9) means the absolute dimensions of the mouse pointer in the field of view of the microscope, expressed in virtual numerical units.

(a)

Figure 9. *Cont.*

(b)
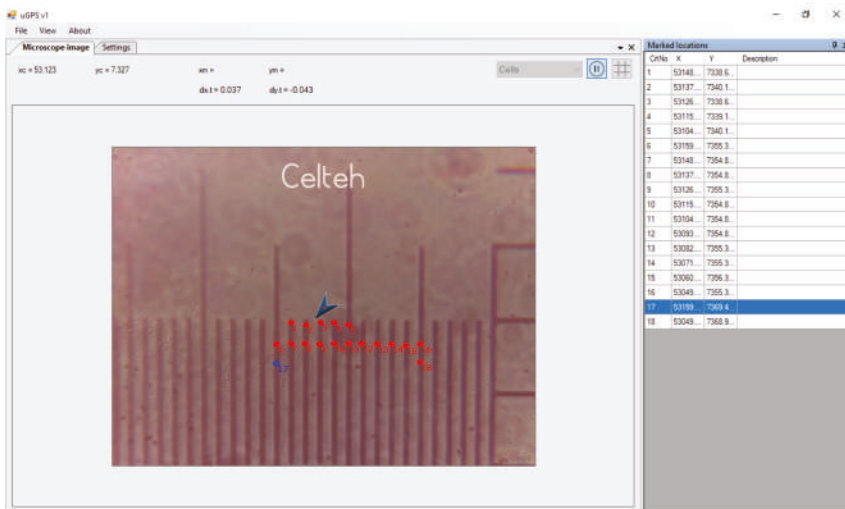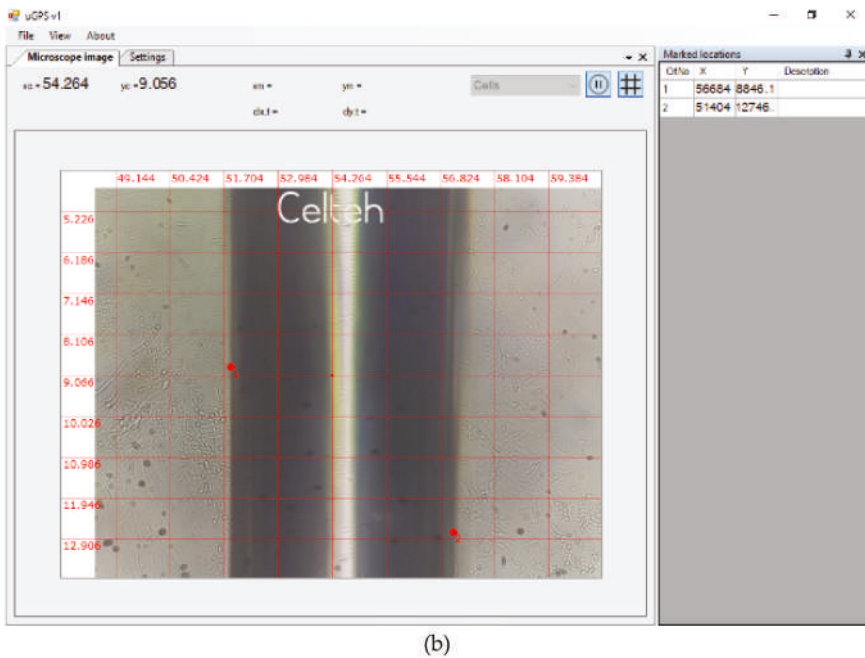
Figure 9. Calibration method of the DTCS system: (**a**) screen-print showing 11 red points (virtual units) representing 10 μm in length, (**b**) screen-print showing the measurement of the cannel diameter of 120 μm by DTCS.

The measurements required for the experiment represent specific distances within the field of vision, i.e., the distance between two liquid fronts generated by the MPWM signal. Figure 9 shows, for example, the calibration mode of the DTCS system. After the calibration it was found that: 11 vital points (units) represent 10 μm (Figure 9a) and thus, with this calibrated system we were able to determine the diameter of the microchannel (Figure 9b). The DTCS system can associate a number of arbitrary units to a real calibration distance (μm) so that distances of less than 1 μm can be measured with very precise approximations.

In Figure 9a calibration of 11 virtual units at a real distance of 10 μm is rendered by a specially graded ruler for this purpose. The meaning of the two red dots (1 and 2) from Figure 9b represents the positions of reaching the extremities of the visualized channel for measurement. The clarity of the image allows a precise reproduction of these edges, which leads to a precise diameter calculation. In order to determine the diameter of the microfluidic channel, a calibration of the DTCS with 44 virtual numerical units for a real distance of 10 μm was used. As can be seen in the image of the DTCS interface in Figure 9, the two reference points recorded by the human operator with the mouse on the screen, have the absolute x coordinates of $x_1 = 56.684$ and $x_2 = 51.404$. The difference between the two coordinates is the diameter of the channel expressed in virtual numerical units, i.e., 5280.

4.2. Results of Experimental Tests for Microfluidic Transportation

The microfluidic transportation was tested to determine the deviation of measurements obtained when small volumes were generated by the MPWM system, using the calibration function obtained previously. Using a frequency of 100 Hz, a pressure of 256 mbar and a duty cycle of 14.5%, a fixed number of MPWM steps were generated in order to obtain a given total volume. This volume

is obtained by multiplying the volume corresponding to a single MPWM step by the number of steps required.

To determine the deviation of measurements due to execution, we repeated 10 times each test. Thus, the repeatability of the volumes generated by the MPWM system was obtained for a given parametric conditions. According to the calibration function, the volume for a single step under the given conditions is 67.5 pL. The four experimental tests corresponding to 5, 10, 15 and 20 MPWM steps have been generated. Tables 3 and 4 shows the values of volumes measured in these experiments and the associated deviation of measurements (dev. of meas.) with respect to the reference volume calculated with the calibration function.

Table 3. MPWM repeatability experimental tests performed for 5 and 10 steps in the following conditions: frequency of 100 Hz, pressure of 256 mbar and duty cycle of 14.5%.

Measurement	5 Steps					10 Steps				
No.	Start, Virtual Unit	Stop, Virtual Unit	V_{5step} (pL)	Fitted Volume (pL)	Dev. of meas. (%)	Start, Virtual Unit	Stop, Virtual Unit	V_{10step} (pL)	Fitted Volume (pL)	Dev. of Meas. (%)
1	87754	86371	355.48	337.50	5.33	87932	85181	707.11	675.00	4.76
2	87762	86520	319.24	337.50	5.41	87928	84966	761.34	675.00	12.79
3	87744	86308	369.11	337.50	9.37	87931	85125	721.25	675.00	6.85
4	87749	86313	369.11	337.50	9.37	87940	85372	660.07	675.00	2.21
5	87758	86498	323.87	337.50	4.04	87937	85023	749.01	675.00	10.96
6	87751	86245	387.10	337.50	14.70	87923	85356	659.81	675.00	2.25
7	87757	86391	351.12	337.50	4.03	87940	85264	687.83	675.00	1.90
8	87745	86379	351.12	337.50	4.03	87626	84715	748.24	675.00	10.85
9	87750	86419	342.12	337.50	1.37	87932	84941	768.80	675.00	13.90
10	87754	86248	387.10	337.50	14.70	87946	85327	673.18	675.00	0.27

Table 4. MPWM repeatability experimental tests performed for 15 and 20 steps in the following conditions: frequency of 100 Hz, pressure of 256 mbar and duty cycle of 14.5%.

Measurement	15 Steps					20 Steps				
No.	Start, Virtual Unit	Stop, Virtual Unit	V_{15step} (pL)	Fitted Volume (pL)	Dev.of Meas. (%)	Start, Virtual Unit	Stop, Virtual Unit	V_{20step} (pL)	Fitted Volume (Pl)	Dev.of Meas. (%)
1	88025	83882	1064.91	1012.50	5.18	88405	82843	1429.65	1350.00	5.90
2	88085	83941	1065.17	1012.50	5.20	88449	82887	1429.65	1350.00	5.90
3	88021	83535	1153.07	1012.50	13.88	88412	82991	1393.40	1350.00	3.22
4	88015	83818	1078.79	1012.50	6.55	88394	83325	1302.93	1350.00	3.49
5	88019	83560	1146.13	1012.50	13.20	88403	82700	1465.89	1350.00	8.58
6	88035	83918	1058.23	1012.50	4.52	88396	82872	1419.88	1350.00	5.18
7	88064	83764	1105.26	1012.50	9.16	88423	83318	1312.18	1350.00	2.80
8	88059	84130	1009.90	1012.50	0.26	88452	82996	1402.40	1350.00	3.88
9	88044	83640	1132	1012.50	11.80	88405	82207	1593.12	1350.00	18.01
10	88075	84249	983.43	1012.50	2.87	88389	82651	1474.89	1350.00	9.25

The next parameter of interest is the standard deviation. In Figure 10, the values of the standard deviation for each of the four volumes tested under the same conditions 10 times are shown.
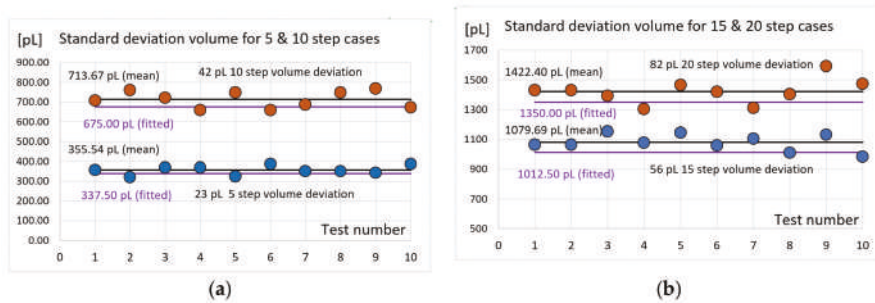
Figure 10. The values of the standard deviation for the four expected volumes: (**a**) for 5 and 10 steps cases; (**b**) for 15 and 20 steps cases.

5. Conclusions

A PDMS-based sensor with circular cross section microchannel was obtained experimentally through the microwire molding technique. PDMS-based sensor with microchannels of 63, 120 and 190 μm were obtained and analyzed to check the quality and the micrometric sizes of the channels. The PDMS-based sensor was integrated with the MPWM system and the methodology to measure picoliter volumes was developed. We have shown that the set-up proposed in this paper was able to verify the microinjection-sensor conceptualization. The measuring process was based on absolute coordinates in the microscopic field of view with high magnification factors. The DTCS system works based on image processing, mainly detecting the mouse pointer coordinates. The human operator had the possibility to pointing with the mouse pointer, into a very precise mode, different positions of fluid boundary as main trackable element from the micro-channel, watching it on the video-microscope screen during experiments. For microfluidic volumes circulation a very high precise generator was used.

The sensor was specially designed to allow the calibration of the MPWM-injector. MPWM is based on a PWM signal that is, in fact, a "train" of identical pulses. One single pulse is a passed volume through the on/off valve. Certain applications request a given precise fluid quantity to be transported, i.e., an MPWM fluidic signal with a given number of pulses. Once the unique relation between a volume (value) and a pulse (pressure, frequency, duty cycle) is established, it only remains to generate a precise PWM signal with a precise number of pulses. In addition to the calibration of the device, the sensor serves the MPWM system to close the control loop in the microinjection process. This is why this novel PDMS-based sensor system for MPWM measurements of microfluidic (nanoliter to picoliter) volumes contributes to the advancement of intelligent control methods and techniques, and could lead to new developments in the design, control, and in applications of real-time intelligent sensor system control.

Following the experiments, the MPWM calibration function was determined, using the relationship between the microfluidic (nanoliter to picoliter) volumes and the duty cycle of the system, thus obtaining the MPWM experimental calibration diagram. Following the calibration, at 100 Hz, 14.5 duty factor and 256 mbar the microfluidic transport and the resulting picoliter volumes were tested using the pointing method. The results of the experiments confirm the quality and precision of the measurement at picoliter volumes, making it possible, in future system developments, to be applied in high-end research fields such as the automatic microinjection of biological cells.

References

1. Ellson, R.N. Nanoliter and picoliter liquid handling for life science applications. In Proceedings of the Nanotech and Biotech Convergence Conference, Stanford, CA, USA, May 2003; p. 7.

2. McDonald, J.C.; Duffy, D.C.; Anderson, J.R.; Chiu, D.T.; Wu, H.; Schueller, O.J.; Whitesides, G.M. Fabrication of microfluidic systems in poly(dimethylsiloxane).Pdf. *Electrophoresis* **2000**, *21*, 14. [CrossRef]

3. Niu, X.; Zhang, M.; Peng, S.; Wen, W.; Sheng, P. Real-time detection, control, and sorting of microfluidic droplets. *Biomicrofluidics* **2007**, *1*, 44101. [CrossRef] [PubMed]

4. Oliva, P.; Bircher, B.A.; Schoenenberger, C.A.; Braun, T. Array based real-time measurement of fluid viscosities and mass-densities to monitor biological filament formation. *Lab Chip* **2019**, *19*, 1305–1314. [CrossRef] [PubMed]

5. Salafi, T.; Zeming, K.K.; Lim, J.W.; Raman, R.; Seah, A.W.R.; Tan, M.P.; Zhang, Y. Portable smartphone-based platform for real-time particle detection in microfluidics. *Adv. Mater. Technol.* **2019**, *4*, 1800359. [CrossRef]

6. Shameer, K.; Badgeley, M.A.; Miotto, R.; Glicksberg, B.S.; Morgan, J.W.; Dudley, J.T. Translational bioinformatics in the era of real-time biomedical, health care and wellness data streams. *Brief Bioinform* **2017**, *18*, 105–124. [CrossRef] [PubMed]

7. Whitesides, G.M. The origins and the future of microfluidics. *Nature* **2006**, *442*, 368–373. [CrossRef]

8. Abgrall, P.; Gué, A.M. Lab-on-chip technologies: Making a microfluidic network and coupling it into a complete microsystem—A review. *J. Micromech. Microeng.* **2007**, *17*, R15–R49. [CrossRef]

9. Liu, L.; Cao, W.; Wu, J.; Wen, W.; Chang, D.C.; Sheng, P. Design and integration of an all-in-one biomicrofluidic chip. *Biomicrofluidics* **2008**, *2*, 34103. [CrossRef]

10. Oh, K.W. *Lab-on-Chip (Loc) Devices and Microfluidics for Biomedical Applications*; Woodhead Publishing Limited: Sawston, UK, 2012; pp. 150–171.

11. Gan, L.T.; Xia, Z.N.; Chen, H.; Yu, Y.L. A novel method for fabricating polydimethylsiloxane microfluidic chip master molds. *J. Chongqing Univ. (Engl. Ed.)* **2009**, *8*, 6.

12. Halldorsson, S.; Lucumi, E.; Gomez-Sjoberg, R.; Fleming, R.M.T. Advantages and challenges of microfluidic cell culture in polydimethylsiloxane devices. *Biosens Bioelectron* **2015**, *63*, 218–231. [CrossRef]

13. Liu, Q.; Wu, C.; Cai, H.; Hu, N.; Zhou, J.; Wang, P. Cell-based biosensors and their application in biomedicine. *Chem Rev* **2014**, *114*, 6423–6461. [CrossRef] [PubMed]

14. Chen, J.; Li, J.; Sun, Y. Microfluidic approaches for cancer cell detection, characterization, and separation. *Lab Chip* **2012**, *12*, 1753–1767. [CrossRef] [PubMed]

15. Permana, S.; Grant, E.; Walker, G.M.; Yoder, J.A. A review of automated microinjection systems for single cells in the embryogenesis stage. *IEEE/ASME Trans. Mechatron.* **2016**, *21*, 2391–2404. [CrossRef]

16. Torino, S.; Corrado, B.; Iodice, M.; Coppola, G. Pdms-based microfluidic devices for cell culture. *Inventions* **2018**, *3*, 65. [CrossRef]

17. Vaughan, D.A.; Sakkas, D. Sperm selection methods in the 21st century. *Biol Reprod* **2019**. [CrossRef]

18. Zhou, Y.; Basu, S.; Wohlfahrt, K.J.; Lee, S.F.; Klenerman, D.; Laue, E.D.; Seshia, A.A. A microfluidic platform for trapping, releasing and super-resolution imaging of single cells. *Sens Actuators B Chem* **2016**, *232*, 680–691. [CrossRef]

19. Jokerst, J.C.; Emory, J.M.; Henry, C.S. Advances in microfluidics for environmental analysis. *Analyst* **2012**, *137*, 24–34. [CrossRef]

20. Oscar, S.V.; Fernando, O.L.; Del Pilar, C.M. Total polyphenols content in white wines on a microfluidic flow injection analyzer with embedded optical fibers. *Food Chem* **2017**, *221*, 1062–1068. [CrossRef]

21. Visser, C.W.; Kamperman, T.; Karbaat, L.P.; Lohse, D.; Karperien, M. In-air microfluidics enables rapid fabrication of emulsions, suspensions, and 3d modular (bio)materials. *Sci. Adv.* **2018**, *9*. [CrossRef]

22. Nge, P.N.; Rogers, C.I.; Woolley, A.T. Advances in microfluidic materials, functions, integration, and applications. *Chem Rev* **2013**, *113*, 2550–2583. [CrossRef]

23. Jovic, A.; Howell, B.; Takayama, S. Timing is everything: Using fluidics to understand the role of temporal dynamics in cellular systems. *Microfluid. Nanofluidics* **2009**, *6*, 717–729. [CrossRef]

24. da Silva, E.N.T.; Ferreira, V.S.; Lucca, B.G. Rapid and inexpensive method for the simple fabrication of pdms-based electrochemical sensors for detection in microfluidic devices. *Electrophoresis* **2019**, *40*, 1322–1330. [CrossRef] [PubMed]

25. Koo, H.J.; Velev, O.D. Design and characterization of hydrogel-based microfluidic devices with biomimetic solute transport networks. *Biomicrofluidics* **2017**, *11*, 024104. [CrossRef] [PubMed]

26. Makamba, H.; Kim, J.H.; Lim, K.; Park, N.; Hahn, J.H. Surface modification of poly(dimethylsiloxane) microchannels. *Electrophoresis* **2003**, *24*, 3607–3619. [CrossRef] [PubMed]

27. Lee, J.N.; Park, C.; Whitesides, G.M. Solvent compatibility of poly(dimethylsiloxane)-based microfluidic devices. *Anal. Chem.* **2003**, *75*, 11. [CrossRef] [PubMed]

28. Tjerkstra, R.W.; De Boer, M.; Berenschot, E.; Gardeniers, J.G.; van den Berg, A.; Elwenspoek, M.C. Etching technology for chromatography microchannels. *Electrochim. Acta* **1997**, *42*, 8. [CrossRef]

29. Becker, H.; Heim, U. Hot embossing as a method for the fabrication of polymer high aspect ratio structures. *Sens. Actuators* **2000**, *83*, 6. [CrossRef]

30. Fiddes, L.K.; Raz, N.; Srigunapalan, S.; Tumarkan, E.; Simmons, C.A.; Wheeler, A.R.; Kumacheva, E. A circular cross-section pdms microfluidics system for replication of cardiovascular flow conditions. *Biomaterials* **2010**, *31*, 3459–3464. [CrossRef]

31. Lee, D.-K.; Kwon, J.Y.; Cho, Y.H. Fabrication of microfluidic channels with various cross-sectional shapes using anisotropic etching of si and self-alignment. *Appl. Phys. A* **2019**, *125*. [CrossRef]

32. Lin, C.-H.; Lee, G.-B.; Chang, B.-W.; Chang, G.-L. A new fabrication process for ultra-thick microfluidic microstructures utilizing su-8 photoresist. *J. Micromech. Microeng.* **2002**, *12*, 590–597. [CrossRef]

33. Wang, G.J.; Ho, K.H.; Hsu, S.H.; Wang, K.P. Microvessel scaffold with circular microchannels by photoresist melting. *Biomed Microdevices* **2007**, *9*, 657–663. [CrossRef] [PubMed]

34. Wilson, M.E.; Kota, N.; Kim, Y.; Wang, Y.; Stolz, D.B.; LeDuc, P.R.; Ozdoganlar, O.B. Fabrication of circular microfluidic channels by combining mechanical micromilling and soft lithography. *Lab Chip* **2011**, *11*, 1550–1555. [CrossRef] [PubMed]

35. Yan, S.; Li, Y.; Zhao, Q.; Yuan, D.; Yun, G.; Zhang, J.; Wen, W.; Tang, S.Y.; Li, W. Liquid metal-based amalgamation-assisted lithography for fabrication of complex channels with diverse structures and configurations. *Lab Chip* **2018**, *18*, 785–792. [CrossRef] [PubMed]

36. Takeuchi, S.; Garstecki, P.; Weibel, D.B.; Whitesides, G.M. An axisymmetric flow-focusing microfluidic device. *Adv. Mater.* **2005**, *17*, 6. [CrossRef]

37. Wu, J.; Gu, M. Microfluidic sensing- state of the art fabrication and detection techniques. *J. Biomed. Opt.* **2011**, *16*, 13. [CrossRef] [PubMed]

38. Kim, M.; Moon, B.-U.; Hidrovo, C.H. Enhancement of the thermo-mechanical properties of pdms molds for the hot embossing of pmma microfluidic devices. *J. Micromech. Microeng.* **2013**, *23*, 095024. [CrossRef]

39. Qi, S.; Liu, X.; Ford, S.; Barrows, J.; Thomas, G.; Kelly, K.; McCandless, A.; Lian, K.; Goettert, J.; Soper, S.A. Microfluidic devices fabricated in poly(methyl methacrylate) using hot-embossing with integrated sampling capillary and fiber optics for fluorescence detection. *Lab Chip* **2002**, *2*, 88–95. [CrossRef]

40. Attia, U.M.; Marson, S.; Alcock, J.R. Micro-injection moulding of polymer microfluidic devices. *Microfluid. Nanofluidics* **2009**, *7*, 1–28. [CrossRef]

41. Ardeleanu, M.N.; Ivan, I.A.; Despa, V. Rapid prototyping technologies used for a microgripper frameworks fabrication. 2013, 44, 3. *Rom. Rev. Precis. Mech. Optics Mechatron.*.

42. Rossier, J.; Reymond, F.; Michel, P.E. Polymer microfluidic chips for electrochemical and biochemical analyses. *Electrophoresis* **2002**, *23*, 10. [CrossRef]

43. Cao, H.; Tegenfeldt, J.O.; Austin, R.H.; Chou, S.Y. Gradient nanostructures for interfacing microfluidics and nanofluidics. *Appl. Phys. Lett.* **2002**, *81*, 3058–3060. [CrossRef]

44. Abdelgawad, M.; Wu, C.; Chien, W.Y.; Geddie, W.R.; Jewett, M.A.; Sun, Y. A fast and simple method to fabricate circular microchannels in polydimethylsiloxane (pdms). *Lab Chip* **2011**, *11*, 545–551. [CrossRef] [PubMed]

45. Mappes, T.; Achenbach, S.; Mohr, J. X-ray lithography for devices with high aspect ratio polymer submicron structures. *Microelectron. Eng.* **2007**, *84*, 1235–1239. [CrossRef]

46. Lorenzini, M.; Suzzi, N. The influence of geometry on the thermal performance of microchannels in laminar flow with viscous dissipation. *Heat Transf. Eng.* **2016**, *37*, 1096–1104. [CrossRef]

47. Parker, B.; Ahmadi, A.; Samanipour, R.; Kim, K. Rapid fabrication of circular channel microfluidic flow-focusing devices for hydrogel droplet generation. *Micro Nano Lett.* **2016**, *11*, 41–45. [CrossRef]

48. Das, A.; Bhaumik, S.K. Fabrication of cylindrical superhydrophobic microchannels by replicating lotus leaf structures on internal walls. *J. Micromech. Microeng.* **2018**, *28*, 045011. [CrossRef]

49. Lu, X.; Liu, C.; Hu, G.; Xuan, X. Particle manipulations in non-newtonian microfluidics: A review. *J Colloid Interface Sci* **2017**, *500*, 182–201. [CrossRef]

50. Bliss, C.L.; McMullin, J.N.; Backhouse, C.J. Rapid fabrication of a microfluidic device with integrated optical waveguides for DNA fragment analysis. *Lab Chip* **2007**, *7*, 1280–1287. [CrossRef]

51. Xiang, N.; Dai, Q.; Han, Y.; Ni, Z.H. Circular-channel particle focuser utilizing viscoelastic focusing. *Microfluid. Nanofluidics* **2019**, *23*. [CrossRef]

52. Effati, E.; Pourabbas, B. New portable microchannel molding system based on micro-wire molding, droplet formation studies in circular cross-section microchannel. *Mater. Today Commun.* **2018**, *16*, 119–123. [CrossRef]

53. Jia, Y.F.; Jiang, J.H.; Ma, X.D.; Li, Y.; Huang, H.M.; Cai, K.B.; Cai, S.X.; Wu, Y.P. Pdms microchannel fabrication technique based on microwire-molding. *Chin. Sci. Bull.* **2008**, *53*, 3928–3936. [CrossRef]

54. Song, S.-H.; Lee, C.-K.; Kim, T.-J.; Shin, I.-c.; Jun, S.-C.; Jung, H.-I. A rapid and simple fabrication method for 3-dimensional circular microfluidic channel using metal wire removal process. *Microfluid. Nanofluidics* **2010**, *9*, 533–540. [CrossRef]

55. Verma, M.K.; Majumder, A.; Ghatak, A. Embedded template-assisted fabrication of complex microchannels in pdms and design of a microfluidic adhesive. *Langmuir* **2006**, *22*, 5. [CrossRef] [PubMed]

56. Ardeleanu, M.N.; Mihai, S.; Vidu, R.; Diaconu, E.M.; Popescu, I.N. Design of microfluidic device and measurements of mpwm for single cell /particle manipulation. *Sci. Bull. Valahia Univ. Mater. Mech.* **2019**, *17*, 39–43. [CrossRef]

57. Dahlin, A.P.; Wetterhall, M.; Liljegren, G.; Bergstrom, S.K.; Andren, P.; Nyholm, L.; Markides, K.E.; Bergquist, J. Capillary electrophoresis coupled to mass spectrometry from a polymer modified poly(dimethylsiloxane) microchip with an integrated graphite electrospray tip. *Analyst* **2005**, *130*, 193–199. [CrossRef]

58. Richter, M.; Woias, P.; Weib, D. Microchannels for applications in liquid dosing and flow-rate measurement. *Sens. Actuators A* **1997**, *62*, 4. [CrossRef]

59. Ainla, A.; Gozen, I.; Orwar, O.; Jesorka, A. A microfluidic diluter based on pulse width flow modulation. *Anal. Chem.* **2009**, *81*. [CrossRef]

60. Unger, M.; Lee, S.S.; Peter, M.; Koeppl, H. Pulse width modulation of liquid flows: Towards dynamic control of cell microenvironments. In Proceedings of the 15th International Conference on Miniaturized Systems for Chemistry and Life Sciences, Seattle, Washington, USA, 2–6 October2011; p. 3.

61. Ardeleanu, M.; Ionita, M.; Ivan, A.; Gurgu, V. Innovative bidimensional absolute transducer based on video detection for positioning into micro assembly processes. *Appl. Mech. Mater.* **2014**, *658*, 535–540. [CrossRef]

62. Experimental Models. Available online: http://www.celteh.ro/proiecte/proiecte-2/ (accessed on 14 September 2019).

The Impact of Technology on People with Autism Spectrum Disorder: A Systematic Literature Review

Katherine Valencia *, Cristian Rusu *, Daniela Quiñones and Erick Jamet

School of computer Engineering, Pontificia Universidad Católica de Valparaíso, Valparaíso 2340000, Chile; daniela.quinones@pucv.cl (D.Q.); erick.jamet@gmail.com (E.J.)

* Correspondence: katherinevalencia25@gmail.com (K.V.); cristian.rusu@pucv.cl (C.R.); Tel.: +56-942-954-648 (K.V.); +56-988-244-506 (C.R.)

Abstract: People with autism spectrum disorder (ASD) tend to enjoy themselves and be engaged when interacting with computers, as these interactions occur in a safe and trustworthy environment. In this paper, we present a systematic literature review on the state of the research on the use of technology to teach people with ASD. We reviewed 94 studies that show how the use of technology in educational contexts helps people with ASD develop several skills, how these approaches consider aspects of user experience, usability and accessibility, and how game elements are used to enrich learning environments. This systematic literature review shows that the development and evaluation of systems and applications for users with ASD is very promising. The use of technological advancements such as virtual agents, artificial intelligence, virtual reality, and augmented reality undoubtedly provides a comfortable environment that promotes constant learning for people with ASD.

Keywords: user experience; accessibility; autism spectrum disorder; game-based learning; systematic literature review

1. Introduction

Currently, autism spectrum disorder (ASD) affects a significant number of people who have difficulties with communication and socialization, which results in complexities for their learning. Studies have examined the use of technology and computer-based interventions to teach people with ASD language and social skills [1]. Specifically, students on the autism spectrum enjoy playing games, which provides a safe environment [2]. Thus, we reviewed the existing literature about the relationship between technology, games, user experience, accessibility, and the education and skill development of people with ASD. This article is organized as follows: Section 2 presents the theoretical background, Section 3 describes the research methodology, Section 4 analyzes the results obtained, and finally, Section 5 highlights the conclusions and recommendations for future work.

2. Theoretical Background

2.1. Autism Spectrum Disorder

Asperger's syndrome was defined in 1944 by Hans Asperger [3]. The fifth edition of the Diagnostic and Statistical Manual of Mental Disorders (DSM-5) [4] defines autism spectrum disorder (ASD) as a condition characterized by deficits in two core domains: (1) social communication and social interaction and (2) restricted repetitive patterns of behavior, interests, and activities. Since 2013, the DSM-5 has recognized Asperger's disorder, childhood disintegrative disorder, Rett's disorder, and several other related disorders, as part of ASD. However, many studies still use the Asperger's syndrome and ASD almost interchangeably.

In a study carried out by the National Institute of Health (NIH) of the USA [5] published in June 2018, it was estimated that 2.41% of children in the United States of America have an autism spectrum disorder. This shows an increase of 0.94% compared to 2010.

2.2. User Experience

The international standard on ergonomics of human system interaction, ISO 9241-210 [6], defines user experience as "user's perceptions and responses that result from the use and/or anticipated use of a system, product or service". In other words, the user experience is the degree of "satisfaction" that the end user has with the system or service after using it, that is based on each of the interactions that he or she has.

According to Peter Morville [7], user experience is meaningful and valuable when a product, service or system is useful (that is, its content is original and satisfies a need), usable (the product is easy to use), desirable (the image, identity, brand, and other design elements produce positive emotions towards the product), locatable (the content is accessible to people with disabilities), credible (users have confidence in the product), and valuable (an added value is generated from the product).

2.3. Accessibility

The international standard on ergonomics of human system interaction, ISO 9241-171 [8] defines accessibility as the "extent to which products, systems, services, environments, and facilities can be used by people from a population with the widest range of user needs, characteristics and capabilities to achieve identified goals in identified contexts of use". In other words, accessibility is the condition that environments, services, processes, and objects (everything that involves an interaction) must meet, which must be understandable and usable by the broadest range of people, regardless of their capabilities.

2.4. Game-Based Learning

Games that use technology are widely used to teach people conceptual knowledge and skills. There are different implementations of such games, such as serious games, gamification, and e-learning.

2.4.1. Serious Games

Serious games are games whose main objective is not fun or entertainment but the learning or practice of skills. In 1970, Clark Abt [9] defined this concept as follows in his book called "Serious Games"—"games that have an explicit and carefully thought-out educational purpose and are not intended to be played primarily for amusement. This does not mean that serious games are not, or should not be, entertaining".

2.4.2. Gamification

The concept of gamification was developed in 2003, and its use became widespread in 2010 through the work of multiple professionals. Gamification is formally defined as "the use of game elements and game design techniques in nongame contexts" [10]. When we talk about gamification, we tend to interpret it as a methodology where the purpose is to provide rewards to users to inspire personal and collective commitment, but this interpretation is very far from reality. Many authors maintain that the success of a gamified system or process lies in good design and adequate feedback, among many other factors. Other authors have supported this argument: for example, Kapp [2] stated, "Do not think of gamification as only the use of badges, rewards, and points. Instead, think of the engaging elements of why people play games—it is not just for the points—its [sic] for the sense of engagement, immediate feedback, and the success of striving against a challenge and overcoming it".

2.4.3. E-Learning

The term "e-Learning" comes from the abbreviation of "electronic learning". Khan [11] defined e-Learning as "a hypermedia instructional program that uses the attributes and resources of the Internet to create meaningful learning environments." That is, e-Learning refers to online teaching and learning through the Internet and technology.

2.5. Game Elements

Game elements are the components that make up a game to create an attractive experience for players. Werbach [10] described 25 such game elements. For the purpose of our study, we identify the relevant game elements are as follows:

- Narrative: Telling of a coherent story.
- Progression: Player growth and development.
- Challenges: Tasks that require an effort to perform.
- Competition: Players or groups that win or lose.
- Rewards: Benefits granted after a certain action.
- Feedback: Information about how the player is performing.
- Avatars: Visual representation of a player character.
- Collections: Set of items that can be accumulated.
- Levels: Steps defined in the progression of a player.
- Leaderboard: Visual representation of the player's progression with respect to others.
- Points: Numerical representation of the player's progression.
- Achievements: Accomplishment of defined objectives.
- Teams: Group of players who work together to achieve a common goal.

3. Research Methodology

This systematic literature review was carried out following the process proposed by Kitchenham [12]. Kitchenham outlined three fundamental phases for conducting a review of the literature: (1) planning the review, which includes creating the research questions and reviewing the protocol; (2) conducting the review, which includes the review, the selection and quality of studies, data extraction and data synthesis; and (3) publicizing the results after the review. Next, we detail the process followed for this document.

3.1. Research Questions

To cover every topic of interest in this systematic literature review, we formulated three research questions. These questions consider relevant and general aspects important for comprehending the concepts that we think are important for this study. These questions can be seen in Table 1.

Table 1. Research questions for the systematic literature review.

ID	Research Question (RQ)
RQ1	In what way does the use of technology contribute to the education of people with autism spectrum disorder?
RQ2	Which user experience and accessibility elements/methods are considered when analyzing the impact of technology on people with autism spectrum disorder?
RQ3	Which game elements are considered when using gamification or serious games in the education of people with autism spectrum disorder?

3.2. Data Sources and Search Strategies

To conduct this systematic literature review, we searched for scientific papers on five databases: IEEE Xplore Digital Library, ACM Digital Library, Science Direct, Scopus, and Web of Science. For

these sources, we considered only documents that were relevant in computer-related categories, such as technology, engineering and computer science, excluding categories related to medicine or chemistry. Additionally, we selected articles published during the last 10 years, between January 2009 and June 2019.

3.3. Article Selection

Once we chose the databases to search, we determined the specific search strings to find articles to answer the research questions and defined the exclusion and inclusion criteria to refine and filter the articles found.

3.3.1. Search Strings

We formulated the search strings based on the relevant topics to our systematic literature review. We determined a set of specific keywords to use in our queries, i.e., "Autism Spectrum Disorder", "Accessibility", "User Experience", "Gamification", "Serious Games", and "Game Elements" that would be useful to answer our research questions.

These strings were focused on finding studies that analyzed or experimented with the use of games with people with ASD, considering aspects such as the user experience, accessibility, and game elements. In Table 2, we present the specific search strings that were used in the selected databases.

Table 2. Search strings.

ID	Search Strings
SS1	("Autism spectrum disorders" OR ASD OR Autism) AND (Accessibility OR "User experience") AND ("game elements" OR gamification OR "Serious game" OR "game-based learning")
SS2	("Autism spectrum disorders" OR ASD OR Autism) AND (Accessibility OR "User experience")
SS3	("Autism spectrum disorders" OR ASD OR Autism) AND ("game elements" OR gamification OR "Serious game" OR "game-based learning")

3.3.2. Study Selection Criteria

To answer the research questions based on the selected articles and develop a general knowledge of the concepts that we were working with, we included the conditions listed in Table 3.

Table 3. Inclusion criteria.

ID	Inclusion Criteria
IN1	Studies published over the last 10 years, between January 2009 and June 2019.
IN2	Journal articles and conference papers.
IN3	Studies with a focus on autism spectrum disorder.
IN4	Studies related to the usage of technology.
IN5	Studies performed in an educative context or focused on teaching.

The types of papers presented in Table 4 were excluded.

Table 4. Exclusion criteria.

ID	Exclusion Criteria
EX1	Studies with an exclusive medical focus or a focus on the diagnosis of autism spectrum disorder.
EX2	Studies that do not directly aim to help people with autism spectrum disorder but rather the people who work with them.
EX3	Studies that consider user experience and accessibility in contexts that do not involve the use of technology.

3.4. Document Selection

Applying the selection criteria, we gathered a total of 94 articles. Figure 1 shows the general process flow of the search and study selection for this review, detailing the inclusion and exclusion criteria applied in each step.
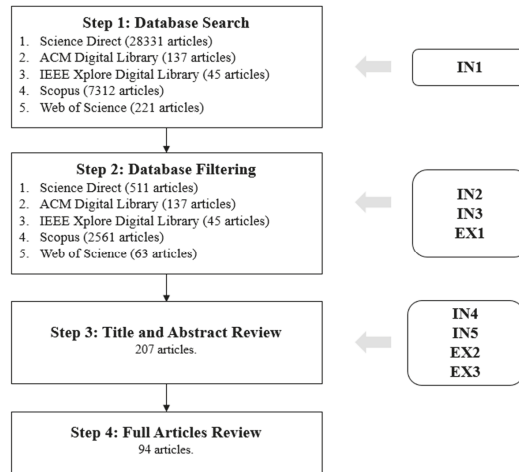
Figure 1. Flow chart with the results of the article selection process.

3.5. Data Synthesis

After the search, we extracted the information from each of the 94 studies, summarizing and tabulating the information based on different metrics, such as the year published, document type and paper category. In the following steps, we detail each of the metrics.

3.5.1. Year of Publication

As detailed above in the inclusion criteria section, we considered studies published during the last 10 years, between 2009 and 2019. As shown in Figure 2, we plotted the number of studies that were found that were published between 2009 and 2018, and we observed an increase in publication on this topic over this period. The studies found in 2019 are not presented in this plot because it would have been misleading to show incomplete data, as this review was finished in June 2019. Seventeen studies published in 2019 were found (almost equal to the number of publications in 2018), which led us to believe that this number will undoubtedly increase significantly during the remaining months of 2019.
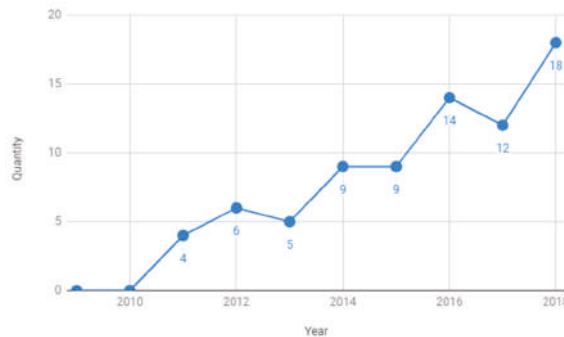
Figure 2. Year of publication.

3.5.2. Document Type

We analyzed the origin of the studies reviewed and determined whether they were conference proceedings or had been submitted to a scientific journal. Figure 3 shows a relative balance between the number of papers that were published as conference proceedings and in journals.

Document Type

44.1%

55.9%

● Journal Paper ● Conference Paper
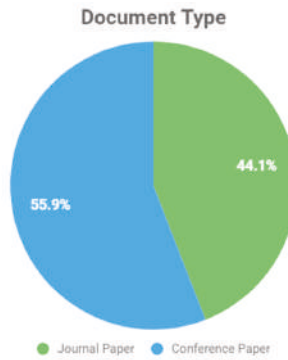
Figure 3. Document Type.

3.5.3. Document Categories

The studies were categorized as follows:

- Review: An updated summary of a particular topic is provided.
- Case Study: A solution is given to a presented problem based on a tool, methodology, etc.
- Empirical Data: A context or situation is analyzed based on historical data.

Figure 4 shows that 74.5% of the studies analyzed were case studies. It is believed that this is because the researchers were focused mainly on conducting investigations and accomplishing their study objectives, such as teaching conceptual skills.

Document Categories

11.7%   13.8%

74.5%

● Review ● Case Study ● Empirical Data
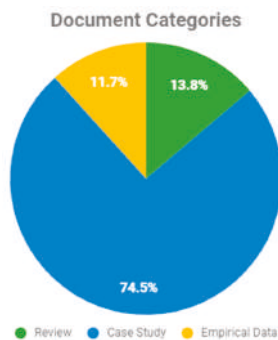
Figure 4. Document Categories.

4. Results and Discussion

After applying each of the filters described in the "Study Selection Criteria" section, as shown in Figure 1, a total of 94 studies were obtained. These studies were analyzed under different metrics, as seen in the "Data Synthesis" section. Based on our review of these studies, we now answer our research questions, considering those studies that are relevant to the specific context of each question.

RQ1. In what way does the use of technology contribute to the education of people with autism spectrum disorder?

As mentioned in the previous sections, ASD is a condition that is categorized as a disability due to the cognitive disorders that people with ASD face [13]. Several studies showed that most people with autism show a natural affinity for technology and a good disposition for using technology and learning through the use of computers [14]. This is because the environment and context that these experiences provide are predictable and structured, which helps people with ASD to maintain their routines and repetitive behaviors without affecting their comfort [15].

Several studies proposed the use of modern technologies to help teach skills to people with ASD. Some interesting examples of new technological approaches are the use of sensors, virtual reality, virtual agents, augmented reality, geolocation, and Kinect, as presented in the following studies. Wojciechowski et al. [15] developed a mobile application that, in conjunction with the use of Estimote Beacon sensors to identify objects, supports children with ASD in pronouncing new words and identifying their meanings. Lorenzo et al. [16] proposed an application that uses virtual reality and robots with cameras to detect children's emotions, adapt system interactions and thus develop social skills in students with autism spectrum disorder. Bernardini et al. [17] presented ECHOES, which is a serious game that focuses on the development of activities to promote social communication in children with ASD using an autonomous virtual agent that acts as a companion for children during their interactions with the system. Sorce et al. [18] developed an exploratory study to evaluate the effectiveness of the use of Kinect as a tool to allow people with ASD to explore works of art in a touchless virtual environment and assess whether this generates greater interest in them. Escobedo et al. [19] presented the Mobile Social Compass (MOSOCO) application, which makes use of augmented reality through a mobile device camera to include game elements in real social situations with the aim of developing social skills in children with ASD. Silva et al. [20] presented a serious game that, through geolocation, virtual reality and augmented reality, creates a virtual environment with 3D virtual monsters positioned all over the world that aim to teach children with ASD relevant educational content, such as vocabulary.

In addition to examining the studies from a technological perspective, we categorized the 94 studies based on the following learning topics with the goal of understanding the contribution of technology to education for people with ASD in terms of the specific skills that they focus on teaching: Conceptual Skills (subtopics: Language, Money, Colors, Mathematics, Programming, and Science), Practical Skills (subtopics: Health, Daily Life, and Transportation), Social Skills (subtopics: Communication, Emotions, and Interpersonal Relationships) and General Skills (subtopic: General). Table 5 shows the percentage of studies for each of the topics and subtopics, and in the same way, Table A1 (available in Appendix A) details each of the topics and subtopics according to which the articles were categorized. The results obtained after categorizing the studies are presented in the following sections.

Table 5. Learning Topic.

Topic	Subtopic	Percentage by Subtopic	Percentage by Topic
Conceptual Skills	Language	14.89%	25.53%
	Money	1.06%	
	Colors	1.06%	
	Math	5.32%	
	Programming	2.13%	
	Science	1.06%	
Practical Skills	Healthcare	2.13%	8.51%
	Daily Living	3.19%	
	Transportation	3.19%	
Social Skills	Communication	9.57%	36.17%
	Emotions	12.77%	
	Interpersonal Relationships	13.83%	
General Skills	General	29.79%	29.79%

4.1. Conceptual Skills

First, 25.53% of the studies focused on analyzing and fostering skills within the range of Conceptual Skills. Studies in the Language subcategory focused on promoting the learning of expressions, thoughts and feelings through words. Examples of this include studies [13,21]. Arciuli and Bailey [13] analyzed a small group of children with ASD that were literate using the ABRACADABRA application and observed significant improvements in reading accuracy in participants who interacted with the system but not in children who did not use the application. For the children who did not use the application, their lack of improvement was believed to be due to their lack of socialization aspects that children must exhibit when interacting with a teacher to develop reading ability. Khowaja et al. [21] developed a prototype of a serious game for children with ASD to learn vocabulary. The effectiveness of the game was assessed through the comparison of children's performance at the beginning of the intervention, after the use of the prototype and 1–2 weeks after the use of the prototype, which enabled the researchers to track the improvement in the children's vocabulary.

Another subcategory of Conceptual Skills is the Money subcategory and only one study [22] was assigned to this subcategory. Caria et al. presented the design of a game that helps people with autism spectrum disorder acquire skills to help them understand the concept of money and its applications in real life, which was tested by obtaining positive and promising results.

In addition, like the Money subcategory, the Colors subcategory also included only one study, [23]. In this study, based on cognitive theories, Tuğbagül et al. developed a computer interface for students with ASD and mild mental disability that used their preferred colors and helped them maintain their concentration.

Additionally, the studies in the Math subcategory aimed to develop skills related to numbers. Examples of studies in this subcategory are [24,25]. Tashnim et al. [24] developed the Play and Learn Number (PLaN) application, which teaches arithmetic and calculus to children who have ASD and helps children memorize and recognize numbers (in or not in sequences) through animated images. Muñoz-Soto [25] developed an application to support professionals in teaching functional mathematics and calculus to children with ASD. Through tests, it was possible to demonstrate that this application promotes the development of mathematical skills. However, it was suggested that the application should be tested by more users and in different institutions.

The Programming subcategory included studies that aimed to develop skills related to computational programming, for example, to design and order actions and commands. Only two studies were assigned to this subcategory, i.e., [26,27]. Eiselt and Carter [26] planned and conducted programming classes through Scratch for children with ASD with the aim of developing their technical and social skills. Despite their efforts, no real evidence of an increase in students' social learning or behavior was found. However, while the students did not develop social skills as expected, the authors suggested that the students knew more about programming after the experiment since at the beginning, they did not have any notion of programming, but after the experiment, they could read and write processing programs. Schmidt and Beck [27] proposed a learning intervention based on digital games for young people with ASD to develop their social skills as they worked on teams to solve introductory computer programming problems with virtual and programmable robots. According to the authors, this intervention has the potential to help participants develop social skills, however, because this study was only concerned with the initial stages of development, there was no analysis of the data, so conclusions regarding cognitive skills could not be made with certainty.

Finally, the studies in the Science subcategory investigated and interpreted natural, social, and artificial phenomena. For this subcategory, we found only one study [28], in which Eder et al. developed a mobile game application as a complementary learning material to teach children with ASD parts of the human body. After the intervention, it was observed that the application was very useful for teaching and that the motivation levels of the participants increased significantly.

4.2. Practical Skills

Second, the Practical Skills category included only 8.51% of the identified studies and was subdivided into several subcategories. First, the Healthcare subcategory concerned teaching about the health care that people should have. An example of a study in this subcategory is [29]. De Urturi et al. [29] developed a system consisting of a set of serious games aimed at teaching first aid (such as what to do in certain situations and basic knowledge about medical care and medical specialties) to people with ASD. Because the application was still in development, only partial results were available, so to determine if these results were promising, the authors administered a simple questionnaire to the participants, as they obtained positive results, they decided to continue developing the project.

Another subcategory of Practical Skills is the Daily Living subcategory. The studies in this subcategory focused on building knowledge about the development of daily recurring activities, and examples are [30,31]. Pérez-Fuster et al. [30] analyzed the impact of an intervention with digital technology (DT) compared to that of a treatment-as-usual (TAU) intervention on adults with ASD. The DT intervention sought to improve daily life skills, such as washing dishes and washing clothes. The results showed that the DT intervention significantly improved the daily life skills of the participants and was more effective than the TAU intervention. Santarosa and Conforto [31] presented a tablet application for children with ASD and children with intellectual disability (ID) that seeks to teach and develop routines in the classroom and verbal communication by directly involving teachers and assistants in schools. Children with ASD successfully adapted to the application, and their socioadaptive behaviors both in the classroom and related to verbal communication improved greatly. On the other hand, children with ID did not achieve autonomous use of the application, and they only had improvements in nonverbal classroom routines.

The final subcategory within the Practical Skills category is the Transportation subcategory. The studies in this category were concerned with teaching the necessary knowledge that individuals need to be able to transport themselves effectively. Some examples of this are found in [32,33]. McKissick et al. [32] investigated the impact of a computer instruction package to teach map-reading skills to three elementary students with ASD. Very promising results were obtained for interventions that used technology with children with ASD, such as increased levels of learning and improved learning habits among students. De Los Rios [33] proposed a draft of a study to evaluate platforms and interfaces that help users transport themselves, such as Google Maps or Apple Maps with eye tracking. They compared these platforms and interfaces with a proposed system that would provide a more personalized environment that is adapted and accessible to the needs of people with ASD.

4.3. Social Skills

Third, the Social Skills category included 36.17% of the total resulting studies and was subdivided into three subcategories. The studies in the first subcategory, Communication, focused on the development of skills such as exchanging information between two or more individuals and examples from this subcategory are found in [34,35]. Milne et al. [34] investigated the use of autonomous virtual humans (self-directed) to teach and facilitate the practice of basic social skills in greetings, conversation, listening, and shifts in conversation to people with ASD. The results were positive, as users increased their knowledge and development of social skills. In addition, it has been indicated that this approach was well received by participants and caregivers. Ribeiro and Barbosa [35] developed a game called ComFiM, which aims to encourage communication between people with severe degrees of autism. The game was evaluated based on the perceptions of the interlocutors of each player and the communication intentions observed between the players to collaborate with each other and the results showed that the application positively influenced the communication intentions of the players.

The Emotions subcategory included studies that examined the development of skills such as the identification of facial emotions. Some studies from this subcategory are [36,37]. Romero [36] carried out a computer-based intervention to teach the recognition of emotions to students with communication and social skill deficits. All participants showed improvements when assessing and

recognizing emotions on faces, but it was suggested that the effectiveness of the intervention should be tested in a larger population. Christinaki et al. [37] presented a serious game with a natural user interface (NUI) interaction that aims to teach young children with ASD to recognize and understand different facial emotions. The authors concluded that technological interventions with NUI improve the learning process and indicated that the emotional state of the players is directly related to their learning skills.

Additionally, the studies in the Interpersonal relationships subcategory emphasized individuals' development of relationships. Some of the studies that were assigned to this subcategory are [38,39]. Boyd et al. [38] described how collaborative assistance technologies, such as the Zody collaborative game, can be used to facilitate social relationships in children with ASD. They discussed how design can foster three levels of social relationship, i.e., membership, partnership, and friendship, even without the help of adults. The results indicate that collaborative technologies provide support for the development of social skills at different levels of intimacy between players without a mediator during the intervention. Hourcade et al. [39] conducted an intervention with multitouch tablets with children with ASD to promote their social skills and help them develop their creativity, alter their interests, and be able to understand emotions. The result of the intervention was that it increased pro-social behaviors, such as collaboration, coordination, and interest in social activities, in children with ASD.

4.4. General Skills

Finally, the General Skills category included 29.79% of the studies. As this category referred to a range of topics, we defined only one subcategory, the General subcategory; some example studies are [40,41]. Backman et al. [40] investigated a method of evaluating children on the autism spectrum through computer games, which provide an objective, motivating, and safe evaluation of the participants. Although more research was recommended, the results showed that computer games have great potential in special education as an evaluation tool to clarify the difficulties associated with ASD. Hulusic and Pistoljevic [41] presented the initial development process of the LeFCA framework, which was used to teach children with ASD basic skills and concepts. LeFCA consists of four games that focus on developing basic skills (such as labeling, pointing and pairing in reference to visual and auditory stimuli) necessary for learning. Each of the participants was constantly motivated to play, and the skills learned could be extrapolated to new media or environments without the need for any training.

After reviewing all the studies and classifying them based on their learning topics, as shown in Table 5, we can see that there are a few studies that used modern and/or complex technologies, such as virtual reality or sensors. These technological approaches are interesting examples of how this area is developing in innovative ways.

Notably, most of the studies focused on teaching Social Skills, such as Emotions (12.77%), Communication (9.57%), and Language (14.89%), which are the most important areas that people with ASD have difficulties with.

RQ2. Which user experience and accessibility elements/methods are considered when analyzing the impact of technology on people with autism spectrum disorder?

Although many of the studies suggested that accessibility and user experience are fundamental concepts for interventions with people who have ASD, these aspects were not treated with the importance that they should be.

Several of the studies that were reviewed from the pool of articles reported having used and/or considered user experience and/or accessibility, but most of these studies did not provide enough detail about the use of these concepts. Table A2 (available in Appendix A) shows a total of 23 studies that in some way used and/or provided "detail" on the use of these concepts in their research. We can see that the most recurrent terms used in the studies were user experience, usability, and accessibility.

For instance, many of the studies claimed to have focused on accessibility when developing touchscreen applications, such as [23,24,39,42,43]. However, the authors' affirmations were not supported by empirical evidence or other details.

On the other hand, other studies such as [27,33] proposed the evaluation of the usability and/or user experience of the systems in future works. De Los Rios [33] suggested evaluating the usability of the application based on eye tracking. Schmidt and Beck [27] proposed the use of eye-tracking, electroencephalogram (EEG) scanning, and focus group interviews to evaluate the usability of the system.

Studies such as [28,40,42,44,45] aimed to evaluate usability and user experience based on post-intervention questionnaires with users, as well as with the people around them (such as their teachers or parents). These studies worked with control and test groups of children with and without ASD. Few studies indicated the number of subjects involved in the experiments: 14 in [42], 11 in [28], and 30 in [40]. Forty teachers were also involved in the experiment described in [42]. In addition to the questionnaires, Santarosa and Conforto [45] and Backman et al. [40] carried out methods such as focus groups in their interventions to be able to evaluate the usability and user experience.

Additionally, in studies such as those by Khowaja and Salim [46] and Naziatul et al. [47], the proposed systems were evaluated based on heuristic evaluations. In these studies, the authors adapted the heuristics proposed by Nielsen [48] to the contexts of their interventions. In both cases, three experienced evaluators assessed the system usability.

In addition, in the study by Vallefuoco et al. [49], a usability user test was carried out with 10 children aged between 5 and 12 under the methodology proposed by Moreno Ger [50] to evaluate the system, its usability, and the effectiveness of the customized elements developed to fulfill the objective of the study.

Finally, Caria et al. [22] worked with children with ASD between 16 and 22 years old, and Almeida et al. [51] worked with 40 children between 3 and 13 using the "System Usability Scale" (SUS) to evaluate the usability of their applications.

As we can see, few studies provided details about how they used concepts such as usability, user experience and accessibility, how these concepts were evaluated, and what kind of users were involved in their experiments. We think that it is important to consider all these concepts when developing new solutions.

RQ3. Which game elements are considered when using gamification or serious games in the education of people with autism spectrum disorder?

Several of the identified studies described the use of game-based learning (mostly serious games), but they did not specify and/or provide details about the elements of the games that were used. However, a significant number of studies explicitly presented some game elements that allow these systems to be more attractive and engaging for users. In Table A3 (available in Appendix A), we can see the game elements used in the studies, where the most frequent elements were points, levels, and rewards. Brief definitions of the game elements, as presented by Werbach [10], are presented in Section 2.5.

For example, Vallefuoco et al. [49] analyzed a serious game that focused on improving math skills in children with ASD and for which one of the main elements was feedback. Likewise, Sorce et al. [18] used avatars in an application with Kinect to foster the interest of participants with ASD in digital representations of works of art, paintings, and sculptures. In addition, Romero [36] carried out a computer-based intervention with intrinsic rewards and points to teach the recognition of emotions. Similarly, Chen et al. [52] designed and developed a computer game with points and rewards to develop and evaluate emotional skills and conceptual comprehension skills (such as recognizing fruits) in children with autism spectrum disorder. Additionally, Harrold et al. [53] added to the concepts described above through the use of levels in CopyMe, a serious game for iPad, which provides children with ASD with a means to learn emotions through observation and mimics. In the same way, Sturm et al. [42] used stories in addition to rewards, points, and levels in a game with Kinect technology that aims to promote the recognition of emotions and encourage collaboration between people with ASD and their peers. Finally, Boyd et al. [38] described the use of Zody, as a collaborative

assistance application, to teach social relations to children with ASD through the use of collaboration, points, levels, and rewards.

Most of the studies considered in this review did not explicitly identify which game elements they used in the development of their solutions. Even when they did, they did not give enough details on the effectiveness of the specific game elements. Although some authors claimed that their users were more engaged with the solutions they proposed, they did not provide empirical evidence to support such claims.

5. Conclusions and Future Work

Our systematic literature review focuses on analyzing the impact of technology on people with autism spectrum disorder based on research published during the last 10 years and available on the relevant scientific databases. The analysis shows an increase in the papers published on this topic over the years, which indicates an increasing research interest in the area. Interestingly, the highest percentage of the papers presented are case studies (74%). The studies were categorized into four categories: Conceptual Skills, Practical Skills, Social Skills, and General Skills. Studies that focus on Social Skills are predominant (36.17%).

Regarding RQ1, we observe that new research has focused on supporting children with ASD by using technologies such as virtual reality, augmented reality, virtual agents, sensors, and geolocation through educational games. These studies emphasize teaching different skills to people with ASD in educational contexts, with a higher percentage of studies focusing on Social Skills (36.17%) than on Conceptual (25.53%) or Practical Skills (8.51%), which shows a need for more research and development of new solutions for teaching such important topics. Exploring these alternatives and expanding the technological solutions to teach skills to people with ASD seem to be promising research topics.

The results related to RQ2 show that several studies mention that aspects such as user experience, usability, and accessibility are crucial when working with people with ASD. However, these aspects are usually not considered or validated in detail. Although the use of new technologies, such as EEG scanning and eye tracking in [27], to evaluate the usability of their systems is indeed interesting, studies have shown that brain activity may be negatively correlated with the Asperger questionnaire [54] and may be weaker for individuals with ASD when observing other people's actions [55]. Future studies should be careful with the use of such technological approaches, as brain activity may be misleading when working with people with ASD, especially in tasks that require recognizing emotions from facial expressions or movements. We believe that user experience is important and that future studies should consider accessibility and usability tests to ensure positive experiences and comfort with the use of their solutions, as there is a lack of research that applies these concepts correctly and that provides details about the user groups that participate in interventions.

Regarding RQ3, we have observed in the literature that game elements are a good way to engage users with learning and enhance the effectiveness of teaching approaches for people with ASD, but our findings show that there is a lack of evidence about the effect of the use of game elements in gamification, e-learning, and serious game solutions. We believe that future studies should consider and validate the use of game elements. Werbach [10] highlighted that game elements are effective, have a positive relation with users' engagement, and have been widely used with promising results.

We think that the use of technologies in conjunction with suitable game elements and user experience and accessibility design and evaluation are promising research topics related to teaching people with ASD.

Author Contributions: Conceptualization, K.V.; methodology, K.V. and C.R.; validation, K.V., C.R., D.Q., and E.J.; formal analysis, K.V. and C.R.; investigation, K.V.; resources, K.V. and E.J.; writing—original draft preparation, K.V.; writing—review and editing, C.R., D.Q., and E.J.; visualization, K.V.; supervision, C.R.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Identified Learning Topics.

Topic	Subtopic	Authors
Conceptual Skills	Language	Arciuli. J, Bailey. B. (2019) [13]
		Lin. C, Chang. S, Liou. W, Tsai. Y. (2013) [14]
		Wojciechowski. A., Al-Musawi. R. (2017) [15]
		Magaton. H, Bim. Silvia. (2017) [56]
		Mendonça. V, Coheur. L, Sardinha. A. (2015) [44]
		Wilson. C, Brereton. M, Ploderer. B, Sitbon. L. (2018) [57]
		Alvarado. C, Muñoz. R, Villarroel. R, et al. (2017) [58]
		Rasche. N, Pourcho. J, Wei. S, Qian. C.Z., Chen. V.Y. (2013) [59]
		Cunha. R.M., Barbosa. S.D.J. (2012) [60]
		Gomez. J, Jaccheri. L, Torrado. J.C, Montoro. G. (2018) [61]
		Khowaja. K, Salim. S.S, Al-Thani. D. (2018) [62]
		Khowaja. K, Salim. SS. (2019) [46]
		Khowaja. K, Al-Thani. D, Salim. SS. (2018) [21]
		Frutos. M, Bustos. I, Zapirain. B.G, Zorrilla. A.M. (2011) [63]
	Money	Caria. S, Paternò. F, Santoro. C, Semucci. V. (2018) [22]
	Colors	Tuğbagül Altan. N, Göktürk. M. (2018) [23]
	Math	Aziz. N.S.A, Ahmad. W.F.W, Hashim. A.S. (2016) [64]
		Naziatul. A.A, Wan. W.A, Ahmad. H. (2016) [47]
		Tashnim. A, Nowshin. S, Akter. F, Das. A.K. (2018) [24]
		Muñoz-Soto. R, Becerra. C, Noël. R., et al. (2016) [25]
		Aziz. N.S.A, Ahmad. W.F.W, Zulkifli. N.J.B. (2015) [65]
	Programming	Eiselt. K, Carter. P. (2019) [26]
		Schmidt. M, Beck. D. (2016) [27]
	Science	Eder. MS, Díaz. JML, Madela. JRS, Mag-usara. MU, Sabellano. DDM. (2016) [28]
Practical Skills	Healthcare	De Urturi. ZS, Méndez. A, García. B. (2011) [29]
		De Urturi. ZS, Méndez. A, García. B. (2012) [66]
	Daily Living	Pérez-Fuster. P, Sevilla. J, Herrera. G. (2019) [30]
		Santarosa. L.M.C, Conforto. D. (2016) [31]
		Lee. D, Frey. G, Cheng. A, Shih. PC. (2018) [67]
	Transportation	McKissick. B, Spooner. F, Wood. C.L, Diegelmann. K.M. (2013) [32]
		De Los Rios. P. C. (2018) [33]
		Rector. K. (2018) [68]

Table A1. *Cont.*

Topic	Subtopic	Authors
Social Skills	Communication	Milne. M, Raghavendra. P, Leibbrandt. R, Powers. D.M.W. (2018) [34]
		Bringas. J.A.S, León. M.A.C, Cota. I.E, Carrillo. A.L. (2016) [69]
		Hussain. A, Abdullah. A, Husni. H, Mkpojiogu. E.O.C. (2016) [70]
		Bernardini. S, Porayska-Pomsta. K, Smith. T.J. (2014) [17]
		Tang H.H, Jheng. C.M, Chien. M.E, Lin. N.M, Chen. M.Y. (2013) [71]
		El-Seoud. M.S.A, Karkar. A.G, Al Ja'am. J.M, Karam. O.H. (2015) [72]
		Baldassarri. S, Passerino. L, Ramis. S, Riquelme. I, Perales. FJ. (2018) [73]
		Cabielles-Hernández. D, Pérez-Pérez. J.-R, Paule-Ruiz. M, Fernández-Fernández. S. (2017) [74]
		Ribeiro. PC, Fox. AB. (2014) [35]
	Emotions	Romero. N.I. (2017) [36]
		Sturm. D, Kholodovsky. M, Arab. R, et al. (2019) [42]
		Papoutsi. C, Drigas. A, Skianis. C. (2018) [75]
		Lorenzo. G, Lledó. A, Pomares. J, Roig. R. (2016) [16]
		Leijdekkers. P, Gay. V, Wong. F. (2013) [76]
		Tinnunnem. S.G, Shah. G, Lahiri. U. (2017) [77]
		Harrold. N, Tan. C.T, Rosser. D, Leong. T.W. (2014) [53]
		Hyun. P.J, Abirached. B, Zhang. Y. (2012) [78]
		Castillo, T.A, Pérez de Celis. C, et al. (2016) [79]
		Almeida. LM, Silva. DPD, Theodório. DP, et al. (2019) [51]
		Cisnero. A.Q, Juárez-Ramírez. R., Figueroa. A.M. (2016) [80]
		Christinaki. E, Vidakis. N, Triantafyllidis. G. (2014) [37]
	Interpersonal Relationships	DiGennaro. F.D, Hyman. S.R, Hirst. J.M. (2011) [81]
		Boyd. L.E, Ringland. K.E, Haimson. O.L, Fernandez. H, Bistarkey. M, Hayes. G.R. (2015) [38]
		Muñoz. R, Morales. C, Villarroel. R, Quezada. A, De Albuquerque. V.H.C. (2019) [82]
		Rapp. A, Cena. F, Castald., R, Keller. R, Tirassa. M. (2018) [83]
		Sturm. D, Gillespie-Lynch. K, Kholodovsky. M. (2017) [84]
		Escobedo. L, Nguyen. D.H, Boyd. L.A, et al. (2012) [19]
		Hourcade. J.P, Bullock-Rest. N.E, Hansen. T.E. (2012) [39]
		Grossard. C, Grynspan. O, Serret. S, Jouen. A.L, Cohen. D. (2017) [85]
		Hughes. D.E, Vasquez E, Nicsinger. E. (2016) [86]
		Hani. H, Abu-Wandi. R. (2015) [87]
		Dehkordi. SR, Rias. RM. (2015) [88]
		Aziz. MZA, Abdullah. SAC, Adnan. SFS, Mazalan. L. (2014) [89]
		Jeekratok. K, Chanchalor. S, Murphy. E. (2014) [90]

Table A1. *Cont.*

Topic	Subtopic	Authors
General Skills	General	Backman. A, Mellblom. A, Norman-Claesson. E, Keith-Bodros. G, Frostvittra. M, Bölte. S, Hirvikoski. T. (2018) [40]
		Hong. E.R, Gong L.Y, Ninci. J, Morin. K, L.Davis. J, Kawaminami. S, Shi Y.G, Noro. F. (2017) [91]
		Still. K, May. R.J, Rehfeldt. R.A, Whelan R, Dymond. S. (2015) [92]
		Smith. K, Abrams. SS. (2019) [93]
		Cinquin. P.-A, Guitton. P, Sauzéon. H. (2019) [94]
		Jingga. F, Meyliana. Hidayanto. AN, Prabowo. H. (2019) [95]
		Constain. M. GE, Collazos O.C, Moreira. F. (2019) [96]
		Chen. J, Wang. G, Zhang. K, Wang. G, Liu. L. (2019) [52] Aziz. NSA, Ahmad. WFW, Hashim. AS. (2019) [97]
		Tsikinas. S, Xinogalos. S, Satratzemi. M, Kartasidou. L. (2018) [98]
		Çorlu. D, Taşel. Ş, Turan. S.G, Gatos. A, Yantaç. A.E. (2017) [99]
		Vallefuoco. E, Bravaccio. C, Pepino. A. (2017) [49]
		Tsikinas. S, Xinogalos. S, Satratzemi. M. (2016) [100]
		Wolff. M, Gattegno. M.P, Adrien. J.-L, Gabeau. C, Isnard. P. (2014) [101]
		Marchetti. E, Valente. A. (2015) [102]
		Alarcon-Licona. S, Loke. L, Ahmadpour N. (2018) [103]
		Sorce. S, Gentile. V, Oliveto. D, Barraco. R, Malizia. A, Gentile. A. (2018) [18]
		Mazon. C, Fage. C, Sauzéon. H. (2019) [104]
		Goosen. L. (2019) [105]
		Santarosa. L.M.C, Conforto. D. (2016) [45]
		Rahman. M.R, Naha. S, Roy. P.C, et al. (2011) [106]
		Silva Sandez. G, Rodriguez Miranda. F.P. (2018) [107]
		Silva. S.D, Neto. F.M.M, De Lima. R.M, De Macedo. F.T, Santo. J.R.S, Silva. W.L.N. (2017) [20]
		Kamaruzaman. N.N, Jomhari. N. (2015) [43]
		Tsikinas. S, Xinogalos. S. (2019) [108]
		Helmi Adly. MN, Faaizah. S, Naim. CP. (2014) [109]
		Boucenna. S, Narzisi., A, Tilmont. E, et al. (2014) [110]
		Hulusic. V, Pistoljevic. N. (2012) [41]

Table A2. Identified User Experience Concepts.

Authors	Identified Concepts
Backman. A, Mellblom. A, Norman-Claesson. E, Keith-Bodros. G, Frostvittra. M, Bölte. S, Hirvikoski. T. (2018) [40]	Usability
Romero. N.I. (2017) [36]	Accessibility
Sturm. D, Kholodovsky. M, Arab. R, et al. (2019) [42]	Usability
Constain. M. GE, Collazos O.C, Moreira. F. (2019) [96]	Usability, accessibility
Muñoz. R, Morales. C, Villarroel. R, Quezada. A, De Albuquerque. V.H.C. (2019) [82]	Usability
Caria. S, Paternò. F, Santoro. C, Semucci. V. (2018) [22]	Usability, accessibility
Milne. M, Raghavendra. P, Leibbrandt. R, Powers. D.M.W. (2018) [34]	Usability

Table A2. *Cont.*

Authors	Identified Concepts
Wojciechowski. A., Al-Musawi. R. (2017) [15]	Usability
Vallefuoco. E, Bravaccio. C, Pepino. A. (2017) [49]	Usability
Naziatul. A.A, Wan. W.A, Ahmad. H. (2016) [47]	Usability
Sorce. S, Gentile. V, Oliveto. D, Barraco. R, Malizia. A, Gentile. A. (2018) [18]	Usability, accessibility
De Los Rios. P.C. (2018) [33]	Usability, accessibility
Mendonça. V, Coheur. L, Sardinha. A. (2015) [44]	User experience
Rector. K. (2018) [68]	Accessibility
Tang H.H, Jheng. C.M, Chien. M.E, Lin. N.M, Chen. M.Y. (2013) [71]	Usability, accessibility, user experience
Muñoz-Soto. R, Becerra. C, Noël. R., et al. (2016) [25]	Usability, accessibility, user experience
Santarosa. L.M.C, Conforto. D. (2016) [45]	Usability, accessibility
Khowaja. K, Salim. SS. (2019) [46]	Usability
Almeida. LM, Silva. DPD, Theodório. DP, et al. (2019) [51]	Usability, accessibility
Lee. D, Frey. G, Cheng. A, Shih. PC. (2018) [67]	Usability, accessibility, user centered
Cabielles-Hernández. D, Pérez-Pérez. J.-R, Paule-Ruiz. M, Fernández-Fernández. S. (2017) [74]	Accessibility
Schmidt. M, Beck. D. (2016) [27]	Usability
Eder. MS, Díaz. JML, Madela. JRS, Mag-usara. MU, Sabellano. DDM. (2016) [28]	Usability

Table A3. Identified Game Elements.

Authors	Game Elements Referenced
Romero. N.I. (2017) [36]	Points, intrinsic rewards
McKissick. B, Spooner. F, Wood. C.L, Diegelmann. K.M. (2013) [32]	Achievement, levels
Lin. C, Chang. S, Liou. W, Tsai. Y. (2013) [14]	Points
Bistarkey. M, Hayes. G.R. (2015) [38]	Points, levels, rewards, collaboration
Sturm. D, Kholodovsky. M, Arab. R, et al. (2019) [42]	Points, levels, rewards, narrative, collaboration
Muñoz. R, Morales. C, Villarroel. R, Quezada. A, De Albuquerque. V.H.C. (2019) [82]	Levels
Chen. J, Wang. G, Zhang. K, Wang. G, Liu. L. (2019) [52]	Points, levels, rewards
Aziz. NSA, Ahmad. WFW, Hashim. AS. (2019) [97]	Levels
Caria. S, Paternò. F, Santoro. C, Semucci. V. (2018) [22]	Levels
Milne. M, Raghavendra. P, Leibbrandt. R, Powers. D.M.W. (2018) [34]	Feedback, rewards
Sturm. D, Gillespie-Lynch. K, Kholodovsky. M. (2018) [84]	Levels, collaboration
Vallefuoco. E, Bravaccio. C, Pepino. A. (2017) [49]	Feedback
Bringas. J.A.S, León. M.A.C, Cota. I.E, Carrillo. A.L. (2016) [69]	Points, levels, feedback, rewards
Lorenzo. G, Lledó. A, Pomares. J, Roig. R. (2016) [16]	Avatars
Bernardini. S, Porayska-Pomsta. K, Smith. T.J. (2014) [17]	Avatars
Sorce. S, Gentile. V, Oliveto. D, Barraco. R, Malizia. A, Gentile. A. (2018) [18]	Avatars
Magaton. H, Bim. Silvia. (2017) [56]	Levels
Tashnim. A, Nowshin. S, Akter. F, Das. A.K. (2018) [24]	Rewards
Alvarado. C, Muñoz. R, Villarroel. R, et al. (2017) [19]	Points, levels
Muñoz-Soto. R, Becerra. C, Noël. R., et al. (2016) [25]	Points, levels
Escobedo. L, Nguyen. D.H, Boyd. L.A, et al. (2012) [19]	Points, levels, rewards
Harrold. N, Tan. C.T, Rosser. D, Leong. T.W. (2014) [53]	Points, levels, rewards
Khowaja. K, Salim. SS. (2019) [46]	Achievement, points, levels, rewards
Almeida. LM, Silva. DPD, Theodório. DP, et al. (2019) [51]	Points, levels, avatars, feedback
Lee. D, Frey. G, Cheng. A, Shih. PC. (2018) [67]	Rewards
Baldassarri. S, Passerino. L, Ramis. S, Riquelme. I, Perales. FJ. (2018) [73]	Points, levels
Hughes. D.E, Vasquez E, Nicsinger. E. (2016) [86]	Achievement, points, avatars
Schmidt. M, Beck. D. (2016) [27]	Team
Eder. MS, Díaz. JML, Madela. JRS, Mag-usara. MU, Sabellano. DDM. (2016) [28]	Levels
Dehkordi. SR, Rias. RM. (2015) [88]	Points, levels, rewards
Ribeiro. PC, Fox. AB. (2014) [35]	Levels, collaboration
De Urturi. ZS, Méndez. A, García. B. (2012) [66]	Levels, avatars

References

1. Grynszpan, O.; Weiss, P.; Perez-Diaz, F.; Gal, E. Innovative technology-based interventions for autism spectrum disorders: A meta-analysis. *Autism* **2014**, *18*, 346–361. [CrossRef] [PubMed]
2. Kapp, K.M. *The Gamification of Learning and Instruction: Game-Based Methods and Strategies for Training and Education*; Pfeifer: San Francisco, CA, USA, 2012.
3. Asperger, H. Die "Autistischen Psychopathen" im Kindesalter. *Arch. Psychiatr. Nervenkrankh.* **1944**, *117*, 76–136. [CrossRef]
4. American Psychiatric Association. *Diagnostic and Statistical Manual of Mental Disorders*, 5th ed.; American Psychiatric Publishing: Arlington, VA, USA, 2013; pp. 853–854.
5. HealthDay, News for Healthier Living. Available online: https://consumer.healthday.com/cognitive-health-information-26/autism-news-51/u-s-autism-rates-may-be-stabilizing-729825.html (accessed on 12 October 2019).
6. International Organization for Standardization. *Ergonomics of Human System Interaction—Part 210: Human-Centered Design for Interactive Systems*; International Organization for Standardization: Geneva, Switzerland, 2018.
7. Sematic Studios. Available online: http://semanticstudios.com/user_experience_design/ (accessed on 12 October 2019).
8. International Organization for Standardization. *Ergonomics of Human System Interaction—Part 171: Guidance on Software Accessibility*; International Organization for Standardization: Geneva, Switzerland, 2008.
9. Abt, C. *Serious Games*; University Press of America: Lanham, MD, USA, 1970.
10. Werbach, K.; Hunter, D. *For the Win: How Game Thinking Can Revolutionize Your Business*; Wharton Digital Press: Philadelphia, PA, USA, 2012.
11. Khan, B.H. Web-Based Instruction: What Is It and Why Is It? In *Web-Based Instruction*; Educational Technology: Englewood Cliffs, NJ, USA, 1997; pp. 5–58.
12. Keele, S. *Guidelines for Performing Systematic Literature Reviews in Software Engineering*; Technical Report Ver. 2.3; EBSE: Durham, UK, 2007.
13. Arciuli, J.; Bailey, B. Efficacy of ABRACADABRA literacy instruction in a school setting for children with autism spectrum disorders. *Res. Dev. Disabil.* **2019**, *85*, 104–115. [CrossRef]
14. Lin, C.; Chang, S.; Liou, W.; Tsai, Y. The development of a multimedia online language assessment tool for young children with autism. *Res. Dev. Disabil.* **2013**, *34*, 3553–3565. [CrossRef]
15. Wojciechowski, A.; Al-Musawi, R. Assisstive technology application for enhancing social and language skills of young children with autism. *Multimed. Tools Appl.* **2017**, *76*, 5419–5439. [CrossRef]
16. Lorenzo, G.; Lledó, A.; Pomares, J.; Roig, R. Design and application of an immersive virtual reality system to enhance emotional skills for children with autism spectrum disorders. *Comput. Educ.* **2016**, *98*, 192–205. [CrossRef]
17. Bernardini, S.; Porayska-Pomsta, K.; Smith, T.J. ECHOES: An intelligent serious game for fostering social communication in children with autism. *Inf. Sci.* **2014**, *264*, 41–60. [CrossRef]
18. Sorce, S.; Gentile, V.; Oliveto, D.; Barraco, R.; Malizia, A.; Gentile, A. Exploring Usability and Accessibility of Avatar-based Touchless Gestural Interfaces for Autistic People. In Proceedings of the 7th ACM International Symposium on Pervasive Displays, PerDis, Munich, Germany, 6–8 June 2018.
19. Escobedo, L.; Nguyen, D.H.; Boyd, L.A.; Hirano, S.; Rangel, A.; Garcia-Rosas, D.; Tentori, M.; Hayes, G. MOSOCO: A mobile assistive tool to support children with autism practicing social skills in real-life situations. In Proceedings of the 30th ACM Conference on Human Factors in Computing Systems, CHI, Austin, TX, USA, 5–10 May 2012.
20. Silva, S.D.; Neto, F.M.M.; De Lima, R.M.; De Macedo, F.T.; Santo, J.R.S.; Silva, W.L.N. Knowledgemon Hunter: A Serious Game with Geolocation to Support Learning of Children with Autism and Learning Difficulties. In Proceedings of the 9th Symposium on Virtual and Augmented Reality, SVR, Curitiba, Brazil, 1–4 November 2017.
21. Khowaja, K.; Al-Thani, D.; Salim, S.S. Vocabulary learning of children with autism spectrum disorder (ASD): From the development to an evaluation of serious game prototype. In Proceedings of the European Conference on Games-based Learning, Sophia Antipolis, France, 4–5 October 2018.

22. Caria, S.; Paternò, F.; Santoro, C.; Semucci, V. The Design of Web Games for Helping Young High-Functioning Autistics in Learning How to Manage Money. *Mob. Netw. Appl.* **2018**, *23*, 1735–1748. [CrossRef]

23. Tuğbagül Altan, N.; Göktürk, M. Providing individual knowledge from students with autism and mild mental disability using computer interface. In Proceedings of the International Conference on Applied Human Factors and Ergonomics, Los Angeles, CA, USA, 17–21 July 2017.

24. Tashnim, A.; Nowshin, S.; Akter, F.; Das, A.K. Interactive interface design for learning numeracy and calculation for children with autism. In Proceedings of the International Conference on Information Technology and Electrical Engineering, ICITEE, Phuket, Thailand, 12–13 October 2017.

25. Muñoz-Soto, R.; Becerra, C.; Noël, R.; Barcelos, T.; Villarroel, R.; Kreisel, S.; Camblor, M. Proyect@ matemáticas: A learning object for supporting the practitioners in autism spectrum disorders. In Proceedings of the 11th Latin American Conference on Learning Objects and Technology, LACLO, San Carlos, Costa Rica, 3–7 October 2016.

26. Eiselt, K.; Carter, P. Integrating Social Skills Practice with Computer Programming for Students on the Autism Spectrum. In Proceedings of the Frontiers in Education Conference (FIE), San Jose, CA, USA, 3–6 October 2018.

27. Schmidt, M.; Beck, D. Computational thinking and social skills in Virtuoso: An immersive, digital game-based learning environment for youth with autism spectrum disorder. In Proceedings of the 2nd International Conference on Immersive Learning Research Network, iLRN, Santa Barbara, CA, USA, 27 June–1July 2016.

28. Eder, M.S.; Díaz, J.M.L.; Madela, J.R.S.; Mag-Usara, M.U.; Sabellano, D.D.M. Fill me app: An interactive mobile game application for children with Autism. *Int. J. Interact. Mob. Technol.* **2016**, *10*, 59–63. [CrossRef]

29. De Urturi, Z.S.; Méndez, A.; García, B. Serious Game based on first aid education for individuals with Autism Spectrum Disorder (ASD) using android mobile devices. In Proceedings of the 16th International Conference on Computer Games CGAMES'2011, Louisville, KY, USA, 27–30 July 2011.

30. Pérez-Fuster, P.; Sevilla, J.; Herrera, G. Enhancing daily living skills in four adults with autism spectrum disorder through an embodied digital technology-mediated intervention. *Res. Autism Spectr. Disord.* **2019**, *58*, 54–67. [CrossRef]

31. Santarosa, L.M.C.; Conforto, D. Tablet-based activity schedule in mainstream environment for children with autism and children with ID. *ACM Trans. Access. Comput.* **2016**, *8*, 9.

32. McKissic, B.; Spooner, F.; Wood, C.; Diegelmann, K.M. Effects of computer-assisted explicit instruction on map-reading skills for students with autism. *Res. Autism Spectr. Disord.* **2013**, *7*, 1653–1662. [CrossRef]

33. De Los Rios, P.C. Adaptable user interfaces for people with autism: A transportation example. In Proceedings of the 16th International Web for All Conference: Addressing Information Barriers, Lyon, France, 23–25 April 2018.

34. Milne, M.; Raghavendra, P.; Leibbrandt, R.; Powers, D.M.W. Personalisation and automation in a virtual conversation skills tutor for children with autism. *Multimodal User Interfaces* **2018**, *12*, 257–269. [CrossRef]

35. Ribeiro, P.C.; Fox, A.B. ComFiM: A game for multitouch devices to encourage communication between people with autism. In Proceedings of the 3rd International Conference on Serious Games and Applications for Health, SeGAH, Rio de Janeiro, Brazil, 14–16 May 2014.

36. Romero, N.I. A Pilot Study Examining a Computer-Based Intervention to Improve Recognition and Understanding of Emotions in Young Children with Communication and Social Deficits. *Res. Dev. Disabil.* **2017**, *65*, 35–45. [CrossRef]

37. Christinaki, E.; Vidakis, N.; Triantafyllidis, G. A novel educational game for teaching emotion identification skills to preschoolers with autism diagnosis. *Comput. Sci. Inf. Syst.* **2014**, *11*, 723–743. [CrossRef]

38. Boyd, L.E.; Ringland, K.E.; Haimson, O.L.; Fernandez, H.; Bistarkey, M.; Hayes, G.R. Evaluating a Collaborative iPad Game's Impact on Social Relationships for Children with Autism Spectrum Disorder. *ACM Trans. Access. Comput.* **2015**, *7*, 3. [CrossRef]

39. Hourcade, J.P.; Bullock-Rest, N.E.; Hansen, T.E. Multitouch tablet applications and activities to enhance the social skills of children with autism spectrum disorders. *Personal Ubiquitous Comput.* **2012**, *16*, 157–168. [CrossRef]

40. Backman, A.; Mellblom, A.; Norman-Claesson, E.; Keith-Bodros, G.; Frostvittra, M.; Bölte, S.; Hirvikoski, T. Internet-delivered psychoeducation for older adolescents and young adults with autism spectrum disorder (SCOPE): An open feasibility study. *Res. Autism Spectr. Disord.* **2018**, *54*, 51–64. [CrossRef]

41. Hulusic, V.; Pistoljevic, N. LeFCA: Learning framework for children with autism. In Proceedings of the 4th International Conference on Games and Virtual Worlds for Serious Applications, VS-GAMES, Genoa, Italy, 29–31 October 2012.

42. Sturm, D.; Kholodovsky, M.; Arab, R.; Smith, D.S.; Asanov, P.; Gillespie-Lynch, K. Participatory Design of a Hybrid Kinect Game to Promote Collaboration between Autistic Players and Their Peers. *Int. J. Hum. Comput. Interact.* **2019**, *35*, 706–723. [CrossRef]

43. Kamaruzaman, N.N.; Jomhari, N. Digital Game-Based Learning for Low Functioning Autism Children in Learning Al-Quran. In Proceedings of the Taibah University International Conference on Advances in Information Technology for the Holy Quran and Its Sciences, NOORIC, Madinah, Saudi Arabia, 22–25 December 2013.

44. Mendonça, V.; Coheur, L.; Sardinha, A. VITHEA-Kids: A Platform for Improving Language Skills of Children with Autism Spectrum Disorder. In Proceedings of the International ACM SIGACCESS Conference on Computers and Accessibility, Lisbon, Portugal, 26–28 October 2015.

45. Santarosa, L.M.C.; Conforto, D. Educational and digital inclusion for subjects with autism spectrum disorders in 1:1 technological configuration. *Comput. Hum. Behav.* **2016**, *60*, 293–300. [CrossRef]

46. Khowaja, K.; Salim, S.S. Serious Game for Children with Autism to Learn Vocabulary: An Experimental Evaluation. *Int. J. Hum. Comput. Interact.* **2019**, *35*, 1–26. [CrossRef]

47. Naziatul, A.A.; Wan, W.A.; Ahmad, H. The design of mobile numerical application development lifecyle for children with autism. *J. Teknol.* **2016**, *78*, 13–20.

48. Nielsen, J.; Molich, R. Heuristic evaluation of user interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing System, Seattle, WA, USA, 1–5 April 1990.

49. Vallefuoco, E.; Bravaccio, C.; Pepino, A. Serious games in autism spectrum disorder: An example of personalised design. In Proceedings of the 9th International Conference on Computer Supported Education, CSEDU, Porto, Portugal, 21–23 April 2017.

50. Moreno-Ger, P.; Torrente, J.; Hsieh, Y.G.; Lester, W.T. Usability testing for serious games: Making informed design decisions with user data. *Adv. Human Comput. Interact.* **2012**, *2012*, 4. [CrossRef]

51. Almeida, L.M.; Silva, D.P.D.; Theodório, D.P.; Silva, W.W.; Rodrigues, S.C.M.; Scardovelli, T.A.; da Silva, A.P.; Bissaco, M.A.S. ALTRIRAS: A Computer Game for Training Children with Autism Spectrum Disorder in the Recognition of Basic Emotions. *Int. J. Comput. Games Technol.* **2019**, *2019*. [CrossRef]

52. Chen, J.; Wang, G.; Zhang, K.; Wang, G.; Liu, L. A pilot study on evaluating children with autism spectrum disorder using computer games. *Comput. Hum. Behav.* **2019**, *90*, 204–214. [CrossRef]

53. Harrold, N.; Tan, C.T.; Rosser, D.; Leong, T.W. CopyMe: A portable real-time feedback expression recognition game for children. In Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems, CHI EA, Toronto, ON, Canada, 26 April–1 May 2014.

54. Herrmann, M.J.; Bogon, J.; Quester, S.; Cordes, A.; Stenneken, P.; Reif, A.; Ehlis, A.C. Serotonin transporter polymorphism modulates neural correlates of real-life joint action. An investigation with functional near-infrared spectroscopy (fNIRS). *Neuroscience* **2015**, *292*, 129–136. [CrossRef]

55. Hauswald, A.; Weisz, N.; Bentin, S.; Kissler, J. MEG premotor abnormalities in children with Asperger's syndrome: Determinants of social behavior? *Dev. Cogn. Neurosci.* **2013**, *5*, 95–105. [CrossRef]

56. Magaton, H.C.; Bim, S.A. The Use of Educational Applications by Children with Autistic Spectrum Disorder: A Case of Study. In Proceedings of the IHC, Brazilian Symposium on Human Factors in Computing Systems, Joinville, Brazil, 23–27 October 2017.

57. Wilson, C.; Brereton, M.; Ploderer, B.; Sitbon, L. MyWord: Enhancing engagement, interaction and self-expression with minimally-verbal children on the autism spectrum through a personal audio-visual dictionary. In Proceedings of the ACM Conference on Interaction Design and Children, Trondheim, Norway, 19–22 June 2018.

58. Alvarado, C.; Muñoz, R.; Villarroel, R.; Acuna, O.; Barcelos, T.S.; Becerra, C. ValpoDijo: Developing a software that supports the teaching of chilean idioms to children with autism spectrum disorders. In Proceedings of the 12th Latin American Conference on Learning Objects and Technologies, LACLO 2017, La Plata, Argentina, 9–13 October 2017.

59. Rasche, N.; Pourcho, J.; Wei, S.; Qian, C.Z.; Chen, V.Y. Literacy LABELS: Emergent literacy application design for children with autism. In Proceedings of the ACM SIGGRAPH, Anaheim, CA, USA, 21–25 July 2013.

60. Cunha, R.M.; Barbosa, S.D.J. Development and evaluation of a computer game for teaching vocabulary to children with autism. In Proceedings of the International Conference on Web Information Systems and Technologies, Porto, Portugal, 18–21 April 2012.

61. Gomez, J.; Jaccheri, L.; Torrado, J.C.; Montoro, G. Leo con lula, introducing global reading methods to children with ASD. In Proceedings of the ACM Conference on Interaction Design and Children, IDC, Trondheim, Norway, 19–22 June 2018.

62. Khowaja, K.; Salim, S.S.; Al-Thani, D. Components to design serious games for children with autism spectrum disorder (ASD) to learn vocabulary. In Proceedings of the 5th International Conference on Engineering Technologies and Applied Sciences, ICETAS, Bangkok, Thailand, 22–23 November 2018.

63. Frutos, M.; Bustos, I.; Zapirain, B.G.; Zorrilla, A.M. Computer game to learn and enhance speech problems for children with autism. In Proceedings of the 16th International Conference on Computer Games: AI, Animation, Mobile, Interactive Multimedia, Educational and Serious Games, CGAMES, Louisville, KY, USA, 27–30 July 2011.

64. Aziz, N.S.A.; Ahmad, W.F.W.; Hashim, A.S. Development phase of mobile numerical application for children with autism: Math4Autism. In Proceedings of the 3rd International Conference on Computer and Information Sciences, ICCOINS, Kuala Lumpur, Malaysia, 15–17 August 2016.

65. Aziz, N.S.A.; Ahmad, W.F.W.; Zulkifli, N.J.B. User experience on numerical application between children with down syndrome and autism. In Proceedings of the International HCI and UX Conference in Indonesia, CHIuXiD, Bandung, Indonesia, 8–10 April 2015.

66. De Urturi, Z.S.; Méndez, A.; García, B. A serious game for android devices to help educate individuals with autism on basic first aid. In Proceedings of the 9th International Conference on Distributed Computing and Artificial Intelligence, DCAI, Salamanca, Spain, 28–30 March 2012.

67. Lee, D.; Frey, G.; Cheng, A.; Shih, P.C. Puzzle walk: A gamified mobile app to increase physical activity in adults with autism spectrum disorder. In Proceedings of the 10th International Conference on Virtual Worlds and Games for Serious Applications, VS-Games, Wurzburg, Germany, 5–7 September 2018.

68. Rector, K. Enhancing Accessibility and Engagement for Those with Disabilities. *IEEE Pervasive Comput.* **2018**, *17*, 9–12. [CrossRef]

69. Bringas, J.A.S.; León, M.A.C.; Cota, I.E.; Carrillo, A.L. Development of a videogame to improve communication in children with autism. In Proceedings of the 11th Latin American Conference on Learning Objects and Technology, LACLO, San Carlos, Costa Rica, 3–7 October 2016.

70. Hussain, A.; Abdullah, A.; Husni, H.; Mkpojiogu, E.O.C. Interaction design principles for edutainment systems: Enhancing the communication skills of children with autism spectrum disorders. *Revista Tecnica de la Facultad de Ingenieria Universidad del Zulia* **2016**, *39*, 45–50.

71. Tang, H.H.; Jheng, C.M.; Chien, M.E.; Lin, N.M.; Chen, M.Y. iCAN: A tablet-based pedagogical system for improving the user experience of children with autism in the learning process. In Proceedings of the 1st International Conference on Orange Technologies, ICOT, Tainan, Taiwan, 12–16 March 2013.

72. El-Seoud, M.S.A.; Karkar, A.G.; Al Ja'am, J.M.; Karam, O.H. A pictorial mobile application for improving communication skills in non-verbal autism. *Int. J. Interact. Mob. Technol.* **2015**, *9*, 49–55. [CrossRef]

73. Baldassarri, S.; Passerino, L.; Ramis, S.; Riquelme, I.; Perales, F.J. Videogame-based case studies for improving communication and attention in children with ASD. In Proceedings of the 19th International Conference on Human-Computer Interaction, Interaccion, Palma, Spain, 12–14 September 2018.

74. Cabielles-Hernández, D.; Pérez-Pérez, J.-R.; Paule-Ruiz, M.; Fernández-Fernández, S. Specialized Intervention Using Tablet Devices for Communication Deficits in Children with Autism Spectrum Disorders. *IEEE Trans. Learn. Technol.* **2017**, *10*, 182–193. [CrossRef]

75. Papoutsi, C.; Drigas, A.; Skianis, C. Mobile applications to improve emotional intelligence in Autism—A review. *Int. J. Interact. Mob. Technol.* **2018**, *12*, 47–61. [CrossRef]

76. Leijdekkers, P.; Gay, V.; Wong, F. CaptureMyEmotion: A mobile app to improve emotion learning for autistic children using sensors. In Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems, CBMS, Porto, Portugal, 20–22 June 2013.

77. Tinnunnem., S.G.; Shah, G.; Lahiri, U. Psycho-physiological implications of computer based social and non-social interactive tasks for children with autism. In Proceedings of the 8th International Conference on Computing, Communications and Networking Technologies, ICCCNT, Delhi, India, 3–5 July 2017.

78. Hyun, P.J.; Abirached, B.; Zhang, Y. A framework for designing assistive technologies for teaching children with ASDs emotions. In Proceedings of the CHI EA '12 CHI '12 Extended Abstracts on Human Factors in Computing Systems, Austin, TX, USA, 5–10 May 2012.

79. Castillo, T.A.; Pérez de Celis, C.; Lara, C.; Somodevilla, M.J.; Pineda, I.H.; de Alba, K.F.; Romero, E. Authic: Computational tool for children with autistic spectrum disorder. In Proceedings of the International Symposium on Computers in Education, SIIE, Salamanca, Spain, 13–15 September 2016.

80. Cisnero, A.Q.; Juárez-Ramírez, R.; Figueroa, A.M. Towards a methodology for the learning of emotions for children with autism. In Proceedings of the 11th Latin American Conference on Learning Objects and Technology, LACLO, San Carlos, Costa Rica, 3–7 October 2016.

81. DiGennaro, F.D.; Hyman, S.R.; Hirst, J.M. Applications of technology to teach social skills to children with autism. *Res. Autism Spectr. Disord.* **2011**, *5*, 1003–1010. [CrossRef]

82. Muñoz, R.; Morales, C.; Villarroel, R.; Quezada, A.; De Albuquerque, V.H.C. Developing a software that supports the improvement of the theory of mind in children with autism spectrum disorder. *IEEE Access* **2019**, *7*, 7948–7956. [CrossRef]

83. Rapp, A.; Cena, F.; Castald, R.; Keller, R.; Tirassa, M. Designing technology for spatial needs: Routines, control and social competences of people with autism. *Int. J. Hum. Comput. Stud.* **2018**, *120*, 49–65. [CrossRef]

84. Sturm, D.; Gillespie-Lynch, K.; Kholodovsky, M. Assessing collaboration between autistic players—An engagement metric. In Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility, Baltimore, MD, USA, 20 October–1 November 2017.

85. Grossard, C.; Grynspan, O.; Serret, S.; Jouen, A.L.; Cohen, D. Serious games to teach social interactions and emotions to individuals with autism spectrum disorders (ASD). *Comput. Educ.* **2017**, *113*, 195–211. [CrossRef]

86. Hughes, D.E.; Vasquez, E.; Nicsinger, E. Improving perspective taking and empathy in children with autism spectrum disorder. In Proceedings of the IEEE International Conference on Serious Games and Applications for Health, SeGAH, Orlando, FL, USA, 11–13 May 2016.

87. Hani, H.; Abu-Wandi, R. DISSERO mobile application for autistic children's. In Proceedings of the International Conference on Intelligent Information Processing, Security and Advanced Communication, IPAC, Batna, Algeria, 23–25 November 2015.

88. Dehkordi, S.R.; Rias, R.M. Using mobile game application to teach children with Autism Spectrum Disorder (ASD) multiple cues responding: A pilot study. In Proceedings of the 3rd International Conference on User Science and Engineering: Experience. Engineer. Engage, i-USEr, Shah Alam, Malaysia, 2–5 September 2014.

89. Aziz, M.Z.A.; Abdullah, S.A.C.; Adnan, S.F.S.; Mazalan, L. Educational app for children with Autism Spectrum Disorders (ASDs). *Procedia Comput. Sci.* **2014**, *42*, 70–77. [CrossRef]

90. Jeekratok, K.; Chanchalor, S.; Murphy, E. Web-based social stories and games for children with Autism. *Int. J. Web Based Learn. Teach. Technol.* **2014**, *9*, 33–49. [CrossRef]

91. Hong, E.R.; Gong, L.Y.; Ninci, J.; Morin, K.L.; Davis, J.; Kawaminami, S.; Shi, Y.G.; Noro, F. A meta-analysis of single-case research on the use of tablet-mediated interventions for persons with ASD. *Res. Dev. Disabil.* **2017**, *70*, 198–214. [CrossRef]

92. Still, K.; May, R.J.; Rehfeldt, R.A.; Whelan, R.; Dymond, S. Facilitating derived requesting skills with a touchscreen tablet computer for children with autism spectrum disorder. *Res. Autism Spectr. Disord.* **2015**, *19*, 44–58. [CrossRef]

93. Smith, K.; Abrams, S.S. Gamification and accessibility. *Int. J. Inf. Learn. Technol.* **2019**, *36*, 104–123. [CrossRef]

94. Cinquin, P.A.; Guitton, P.; Sauzéon, H. Online e-learning and cognitive disabilities: A systematic review. *Comput. Educ.* **2019**, *130*, 152–167. [CrossRef]

95. Jingga, F.; Meyliana Hidayanto, A.N.; Prabowo, H. Computer human interaction research for children with autism spectrum disorder (ASD): A systematic literature review. *Int. J. Mech. Eng. Technol.* **2019**, *10*, 1610–1619.

96. Constain, M.G.E.; Collazos, O.C.; Moreira, F. The gamification in the design of computational applications to support the autism treatments: An advance in the state of the art. In Proceedings of the World Conference on Information Systems and Technologies, WorldCIST, Galicia, Spain, 16–19 April 2019.

97. Aziz, N.S.A.; Ahmad, W.F.W.; Hashim, A.S. A study on mobile applications developed for children with autism. In Proceedings of the 3rd International Conference of Reliable Information and Communication Technology, IRICT, Kuala Lumpur, Malaysia, 23–24 July 2019.

98. Tsikinas, S.; Xinogalos, S.; Satratzemi, M.; Kartasidou, L. Using serious games for promoting blended learning for people with intellectual disabilities and autism: Literature vs reality. In Proceedings of the 11th International Conference on Interactive Mobile Communication Technologies and Learning, IMCL, Hamilton, ON, Canada, 11–12 October 2018; Springer: Cham, Switzerland, 2018.

99. Çorlu, D.; Taşel, Ş.; Turan, S.G.; Gatos, A.; Yantaç, A.E. Involving autistics in user experience studies: A critical review. In Proceedings of the 12th ACM Conference on Designing Interactive Systems, DIS, Edinburgh, UK, 10–14 June 2017.

100. Tsikinas, S.; Xinogalos, S.; Satratzemi, M. Review on serious games for people with intellectual disabilities and autism. In Proceedings of the 10th European Conference on Games Based Learning, ECGBL, Paisley, Scotland, 6–7 October 2016.

101. Wolff, M.; Gattegno, M.P.; Adrien, J.L.; Gabeau, C.; Isnard, P. Contribution of tablets to the support of children and adolescents with autistic disorders: Case studies. *J. Eur. Syst. Automat.* **2014**, *48*, 261–282. [CrossRef]

102. Marchett, E.; Valente, A. A tangible digital installation in the classroom: Role play and autistic children. In Proceedings of the European Conference on Games-Based Learning, Steinkjer, Norway, 8–9 October 2015.

103. Alarcon-Licona, S.; Loke, L.; Ahmadpour, N. From autism educators to game designers: Integrating teaching strategies into game design for autism education support. In Proceedings of the 30th Australian Conference on Computer-Human Interaction, OzCHI, Melbourne, Australia, 4–7 December 2018.

104. Mazon, C.; Fage, C.; Sauzéon, H. Effectiveness and usability of technology-based interventions for children and adolescents with ASD: A systematic review of reliability, consistency, generalization and durability related to the effects of intervention. *Comput. Hum. Behav.* **2019**, *93*, 235–251. [CrossRef]

105. Goosen, L. Information systems and technologies opening new worlds for learning to children with autism spectrum disorders. In Proceedings of the 2nd International conference on Europe Middle East and North Africa Information Systems and Technologies to Support Learning, EMENA-ISTL, Marrakech, Morocco, 21–23 November 2019; Springer: Cham, Switzerland, 2019.

106. Rahman, M.R.; Naha, S.; Roy, P.C.; Ahmed, I.; Samrose, S.; Rahman, M.M.; Ahmed, S.I. A-class: A classroom software with the support for diversity in aptitudes of autistic children. In Proceedings of the IEEE Symposium on Computers and Informatics, ISCI, Kuala Lumpur, Malaysia, 20–23 March 2011.

107. Silva Sandez, G.; Rodriguez Miranda, F.P. A view to ICT in the education of disabilities people and with autism spectrum disorders: A topical and bibliographic analysis. *Educ. Educ. Res.* **2018**, *7*, 43–65.

108. Tsikinas, S.; Xinogalos, S. Studying the effects of computer serious games on people with intellectual disabilities or autism spectrum disorder: A systematic literature review. *J. Comput. Assist. Learn.* **2019**, *35*, 61–73. [CrossRef]

109. Helmi Adly, M.N.; Faaizah, S.; Naim, C.P. Serious game for autism children: Conceptual framework. In Proceedings of the International Conference on Information and Communication Technology for Education, ICTE, Singapore, 1–2 December 2013.

110. Boucenna, S.; Narzisi, A.; Tilmont, E.; Muratori, F.; Pioggia, G.; Cohen, D.; Chetouani, M. Interactive Technologies for Autistic Children: A Review. *Cogn. Comput.* **2014**, *6*, 722–740. [CrossRef]

Connected Bike-smart IoT-based Cycling Training Solution

George Catargiu [1], Eva-H. Dulf [1,2,*] and Liviu C. Miclea [1]

[1]   Department of Automation, Faculty of Automation and Computer Science,
     Technical University of Cluj-Napoca, Memorandumului Str. 28, 400014 Cluj-Napoca, Romania;
     George.CATARGIU@student.utcluj.ro (G.C.); Liviu.Miclea@aut.utcluj.ro (L.C.M.)
[2]   Physiological Controls Research Center, Óbuda University, H-1034 Budapest, Hungary
[*]   Correspondence: Eva.Dulf@aut.utcluj.ro

Abstract: The Connected Bike project combines several technologies, both hardware and software, to provide cycling enthusiasts with a modern alternative solution for training. Therefore, a trainer can monitor online through a Web Application some of the important parameters for training, more specifically the speed, cadence and power generated by the cyclist. Also, the trainer can see at every moment where the rider is with the aid of a GPS module. The system is built out of both hardware and software components. The hardware is in charge of collecting, scaling, converting and sending data from sensors. On the software side, there is the server, which consists of the Back-End and the MQTT (Message Queues Telemetry Transport) Broker, as well as the Front-End of the Web Application that displays and manages data as well as collaboration between cyclists and trainers. Finally, there is the Android Application that acts like a remote command for the hardware module on the bike, giving the rider control over how and when the ride is monitored.

Keywords: connected bike; smart technologies; IoT; personalized training; embedded; back-end; front-end; Android; modules; MQTT; monitoring

1. Introduction

As times change, technology evolves and people migrate from fieldwork to deskwork, a new health threat arises: the sedentary lifestyle. To fight this phenomenon, fitness seems like a handy solution, especially for the advantage of having a personal trainer that can monitor the activity and offer guidance throughout the training. The electric bicycle is a trendy solution, as it is presented in [1]. However, most cyclists either amateur or professionals do not have this opportunity, making it harder to improve and reach specific goals. This paper proposes a technical solution meant to overcome this issue, by using sensors, microcontrollers and microcomputers to read, send, save, and display data. Therefore, any trainer can monitor relevant cycling data gathered from the bike in real time.

Data collection from sensors is widely used today. However, all the data are focused on sharing system development [2–4], sensor network implementation [5,6], safety [7], smart city development [8,9] or intelligent control implementation [10–12]. A series of great results are published considering the positioning method of the sensor system, structural health monitoring, and intelligent vehicle [13–17]. No research focusing on IoT (Internet of Thinks) and smart bike fusion for training has been published.

Nowadays, in cycling, the most common training method is by sheet. This sheet can be custom made by a trainer for one or more riders and holds a training plan by days, weeks and even season periods. However, this old and rigid method has a higher chance of failing to reach its purpose.

The Connected Bike proposed in this paper retrieves and transmits over internet the speed, cadence and cyclist power output both to a server-hosted database and to a web application, from where a trainer can monitor, evaluate and notify possible errors made by the cyclist during the ride.

Furthermore, by persisting data to a server, both the cyclist and the trainer can return over past rides and discuss about them. The rider also has the possibility to control the system by starting, pausing or ending the ride with the help of a mobile application and internet connection.

All the technologies used in the presented Connected Bike project serve the following purposes:

- Collecting data from sensors
- Computing the speed, pedaling cadence and instant power output of the cyclist from collected data
- Real time GPS tracking of the bicycle
- Aggregating and sending data to the web interface for online monitoring
- Persisting training data to a database for later visualization
- Allowing user to control the monitoring system through a mobile Android app
- Implementing security levels to protect users' data

The idea of real time bicycle tracking has been around for a while, and Chris Kiefer and Frauke Behrendt tackled this in [18] with a smart e-bike monitoring system (SEMS). They use custom hardware and a mobile phone with internet connection as an access point to monitor data like the bike's current location and the motor's level of assistance. This project is aimed towards e-bicycle fleets, focusing on scalability and ease of use, thus leaving aside acquisition of training relevant data like speed or cadence. A significant drawback for SEMS is that the battery powering the system discharges quite fast because of the hardware and connected mobile phone.

Neil Savage describes in [19] a bicycle data-monitoring concept named Copenhagen Wheel, by replacing the rear wheel with an intelligent one. This wheel has a modified hub, which encapsulates in addition to a standard one, an electric motor, and an ensemble of electronic components, both powered by a battery. The system is capable of changing gears, adapting the motor's speed and monitoring the torque, all while gathering pollution, humidity, temperature data, and sending it to a server through a mobile phone connected to the internet.

In terms of gathering data, which is a key concept of this research, Maurizio Di Paolo Emilio presented in [20] the main concepts involved in the process. Therefore, he described a data acquisition system as a mean of collecting information about a physical phenomenon through sensors and other electronic components. The sensors can be both digital (1/0, ON/OFF, etc.) and analog, the latter being harder to quantify and needs additional sampling and conversion circuitry. Any data acquisition system must have a computer or microcontroller as a core in charge of processing data and a mean of transportation, storage and display for this data.

In [21], Sarah Al-Mutlaq briefly presented the working principle of a load cell. These sensors translate the load or force acting on it into an electric signal and can be of three types:

1. Hydraulic load cells—determines the applied force by measuring the displacement of a piston in a cylinder and the change of a diaphragm, which modifies the pressure inside the Bourdon tube
2. Pneumatic load cells—measures the force by determining the air pressure applied to one end of the diaphragm, which then leaves the cell through a nozzle on the other end.
3. Strain gauge load cells—mechanical elements, without moving parts that measure the force through the deformation of one or more strain gauges.

All these ideas are embedded in the present paper, leading to an easy-to-use smart bike for personalized training. The novelty of the paper consists in the design and implementation of this smart connected bike, using the interaction of future-oriented mobility and state-of-art data technology, but cheap and accessible elements. All steps are described in detail to be easy to reproduce by the reader. The presented solution has the potential to disrupt existing training solutions.

The paper is structured in three parts. After this short introductory part, Section 2 reveals the used materials and methods, divided in 12 subsections, for each discussed module. Section 3 presents the obtained results. The work ends with concluding remarks.

2. Materials and Methods

The interface for monitoring the cyclist's activity must have an increased portability and must be easy to access and interact. To meet these criteria, a web application was developed. This can be accessed by trainers, as well as by cyclists, having pages and functionalities tailored for the specific user type.

In the web application, a trainer has access to a list of the cyclists to whom they have connected, can select a rider for real time monitoring, or analyze any of the previous rides. They can also stop the collaboration with any rider or establish a new collaboration with another. Last, but not least, the user has access to a page from where they can view and modify their personal account's data. The cyclist user cannot see their performance in real time, having only access to the previous rides. However, they have the possibility to control the monitoring by creating a new ride, pausing or ending the current one, through the help of a mobile phone application that acts as a remote for the module on the bicycle.

From the hardware point of view, the project consists of three modules: a microcontroller module, a microcomputer module and a mixed one. The first one is in charge of reading data from the load cell, measuring the force applied to the pedal and sending the data to the second module through a wireless channel.

The second module consists of a microcontroller, in charge of reading and processing data both from sensors and from the first module and a microcomputer that receives this data, converts it into a standardized message format, and sends it to the server via internet. It also retrieves the current location in terms of latitude and longitude with the help of a GPS module.

Last, but not least, the third module is an always-on microcomputer which hosts the back-end of the web application and the MQTT broker. Two rechargeable batteries power the first two modules, and the mixed module requires internet access that is provided either by a mobile hotspot or by an internet USB Stick.

Taking into account the multiple IoT specific architectures, described by P.P. Ray in [22] and presented shortly in the previous section, the most suitable one for this research proved to be a hybrid architecture, including the sensors, microcontrollers and the web and mobile applications.

As seen in Figure 1, the entire system is structured on three levels: the hardware level, the application level and the network level.

The hardware level consists of two modules that both have an Arduino microcontroller with separate power banks and have wireless communication between each other. The first module is based on an Arduino Micro, having connected an amplification module HX711 and a load cell. This module reads data from the load cell and sends it to the other module via infrared communication. The second module has an Arduino Mega with two Hall Effect sensors, used to determine the speed and cadence. The microcontroller performs data manipulation, afterwards converting it to byte array and sending it to the Raspberry Pi through USB communication. Here, the data is converted again from binary format and is aggregated with latitude and longitude readings from the GPS module. The final data format is a JavaScript Object Notation (JSON) that is periodically published to the server.

The application level includes three main components in charge of the system's business logic, establishing the way the application is handled and how data is stored and displayed. The first component is an Android App that is used to command the Raspberry Pi located on the bicycle, making it easy to manage the ride monitoring. The second component is the Raspberry Pi module placed on the bicycle that implements the system's logic in the form of a state machine, deciding based on the commands from the Android App if it should start reading, managing and sending data. Finally, the third component of this layer is the web application, where users can register, login and view different rides' data, or monitor a ride in real time.
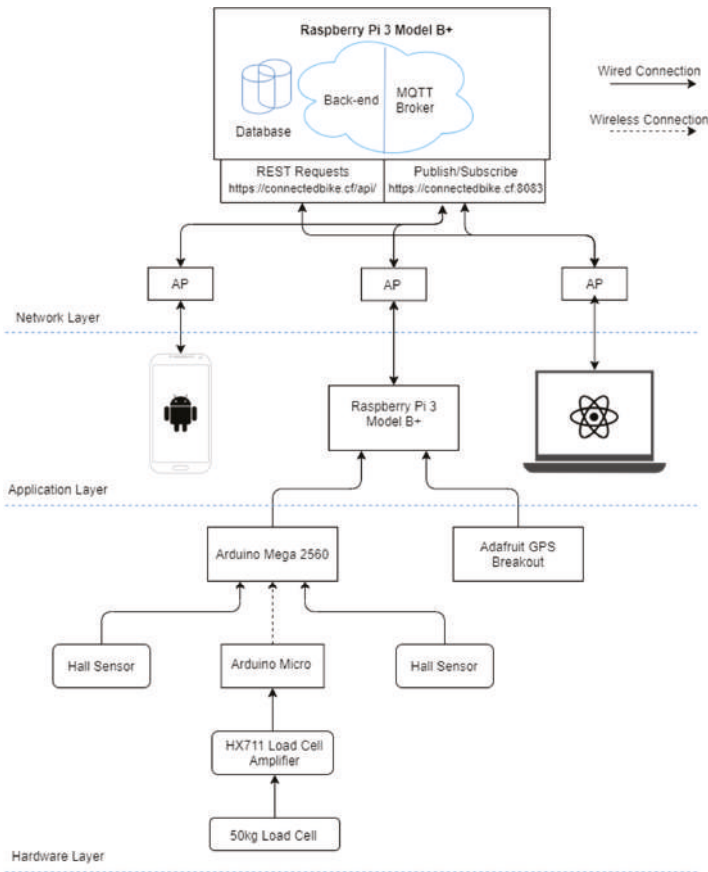
Figure 1. Connected Bike system architecture.

The network level is made out of the actual communication between modules through internet. All the IoT-specific communication in this system is based on the MQTT, each component being connected through an Access Point as clients, except of a Raspberry Pi device, which is the actual server, hosting both the MQTT broker and the HTTP RESTful API server for the web application. There is also HTTP protocol used to establish communication between the front-end and the back-end for the web application.

Seen as a black box, the entire hardware scheme is described in Figure 2, making it easy to observe the way modules communicate with each other. The module blocks are highlighted with blue dotted lines. The communication between the Arduino Mega, Raspberry Pi and the GPS module is realized via USB (for the GPS module, an UART to USB converter was attached). Furthermore, because there are two rotating joints from the pedal to the bicycle's frame, making it harder to use wires, the infrared transmission medium was chosen for its simplicity and robustness. The downside of this approach was the addition of another microcontroller and an extra battery to power it.
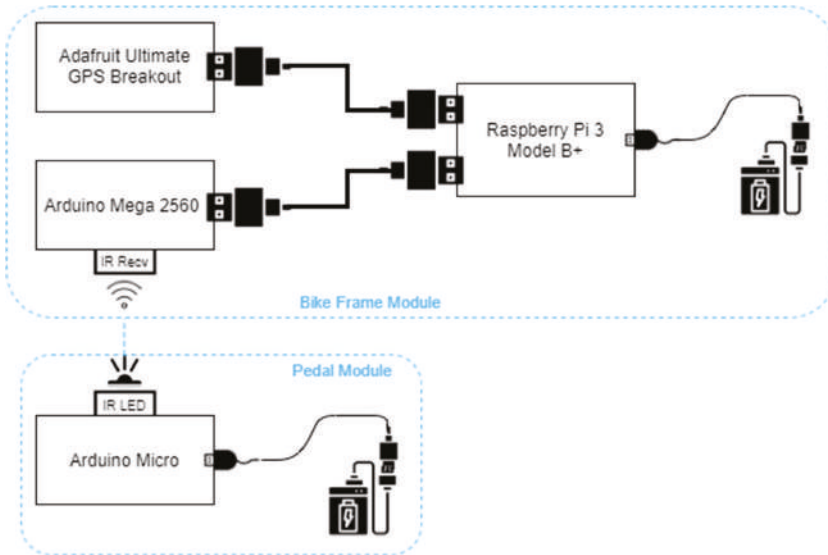
Figure 2. The hardware scheme.

2.1. The Pedal Module

The main functionality of this module is reading the force applied to the pedal by a cyclist. This is possible thanks to a load cell, an amplifying driver, a microcontroller, and a Hall Effect sensor. The data is sent to the frame module with an infrared LED connected to the microcontroller. A rechargeable portable battery powers the entire module. The wiring diagram is shown in Figure 3.
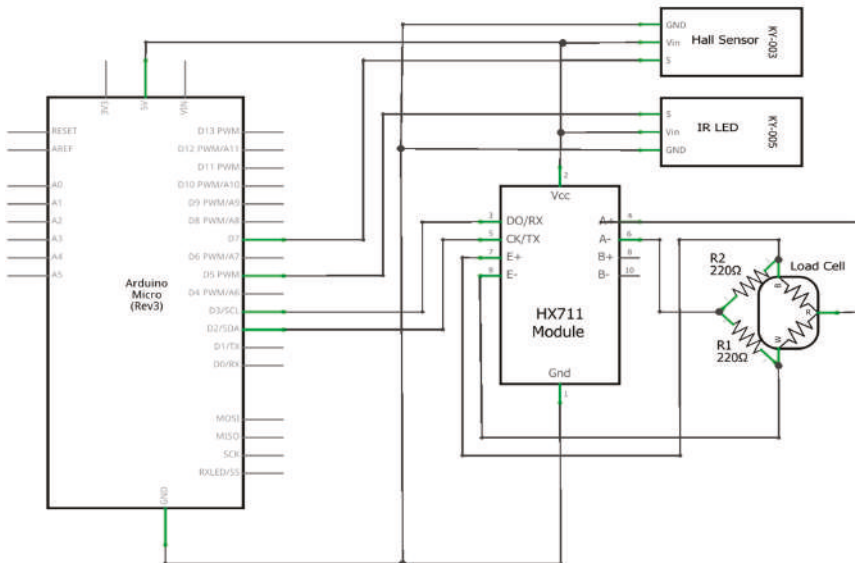

Figure 3. Pedal module wiring diagram.

The microcontroller implements all the logic in this module, being responsible for reading, processing and sending data with a sample frequency of 80 Hz. Because of the fitting space and power limitations, an Arduino Micro was chosen for this module. The communication between the Arduino and the load cell driver is based on I2C (Inter-Integrated Circuit) using the pins 2 and 3 configured as serial data and serial clock. The Hall Effect sensor was connected to General-Purpose Input/Output (GPIO) pin 7, configured as an external interrupt pin and for the infrared LED, the fifth GPIO pin was used.

For this project, a rectangular load cell from SparkFun was used. Thanks to its rectangular form and small dimensions, it fitted directly on the pedal, receiving the full foot force from the cyclist. This load's range is between 0 and 50 kgf and measures with ± 0.5%RO accuracy.

Another particularity of this sensor, in contrast to its alternatives, is the number of wires. Most of the load cells provide four wires, making it easy to use one standalone sensor because it contains a full Wheatstone bridge made out of strain gauges. However, this particular load cell uses three wires and implements half of the bridge from two strain gauges of 1000 Ω each, connected in series.

The HX711 driver was developed for general interfacing between a microcontroller and a load cell, working with a voltage supply of 2.7 to 5 V and has a very simple working principle. Two pins are sending an excitation signal to two sides of the Wheatstone bridge (from the load cell) and two other pins are reading the voltage difference between the other two sides of the bridge. In free mode when no force is applied, the bridge is balanced, thus the difference is zero. When a force is applied onto the cell, the bridge becomes unbalanced and the voltage difference increases. The HX711 reads this difference in mV, amplifies it and maps it to digital values to be sent to the microcontroller.

In order to make this module work with only one three wire load cell, the bridge had to be completed with two 220 Ω static resistors, allowing the driver to read the voltage change from the sensor.

In order for the data to be received from the HX711 driver and processed by the Arduino Micro to reach the frame module, an infrared 5 mm LED was used. Because the pedal is continuously rotating and the pedal module is functioning as a standalone one, a Hall effect sensor was used to determine the position of the pedal arm relative to the bicycle's frame so the Arduino Micro gets notified when the pedal reaches a full circle and passes the magnet placed on the frame.

2.2. Bike Frame Module

This module is used to determine the speed, cadence and rider's power during the ride activity. All this data is serialized and sent via USB to the Raspberry Pi, which acts as a master of the entire system, but also as an internet gateway. Being also a standalone module, it is powered by a rechargeable battery. The wiring diagram is presented in Figure 4.

For data processing, an Arduino Mega 2560 was used, which communicates with the Raspberry Pi via USB. The 16 MHz processing frequency of the microcontroller allows implementation of multiple tasks in a Real Time Operating System (RTOS). The necessity of using an RTOS occurred from the need to read and compute sensor values with higher precision and data transmission.

To monitor the rider's position continuously, a GPS module from Adafruit was used. This device has a sensitivity of −165 dB, 10 readings per second, 66 channels, and the possibility to receive signals from 22 satellites with its own integrated antenna. The latitude and longitude are sent via a FTDI USB-TTL converter to the Raspberry Pi.
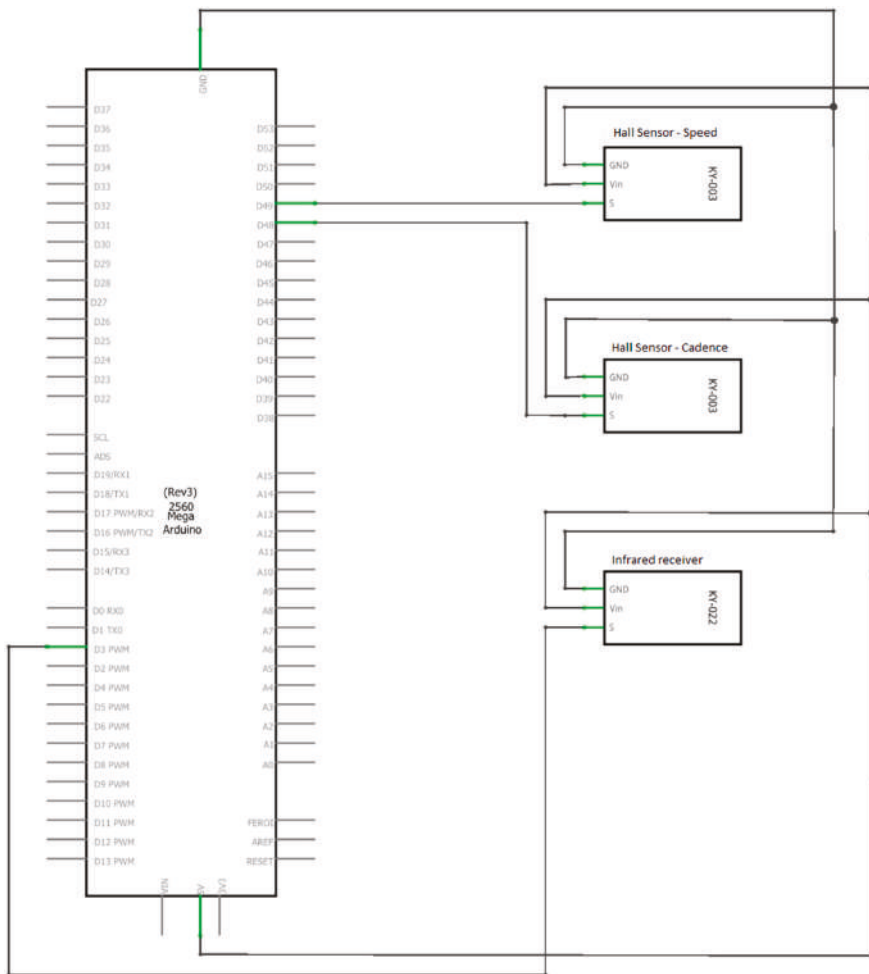
Figure 4. Bike frame module wiring diagram.

2.3. Software

This project was built using multiple programming languages and technologies, addressing different parts and modules.

The users can interact with the system using two applications: An Android app and a web app. Riders can only use the first application while both riders and trainers can use the second app. The user's use cases diagram is presented in Figure 5.

In the following sections, the software implementation is explained, alongside the used technologies for each module.
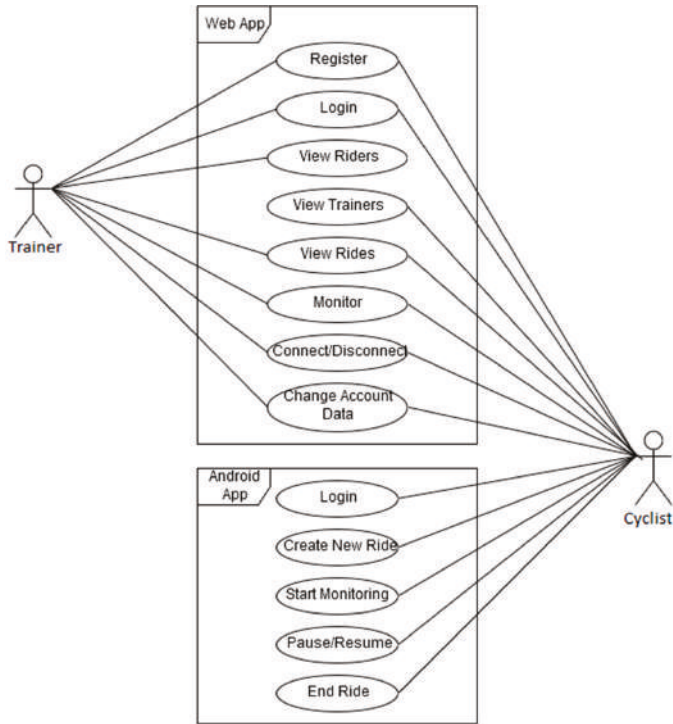
Figure 5. Connected bike use cases.

2.4. The Pedal Software Module

The main component for this module is an Arduino Micro, detailed in the previous hardware chapter. The open source libraries used to implement the software for this microcontroller are HX711.h and IRemote.h. The first one allows interfacing with the HX711 load cell driver while the second one was used for sending encoded message with the infrared LED. The setup function from the Arduino sketch initializes the pin connected to the IR LED and attaches a rising edge interrupt as well as an interrupt service routine to be dispatched on each event. Following this, a HX711 driver initialization and calibration sequence is executed.

Once the setup is complete, the loop function is called, reading the sensors in an infinite loop. The load cell reading is performed by the get_units function from the library mentioned above. The returned value of this function is converted from lbs. to kg and scaled by 10 for a single decimal precision as seen in Equation (1).

$$val[kg] = (val[lbs] \cdot 0.453592) \cdot 0.4 \tag{1}$$

The readings are stored in a three values circular buffer holding the highest values read during a complete pedal cycle. Once the pedal reaches a full rotation and comes next to the magnet placed on the frame, the Hall Effect sensor triggers an interrupt on pin 7, therefore triggering the corresponding interrupt routine. This function computes the average between the values from the buffer, sets the transmission flag and clears the buffer. In the loop function, on each iteration, the transmission flag is checked and if it is set, then the previously computed average is serialized and sent via USB to the Raspberry Pi.

2.5. The Frame Module

The code for the Arduino Mega 2560 in this module is structured in multiple files with the .cpp and .h extensions. The software architecture for the frame module is presented in Figure 6.
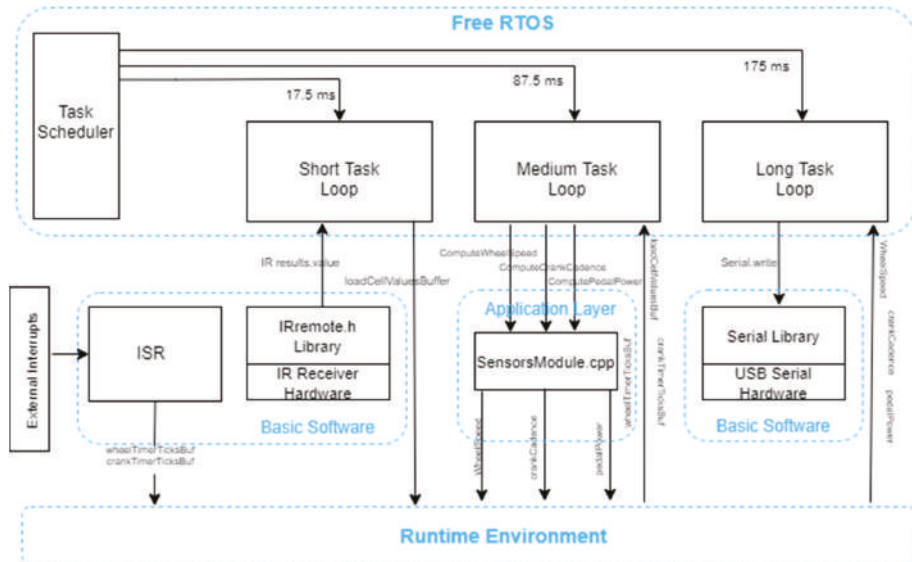

Figure 6. Software architecture.

The code structure and the data manipulation follow a custom-made structure, following guides and patterns from an automotive. Thus, as seen in the above figure, the data transmission and the functionalities are decoupled, data travelling from one component to another through a virtual medium named the Runtime Environment, implementing an AUTOSAR-like (Automotive Open System Architecture) architecture, adapted to the application's context and simplified. The code is structured in four abstraction layers:

- Basic Software—this level is dedicated to hardware interaction between the microcontroller and its connected peripherals. Here can be found the libraries used to interface the sensors and drivers and also the interrupt service routines
- Runtime Environment—as the name suggests, this is a software medium, which only exists at runtime. This holds global variables and value buffers used to update and send data between abstraction layers creating the actual decoupling between layers
- Application Layer—represents the business logic of the program and includes all the functions for computing speed, cadence and pedal power. The raw sensor data is received via runtime environment from the hardware layer, and the computed values are returned back to the RTE from where will they be taken by the transmission function, sending them over USB to the Raspberry Pi
- RTOS (Real-Time Operating System)—is the component in charge of the entire program's execution flow, triggering three periodic tasks based on the priority and task periodicity
- When powering up the microcontroller, the infrared receiver library is instantiated and the setup function is called. Here the following functions are triggered:
- initPinout—initialize pins 48 and 49 for the two Hall effect sensors
- set the pins as digital inputs
- activate the pull-up resistor

- initTimers—timers 4 and 5
- reset the TCCRnA (n = 4,5) registers
- set the prescaler to 1024
- set the capture and overflow interrupt
- reset the interrupt flag register
- irrecv.enableIRIn—activate the infrared receiver
- xTaskCreate—create three tasks for the RTOS and set the execution priority for each
- sei—activate global interrupts
- vTaskStartScheduler—start scheduling component from the RTOS in charge of periodically calling the tasks based on the priority

The initPinout and initTimers macros initialize the pins and timers used by the program and are placed in a separate header file called ucActuatorInit__hh.h using the #*define* directive.

The IRrecv library for the infrared receiver is instantiated at the beginning and is activated by the enableIRIn function. After creating an instance, the object reads in the background the data from the KY-022 module and updates the global results variable, of type decode_results.

In order to compute the speed and cadence, the timers four and five are used. When initialized, the timers are configured to use in normal mode and the capture mode is activated. This means that each timer will increment the TCNTn (Timer Counter Register n = 4, 5) starting from 0 to the maximum value of 65535, and once it reaches this value, the register is reset. Each timer has an allocated pin for the capture mode (pins 48 and 49). Thus, by enabling this mode, the pins' functionality is also changed, allowing them to trigger interrupts on events such as a Hall Effect sensor's impulse, capturing the TCNT's current value in a separate register ICR (Interrupt Capture Register), and an interrupt service routine is dispatched. The timer's working principle is presented in Figure 7, presenting a normal mode of operation and three interrupts on Interrupt Capture Pin.
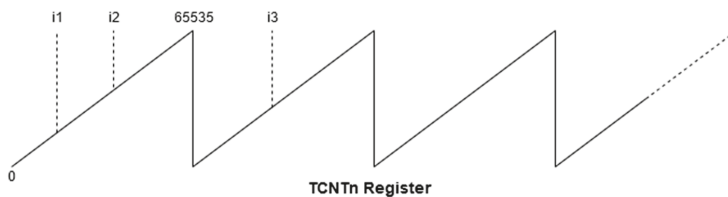

Figure 7. Timer's working principle.

Both the timer 4 and timer 5 routines implement the same operations. First, the tick count is read, then using the tick count from the last interrupt, the elapsed ticks between two consecutive interrupts are determined, using Equation (2), and the result is stored in a circular 10 values buffer from the runtime environment.

$$t_{timer} = i_2(k) - i_1(k-1) \tag{2}$$

where k refers to current ticks and $(k-1)$ to previous ticks.

In the case where the current timer value reading is lower than the previous one, the tick count will be determined with Equation (3).

$$t_{timer} = 65535 - i_2(k-1) + i_3(k) \tag{3}$$

The SensorsModule.cpp file holds the functions to calculate the speed, cadence and power. These functions take the value buffers as parameters and compute the average for the entire set. The value

buffers from RTE are defined as volatile ensuring atomic access and data consistency. The speed computing function determines the bicycle's speed implementing as in Equation (4).

$$v_{wheel} = \frac{C_{wheel}}{t} \cdot 3.6 \cdot 10 \tag{4}$$

where v_{wheel} represents the actual wheel speed, C_{wheel} is the wheel circumference in [mm], and t is the time necessary for rotation in [ms]. The value 3.6 is the 1 m/s converted in 3.6 km/h, and the value 10 is the scaling preventing from working with float values.

Similar to the previous function, the cadence calculus follows Equation (5).

$$Cadence = \frac{time}{\frac{(timerTicksAvg \cdot 64)}{1000}} \tag{5}$$

where time is measured in [ms], while timerTicksAvg in [µs].

The power computing function, in contrast to the previous ones presented above, receives both the 10 values buffer holding the load cell readings and the current cadence, necessary to determine the angular speed used in the final formula. The instantaneous power is determined with Equation (6).

$$P_{pedal} = F_{pedal} \cdot l \cdot \Omega \tag{6}$$

where P_{pedal} represents the pedal power, P_{pedal} the pedal force, l is the crank leg length measured in [cm], and Ω is the angular speed.

At the microcontroller side, the implementation of the power computation is presented in Figure 8.

```
pedalForce = (loadValuesAvg * 98) / 10; // load [kg*10] * factor [9.8*10] => pedalForce scaled with 100
                                        // and then scaled down with 10 [Newtons * 10]

angularSpeed = ((SCALED_PI / 30) * crankCadence) / 100; // compute angular velocity upscaled with 1000
                                                        // and then scaled back down with 100 [rad/s * 10]

pedalPower = ((uint32_t)pedalForce * (uint32_t)CRANK_LEG_LENGTH_CM * (uint32_t)angularSpeed) / 10000;   // 10000 = scaling factor
```

Figure 8. C code implementation of power computing.

The most important software part implemented on the Arduino Mega is the real time operating system. It completely replaces the implicit function loop with periodically executed tasks. At the core of this framework stands the task scheduler. This component dispatches tasks based on their period and priority.

For this project, the FreeRTOS library was used, developed as open source under the MIT license [23]. In order for the library to work on most microcontrollers available on the market, the timer used to determine when a task should be dispatched is the watchdog timer. In addition, because some of the microcontrollers including Arduino only have a single priority level, the library implemented an abstraction layer exposing four virtual priority levels.

In the ucActuator.ino file, inside the setup function, three tasks are created with different periods and priorities, using the xTaskCreate from the FreeRTOS library as follows:

- shortTask

  - 17.5 ms execution period
  - Highest priority
  - Continuously interrogates the infrared receiver and saves the values to the RTE buffer and performs the same for the pedal load cell
  - Operations implemented in this task have the shortest execution time

- mediumTask

- - 87.5 ms execution period
  - Medium priority
  - Calls the functions that compute the speed, cadence and power and returns the results back to the RTE
  - Operations performed in this task have a medium execution time

- longTask

  - 175 ms execution period
  - Lowest priority
  - Checks if the timers used for speed and cadence overflowed more than twice, meaning the movement stopped and sets the corresponding values to 0
  - Calls the function that transmits the serialized data via USB to the Raspberry Pi
  - Data serialization to byte array and data transmission take the longest time

Due to the software priorities, the tasks are executed preemptively. Thus, the medium task is interrupted five times by the shortest task, while the longest task is interrupted 10 times by the short task and twice by the medium task. This allows for a so-called parallel execution on a single core processor given also the reasonable 16 MHz processor frequency of the Arduino Mega. A visual representation of tasks can be observed in Figure 9.
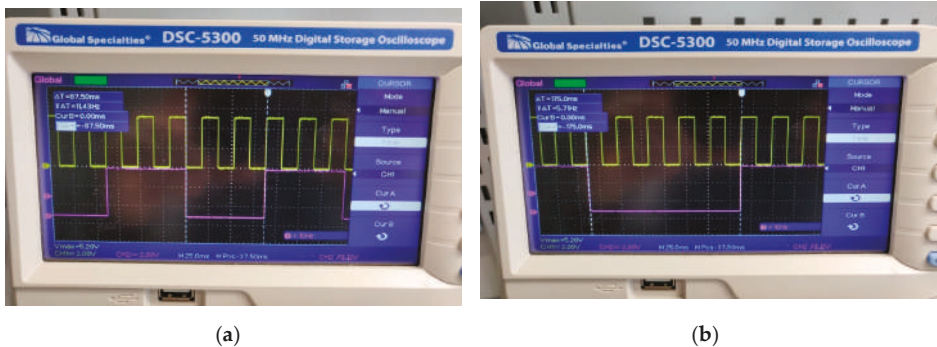

(**a**)   (**b**)

Figure 9. (**a**) Medium (purple) and short task (yellow); (**b**) Long (purple) and short task (yellow).

To facilitate code maintenance, a configuration header was created named ucActuator_kh.h containing pin names and constant values used throughout code. In the event a pin or a parameter shall be changed, all the modifications can be performed into a single file.

2.6. The Raspberry Pi Frame Module

The Raspberry Pi from the frame module is in charge of processing data from Arduino Mega, structuring it into messages and sending them over the internet to the server and front-end for storing and visualization.

The software for the Raspberry Pi application is written in Python, and the communication wireless communication is performed through the MQTT (Message Queueing Telemetry Transport) protocol. As presented on the official web page [24], this protocol is a simple way of sending data between nodes as clients via TCP or Web sockets, used mainly in Internet of Things. The two main entities that are at the core of this protocol are the client and the broker. The latter manages the communication between clients that can publish messages to a topic or subscribe to a topic to consume messages. The Python library that was used to work with this protocol is Paho, which was developed as open source by the Eclipse foundation.

The code was structured by functionalities, in multiple files with the .py extension, also called Python modules. The entry point of the application is the piBrain.py module where two methods are called from the beginning. The first one blocks the program's execution waiting for an internet connection, returning when one is established. The next method is the main one, where the configuration file appconfig.cfg is parsed and the constants are retrieved. In addition, this method instantiates the MqttRemoteThread class and starts the thread.

The MqttRemote.py module contains the MqttRemoteThread class, which extends the Threading class, allowing it to run in a separate thread. The class implements a state machine, listens to commands coming from the Android App through MQTT, and based on them changes the state of the application. Therefore, the cyclist can start, pause or stop the ride using their mobile phone, the thread modifying the entire application's state. A visual representation of the state machine can be seen in Figure 10.
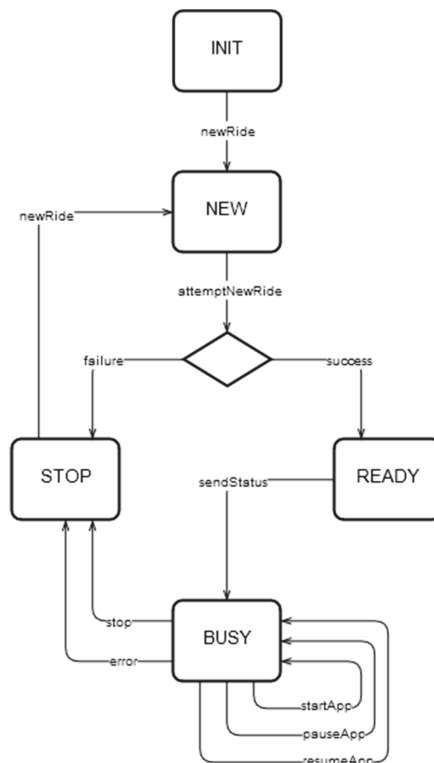

Figure 10. State machine of Raspberry Pi application.

When instantiating the MqttRemoteThread, a new MQTT client is also created that subscribes to the command topics that are sent from the Android App. Paho allows for attaching callback methods to events triggered on-message. The first method of such a kind is named on_connect and is dispatched right after a new connection is established between the client and the broker. Here, the client subscribes to the specific topic. The second callback method is on_message and it is called each time a new message is published to the topic the client is subscribed to. The library also provides a loop_start method that creates a new non-blocking thread that will get messages by polling on the subscribed topics, calling the on_message method once a new message is detected. In this application, the commands that the Android App can send are:

- "NEW"—creates a new ride;

- "START"—starts the current ride;
- "PAUSE" —pauses the current ride,
- "RESUME"—resumes the current ride,
- "STOP"—end the current ride

A new ride is created by the attemptNewRide method, which performs five tries as POST requests to the server. If the server returns 201 success codes together with the newly created ride's primary key, then the program saves this id that is afterwards used to save the ride data to the corresponding ride id. If the server fails to create a new ride, then the application changes its state to "STOP", issuing an event through MQTT back to the Android App to inform the user about the failure.

The run method of the MqttRemoteThread class runs in an infinite loop and consists of if statements that check if a new command was received, changing the application's state accordingly. Once in a new state, the application will perform the required operation and then enter a listening state, ready to accept further commands. This thread is also responsible of sending information back to the Android App to inform the user about the current application's state.

When a user starts the application and sends the "START" command, the startApp method is called from the main thread, which starts three new threads and a thread safe queue object for passing data between them: serialReadThread, gpsPollingThread, mqttPublisherThread.

The SerialReadThread is implemented in the SerialReader.py module, which extends the Threading class and serves the purpose of reading and decoding the data received on the USB bus. On each loop iteration, if the application's state is "RUNNING", the serReader method is called to read six bytes from the input buffer. The raw data is converted into the actual values for speed, cadence and power, which in turn are formatted into a dictionary object that is added to the thread safe queue from where it can be read by other threads.

Inside the GpsPolling.py module there is the GpsPollingThread that is also a separate execution thread where values for latitude and longitude are read using the gps library for Python. The values are updated periodically and stored into a dictionary object, being available through the getLocation method.

Last but not least, the MqttPublisherThread from the MqttPublisher.py module is the execution thread where the final message is created and is sent over MQTT. The speed, cadence and power values as well as the latitude, longitude and the current date-time are retrieved from the thread safe queue object and put together in a JSON message. This message is sent to the cyclist-dependent topic. The way this application's classes interact is presented in Figure 11, which contains the UML classes diagram in Pynthon, used in Raspberry Pi.

In order to start the application at boot time, a Linux shell script was created called raspmodule.sh. Here, the Python App's main module, piBrain.py, is started using its own virtual environment, using the local interpreter and libraries. This shell script is triggered when the Raspberry Pi boots thanks the Cron utility available also on Debian that reads a crontab file where the script was added.
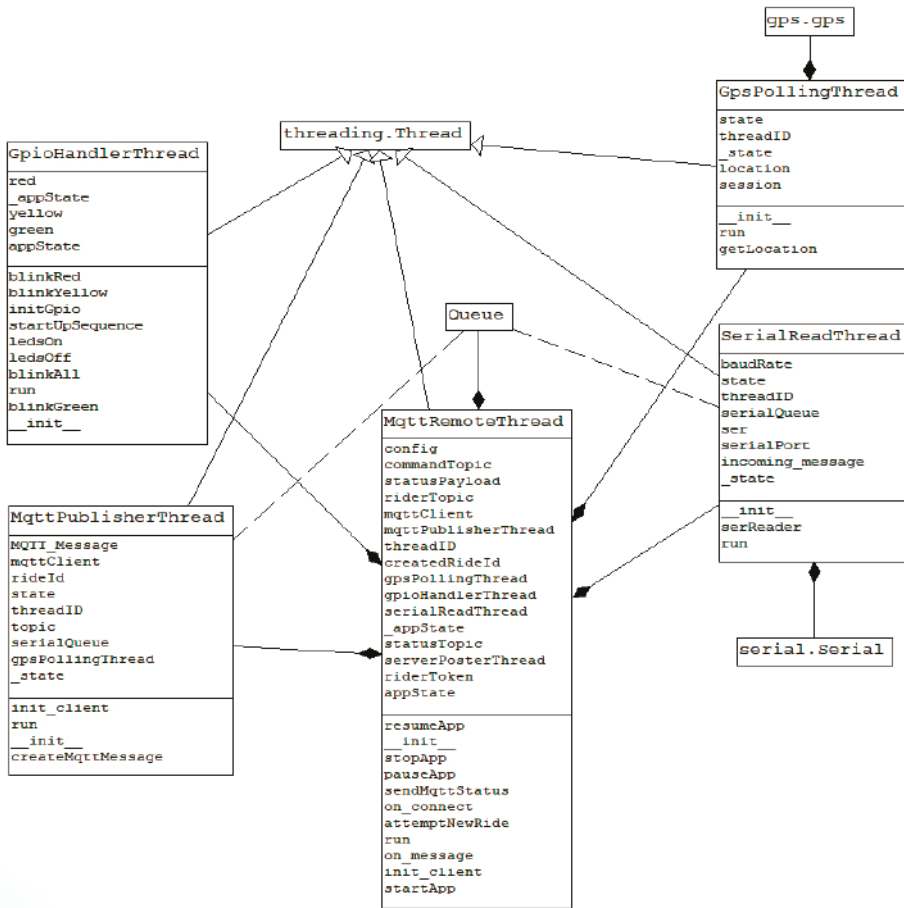
Figure 11. UML classes diagram.

2.7. Android Application

The Android Application has been designed to provide a user-friendly interface and usage. The sole purpose of this application is to inform the user about the state of the bicycle module and to allow them to interact with the system using the phone as a remote control to create a new ride, to stop or resume the current one.

In terms of technical implementation, this app includes two main activities, two classes and one interface. The first activity consists of a TextView that displays in the form of a label the name of the application "Connected Bike Trainer", two EditText elements that allow the user to enter the username and password. It should be noted that in this application, a user could only access with an account that was previously created in the web application. Thus, the last element is a login-only button that triggers the user authentication and if it proves successful, it creates the next activity.

The second activity consists of four buttons and three graphical elements of type TextView. Here, the user can interact with the application, can create a new ride, start, pause, resume, or end the current one.

Of the two classes, MqttHandler is responsible for sending commands and receiving status messages from the Raspberry Pi, via the MQTT communication protocol, while the Service class is responsible for communicating with the server via HTTP protocol.

When a user accesses the application, the first activity to be started is the Login one. Attached to the login button in this activity, is a method that will be called when the button is pressed. This method, attemptLogin, performs the necessary operations to validate the account data end and prevents sending erroneous data to the server such as blank credentials, or special characters. If the validation passes, the UserLoginTask class is instantiated and the execute method is called. This class, representing an asynchronous task, extends the AsyncTask class and contains four methods. The onPreExecute method is automatically called first and it makes the loading component available on the phone's screen. Next, the doInBackground method is called, which has the role of logging the user into the application asynchronously. This is the main component of this task, executing the instructions in a separate thread without blocking the main thread or the graphical user interface. Here, the RequestLogin method is called from the Services class, sending the username, password and an instance of the RestCallback as parameters. onSuccess, onFailure and onNotRider are the three interface methods that are asynchronously called based on the server's response. If the login I is successful, the current activity is replaced by the RiderActivity, which is the core of this application.

The Services class is utilized just by the login activity, implementing the RequestLogin and RequestUserData methods. The first one creates a JSON object with the user's credentials and makes a POST request to the server. The returned response is valid and successful if it has the "token" field, representing the authorization key. Moreover, if the logged-in user is a cyclist, the onSuccess method is called, otherwise the onNotRider is dispatched, displaying a message to inform the user that only cyclists can use this Android Application to control their rides. The second method performs a GET request to the server, using the token received from the previous request and the response is valid if it contains the "topic" field. The synchronization between the asynchronous tasks and the execution threads is done with a CountDownLatch instance that has also been configured with a 5 s timeout to prevent blocking connections.

The methods by which the user can interact with the bicycle system are implemented in RiderActivity. The four buttons available to the user have those methods attached in order to send messages and change the state of the frame module application, running on the Raspberry Pi. This is possible by calling the sendCommand method implemented in the MqttHandler class. Here, the command is converted to JSON and sent on the dedicated command topic to the frame module. Also, being a two-way communication between the Android App and the Raspberry Pi on the bicycle, in order to receive and display information about the state of the frame module, when the eclipse.paho library is initialized, event handlers are attached for events such as lost connection or successful message delivery. When a new message is received from the frame module, the messageArrived is called, retrieving it and sending it for displaying in the RiderActivity to inform the cyclist regarding the application's state. The MQTT connection to the broker is established in the constructor of the MqttHandler class. For a better understanding of the Android Application, a UML class diagram was developed, shown in Figure 12.
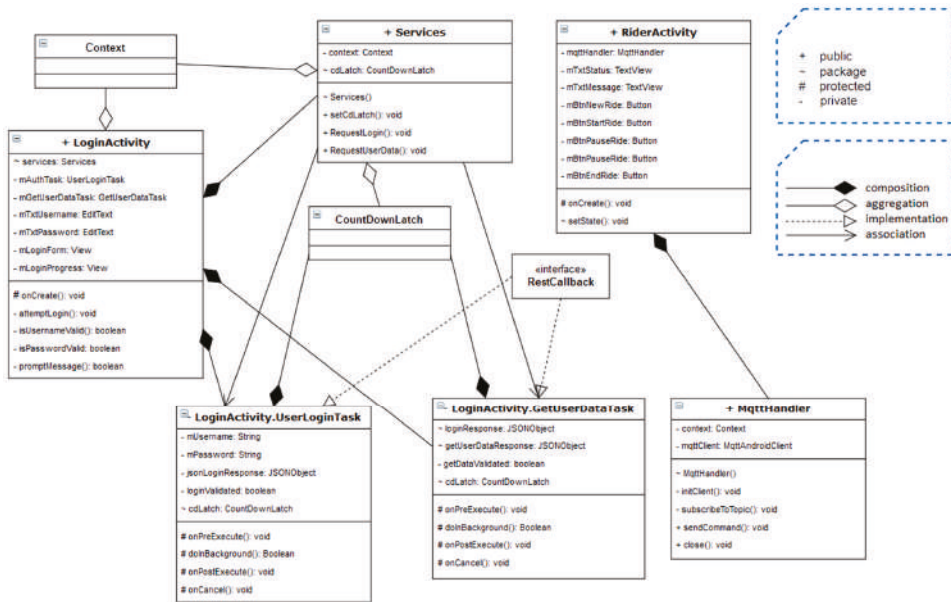
Figure 12. UML class diagram in Android application.

2.8. The Server

As most IoT systems, the Connected Bike project also has a server, which in this case consists of two main parts. The first part is the MQTT broker, which is necessary for the intermediation of messages between clients, while the second part is the Back-End for the web application, having both the role of persisting data transmitted from the bicycle and providing functionality to the Front-End.

Being a vital component in the operation of the entire system, the server must be switched on and have an uninterrupted Internet connection. The optimal solution was to use a Raspberry Pi 3 Model B+ to host the server for this project, being both low-cost and reliable. It is always powered on and connected to the internet, serving both the Back-End and the MQTT broker. In order to be accessible to the other modules, port forwarding rules were set on the router to which the Raspberry Pi is connected and the device received a static IP in the Local Area Network. Therefore, the other components of the system are able to access the services available on the server using the public IP of the router and the specified port. Next, are presented the two components that run on the server explaining them in more detail.

2.9. Back-End

The Back-End component is the one that directly accesses the database and implements the logic behind the web interface so that the data requested in the application is formatted and sent to the requester. In this project, the Back-End serves both the Front-End of the application and the Android Application. The communication protocol is HTTP, currently being one of the most used data transmission protocols over the internet.

The programming language in which the Back-End part was implemented is Python 3.6, using the open-source flask and flask-restful libraries. The design pattern used is Model-View-Controller (MVC), but only the Model and Controller are implemented on the Back-End side, because the Front-End is in charge of the View. In the development of the Back-End, the steps in the tutorial [25] have been followed and adapted.

The database used is PostgreSQL where all the users' data as well as the rides' data are stored. The tables and the relationships can be seen in Figure 13.

The database consists of two tables that maintain records of all the rides for each cyclist (ride_table) and the data recorded for each ride (ride_data). General user data is kept specific to all users, regardless of profile, such as username, password, and role. Because users may have different roles, namely cyclist or trainer, it means that the specific data for each category will be different. Therefore, the rider_info table was created to keep specific information for cyclists such as name, weight, bicycle, and topic, the last one being used to send commands from the Android App to the rider's bike frame module. The trainer_info keeps information about coaches such as name, discipline, level, and certificate.

Because a cyclist can have multiple trainers and a trainer can collaborate with multiple cyclists, there is a many-to-many relationship between the two entities. This was translated into a relationship table trainer_to_rider. This table keeps in each record the relation between a trainer and a cyclist through their primary keys.
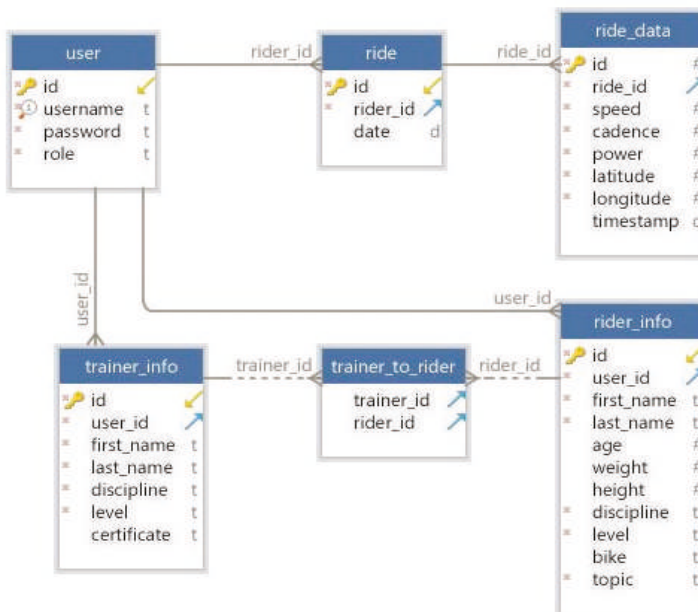

Figure 13. Database structure.

The database was defined first by writing the tables and relationships as model classes in separate Python modules, all centralized in a package named models. After this, the database was created directly from the defined classes using the SQLAlchemy library with the Code-First approach. This library is an Object Relational Mapper (ORM) designed for Python that allows users to map database tables and relationships to classes and work with entities to perform CRUD operations (Create Read Update Delete). Everything from database access to creating queries are performed in the background, allowing the developer to write complex database operations that are translated into the database-specific queries.

The other Python package is named controller. Here, the Back-End is divided into two functionalities. The first one is dedicated to users. The UserView.py module defines all the methods that provide functionality for mobile and web applications using the table dedicated to users. These methods can create a new account, modify user data, and validate login credentials by querying the user, trainer_info, rider_info and trainer_to_rider tables with the help of SQLAlchemy. Similar to

UserView.py, the RideView.py module contains all the methods dedicated to rides and rides data respectively, performing queries on the ride and ride_data tables. All the functions described above have a decorator that specifies the URL path and the type of REST (Representational state transfer) method through which they can be accessed remotely. Also, some methods may have more decorators depending on the access level. For example, the delete_ride method has both the *@Auth.auth_required* decorator, which allows it to be called only if the request has a token corresponding to an authenticated user, and the *@Auth.rider_required* decorator, which checks if the requester is a cyclist. These decorators are in fact other methods called implicitly to verify various requirements, such as whether the user is authenticated or has a certain role to access a resource. If the access criteria are satisfied, the decorated method will perform its task. All the decorators as well as the methods used to generate and decode JWT tokens are in the Authentication.py module. The encryption algorithm is SHA-256 and the Unicode standard is UTF-8.

In order for all the Back-End methods written in Python and described above to be remotely accessible over the Internet, a Web Server Gateway Interface (WSGI) is required. This mechanism handles all the HTTP requests coming from the Internet, routing them to the appropriate methods from the local machine. For this application, a WSGI server called Gunicorn was used, which is specifically designed to work with the Flask framework. It was configured to work as a daemon with three workers, so it did not interfere with the other running processes on the machine. These workers are actually threads in a pool, that are assigned per request to route to the corresponding method.

2.10. MQTT Broker

The communication between the MQTT clients was made possible by the Mosquitto broker, developed by Eclipse Foundation and released for public use for free. Once installed and configured, the broker starts as a background process when the Raspberry Pi finishes booting. The settings were configured in the mosquitto.conf file, enforcing the use of a username and password for all MQTT clients in this project for increased security. In addition, two listeners are configured here. The first uses the 1883 port for TCP/IP connection, which is used both for communication with the frame module and for sending commands and status information between the frame module and the Android Application. The second listener exposes the 8083 port for websockets connection, used by the Web Application for receiving online ride data from the frame module. Moreover, in this configuration file, the SSL certificates are also added for a more secure connection and the location where logs from Mosquitto broker will be stored.

2.11. Other Components

In addition to the Back-End and the MQTT broker, there are other auxiliary components that fulfill various roles on the Raspberry Pi server.

The first component is the Python service that subscribes to the data topic and retrieves ride data messages transmitted by the bicycle's frame module and saves it to the database based on the ride id from the message. This service started at the Raspberry Pi's boot time is waiting for the internet connection to be established, then it initializes the MQTT client and starts an infinite loop thread waiting for messages on the *connectedbike/data/#* topic. The hashtag character is a wildcard, meaning that all messages that are sent on the topic *connectedbike/data/{user_id}* will be received, regardless of the user id. This is advantageous as the ride_data table does not require a user id but only ride id, and this id is sent along with the data in the MQTT message.

Another important part of running the Back-End is the reverse proxy component. Its purpose is to hide implementation details and protect the Back-End component. All requests from clients are received by the reverse proxy that routes them to the WSGI and then routes back the response to the caller. For this project, NGINX was used as a reverse proxy, being a free and open-source software, supported by numerous contributors. After installation, it has been configured in the connectedbike.conf. Here were defined the access domains, the path to the Back-End, the files for

storing logs, and the server that contains the parameters and the proxy and access control headers. Certbot certificates for SSL encryption are also configured here. Thus, all requests received on the public domain connectedbike.cf or www.connectedbike.cf are forwarded by the proxy to the Gunicorn WSGI serving the Back-End. The operation of this assembly can be seen in Figure 14.

After installation, the NGINX server automatically starts as a daemon service when the Raspberry Pi is turned on. Also, at the time of activation, NGINX takes over and sets all the configurations identified in the file with the .conf extension from the /etc/nginx/sites-available/ folder, in this case being connectedbike.conf.

The third server component is the Cron utility. This is a simple service offered on most Linux operating systems. To configure the crontab file, which the service uses as a look-up file for scheduling tasks, the *crontab -e* command line is called from the command line. For this application, a command is added to the crontab to run at boot time on the Raspberry Pi and to call the mqttdataposter.sh script. This shell script changes the current working directory to the root of the Application folder containing the MqttDataPoster.py. Afterwards, the Application is started from inside its Python virtual environment.
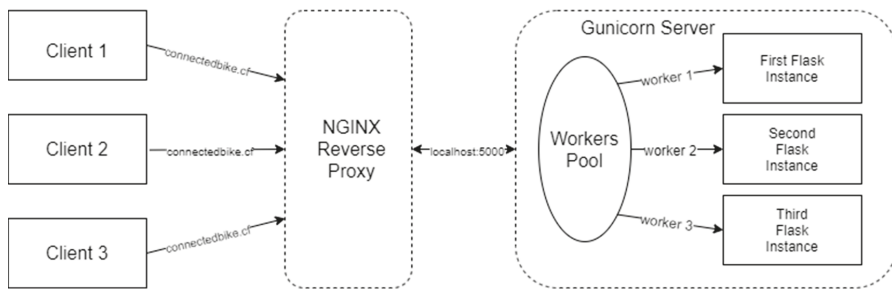

Figure 14. NGINX reverse proxy operation scheme.

Both the MQTT Mosquitto broker and the NGINX reverse proxy server are automatically started at boot time and run as background services. Gunicorn, however, does not offer this possibility, which is why a forth auxiliary component is required, namely the creation of a Systemd unit with the .service extension, that handles the activation of the Gunicorn server at boot time. Systemd is a utility offered within the Debian operating system that allows the creation of user-defined services, running in the background. The service created for Gunicorn is called connectedbike.service and consists of three parts.

The first part, defined by the [Unit] tag, contains a description of the service and specifies the dependencies required to run the service. In this case, they are network-online.target and network.target.

The second part represented by the [Service] tag specifies the details for starting the service. Here the following are defined:

- *User = pi:* the Linux user running the service
- *Group = www-data*: the user group that runs the process
- *PIDFile = /tmp/gunicorn.pid*: the file that stores the process id
- Environment="PATH=/home/pi/Programming/licentabackend/venv/bin" "FLASK_ENV=production" "DATABASE_URL=postgres://user:password@localhost:5432/db_ licenta_deployment" "JWT_SECRET_KEY=encodingstring": the virtual environment for the Back-End and the environment variables
- *WorkingDirectory = /home/pi/Programming/licentabackend*: the folder in which the Flask application is located

- ExecStart={path_to_directory}/venv/bin/gunicorn –workers 3 – access-logfile {path_to_directory}/ logs/gunicorn-access.log –error-logfile {path_to_directory}/logs/gunicorn-error.log -b localhost: 5000 wsgi: the command to be executed when calling the service. This command calls the WSGI file that instantiates the Flask application, using Gunicorn and the virtual environment. The server will start running locally, serving port 5000 with three threads available for handling requests
- *ExecReload = /bin/kill -s HUP $MAINPID*: specifies how the service can be restarted
- *ExecStop = /bin/kill -s HUP $MAINPID*: specifies how the service can be terminated

The third section defined by the [Install] tag allows the service to be enabled or disabled. So, using the WantedBy=multi-user.target directive, a folder with the same name is created that persists even if the device is restarted and activates the service at boot time. In order for the service to be active and running every time the Raspberry Pi is started, the shell command *sudo systemctl enable connectedbike.service* is required. The same command, but replacing the enable keyword with start or stop, can be called to begin or terminate the service.

Last but not least, the Raspberry Pi server has allowed firewall access to the ports required by the application. Also, for a more efficient and user-friendly use, a free domain name (connectedbike.cf) has been assigned to the server's IP.

2.12. Web Application

In this project, the web application has the role of displaying in a clear and easy way the data read by the module on the bicycle, giving the trainer the possibility to monitor the evolution of a cyclist online. In addition, the application offers the possibility of both categories of users to analyze previous rides. Also from the app, users can register a new account, login or connect with each other for collaboration. To achieve this, the ReactJS Javascript framework, developed and released free by Facebook, was used.

The application uses multiple components and elements to achieve the desired graphical user interface and user experience. For an easier state management, the react-redux library was used. It allows for centralizing the state for the entire application and passing only parts of it to components that specifically need them. The global application state can be modified only by functions named reducers. They receive as a parameter the current state of the application and the action based on which the decision to change the global state is taken.

For the implementation of this application, the tutorial presented by Jason Watmore in [26] is followed. The structure is divided into pages, auxiliary components and Redux components. In total, there are five pages: LoginPage, RegisterPage, RiderPage, TrainerPage and MonitoringPage. These in turn contain sub-pages, components rendered in the main page, which change according to the selected tab in the left-side menu. This allows conditional display of several different information, using the same URL. The user interface (buttons, labels, graphic elements) was created with the help of the MaterialUI library, which offers a wide range of free graphic elements. The monitoring page is accessible by both the trainer and the cyclist, being the place where the data is displayed, whether they come online or are uploaded from the Back-End. When the monitoring page is rendered in the browser, a props check is performed. If a ride id is found, then a function is called to bring the data through a GET request to the server. In the absence of a ride id, riderTopic is received in the props and the component subscribes to this topic to receive the data from the bicycle frame module in real time. To receive data through MQTT, the paho-mqtt library for React is used and the protocol supported by JavaScript is Websockets, which is why the broker on the server exposes both TCP/IP connection port and a Websocket connection port.

Because the data from the bicycle is transmitted and records each second, the number of entries in the database increases drastically. Thus, for the data to be fully displayed and analyzed, without occupying a large portion of the graphical interface, the Highcharts library for React was used. Three Highcharts charts were configured with this, one for speed, one for cadence and one for power. To each chart was attached a Navigator element available in the library that allows selection of a portion from a

large data set and the graphical visualization of that part only, thus facilitating a thorough study of the ride in more detail. Also on the monitoring page was added a map using the GoogleMaps library for React, that displays markers with latitude and longitude received from the GPS module on the bicycle.

The components folder also includes AlertBar.js, which is used to display labels with various user information about in-app events such as failed login or failed connection to Back-End. Also here have been defined a set of constants for enforcing private routes, respectively limiting access to the application for users who have not logged in. So, anyone can access the Login or Register page, but only users who have logged in and received an authorization token from the server can navigate through the rest of the application. In addition, these constants decide based on the response from the server whether the user is a cyclist or coach and redirects it to the appropriate page.

Finally, yet importantly, the Redux part is composed of three parts:

1. Actions—here are the functions that can be called to change the overall state of the application. These have been divided into functionalities for user, rider, trainer, and alert. Within them, in addition to the implemented logic, is called the dispatch method in which a constant is sent as a parameter, which is respectively the part of the global state that must be modified;
2. Constants—here are defined the constants based on which the behavior of the global state is decided. Similar to actions, constants are also divided into functionalities for user, trainer, rider, and alert;
3. Reducers—represent the functions that perform the actual change of the global state. They are called with the initial state or the action as parameters. Based on one of the constants in point two, which is transmitted in the body of the action, the state is modified with the new value sent in the body of the action;

The engine behind these three parts is the store. It is created using the createStore function in the redux library. This function is called with a set of parameters such as persistedReducer or middleware. In order to maintain the entire state of the application, even after the user refreshes the browser page, the redux-persist library is used. With it, the entire status hierarchy of the application is saved in Local Storage as JSON, where it can be retrieved again after refreshing the web page.

In the application development stage, the Node.js server was used. It has the role of running a local instance (localhost) at a predefined port, in this case, port 3000, where the developer can observe the changes made to the code. To expose the application on the internet and for anyone to access it, the npm run build command was run, creating a new build folder containing all the files and packages needed to run the application in production. This build was uploaded for free to Netlify, making it accessible to everyone on the internet.

3. Results

The main distinguishing features of the system are presented in the following:

*   The architecture of the project allows the system to be divided into dedicated modules, each implementing both data computing and transmission operations. This approach offers the advantage of reducing computational demand on a single module by distributing specific operations such as conversions or scaling to each module separately;
*   By using a real-time operating system on the Arduino Mega from the bicycle frame module, a clear definition of the tasks and their separation according to the fixed duration are obtained;
*   The data transmission between the pedal module to the frame module is done via infrared communication, overcoming the wires limitation imposed by the two rotational couplings from pedal to bike frame;
*   In order to use a single three-wire load cell, it was necessary to close the Wheatstone bridge with the resistors of equal value and minimum tolerances;
*   The Raspberry Pi was turned into a control component capable of stopping or starting recording ride activity depending on the commands received from the Android App;

- The MQTT Protocol was used as the main communication medium in the system, being used both for transmitting on-line data collected from sensors and for full-duplex communication between the bicycle module and the mobile phone. This reduced the complexity of the transmission and increased the robustness of the whole assembly;
- Multiple security levels were implemented such as password encryption with secret key, JWT authorization tokens for Back-End, restricting access to MQTT channels by username and password and implementing SSL certificates on the server for both Back-End and MQTT broker;
- The Redux library was used to manage the entire state of the application and its persistence in the local memory of the browser so that it is not reset between refreshes.

The pedal module, as mounted on the bicycle, can be seen in Figure 15.
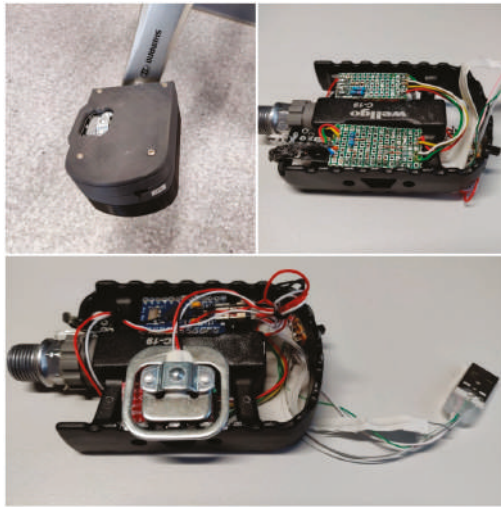
Figure 15. The final pedal module.

Also, the frame module, consisting of Arduino Mega 2560, Raspberry Pi 3 Model B+, Hall sensors, infrared receiver, and GPS module from Adafruit is shown in Figure 16.
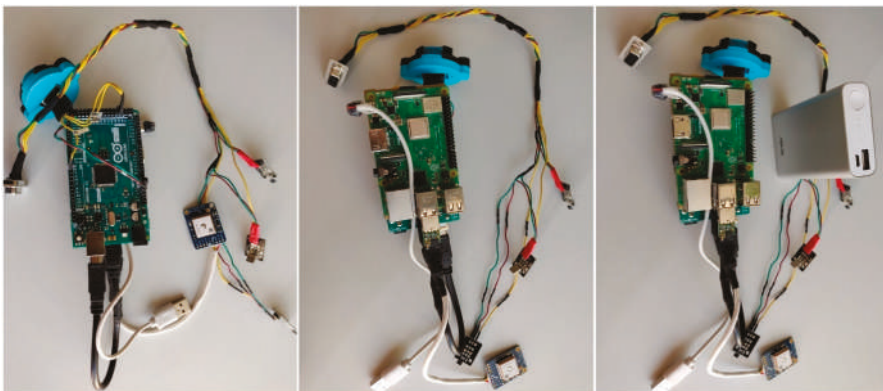
Figure 16. The resulted bicycle frame module.

In its final form, the hardware part of the frame module has been placed inside a bicycle water bottle and the connection with the sensors is made through an RS-232 connector as shown in Figure 17. This allows the bottle to be disconnected from the electronic components on the frame, leaving the sensors in place.

Figure 17. Final form of the Connected Bike

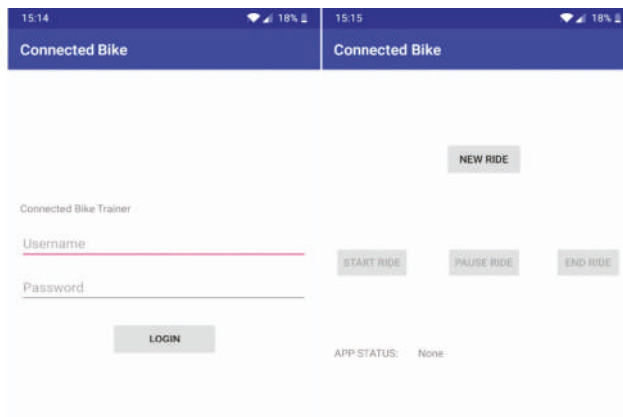The Android App that allows the cyclist to manage the ride data monitoring is presented in Figure 18.

Figure 18. Android application interface

Last but not least, the web application by which users can create their account, monitor the rides or create connections with each other is shown in Figure 19.
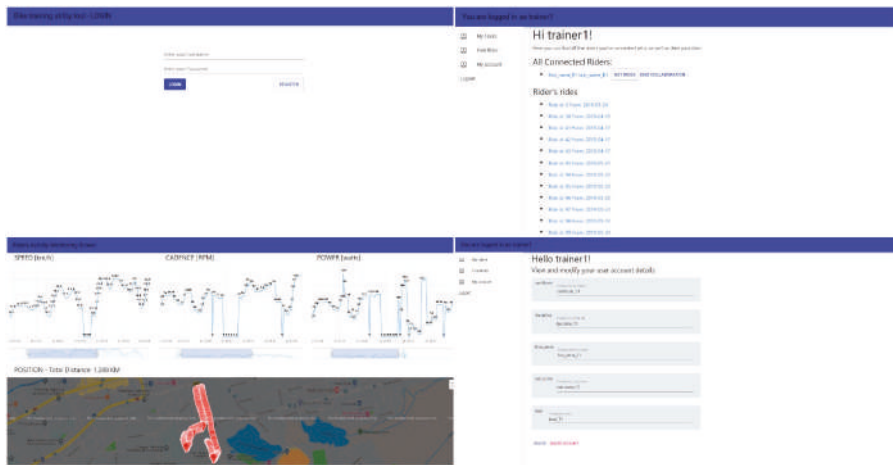
Figure 19. Web application of the Connected Bike

Finally, using the above modules, both hardware and software, the following requirements were achieved:

- Collecting raw data from the two Hall sensors and from the load cell
- Computing the riding speed [km/h] with a precision of two decimals, pedal cadence [rpm] and instantaneous power from the cyclist [watt]
- Track the cyclist's position in real-time by reading latitude and longitude from the GPS module
- Sending with a 1 s period the collected and processed data from sensors to the server to be stored in the database and for online monitoring
- Implementing an Android Application that allows the rider to create a new ride, start or terminate the current one, and also receive information on the state of the bike's frame module
- Creating a Web Application through which the data sent from the bicycle can be viewed both in real time and after the ride has been completed
- Securing the application by working with sessions, authorization tokens, SSL certificates, and credentials for the MQTT clients

4. Conclusions

This connected bike project was born out of a real issue encountered among both amateur and professional cyclists. This problem consists of the limitations existing in the current training methods. In this project, a wide range of technologies have been used, starting from the electronics and hardware side and to the web and mobile applications. Putting all these together, a prototype of a complete solution has been created to help cyclists and facilitate trainers.

Abbreviations

AUTOSAR	Automotive Open System Architecture
CRUD	Create Read Update Delete
GPIO	General-Purpose Input/Output
GPS	Global Positioning System
HTTP	Hypertext Transfer Protocol
ICR	Interrupt Capture Register
IoT	Internet of things
IP	Internet Protocol
IR	Infrared
JSON	JavaScript Object Notation
JWT	JSON Web Token
LED	Light-Emitting Diode
MIT	Massachusetts Institute of Technology
MQTT	Message Queues Telemetry Transport
MVC	Model-View-Controller
NGINX	engine X
ORM	Object Relational Mapper
RST	Representational state transfer
RTE	Runtime Environment
RTOS	Real Time Operating System
SEMS	smart e-bike monitoring system
TCNT	Timer Counter Register
TCP	Transmission Control Protocol
UART	Universal Asynchronous Receiver-Transmitter
UML	Unified Modeling Language
URL	Uniform Resource Locator
USB	Universal Serial Bus
WSGI	Web Server Gateway Interface

References

1. Salmeron-Manzano, E.; Manzano-Agugliaro, F. The Electric Bicycle: Worldwide Research Trends. *Energies* **2018**, *11*, 1894. [CrossRef]
2. Zguira, Y.; Rivano, H.; Meddeb, A. Internet of Bikes: A DTN Protocol with Data Aggregation for Urban Data Collection. *Sensors* **2018**, *18*, 2819. [CrossRef] [PubMed]
3. Shen, S.; Wei, Z.; Sun, L.; Su, Y.; Wang, R.; Jiang, H. The Shared Bicycle and Its Network—Internet of Shared Bicycle (IoSB): A Review and Survey. *Sensors* **2018**, *18*, 2581. [CrossRef] [PubMed]
4. Zguira, Y.; Rivano, H.; Meddeb, A. IoB-DTN: A lightweight DTN protocol for mobile IoT applications to smart bike sharing systems. In Proceedings of the 2018 IEEE Wireless Days (WD), Dubai, UAE, 3–5 April 2018. [CrossRef]
5. Nakamura, T.; Kikuya, Y.; Arakawa, Y.; Nakamura, M.; Higashijima, Y.; Maruo, Y.Y.; Nakamura, M. Proposal of web framework for ubiquitous sensor network and its trial application using no2 sensor mounted on bicycle. In Proceedings of the 12th International Symposium on Applications and the Internet (SAINT) 2012 IEEE/IPSJ, Izmir, Turkey, 16–20 July 2012; pp. 83–90.
6. Qiu, P.; Liu, X.; Wen, S.; Zhang, Y.; Winfree, K.N.; Ho, C. The Development of an IoT Instrumented Bike: For Assessment of Road and Bike Trail Conditions. In Proceedings of the 2018 International Symposium in Sensing and Instrumentation in IoT Era (ISSI), Shanghai, China, 6–7 September 2018. [CrossRef]
7. Archana, D.; Boomija, G.; Manisha, J.; Kalaiselvi, V.K.G. Mission on! Innovations in bike systems to provide a safe ride based on IOT. In Proceedings of the 2017 2nd International Conference on Computing and Communications Technologies (ICCCT), Chennai, Tamil Nadu, India, 23–24 February 2017. [CrossRef]
8. Chen, L.; Tsao, C.; Li, C.; Lo, Y.; Huang, W.; Chen, H. Demo: All-You-Can-Bike—A Street View and Virtual Reality Based Cyber-Physical System for Bikers through IoT Technologies. In Proceedings of the 25th Annual

International Conference on Mobile Computing and Networking, Los Cabos, Mexico, 21–25 October 2019; ACM Digital Library, Article No.: 60. pp. 1–3. [CrossRef]

9. Wang, C.; Kou, S.; Song, Y. Identify Risk Pattern of E-Bike Riders in China Based on Machine Learning Framework. *Entropy* **2019**, *21*, 1084. [CrossRef]

10. Danku, A.; Kovari, A.; Miclea, L.C.; Dulf, E.H. Intelligent Control of an Aerodynamical System. In Proceedings of the 2019 IEEE 15th International Conference on Intelligent Computer Communication and Processing (ICCP), Cluj-Napoca, Romania, 5–7 September 2019.

11. Dulf, E.H.; Timis, D.D.; Szekely, L.; Miclea, L.C. Adaptive Fractional Order Control Applied to a Multi-rotor System. In Proceedings of the IEEE 2019 22nd International Conference on Control Systems and Computer Science (CSCS), Bucharest, Romania, 28–30 May 2019.

12. Dulf, E.H.; Timis, D.; Muresan, C.I. Robust fractional order controllers for distributed systems. *Acta Polytech. Hung.* **2017**, *14*, 163–176.

13. Dong, J.L.; Hu, C.Q.; Tong, J.X.; Liu, F.Y. Velocity-free MS/AE source location method for three-dimensional hole-containing structures. *Engineering* **2020**. [CrossRef]

14. Dong, J.; Sun, D.J.; Han, G.J.; Li, X.B.; Hu, Q.C.; Shu, L. Velocity-free Localization of Autonomous Driverless Vehicles in Underground Intelligent Mines. *IEEE Trans. Veh. Technol.* **2020**. [CrossRef]

15. Bao, Y.; Chen, Z.; Wei, S.; Xu, Y.; Tang, Z.; Li, H. The state of the art of data science and engineering in structural health monitoring. *Engineering* **2019**, *5*, 234–242. [CrossRef]

16. Li, D.; Gao, H. A hardware platform framework for an intelligent vehicle based on a driving brain. *Engineering* **2018**, *4*, 464–470. [CrossRef]

17. Pan, Y. Heading toward artificial intelligence 2.0. *Engineering* **2016**, *2*, 409–413. [CrossRef]

18. Frauke, B. Cycling the Smart and Sustainable City: Analyzing EC Policy Documents on Internet of Things, Mobility and Transport, and Smart Cities. *Sustainability* **2019**, *11*, 763. [CrossRef]

19. Savage, N. Cycling Through Data. *Commun. ACM* **2010**, *53*, 16–17. [CrossRef]

20. Emilio, M.D.P. *Data Acquisition Systems from Fundamentals to Applied Design*; Springer: Pescara, Italy, 2013.

21. Al-Mutlaq, S.; Tutorials, S. SparkFun Electronics [Interactive]. Available online: https://learn.sparkfun.com/tutorials/getting-started-with-load-cells/all (accessed on 20 April 2019).

22. Ray, P. A survey on Internet of Things architectures. *J. King Saud Univ. Comput. Inf. Sci.* **2018**, *30*, 291–319.

23. The FreeRTOS Kernel [Interactive]. Available online: https://www.freertos.org/ (accessed on 5 June 2019).

24. MQTT [Interactive]. Available online: http://mqtt.org/ (accessed on 5 June 2019).

25. Aladeusi, O. Building a RESTful Blog APIs Using python and flask, 17 06 2018. [Interactive]. Available online: https://www.codementor.io (accessed on 11 February 2019).

26. Arduino Mega 2560 [Interactive]. Available online: https://store.arduino.cc/mega-2560-r3 (accessed on 3 June 2019).

High Precision Positioning with Multi-Camera Setups: Adaptive Kalman Fusion Algorithm for Fiducial Markers

Dragos Constantin Popescu [1,2,*]**, Ioan Dumitrache** [1,3]**, Simona Iuliana Caramihai** [1]
and Mihail Octavian Cernaianu [2,*]

[1] Faculty of Automatic Control and Computers, University Politehnica of Bucharest, Splaiul Independentei No. 313, 060042 Bucharest, Romania; ioan.dumitrache@acse.pub.ro (I.D.); simona.caramihai@aii.pub.ro (S.I.C.)

[2] Extreme Light Infrastructure-Nuclear Physics (ELI-NP)/Horia Hulubei National Institute for R&D in Physics and Nuclear Engineering (IFIN-HH), Str. Reactorului No. 30, Magurele, 077125 Bucharest, Romania

[3] Department of Science and Information Technology of the Romanian Academy, Cal. Victoriei No. 125, 010071 Bucharest, Romania

[*] Correspondence: dragos.popescu@acse.pub.ro (D.C.P.); mihail.cernaianu@eli-np.ro (M.O.C.)

Abstract: The paper addresses the problem of fusing the measurements from multiple cameras in order to estimate the position of fiducial markers. The objectives are to increase the precision and to extend the working area of the system. The proposed fusion method employs an adaptive Kalman algorithm which is used for calibrating the setup of cameras as well as for estimating the pose of the marker. Special measures are taken in order to mitigate the effect of the measurement noise. The proposed method is further tested in different scenarios using a Monte Carlo simulation, whose qualitative precision results are determined and compared. The solution is designed for specific positioning and alignment tasks in physics experiments, but also, has a degree of generality that makes it suitable for a wider range of applications.

1. Introduction

The evolution of technology has led to an increasing demand for solving complex problems which may be viewed as attempts to control and direct system behaviours towards desired states. The inherent complexity of problems and processes requires new approaches both in system modelling and in defining the emergent interaction with a highly dynamical and sparsely defined environment.

Cognitive approaches are successfully used in contexts where the boundary between the systems and the environment is fuzzy. However, they exhibit strong interrelation and interconnection, assisted by specific perception mechanisms [1]. Advances in complex control applications can be achieved only by considering adequate design approaches for sensory systems, especially in domains like environmental applications [2], health [3], industrial control, agriculture, etc.

Although new technologies such as wireless sensor networks or Internet of Things (IoT) are providing valuable solutions for appropriate perception mechanisms, complex issues are raised with the inclusion of data fusion, reliability, flexibility, reconfigurability, and cost of the measurement system. If some of these problems can be addressed during the modelling process, there are others (e.g., sensor positioning and sensor dynamics) that have a bigger impact on the generality of the overall solution.

These issues are specific in critical infrastructures, research institutes, power systems, e-health, mining, and traffic control, where the multitude of concurrent dynamics generating a large amount

of information requires adequate solving methods that cover aspects such as relevance, efficiency, and cost optimisation.

This paper follows the development of a measurement method used in a very precise and yet flexible and portable positioning system. Firstly, the method is an answer to a specific application: Automatic alignment with high precision of various instruments which have to be remotely operated in highly flexible, open, and dynamic configurations for physics experiments. However, it can be used in any application where high precision positioning over a large working area is required, where no absolute reference can be defined, or when it is necessary to simultaneously align multiple features in relative positions, with high reliability. Such applications include large scale construction sites where multiple components need to be positioned in precise locations, manufacturing facilities where large machines are assembled or spacecraft docking in zero gravity environments [4].

The article has the following structure. The specific application and context addressed in this work are presented in Section 2. The theoretical context behind the proposed algorithm is described in Section 3. It it then used to build two essential procedures for the positioning system. The precision of the proposed method is assessed in the Section 4 using a Monte Carlo simulation. The conclusions and future developments are detailed in Section 5.

2. Description of the Initial Problem

The large number of high repetition rate, ultrashort pulse, and high power laser facilities that will come online all around the world will require state-of-the-art tools to allow the harnessing of their full potential [5]. The high power lasers of the ELI-NP (Extreme Light Infrastructure-Nuclear Physics) user facility will be employed for a wide range of research topics like studies of nuclear induced reactions, dark matter search, material irradiation, or medical applications [6]. The development of the experimental setups for studying these topics requires specific instrumentation, while also having strict needs in terms of positioning and alignment, in order to ensure optimal experimental conditions.

As the setups are continuously changing, absolute position referencing is hard to achieve. This is a necessity, as multiple instruments need to be precisely positioned relative to each other during the experiment. Figure 1 displays an example of a setup for solid target alignment in which multiple optical diagnostics are positioned using motorised manipulators that have 3 to 6 degrees of freedom (DOF).
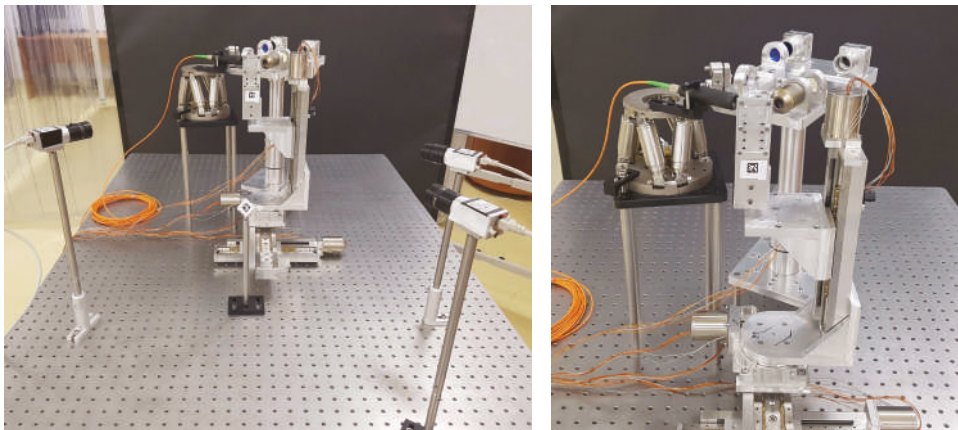

Figure 1. Solid target alignment setup example where multiple instruments are positioned using motorised manipulators.

Apart from the precision requirements, additional ones should be taken into consideration. The alignment should be done remotely because the experiments take place inside vacuum chambers and behind concrete walls used for radiation shielding. Moreover, to take advantage of the high

repetition rate of the lasers and to maximise the beam-time available for the users, the positioning has to be performed in a limited amount time. In a previous work [7], we addressed this series of requirements and constraints by developing an automatic alignment system that is based on relative position measurements using imaging cameras and compact and flat fiducial markers, due to space and visibility constraints. The alignment algorithm was based on a real-time optimisation procedure which is the subject of a patent [8].

Although the fiducial markers were initially developed and used in augmented reality applications [9], due to their versatility in determining their position in a non-invasive manner (by imaging them with a camera), they were quickly adopted for a different range of applications. These include kinematic calibration and visual servoing for industrial robots [10,11], robot localisation and navigation [12–14], SLAM (simultaneous localisation and mapping) [15–17] and sensor fusion [18–20].

The motivation behind this work is to combine the measurements from multiple cameras in order to increase the working area of the system and to maximise the positioning precision. Simultaneously, the main focus is to present and test the precision of the proposed methods and algorithm, comparing the results with those recorded while using only one camera [21].

3. Method

The basic approach behind the proposed method is to use multiple cameras in order to improve the precision of estimating the pose of fiducial markers and to extend the working area of the system. It begins with the assumption that each camera provides measurements that are erroneous and noisy. The problem can be conceived through an analogy with the GPS system where distance measurements from multiple satellites are used to estimate the position of the receiver module with high precision. The key ingredient in the GPS system is to know very precisely the position of the satellites. The proposed positioning system needs to meet the same requirement for its cameras, but wasting too much effort on this task diminishes its practicality. Consequently it is only assumed that the cameras have unknown but fixed positions.

A related approach that has similar objectives can be found in [22]. The main benefit of the method is the possibility to calibrate the setup of cameras using a 3D feature with fiducial markers having unknown configuration. However, it is not meant for estimating the pose of single markers. In order to achieve this, additional methods are required.

Our method is developed using the ArUco fiducial markers [23,24]. The ArUco library detects and estimates the pose of the marker with respect to the camera using the solvepnp algorithm in which the pinhole camera model is data fitted using a Levenberg–Marquardt non-linear optimisation procedure. The output is represented by the extrinsic camera parameters which can be expressed in terms of a homogenous transformation T_C^M (the transformation between the camera attached reference frame and the marker attached one) detailed in [7].

3.1. Adaptive Kalman Fusion Algorithm for Multiple Cameras and Fiducial Markers

The method consists of fusing noisy measurements from multiple cameras, in order to improve the precision of estimating the exact position of the fiducial marker. In the scientific literature, multiple sensor fusion and noise filtering methods have been developed over the years. Among them there is the Kalman filter [25], which is the most widely used. The noise effect is mitigated by using a dynamical model for the physical process involved and a statistical model (covariance matrix) for the noise in the measurement process.

For a discrete linear state-space dynamical model in Equation (1), the Kalman filter estimates the value of the state vector x using noisy measurements for the input u and the output y.

$$\begin{cases} x_k = A \cdot x_{k-1} + B \cdot u_{k-1} + q_{k-1} \\ y_k = C \cdot x_k + r_k \end{cases}. \tag{1}$$

In standard data fusion applications that use the Kalman filter, the pose of various objects is estimated using the input from accelerometer, gyroscope, and magnetometer sensors [26,27] and the output from distance measurement devices [28,29]. Our application makes impossible to use any type of attached sensors and, hence, we only rely on the "non-invasive" pose measurement using the solvepnp algorithm and imaging cameras.

Our approach is to employ a state-space model with free dynamics (where u is zero) and with an identity state matrix (A). In this respect, each measurement is made using only one snapshot image from all the cameras that are synchronised with the help of an electrical trigger signal. In this way, the position of the fiducial marker during the measurement is considered fixed. On the same set of images, the solvepnp algorithm is applied multiple times and, thus, the evolution of the measurement is caused only by the noise and not by the movement of the fiducial marker. Furthermore, the measurements are used to iteratively estimate the real position using the proposed algorithm.

In order to improve the results, we propose a procedure designed to update the statistical model of the noise. The Q and R covariance matrices (which are discussed below) are continuously adjusted using the newly acquired data. Any change of the noise behaviour and any existing correlations are captured and thus, the effect of the noise is mitigated more efficiently. Consequently, the proposed algorithm is an adaptive Kalman version.

The homogeneous transformation representation (T_C^M) has numerous practical benefits, especially for pose composition and inversion operations, but it is not suitable in this circumstance because it is redundant (16 numerical values for expressing a 6 DOF pose). The solution is to use an equivalent representation which is composed of the set of translation coordinates (X, Y, and Z) and the set of Euler angles (AX, AY, and AZ).

In this particular case, the structure of the state vector is the real pose of the fiducial marker defined in Equation (2), where the subscript k denotes the present discrete-time sample of the state vector.

$$x_k = \begin{bmatrix} X_k \\ Y_k \\ Z_k \\ AX_k \\ AY_k \\ AZ_k \end{bmatrix}. \tag{2}$$

The output vector is composed of the pose elements measured using all the cameras available. The structure of the output vector is defined in Equation (3), where the superscript i is indicating the index of the camera considered.

$$y_k = \begin{bmatrix} X_k^1 \\ Y_k^1 \\ Z_k^1 \\ AX_k^1 \\ AY_k^1 \\ AZ_k^1 \\ \vdots \end{bmatrix} \begin{matrix} \vdots \\ X_k^i \\ Y_k^i \\ Z_k^i \\ AX_k^i \\ AY_k^i \\ AZ_k^i \\ \vdots \end{matrix}. \tag{3}$$

The discrete-time state-space model considered is defined in Equation (4), where I_6 is the identity matrix of rank 6, q_{k-1} is a random noise signal with normal distribution (white noise) that is quantifying the false evolution induced by the noisy measurement, the H is the measurement model matrix defined in Equation (5), and r_k is the measurement noise also considered white noise.

$$\begin{cases} x_k = I_6 \cdot x_{k-1} + q_{k-1} \\ y_k = H \cdot x_k + r_k \end{cases} \tag{4}$$

$$H = \begin{bmatrix} I_6^1 \\ \vdots \\ I_6^i \\ \vdots \end{bmatrix}. \tag{5}$$

The equations that describe the Kalman filter are presented in Equation (6). The first two equations give a rough state estimate using the dynamical model while the last 5 equations are used for improving the estimate using the newly acquired output sample. P is a covariance matrix that expresses the confidence degree of the state estimation, which is updated during the algorithm iterations, Q and R are the covariance matrices associated with the noises q and r respectively, v_k is the rough estimation error (difference between the measured output and the predicted one using the first two equations), S_k is the covariance matrix associated with the predicted output, and K_k is the Kalman state update.

Predict the state

1. $\hat{x}_{k|k-1} = A \cdot \hat{x}_{k-1|k-1}$
2. $P_{k|k-1} = A \cdot P_{k-1|k-1} \cdot A^T + Q_{k-1}$

Update the prediction

3. $v_k = y_k - H_k \cdot \hat{x}_{k|k-1}$
4. $S_k = H_k \cdot P_{k|k-1} \cdot H_k^T + R_k$
5. $K_k = P_{k|k-1} \cdot H_k^T \cdot S_k^{-1}$
6. $\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \cdot v_k$
7. $P_{k|k} = P_{k|k-1} - K_k \cdot S_k \cdot K_k^T$

$$\text{(6)}$$

The filter requires good estimates for the initial state x_0 and the covariance matrices Q and R. In the proposed algorithm this is achieved using an initialisation procedure. The position of the fiducial marker is measured for N_i number of sampling times. The result is the set of samples defined in Equation (3) for $k = 1, \ldots, N_i$ (the length of the initialisation) and $i = 1, \ldots, n$ (the number of cameras available).

Averaging the samples gives a good estimate for the initial state, which is built according to Equation (7), where $E[\cdot]$ is the mean operator (expected value). The Q and R matrices are computed from the same set of samples, assuming that there is no correlation between the noises affecting the measurements.

$$x_0 = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ AX_0 \\ AY_0 \\ AZ_0 \end{bmatrix} = \begin{bmatrix} \underset{\substack{1 \le k \le N_i \\ 1 \le i \le n}}{E}\left[X_k^i\right] \\ \underset{\substack{1 \le k \le N_i \\ 1 \le i \le n}}{E}\left[Y_k^i\right] \\ \underset{\substack{1 \le k \le N_i \\ 1 \le i \le n}}{E}\left[Z_k^i\right] \\ \underset{\substack{1 \le k \le N_i \\ 1 \le i \le n}}{E}\left[AX_k^i\right] \\ \underset{\substack{1 \le k \le N_i \\ 1 \le i \le n}}{E}\left[AY_k^i\right] \\ \underset{\substack{1 \le k \le N_i \\ 1 \le i \le n}}{E}\left[AZ_k^i\right] \end{bmatrix} \tag{7}$$

$$Q_0 = diag \left(\underset{\substack{1 \leq i \leq n \\ 1 \leq k \leq N_i}}{E} \left[\underset{1 \leq k \leq N_i}{var} \left(X_k^i \right) \right], \underset{\substack{1 \leq i \leq n}}{E} \left[\underset{1 \leq k \leq N_i}{var} \left(Y_k^i \right) \right], \underset{\substack{1 \leq i \leq n}}{E} \left[\underset{1 \leq k \leq N_i}{var} \left(Z_k^i \right) \right], \right.$$

$$\left. \underset{1 \leq i \leq n}{E} \left[\underset{1 \leq k \leq N_i}{var} \left(AX_k^i \right) \right], \underset{1 \leq i \leq n}{E} \left[\underset{1 \leq k \leq N_i}{var} \left(AY_k^i \right) \right], \underset{1 \leq i \leq n}{E} \left[\underset{1 \leq k \leq N_i}{var} \left(AZ_k^i \right) \right] \right) \quad (8)$$

$$R_0 = diag \left(R_0^1, \dots, R_0^i, \dots, R_0^n \right) \quad (9)$$

$$R_0^i = diag \left(\underset{1 \leq k \leq N_i}{var} \left(X_k^i \right), \underset{1 \leq k \leq N_i}{var} \left(Y_k^i \right), \underset{1 \leq k \leq N_i}{var} \left(Z_k^i \right), \right.$$

$$\left. \underset{1 \leq k \leq N_i}{var} \left(AX_k^i \right), \underset{1 \leq k \leq N_i}{var} \left(AY_k^i \right), \underset{1 \leq k \leq N_i}{var} \left(AZ_k^i \right) \right). \quad (10)$$

Q is a diagonal matrix defined in Equation (8) built using the mean variance of the pose elements (X, Y, Z, AX, AY, and AZ) along all the cameras. The R matrix is defined in Equation (9) where R^i is defined in Equation (10), built using the variance for each pose element measured by each camera.

After the initialisation is finished, the Kalman algorithm is iterated for N_e number of sampling times, while at each step, a new set of samples y_k in the form of Equation (3) is measured and an estimated state trajectory is built ($\hat{x}_{k|k}$ for $k = N_i + 1, \dots, N_i + N_e$).

Newly measured system outputs can be used to improve the statistical models of the noises for increased performance. Thereby, at each iteration, every new set is added to the initialisation set and the covariance matrices Q and R are updated using Equations (11)–(13) for $k = 1, \dots, N_e$.

$$Q_k = diag \left(\underset{1 \leq i \leq n}{E} \left[\underset{1 \leq j \leq N_i+k}{var} \left(X_j^i \right) \right], \underset{1 \leq i \leq n}{E} \left[\underset{1 \leq j \leq N_i+k}{var} \left(Y_j^i \right) \right], \underset{1 \leq i \leq n}{E} \left[\underset{1 \leq j \leq N_i+k}{var} \left(Z_j^i \right) \right], \right.$$

$$\left. \underset{1 \leq i \leq n}{E} \left[\underset{1 \leq j \leq N_i+k}{var} \left(AX_j^i \right) \right], \underset{1 \leq i \leq n}{E} \left[\underset{1 \leq j \leq N_i+k}{var} \left(AY_j^i \right) \right], \underset{1 \leq i \leq n}{E} \left[\underset{1 \leq j \leq N_i+k}{var} \left(AZ_j^i \right) \right] \right) \quad (11)$$

$$R_k = diag \left(R_k^1, \dots, R_k^i, \dots, R_k^n \right) \quad (12)$$

$$R_k^i = diag \left(\underset{1 \leq j \leq N_i+k}{var} \left(X_j^i \right), \underset{1 \leq j \leq N_i+k}{var} \left(Y_j^i \right), \underset{1 \leq j \leq N_i+k}{var} \left(Z_j^i \right), \right.$$

$$\left. \underset{1 \leq j \leq N_i+k}{var} \left(AX_j^i \right), \underset{1 \leq j \leq N_i+k}{var} \left(AY_j^i \right), \underset{1 \leq j \leq N_i+k}{var} \left(AZ_j^i \right) \right). \quad (13)$$

Considering that the estimated state trajectory ($\hat{x}_{k|k}$) belongs to a system with free dynamics where, in the absence of the noise effects, the state should be constant, a final estimate with a better precision can be achieved by averaging the values of the estimated state trajectory according to Equation (14). The timeline of all the procedures that are involved in the proposed algorithm is presented in Figure 2. In Algorithm 1 the proposed procedure is summarised.

$$\hat{x} = \underset{N_i+1 \leq k \leq N_i+N_e}{E} \left[\hat{x}_{k|k} \right] \quad (14)$$
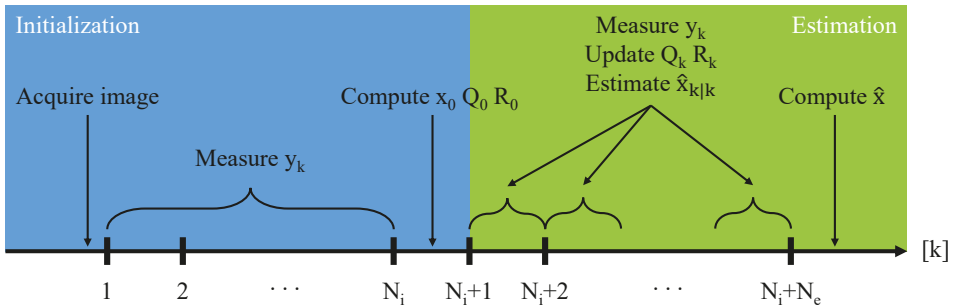
Figure 2. The sequence of procedures involved in the proposed algorithm.

Algorithm 1: The proposed algorithm for multi-camera pose fusion.

---

1 Acquire one image from all n available cameras;
2 Apply solvepnp algorithm on each image for N_i times and build the initialisation set from
   Equation (3), where $k = 1, \ldots, N_i$;
3 Compute x_0 using Equation (7);
4 Compute Q_0 using Equation (8);
5 Compute R_0 using Equations (9) and (10);
6 **for** N_e *times* **do**
7 ⎪   Compute $\hat{x}_{k|k-1}$ and $P_{k|k-1}$ using Equations (6).1 and (6).2 respectively;
8 ⎪   Apply solvepnp algorithm on each image and append a new set from Equation (3) to the
   ⎪   initialisation set;
9 ⎪   Update Q using Equation (11);
10 ⎪   Update R using Equations (12) and (13);
11 ⎪   Compute $\hat{x}_{k|k}$ and $P_{k|k}$ using Equations (6).3–7;
12 ⎪   Add $\hat{x}_{k|k}$ to the estimated state trajectory;
13 Compute $\hat{x}$ using Equation (14);

3.2. Setup Calibration Procedure

Given a set of n cameras in a pre-defined fixed configuration, having unknown absolute positions, the purpose of the setup calibration procedure is to determine, in a precise manner, their relative positions. As depicted in Figure 3, the goal is to determine the set of homogenous transformations T_{Ci}^{Cj} where $i = 1, \ldots, n$ and $j = i + 1, \ldots, n$.
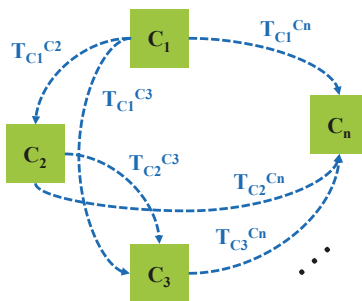
Figure 3. The calibration of a set of n cameras requires to determine their relative positions T_{Ci}^{Cj}.

This requires the use of a precision gauge, which can be manufactured in the form of a cube of fiducial markers like in [21] or in any 3D shape where the markers can be viewed all around. The gauge needs to be manufactured or calibrated with increased precision. It can be seen as the absolute precision reference used to calibrate the entire positioning system. In Figure 4, a conceptual diagram of a gauge containing m fiducial markers is presented. The set of homogenous transformations between all the markers (T_{Mi}^{Mj} where $i = 1, \ldots, m$ and $j = i + 1, \ldots, m$) is precisely known.
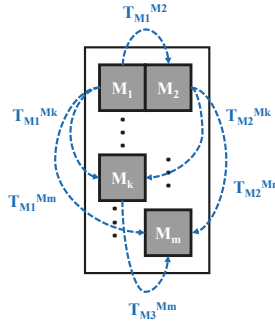
Figure 4. A setup of cameras is calibrated using a precision gauge. It is a manufactured feature having m fiducial markers where all their relative positions are precisely known.

The camera setup calibration procedure is done simultaneously for all the camera pairs C_i, C_j where $i = 1, \ldots, n$ and $j = i + 1, \ldots, n$, with the goal of estimating the homogenous transformation T_{Ci}^{Cj} (as presented in Figure 5). It starts with placing the precision gauge inside the environment. Depending on the orientation, each camera sees a different portion of the gauge, i.e., from all n fiducial markers, C_i can measure the position of n_i markers and C_j can measure only n_j, where $n_i \leq n$ and $n_j \leq n$. It is preferable that at least one fiducial marker M_k is seen by both cameras, otherwise, it can be virtually determined using the gauge transformations.

The proposed algorithm estimates the T_{Ci}^{Mk} homogenous transformation using the noisy measurements from all n_i markers using the C_i camera. This transformation is expressed multiple times in terms of T_{Ci}^{Ml} pose measurement (where $l = 1, \ldots, n_i$) using Equation (15). The T_{Ml}^{Mk} transformation is precisely known from the gauge.

$$T_{Ci}^{Mk} \approx T_{Ci\ l}^{\hat{M}k} = T_{Ci}^{Ml} \cdot T_{Ml}^{Mk} \tag{15}$$

The resulting $T_{Ci\ l}^{\hat{M}k}$ pose is converted to translations and Euler angles which are used for building the output measurement vector of Equation (3).
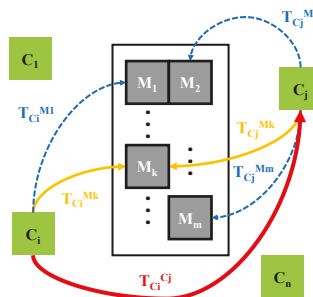

Figure 5. The process of calibrating all the pairs of n cameras, which illustrates all the homogenous transformations involved in finding the relative position of C_i and C_j cameras.

The proposed algorithm is further applied and the result is an estimation of T_{Ci}^{Mk} having a higher degree of precision and being closer to the real value. For the C_j camera, T_{Cj}^{Mk} is estimated in a similar manner. Consequently, T_{Ci}^{Cj} transformation is computed using Equation (16).

$$T_{Ci}^{Cj} = T_{Ci}^{Mk} \cdot \left(T_{Mk}^{Cj} \right)^{-1} \tag{16}$$

3.3. Position Estimation Procedure

Having a calibrated camera setup, the aim of this procedure is to estimate the position of a fiducial marker attached to a specific instrument, according to the application.

In Figure 6, the conceptual diagram of this procedure is depicted. Depending on the configuration, the M marker can be seen only by a number of n_m out of n cameras ($n_m \leq n$). An arbitrary camera (C_r) is chosen, which is considered the positioning reference.
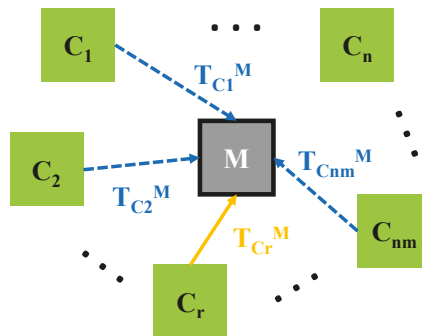
Figure 6. The homogenous transformations involved in estimating the position of a fiducial marker placed in the environment of a calibrated camera setup.

The proposed algorithm is estimating the T_{Cr}^{M} homogenous transformation using the noisy measurements from each of the n_m cameras. In a similar manner to the setup calibration procedure, the T_{Cr}^{M} transformation is expressed multiple times in terms of T_{Ck}^{M} pose measurement (where $k = 1, \ldots, n_m$) using Equation (17). The T_{Cr}^{Ck} is precisely known from the setup calibration.

$$T_{Cr}^{M} \approx T_{Crk}^{\hat{M}} = T_{Cr}^{Ck} \cdot T_{Ck}^{M}. \tag{17}$$

The resulting $T_{Crk}^{\hat{M}}$ pose is converted to translations and Euler angles and used for building the output measurement vector of Equation (3). The proposed algorithm is further applied and the result is an estimation of T_{Cr}^{M} which has a higher degree of precision and is closer to the real value.

This approach can be used in a similar manner for measuring the relative position of multiple fiducial markers in the environment, which is very useful in alignment tasks as presented in [7].

4. Simulation Results

4.1. Monte Carlo Setup

There are two factors that contribute and affect the precision of estimating the pose of the fiducial marker. First, there are the physical and environment-related aspects, which include the optical specifications of the imaging system (the sensor and optical resolution, the focal length, the depth of field, and the field of view), the environment illumination conditions (how strong, uniform, and consistent the lighting is) and if the cameras are optimally positioned so as to achieve good viewing angles and maintain consistent accuracy along the working area. Secondly, there are

factors related to the algorithms regarding how precise the pose can be estimated and how much the effect of the noise can be mitigated during the data fusion. In this paper we decouple the two types of factors and only consider the behaviour of our proposed method, so as to have a first qualitative assessment regarding the precision.

Consequently, we developed a simulation environment that aims to replicate how our method performs in a real setup, considering the noisy pose estimations it receives from the solvepnp algorithm. For an increased confidence in the results, we adopted a Monte Carlo approach in which we statistically analysed how the noise is propagated through our method and how the precision is affected. The simulation is implemented in MATLAB where positions of multiple cameras, precision gauges, and fiducial markers are virtually defined. In order to simulate the noisy input from the solvepnp, each position (that is supposed to be measured in the real environment) is disturbed with additive random noise having normal distribution. The noise is configured considering the precision limits we determined experimentally for one camera in our previous work [21]: for X and Y, 75 µm, for Z, 300 µm, and for AX, AY and AZ, 0.02°. The mean value of the noise is 0 while the standard deviation (σ) was configured in such a way that 95.45% of the noise values are within the experimental limits (inside $[-2\sigma, 2\sigma]$).

The simulation puts in place three scenarios which are additionally used to assess the different contributions between the number of cameras and the number of the fiducial markers from the gauge:

1. 3 cameras, a precision gauge with 5 markers, and one marker whose position must be estimated;
2. 5 cameras, a precision gauge with 3 markers, and one marker whose position must be estimated;
3. 5 cameras, a precision gauge with 5 markers, and one marker whose position must be estimated.

The results are compared against estimating the pose using only one camera in the same environment. The Monte Carlo simulation performs 5000 iterations where, in each run and for each of the scenarios, the setup is calibrated using the gauge. The calibration is further used to estimate the pose of the fiducial marker which is compared with the true, predefined one. The N_i and N_e parameters are configured to 20 and 50 respectively. In a real application, the choice of these two parameters is a matter of cost optimisation, considering the computational effort, the resources available, and the required measuring frequency.

Compared with a real setup, we adopted one simplifying hypothesis. The angle of incidence of the marker relative to the camera, as we showed in [21], has an influence on the precision. Close to normal angles of incidence tend to bring more noise in the estimation. In this study, we only consider that the precision of the solvepnp algorithm to be consistent, regardless of the angle. However, in other respects, the simulation is considering the worst case scenario because of the following arguments:

*   Noises from different cameras and from different elements of the pose (X, Y, Z, AX, AY, and AZ) are considered completely uncorrelated. In real circumstances this might not be the case (e.g., the noise induced by the environment illumination which affects all the measurements in a similar manner) thus, any relaxed conditions are contributing to an increased precision of the estimation. This additional information is harnessed using the update procedure for Q and R covariance matrices (which in this case would no longer be diagonal);
*   The precision of one camera measurement along the Z axis is 5 times lower compared with the X and Y axis. In the simulation this is taken as it is, but in a real situation, this effect can be diminished by placing the camera setup in an optimal configuration. For instance, the measurement along the Z axis from one camera can be replaced by measurements from two cameras placed in lateral positions, for which the Z axis motion is decomposed in X and Y components that have a higher precision.

4.2. Results

Figures 7–12 presents the simulation results for estimating X, Y, and Z coordinates and AX, AY, and AZ orientation angles, which are given in terms of the probability density function (PDF) and

the standard deviation (SD) of the error. The first plot from each Figure depicts the estimation error achieved using only one camera. The following three plots give the results achieved using the setups from each of the above-mentioned scenarios.

The results show that the proposed algorithm achieves an increase in precision which is close to an order of magnitude. It can also be observed that it is of greater importance to have more fiducial markers in the precision gauge instead of more cameras. In scenario #3, a slight decrease of precision is experienced in comparison to the #1 scenario. This is to be expected as each added camera is an additional noise source. However, the benefit of achieving a larger working area is far more important. In Table 1 the results are summarised and compared with regards to the limits of the variation interval $[-2\sigma, 2\sigma]$ where it is expected that 95.45% of the errors will occur. This can be considered the precision that the positioning system is expected to achieve when using th proposed method.
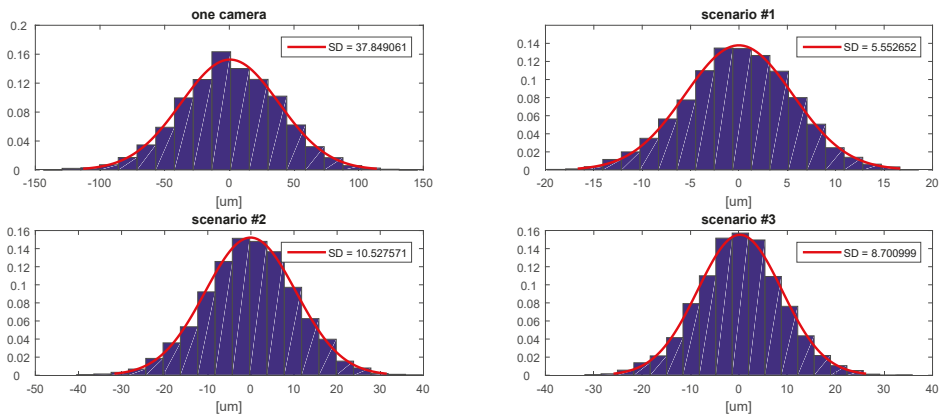
Figure 7. The probability density function (PDF) and the standard deviation (SD) of the error for estimating the *X* coordinate.
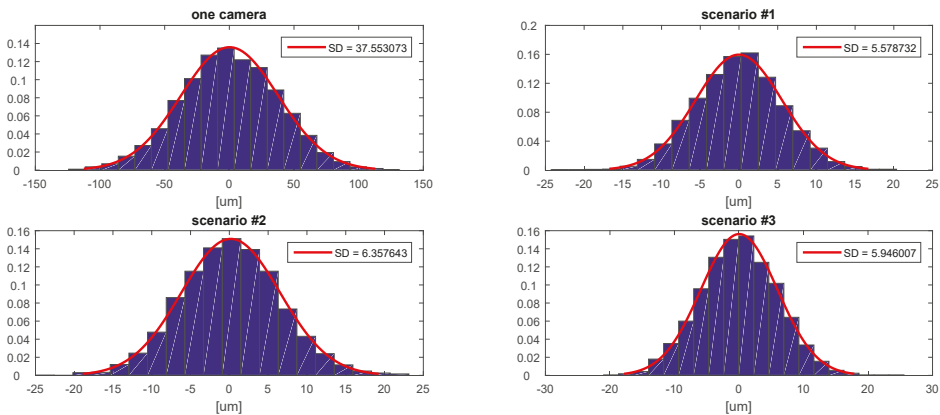
Figure 8. The probability density function (PDF) and the standard deviation (SD) of the error for estimating the *Y* coordinate.
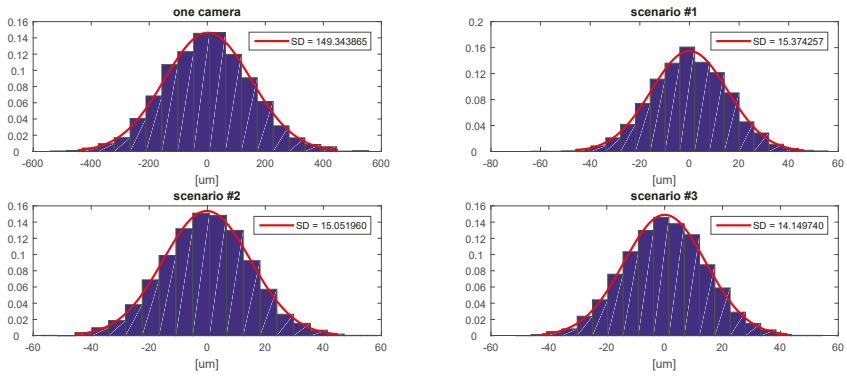
Figure 9. The probability density function (PDF) and the standard deviation (SD) of the error for estimating the *Z* coordinate.
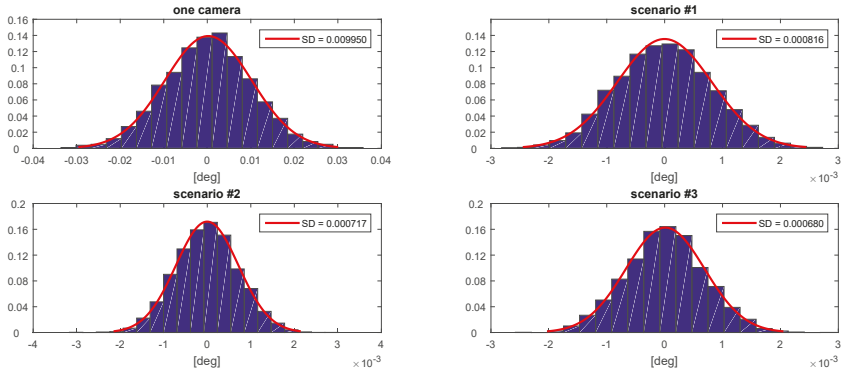

Figure 10. The probability density function (PDF) and the standard deviation (SD) of the error for estimating the *AX* angle.
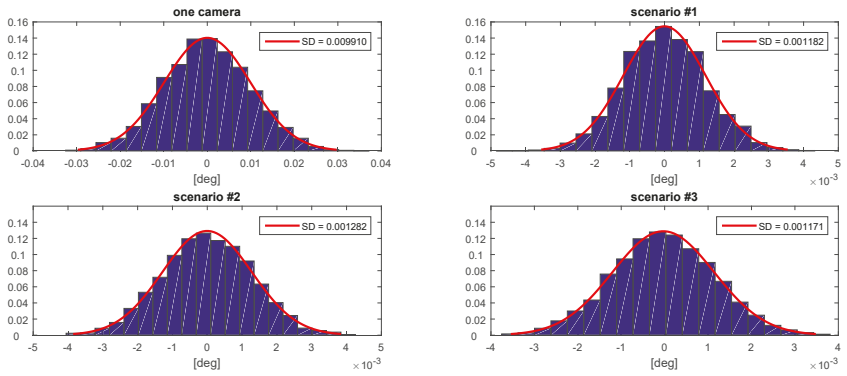

Figure 11. The probability density function (PDF) and the standard deviation (SD) of the error for estimating the *AY* angle.
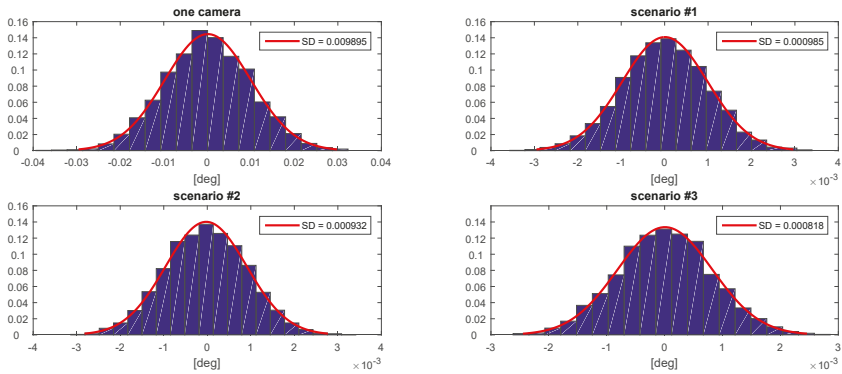
Figure 12. The probability density function (PDF) and the standard deviation (SD) of the error for estimating the *AZ* angle.

Table 1. The simulation results for estimating the pose using only one camera and multiple cameras in three scenarios. The value is the boundary of the error variation interval (2σ).

Pose Element	UM	One Camera	Scenario #1	Scenario #2	Scenario #3
X	(μm)	75.68	11.1	21.04	17.4
Y	(μm)	75.1	11.14	12.7	11.88
Z	(μm)	298.68	30.74	30.01	28.28
AX	(deg)	0.019	0.0016	0.0014	0.0013
AY	(deg)	0.019	0.0022	0.0024	0.0022
AZ	(deg)	0.019	0.0019	0.0018	0.0016

In order to further emphasise the simulation results achieved by the proposed method, Figures 13–18 depict a set of examples from the 3rd scenario which show how the estimation of the state vector elements is evolving. Each of the figures presents: The noisy measurement from the five cameras, the value of the state vector computed after the initialisation procedure, the evolution of the state vector during the Kalman estimation, and the final state estimation which falls close to the true value. Although the noise in the measurements is amplified because the reconstruction of the position of the marker in different cameras, the evolution of the Kalman estimation shows a clear noise damping effect.
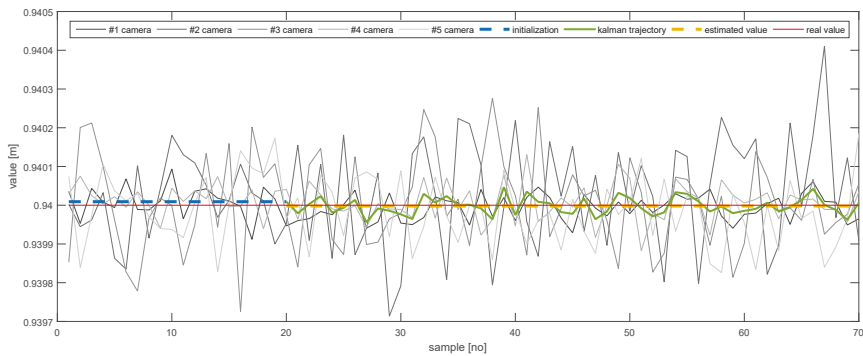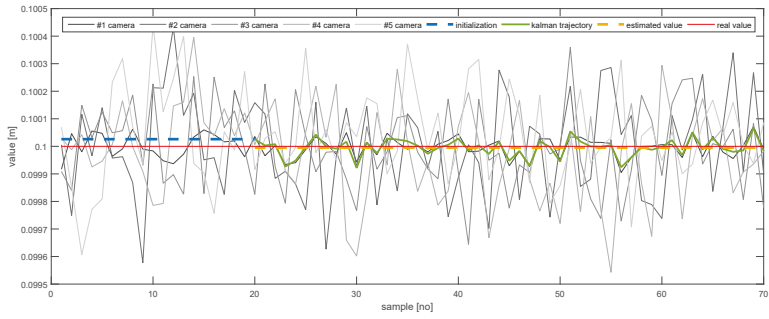

Figure 13. The evolution of the estimation of *X*.

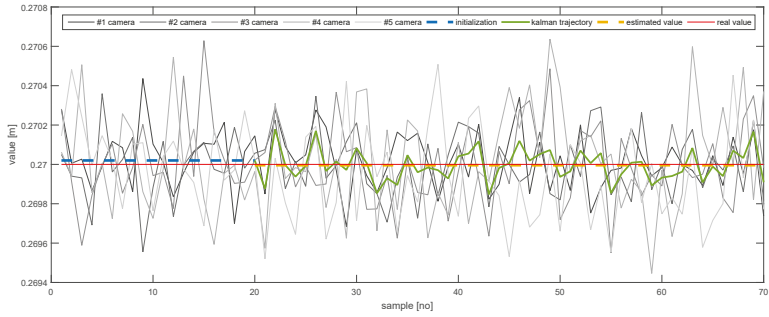**Figure 14.** The evolution of the estimation of Y.

Figure 15. The evolution of the estimation of Z.
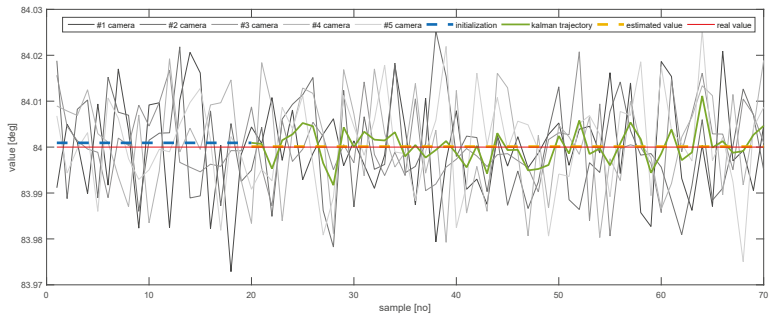

Figure 16. The evolution of the estimation of AX.
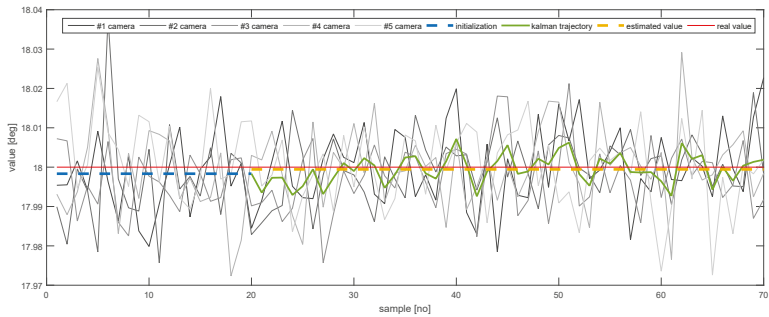

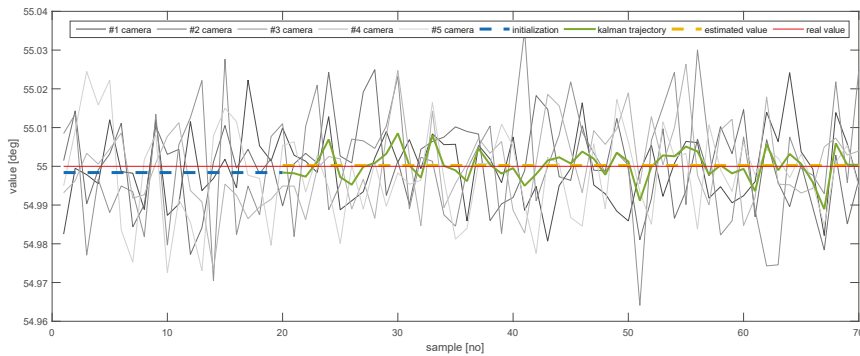**Figure 17.** The evolution of the estimation of AY.

Figure 18. The evolution of the estimation of *AZ*.

5. Conclusions

With respect to the considered case study, the results show that the proposed method and algorithm have managed to successfully meet the objectives. The working area could be increased in accordance with the number of cameras in the setup. This is a decision-making procedure that needs to take into account the cost relative to the working area and the precision required. For our simulation scenarios, the precision increase was close to an order of magnitude, which was around 10–15 μm for *X* and *Y* coordinates, 30 μm for *Z* and $0.002°$ for *AX*, *AY*, and *AZ* orientation angles.

The cost is an important benefit of the system compared to other solutions like laser trackers which tend to be extremely expensive. In addition to that, our proposed method could achieve relative and simultaneous positioning of multiple fiducial markers, which supports the development of advanced applications.

Future work will include a complete analysis of the method in a real environment where all physical and algorithm-related factors are considered, and a precision comparison against other methods presented in the literature.

References

1. Albus, J.S.; Barbera, A.J. RCS: A cognitive architecture for intelligent multi-agent systems. *IFAC Proc. Vol.* **2004**, *37*, 1–11. [CrossRef]
2. Prasanna, S.; Rao, S. An overview of wireless sensor networks applications and security. *Int. J. Soft Comput. Eng.* **2012**, *2*.
3. Apolle, R.; Brückner, S.; Frosch, S.; Rehm, M.; Thiele, J.; Valentini, C.; Lohaus, F.; Babatz, J.; Aust, D.E.; Hampe, J.; et al. Utility of fiducial markers for target positioning in proton radiotherapy of oesophageal carcinoma. *Radiother. Oncol.* **2019**, *133*, 28–34, doi:10.1016/j.radonc.2018.12.016. [CrossRef]

4.	Fiala, M. Artag fiducial marker system applied to vision based spacecraft docking. In Proceedings of the International Conference on Intelligent Robots and Systems (IROS) 2005 Workshop on Robot Vision for Space Applications, Edmonton, AB, Canada, 2 August 2005; pp. 35–40.

5.	Gales, S.; Tanaka, K.A.; Balabanski, D.L.; Negoita, F.; Stutman, D.; Tesileanu, O.; Ur, C.A.; Ursescu, D.; Andrei, I.; Ataman, S.; et al. The extreme light infrastructure-nuclear physics (ELI-NP) facility: New horizons in physics with 10 PW ultra-intense lasers and 20 MeV brilliant gamma beams. *Rep. Prog. Phys. Phys. Soc.* **2018**, *81*, 094301. [CrossRef] [PubMed]

6.	Tanaka, K.A.; Spohr, K.M.; Balabanski, D.L.; Balascuta, S.; Capponi, L.; Cernaianu, M.O.; Cuciuc, M.; Cucoanes, A.; Dancus, I.; Dhal, A.; et al. Current status and highlights of the ELI-NP research program. *Matter Radiat. Extrem.* **2020**, *5*, 024402, doi:10.1063/1.5093535. [CrossRef]

7.	Popescu, D.C.; Cernaianu, M.O.; Dumitrache, I. Automatic rough alignment for key components in laser driven experiments using fiducial markers. *J. Phys. Conf. Ser.* **2018**, *1079*, 012013, doi:10.1088/1742-6596/1079/1/012013. [CrossRef]

8.	Popescu, D.C.; Cernăianu, M.O. *Method and System for Automatic Positioning of Elements for Configurations of Experiments with High-Power Lasers*; Romanian patent RO132327B1; Library Catalog: ESpacenet; OSIM: Bucharest, Romania, 2019.

9.	Fiala, M. ARTag, a fiducial marker system using digital techniques. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 2, pp. 590–596, ISSN: 1063-6919, doi:10.1109/CVPR.2005.74. [CrossRef]

10.	Boby, R.A.; Saha, S.K. Single image based camera calibration and pose estimation of the end-effector of a robot. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 2435–2440. [CrossRef]

11.	Cai, C.; Dean-León, E.; Somani, N.; Knoll, A. 6D image-based visual servoing for robot manipulators with uncalibrated stereo cameras. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 736–742, ISSN: 2153-0866, doi:10.1109/IROS.2014.6942640. [CrossRef]

12.	Chen, D.; Peng, Z.; Ling, X. A low-cost localization system based on artificial landmarks with two degree of freedom platform camera. In Proceedings of the 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014), Bali, Indonesia, 5–10 December 2014; pp. 625–630. [CrossRef]

13.	Babinec, A.; Jurišica, L.; Hubinský, P.; Duchoň, F. Visual Localization of Mobile Robot Using Artificial Markers. *Procedia Eng.* **2014**, *96*, 1–9. [CrossRef]

14.	Bi, S.; Yang, D.; Cai, Y. Automatic Calibration of Odometry and Robot Extrinsic Parameters Using Multi-Composite-Targets for a Differential-Drive Robot with a Camera. *Sensors* **2018**, *18*, 3097, doi:10.3390/s18093097. [CrossRef]

15.	Houben, S.; Droeschel, D.; Behnke, S. Joint 3D laser and visual fiducial marker based SLAM for a micro aerial vehicle. In Proceedings of the 2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Baden-Baden, Germany, 19–21 September 2016; pp. 609–614. [CrossRef]

16.	Muñoz-Salinas, R.; Marín-Jimenez, M.J.; Medina-Carnicer, R. SPM-SLAM: Simultaneous localization and mapping with squared planar markers. *Pattern Recognit.* **2019**, *86*, 156–171. [CrossRef]

17.	Muñoz-Salinas, R.; Medina-Carnicer, R. UcoSLAM: Simultaneous localization and mapping by fusion of keypoints and squared planar markers. *Pattern Recognit.* **2020**, *101*, 107193. [CrossRef]

18.	Xing, B.; Zhu, Q.; Pan, F.; Feng, X. Marker-Based Multi-Sensor Fusion Indoor Localization System for Micro Air Vehicles. *Sensors* **2018**, *18*, 1706, doi:10.3390/s18061706. [CrossRef] [PubMed]

19.	Poulose, A.; Han, D.S. Hybrid Indoor Localization Using IMU Sensors and Smartphone Camera. *Sensors* **2019**, *19*, 5084, doi:10.3390/s19235084. [CrossRef] [PubMed]

20.	Wu, Y.; Niu, X.; Du, J.; Chang, L.; Tang, H.; Zhang, H. Artificial Marker and MEMS IMU-Based Pose Estimation Method to Meet Multirotor UAV Landing Requirements. *Sensors* **2019**, *19*, 5428, doi:10.3390/s19245428. [CrossRef] [PubMed]

21.	Popescu, D.C.; Cernaianu, M.O.; Ghenuche, P.; Dumitrache, I. An assessment on the accuracy of high precision 3D positioning using planar fiducial markers. In Proceedings of the 2017 21st International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, Romania, 19–21 October 2017; pp. 471–476. [CrossRef]

22. Sarmadi, H.; Muñoz-Salinas, R.; Berbís, M.A.; Medina-Carnicer, R. Simultaneous Multi-View Camera Pose Estimation and Object Tracking With Squared Planar Markers. *IEEE Access* **2019**, *7*, 22927–22940, doi:10.1109/ACCESS.2019.2896648. [CrossRef]

23. Garrido-Jurado, S.; Muñoz-Salinas, R.; Madrid-Cuevas, F.J.; Marín-Jiménez, M.J. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognit.* **2014**, *47*, 2280–2292. [CrossRef]

24. Garrido-Jurado, S.; Muñoz-Salinas, R.; Madrid-Cuevas, F.J.; Medina-Carnicer, R. Generation of fiducial marker dictionaries using Mixed Integer Linear Programming. *Pattern Recognit.* **2016**, *51*, 481–491. [CrossRef]

25. Kalman, R.E. A New Approach to Linear Filtering and Prediction Problems. *J. Basic Eng.* **1960**, *82*, 35–45, doi:10.1115/1.3662552. [CrossRef]

26. Hao, Y.; Xu, A.; Sui, X.; Wang, Y. A Modified Extended Kalman Filter for a Two-Antenna GPS/INS Vehicular Navigation System. *Sensors* **2018**, *18*, 3809, doi:10.3390/s18113809. [CrossRef]

27. Hosseinyalamdary, S. Deep Kalman Filter: Simultaneous Multi-Sensor Integration and Modelling; A GNSS/IMU Case Study. *Sensors* **2018**, *18*, 1316, doi:10.3390/s18051316. [CrossRef]

28. Bischoff, O.; Wang, X.; Heidmann, N.; Laur, R.; Paul, S. Implementation of an ultrasonic distance measuring system with kalman filtering in wireless sensor networks for transport logistics. *Procedia Eng.* **2010**, *5*, 196–199. [CrossRef]

29. Li, S.E.; Li, G.; Yu, J.; Liu, C.; Cheng, B.; Wang, J.; Li, K. Kalman filter-based tracking of moving objects using linear ultrasonic sensor array for road vehicles. *Mech. Syst. Signal Process.* **2018**, *98*, 173–189. [CrossRef]

Automatic Distortion Rectification of Wide-Angle Images Using Outlier Refinement for Streamlining Vision Tasks

Vijay Kakani [1], Hakil Kim [1,*], Jongseo Lee [2], Choonwoo Ryu [2] and Mahendar Kumbham [3]

[1] Information and Communication Engineering, Inha University, 100 Inharo, Nam-gu Incheon 22212, Korea; vijaykakanivja@gmail.com

[2] Future Vehicle Engineering, Inha University, 100 Inharo, Nam-gu Incheon 22212, Korea; jongseo@inha.edu (J.L.); cwryu@inha.ac.kr (C.R.)

[3] Valeo Vision Systems, Dunmore Road, Tuam, Co. Galway H54, Ireland; mahendar.kumbham@valeo.com

* Correspondence: hikim@inha.ac.kr; Tel.: +82-32-860-7385

Abstract: The study proposes an outlier refinement methodology for automatic distortion rectification of wide-angle and fish-eye lens camera models in the context of streamlining vision-based tasks. The line-members sets are estimated in a scene through accumulation of line candidates emerging from the same edge source. An iterative optimization with an outlier refinement scheme was applied to the loss value, to simultaneously remove the extremely curved outliers from the line-members set and update the robust line members as well as estimating the best-fit distortion parameters with lowest possible loss. The proposed algorithm was able to rectify the distortions of wide-angle and fish-eye cameras even in extreme conditions such as heavy illumination changes and severe lens distortions. Experiments were conducted using various evaluation metrics both at the pixel-level (image quality, edge stretching effects, pixel-point error) as well as higher-level use-cases (object detection, height estimation) with respect to real and synthetic data from publicly available, privately acquired sources. The performance evaluations of the proposed algorithm have been investigated using an ablation study on various datasets in correspondence to the significance analysis of the refinement scheme and loss function. Several quantitative and qualitative comparisons were carried out on the proposed approach against various self-calibration approaches.

Keywords: automatic distortion rectification; wide-angle lens; fish-eye lens; advanced driver-assistance system (ADAS); video-surveillance; vision tasks

1. Introduction

The usage of wide-angle camera lenses in vision-based applications demands greater precision in terms of image projection geometry such as distortion compensation and maintaining pixel consistency. There appears to be a plethora of challenges involved in the context of employing wide-angle lens models for applications such as advanced driver-assistance system (ADAS) and video surveillance.

1.1. Challenges

The image projections from the wide-angle and fish-eye lens are generally affected by the radial distortions and thereby create a scenario of severe pixel inconsistencies along the edges which depend on the properties of the lens such as horizontal FOV, curvature, etc. [1,2]. This indeed influences the performance of the lens employed in various metric-based tasks such as height estimation and single metrology, and even in geometrical tasks such as camera localization, stereo-vision, etc. This analogy

can be observed in Figure 1, where a self-calibrated camera frame is used for streamlining various vision-based tasks in scenarios of diverse vision applications.
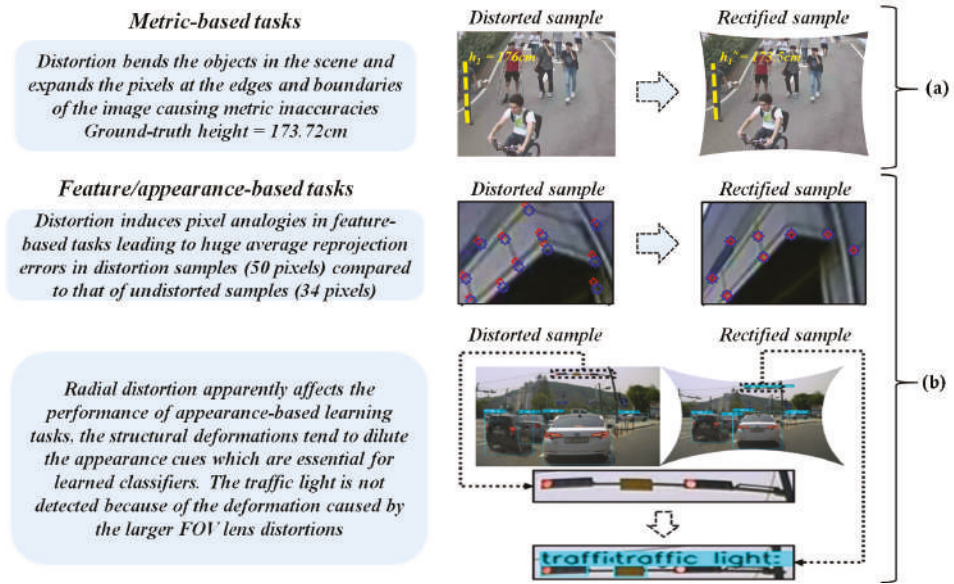

Figure 1. Effects of larger FOV distortions on high-end vision-based tasks. (**a**) Metric-based tasks and (**b**) feature/appearance-based tasks.

The flexibility of handling diverse lens models is another major concern in the formulation of a robust self-calibration technique. The presence of various larger FOV lens models such as fish-eye ($165° <$ FOV $< 190°$), wide-angle ($120° <$ FOV $< 150°$), and super wide-angle ($160° <$ FOV $< 180°$) impose severe challenges in determination of distortion parameters for each class and compensating the specific lens models automatically. The variations in lens models and real-time scenarios are depicted in Figure 2.

The fish-eye and wide-angle lens models are manufactured with a basic notion of the coverage area that the lens can capture. In accordance with that, the lens usually possesses severe distortions due to which the scene aspects on the image plane tend to deviate from the factual representation of a 3D real-world plane. Under such circumstances, the calibration is very important to retrieve the distortion-rectified scene while simultaneously preserving the automatic sense of adaptability without the involvement of any chessboards or objects. Self-calibration totally depends on the scene aspects such as lines, curves, points at infinity, edge candidates, special elements, etc. Several methodologies have been proposed to get past these challenges to formulate robust self-calibration techniques [3–5] but they still get severely caught off with inevitable real-world scenarios such as variation in illuminations, shadow castings, different timings in the day and night, and scenes with limited scene attributes to rely on.
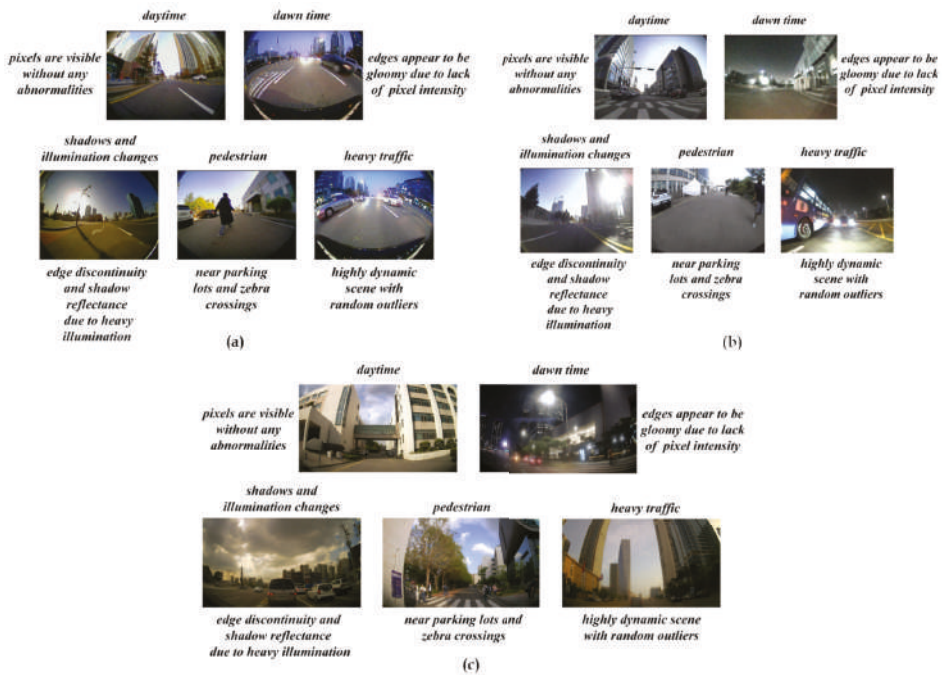
Figure 2. Real-time challenging scenarios: (**a**) fish-eye model (165° < FOV < 190°); (**b**) wide-angle (120° < FOV < 150°); (**c**) super wide-angle (160° < FOV < 180°).

1.2. Purpose of Study

The primary purpose of this work is to develop a flexible automatic distortion rectification methodology that can refine the outliers simultaneously, optimizing the best-fit parameters with minimum error possible. As an underlying investigation, the study has been incorporated by streamlining the distortion-rectified frames for acquiring better performance on tasks such as object detection and fixed monocamera-based height estimation. The two main aspects that this work clearly studies are how the proposed system can be robust towards various real-time scenarios with diverse challenges, and how the streamlining of vision tasks can be done with respect to the distortion-rectified frames. The main contributions are as follows: (1) Proposing an iterative optimization with refinement of the outliers from the pool of robust line-member set; (2) formulating plumbline angular cumulative loss over refined line-member set and investigating the significance through an ablation approach; (3) validating the proposed system with respect to quantitative (accuracy, processing time, practical significance) and qualitative (adaptability, practical significance) aspects on diverse real/synthetic, public and private datasets with respect to ADAS and video-surveillance applications. The scope of this study is targeting the high-end vision-based applications such as intelligent transportation, video surveillance, and advanced driver-assistant systems (ADAS).

The paper is organized as follows. Section 2 extensively discusses the previous works and their characteristics regarding the automatic distortion rectification. Section 3 elaborates on the proposed outlier refinement enabling the automatic distortion rectification process. Section 4 is dedicated to investigating the significance of proposed aspects with respect to various datasets and metrics. Section 5 illustrates the experimental design and evaluation metrics employed in the study. Section 6 reports the outcomes and corresponding discussions based on employed data and evaluations. Finally, Section 7 concludes the paper with a summary.

2. Literature Review

2.1. Automatic Distortion Rectification

In the literature, there are a plethora of studies that were designed to deal with radial distortion rectification via autocalibration of the camera systems [6–8]. Most of these simply followed the approach of employing the calibration object such as a checkerboard or circular patterns [9]. In practice, these camera systems tend to suffer from the variations in the weather conditions with respect to overheating or cold [10,11]. In situations as such, the calibration of the camera must be done to adjust the intrinsic parameters. Automatic distortion rectification, being a more practical approach, can come in handy in such circumstances. Especially, lens models such as fish-eye and wide-angle camera systems demand a better algorithm that can rectify the radial distortions.

Few works like Zhang et al. [12] and Barreto et al. [13] proposed their version of approaches in solving this problem through autocalibration of the visual sensor using scene attributes. However, their approaches demand a specific set of the environment such as precise structured lines (presence of at least three orthogonal straight lines). Brown et al. [14] was the first study to coin the term plumbline, specifying the usage of scene geometry for retrieving the camera's intrinsic parameters. Additionally, this study specified the radial distortions using the polynomial lens distortion model. Later, the one-parameter rational model was proposed by [15,16] which were extensively used in the automatic camera calibration. In literature, the variants of plumbline approaches were used, among which employing of vanishing points to calibrate the camera yielded better results [17]. Yet, their approach was not able to handle wide-angle lens models with heavy distortions.

2.2. Previous Works

The automatic distortion rectification problem can typically be resolved using two main methodologies such as traditional and deep-learning approaches. In the traditional approach, various geometrical aspects are exploited to estimate the distortion parameters of the lens. On the other hand, deep-learning approaches estimate the distortion parameters through learned radial distortion values and image samples. Though there are various algorithms in the above two portfolios, there exist some limitations which make the algorithm venerable towards various real-world conditions.

In the past decade, few remarkable studies were proposed in the context of automatic rectification of wide-angle and fish-eye lens models. A few studies were formulated to explore the arithmetic approach on the line curvatures to estimate the distortions [4]. A few others exploited the scene lines to estimate the parameters with intense iterative optimizations [3,5] within parametric Hough spaces, and a few employed the semiautomatic algebraic approach of tracing line segments over the curved lines for estimating distortions [18]. The semiautomatic study proposed by Alvarez et al. [18] heavily requires user-interaction in the line tracing approach, which is not appropriate for real-time usage. Although Bukhari et al. [4] was able to rectify the distortions with reliable performance for nonsevere distortion cases, it suffers from longer processing times and deformed outputs in the case of heavy distortions. The Hough parametric space approaches from Aleman et al. [3] and Santana et al. [5] were able to rectify the wide-angle and fish-eye lens models with reasonable performance. However, the heavy dependency on hyper-parameters and disability to handle samples acquired using low-quality camera sensors under low-light conditions make it less reliable for ADAS and video surveillance applications. Although, the algorithm proposed by Kakani et al. [19,20] was able to rectify multiple lens models which include a wide-angle and fish-eye lens. Yet, the schematic includes model-specific empirical γ-residual rectification factor for heavy fish-eye distortions with FOV > 165°. The design of this factor requires a certain amount of prior knowledge about the lens models from an optical perspective.

CNN deep-learning approaches such as Bogdan et al. [21] and Lopez et al. [22] cannot rectify the distortion samples with illumination changes, and certain higher distortion ranges cannot be handled with consistency. Additionally, deep GANs such as Liao, Kang et al. [23] are used for generating

corresponding rectified samples for a distorted image. Yet, the trained distortions are confined to certain ranges such as $< -10^{-5}$. Another GAN-based architecture proposed by Park et al. [24] was able to rectify the synthetic distorted samples as well as real sensor data within a specified distortion range. However, in the context of heavy distortion ranges, the model fails to rectify the samples. The major concern regarding these learning approaches is that the training examples must cover almost all the sensor types and ranges of the distortions in order to develop a model that can best rectify all the possible sensor units. In reality, this is not quite possible with the currently available advancements. This raises an issue of using only a certain sensor type and distortion range for a specific application such that one can attain the best performance using learning-based methodologies on that sensor unit. This must be done with each and every sensor unit in correspondence to the use-case that has to be deployed on the rectified frames. Due to this ambiguity, the present proposed work ruled out the learned method in performance evaluations. The details of the summarized state-of-the-art automatic distortion rectification techniques are stated in Table 1.

Table 1. Insights of traditional and learning-based automatic distortion rectification methods.

Algorithm	Method	Dataset	Limitations
Alvarez et al. [18]	2D Euclidean distance	Synthetic dataset with symmetrical patterns	Semiautomatic. Not robust for real-time usage (illumination changes, etc.)
Bukhari et al. [4]	Circular arcs algebra	Synthetic dataset with salient point GT	Severe stretching along the edges. Long processing time for heavy distortion samples
Aleman et al. [3]	Hough parametric space	A private dataset using Nikon D90	Unstable outputs for larger FOV lens camera samples. Heavy hyperparameter-dependent
Santana et al. [5]	Iterative optimization of Hough transforms	Wide angle lens distortion image	Lacking robustness towards blurred images and low-light conditions
Kakani et al. [19]	Straightness cost constraint loss with model-specific empirical γ-residual rectification factor	Real data with varying distortion ranges $120° < FOV < 200°$ Synthetic distorted KITTI samples	Requires prior model-specific knowledge to deal with γ-residual rectification factors
Bogdan et al. [21]	Dual CNN network on radial distortions	Panoramic images of the SUN360 dataset	Fails to rectify samples in illumination changes, motion blur samples
Lopez et al. [22]	CNN Parameterization for radial distortions	SUN360 panorama dataset	Network can only undistort in cropped mode rising an issue of pixel loss $\geq 30\%$
Park et al. [24]	U-Net-based GAN for radial distortions	Real and synthetic distortion dataset	Cannot handle heavy distortions $FOV > 160°$
Liao, Kang et al. [23]	U-Net-based GAN for radial distortions	Synthetic dataset with distortion ranging	Limited distortion ranges (cannot handle distortions $< -10^{-5}$)

This study focuses mostly on the drawbacks encountered in our previous work [19] and proposes a solution to handle heavy distortions without having to use any model-specific residual factors. Especially, this work introduces the outlier refinement scheme in conjunction with the plumbline angular loss function that makes the whole system more robust to outliers and thereby able to handle heavy distortions FOV > 190°. The significance of the novel aspects—such as loss aggregation over line-member sets—of the outlier refinement scheme was extensively tested through ablation study,

and the corresponding results are discussed in Section 4. The major difference between our previous work [19] and the current study is as follows:

- The segregation of robust line candidates was done on the basis of threshold heuristics in the previous work [19], which made some outliers raise some complications while dealing with heavy distortions FOV > 165°, thereby creating a need for model-specific residual factors.
- Unlike [19], the current study employs an iterative outlier refinement scheme which basically considers the aggregation of robust line members into a set and iterating the sets over the plumbline angular loss constraint. The loss over the cumulative line-member sets and corresponding estimated distortion parameters are used to eliminate the outliers, thereby using the new set of robust line candidates to update parameters for distortion rectification.
- The current plumbline angular loss constraint with respect to optimization scheme is analogous to that of [19], but the optimization is altered to consider the loss over the cumulative line-member sets to estimate the distortion parameters with simultaneous outlier elimination.

3. Outlier-Refinement-Enabled Distortion Estimation

3.1. Lens Distortion Parameter Modeling

In this study, the distortion estimation and optimization procedures were followed as per the odd polynomial lens-distortion model with up to two distortion coefficients D_1, D_2 as per the design in our previous work [19], which maps rectified pixel coordinates to the distorted pixel coordinates, as shown in Equation (1) below.

$$
\begin{aligned}
r_{dist} &= r_{undist} + D_1 \cdot r_{undist}^3 + D_2 \cdot r_{undist}^5, \\
r_{dist} &= r_{undist} \left(1 + D_1 \cdot r_{undist}^2 + D_2 \cdot r_{undist}^4\right),
\end{aligned}
\tag{1}
$$

where $r(radius) = \sqrt{(a - a_0)^2 + (b - b_0)^2}$, (a, b) is a point coordinate, (a_0, b_0) is the image center, and $D_1, D_2, \cdots D_N$ are distortion coefficients.

3.2. Plumbline Angular Loss Estimation

The plumbline angular loss is estimated on the robust line-member set, the line members are extracted using parameter-free edge drawing algorithm [25]. Line members emerging from the same edge sources are further filtered based on length threshold heuristics. The line-member set was formed with the elements as line members emerging from same edge. There exists several line-member sets which are to be considered to calculate the cumulative loss on a whole.

The image I_{wxhx3} represents an image and $\bar{n}$ denotes the number of line-member sets within the image I. The collection of all line-member sets as a matrix $L_{\bar{n} \times 4}$, where each line-member set consists of several line members. Each line member is a 4-tuple (x_0, y_0, x_1, y_1), where (x_0, y_0) represent the starting point and (x_0, y_0) represent the ending points of the line member. The grouped line members are collected as

$$
l_{ki} =
\begin{bmatrix}
(x_{0,1}, y_{0,1}, x_{1,1}, y_{1,1}) \\
(x_{0,2}, y_{0,2}, x_{1,2}, y_{1,2}) \\
\vdots \\
\left(x_{0,k}, y_{0,\bar{k}}, x_{1,k}, y_{1,k}\right)
\end{bmatrix},
\tag{2}
$$

$$
L_{\bar{n} \times 4} =
\begin{bmatrix}
l_{k1 \times 4} \\
l_{k2 \times 4} \\
\vdots \\
\vdots \\
l_{k\bar{n} \times 4}
\end{bmatrix},
\tag{3}
$$

where $k \in 1, 2, \ldots \bar{n}$, for instance, $l_{ki} = l_{23}$ indicates that this is the second line-member set and it consists of three line members.

The angular plumbline error α can be estimated through the function $A(l_1, l_2)$ which computes the angular difference between the line members in a set as shown below:

$$A(l_1, l_2) = \begin{cases} \Delta\alpha \\ 360 - \Delta\alpha \quad \text{if } \Delta\alpha > 180 \end{cases}, \tag{4}$$

$$\Delta\alpha = |\alpha_1 - \alpha_2|, \tag{5}$$

$$\alpha_i = \arctan 2(y_1 - y_0, x_1 - x_0). \tag{6}$$

The angular plumbline error α with respect to all N line members is estimated, and an individual line member errors LE for the ith element of the line-member set is calculated by applying cross-entropy of the angular plumbline error:

$$LE_{i\times\bar{n}} = -\frac{1}{N} \begin{pmatrix} |\Delta\alpha_{i,i+1}| \log |\Delta\alpha_{i,i+1}|, |\Delta\alpha_{i,i+2}| \log |\Delta\alpha_{i,i+2}|, \ldots, \\ |\Delta\alpha_{i,i+k}| \log |\Delta\alpha_{i,i+k}|, |\Delta\alpha_{i+1,i+2}| \log |\Delta\alpha_{i+1,i+2}| \\ |\Delta\alpha_{i+1,i+3}| \log |\Delta\alpha_{i+1,i+3}|, \ldots, \\ |\Delta\alpha_{i+1,i+k-1}| \log |\Delta\alpha_{i+1,i+k-1}|, \ldots, \\ |\Delta\alpha_{i+k-2,i+k-1+\bar{n}}| \log |\Delta\alpha_{i+k-2,i+k-1+\bar{n}}| \end{pmatrix}, \tag{7}$$

where $k = |LE_{i\times\bar{n}}|$, i.e, the length of ith row

$$SE = \left(\frac{\sum_1^{|LE_1|} LE_1}{|LE_1|}, \frac{\sum_1^{|LE_2|} LE_2}{|LE_2|}, \ldots, \frac{\sum_1^{|LE_{\bar{n}}|} LE_{\bar{n}}}{|LE_{\bar{n}}|} \right), \tag{8}$$

where SE (line-member set errors) is a row vector of length $\bar{n}$, which represents the average of the ith line-member set.

The mean cumulative loss $SMCE$ which computes the mean errors of a line-member set given by $L_{\bar{n}\times4}$ as follows:

$$SMCE(L_{\bar{n}\times4}) = \frac{\sum_1^{|SE|} SE}{|SE|}, \tag{9}$$

where $|SE|$ is the cardinal set of all line-member set errors.

This overall error loss must be minimized such that we can accomplish two things in one-shot:

- By minimizing error and refining the accumulated line-member set such that the unwanted curves and outliers in the image can be pruned.
- Additionally, through minimizing the error equation, we can estimate the distortion parameter.

3.3. Refinement Optimization Scheme

The Levenberg–Marquardt (LM) optimization, which was employed in the current study, estimates the best fit parameters with simultaneous outlier elimination, where the camera lens parameters are initial with default initial guess:

$$\text{Params} = \begin{bmatrix} f_x \\ f_y \\ c_x \\ c_y \\ D_1 \\ D_2 \end{bmatrix}. \tag{10}$$

Let $f_x, f_y, c_x, c_y, D_1, D_2$ represent the focal length of x (in pixels), the focal length of y (in pixels), the x position of the camera center, the y position of the camera center, and the distortion parameters, respectively.

$$
r_{\bar{n}\times 1} = \begin{bmatrix} r_{1\times 1} \\ r_{2\times 1} \\ \vdots \\ \vdots \\ r_{\bar{n}\times 1} \end{bmatrix} ; \quad x_{\bar{n}\times 1} = \begin{bmatrix} x_{1\times 1} \\ x_{2\times 1} \\ \vdots \\ x_{\bar{n}\times 1} \end{bmatrix} ; \quad y_{\bar{n}\times 1} = \begin{bmatrix} y_{1\times 1} \\ y_{2\times 1} \\ \vdots \\ y_{\bar{n}\times 1} \end{bmatrix}, \tag{11}
$$

where $r'_{\bar{n}\times 1}$ is the column vector of radial distortions for each line member within the line-member set given by $L_{\bar{n}\times 4}$, and $r'_{i\times 1} = \sqrt{x'^2_{i\times 1} + y'^2_{i\times 1}}$ in which $x'_{i\times 1}, y'_{i\times 1}$ are the corresponding x and y coordinates of the ith radial distortion—$i \in \{1, 2, \ldots \bar{n}\}$.

$$
x_i = \frac{x'_i}{\left(1 + D_1 r_i^2 + D_2 r_i^4\right)}, \quad y_i = \frac{y'_i}{\left(1 + D_1 r_i^2 + D_2 r_i^4\right)}, \tag{12}
$$

where undistorted x_i and y_i points are mapped using the distorted parameters D_1 and D_2 with respect to r_i, resulting in distorted points x'_i and y'_i. In addition, $Pts_1 = \begin{bmatrix} L_{\bar{n}\times[1,2]} \\ L_{\bar{n}\times[3,4]} \end{bmatrix}^1_{2\bar{n}\times 2}$ represent the matrix of undistorted start and end points of the line-member set.

$$
L_{\bar{n}\times[1,2]}{}^1 = x_i \times f_x + cx = \begin{bmatrix} (x_{0,1}, y_{0,1}) \\ (x_{0,2}, y_{0,2}) \\ \vdots \\ (x_{0,\bar{n}}, y_{0,\bar{n}}) \end{bmatrix}^1, \tag{13}
$$

$$
L_{\bar{n}\times[3,4]}{}^1 = y_i \times f_y + cy = \begin{bmatrix} (x_{1,1}, y_{1,1}) \\ (x_{1,2}, y_{1,2}) \\ \vdots \\ (x_{1,\bar{n}}, y_{1,\bar{n}}) \end{bmatrix}^1. \tag{14}
$$

Let $L_{\bar{n}\times 4}$ represent a matrix for the set of line members of an image, where l_{ki} is the matrix formed by all the line members. The overall mean cumulative line-member set error ($SMCE$) in the image is estimated using the initial parameters and line members $L_{\bar{n}\times 4}{}^0$:

$$
L_{\bar{n}\times 4}{}^0 = \begin{bmatrix} l_{k1\times 4} \\ l_{k2\times 4} \\ \vdots \\ l_{k\bar{n}\times 4} \end{bmatrix} ; \quad l_{ki} = \begin{bmatrix} (x_{0,1}, y_{0,1}, x_{1,1}, y_{1,1}) \\ (x_{0,2}, y_{0,2}, x_{1,2}, y_{1,2}) \\ \vdots \\ (x_{0,k}, y_{0,k}, x_{1,k}, y_{1,k}) \end{bmatrix}. \tag{15}
$$

The parameters are used to refine the outliers by eliminating unwanted set of line members with respect to minimum error and then an iterative process of elimination takes place to see if the error is getting minimized further by eliminating unwanted outliers ith line member and forming new line-member set $l_{(k-1)i}$ for distortion estimation as shown below:

$$
\text{Err}_{L_{\bar{n}\times 4}}{}^0 = SMCE\left(D_1, D_2, I, L_{\bar{n}\times 4}{}^0\right); \tag{16}
$$

$$l_{(k-1)i} = \begin{bmatrix} (x_{0,1}, y_{0,1}, x_{1,1}, y_{1,1}) \\ (x_{0,2}, y_{0,2}, x_{1,2}, y_{1,2}) \\ \vdots \\ (x_{0,k-1}, y_{0,k-1}, x_{1,k-1}, y_{1,k-1}) \end{bmatrix}. \tag{17}$$

Similarly, $L_{\tilde{n},(j-1)\times 4}$ is the submatrix formed by removing the outliers and retaining $j - 1$ line members from the nth line-member set; thereby, the error $Err_{L\tilde{n},(j-1)\times 4}$ corresponding to the outlier refinement can be estimated simultaneously such that the sequence of submatrices $L_{\tilde{n},(1)\times 4}, L_{\tilde{n},(2)\times 4}, \dots L_{\tilde{n},(j-1)\times 4}$ and their corresponding line-member set errors $Err_{L\tilde{n},(1)\times 4}, Err_{L\tilde{n},(2)\times 4}, \dots Err_{L\tilde{n},(j-1)\times 4}$ are formed:

$$L_{\tilde{n},(j-1)\times 4} = \begin{bmatrix} l_{k1\times 4} \\ l_{k2\times 4} \\ l_{k3\times 4} \\ . \\ . \\ l_{k(j-1)\times 4} \end{bmatrix}, \tag{18}$$

$$Err_{L\tilde{n},(j-1)\times 4} = SMCE\left(D_1, D_2, I, L_{\tilde{n},(j-1)\times 4}\right).$$

The final line-member sets containing refined line members with minimum error are elected for the distortion parameter estimation. The election process of robust line-member set (ELS) is depicted in the Figure 3.

$$ELS = \begin{cases} if \min\left(Err_{L\tilde{n}\times 4}, Err_{L\tilde{n},(j-1)\times 4}\right) = Err_{L\tilde{n},(j-1)\times 4} & L_{\tilde{n},(j-1)\times 4} \\ \\ Otherwise & L_{\tilde{n}\times 4}, \end{cases} \tag{19}$$

where $j \in \{1, 2, \dots i\}; i \in \{1, 2, \dots \tilde{n}\}$.

Figure 3. Outlier refinement scheme based on line-member set aggregations.

4. Ablation Study

Practical Significance Analysis

The ablation study serves as a practical significance analysis investigating the novel aspects introduced in this work. Additionally, this study differentiates the method using straightness loss constraint on individual line candidates [19] from the proposed method of cumulative set aggregation loss and refinement scheme. This investigation will assist in understanding the real significance of using these aspects in the proposed system and their influence on the output performance:

- **Quantitative:** Investigation of proposed cumulative set aggregation loss and refinement scheme with respect to image quality, edge stretching, pixel-point error, and processing time on distorted KITTI dataset and distortion center benchmark.
- **Qualitative:** Investigation of proposed cumulative set aggregation loss and refinement scheme with respect to real-time adaptability and feasible undistortion on severe distortions (FOV: 140° and 165°) with respect to private CV Lab Larger FOV real dataset.

The three Figures 4–6 illustrated below depict the quantitative and qualitative significance analysis of the proposed novel elements over various public and private datasets with respect to diverse metrics. The clear influence of the proposed elements such as cumulative set aggregation loss and refinement scheme can be observed in the qualitative analysis depicted in Figure 6. The following acronyms are used: B—Baseline; B + RO—Baseline + Refined optimization scheme; B + SC—Baseline + Set cumulative aggregation; B + RO + SC—Baseline + Refined optimization scheme + Set cumulative aggregation. Various combinations were used in the ablation study to mainly understand the practical significance of the proposed elements. The clear explanation of the combinations is as follows:

- **B:** Uses the basic straightness loss constraint between line members (without outlier refinement) to estimate distortion parameters.
- **B + RO:** Uses the basic straightness loss constraint between line members (with outlier refinement) to estimate distortion parameters.
- **B + SC:** Uses the basic straightness loss constraint over set cumulative line-member sets (without outlier refinement) to estimate distortion parameters.
- **B + SC + RO:** Uses the basic straightness loss constraint over set cumulative line-member sets (with outlier refinement) to estimate distortion parameters.
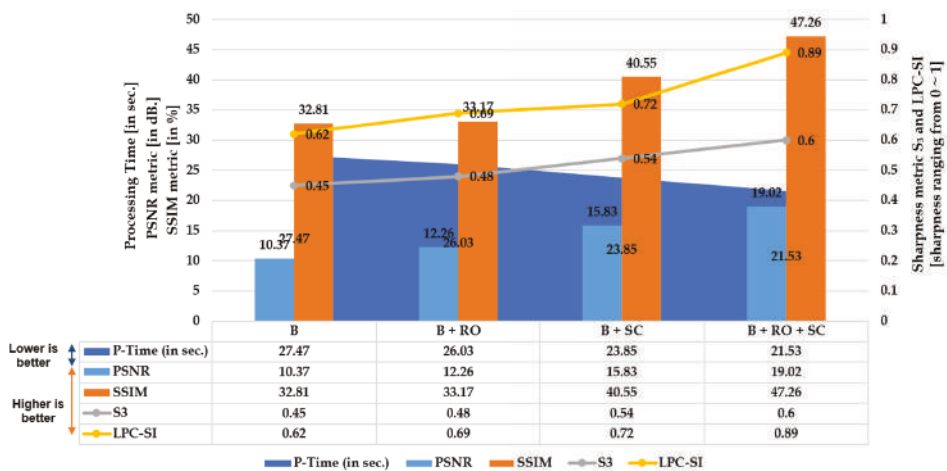

	B	B + RO	B + SC	B + RO + SC
P-Time (in sec.)	27.47	26.03	23.85	21.53
PSNR	10.37	12.26	15.83	19.02
SSIM	32.81	33.17	40.55	47.26
S3	0.45	0.48	0.54	0.6
LPC-SI	0.62	0.69	0.72	0.89

Figure 4. Quantitative: Significance of proposed cumulative set aggregation loss and refinement scheme with respect to image quality, edge stretching, and processing time on distorted KITTI dataset.
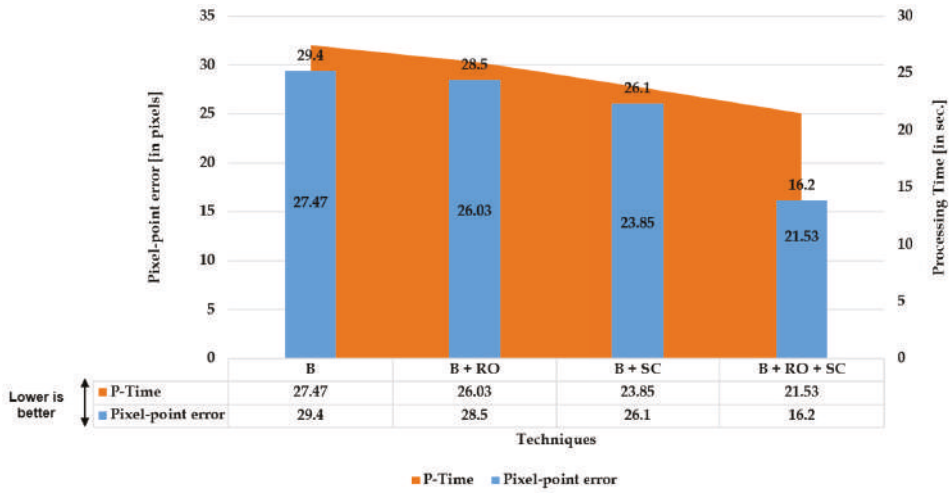
Figure 5. Quantitative: Significance of proposed cumulative set aggregation loss and refinement scheme with respect to pixel-point error and processing time on distortion center benchmark dataset.
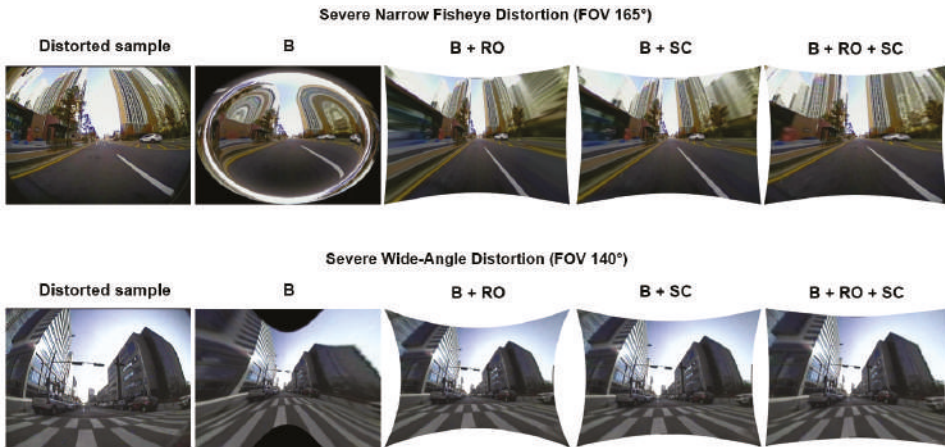

Figure 6. Qualitative: Significance of proposed cumulative set aggregation loss and refinement scheme with respect to severe distortions.

5. Experiments and Evaluations

5.1. Pixel Quality and Consistency Experiments

The experiments were carried out to examine the pixel quality and consistency of the rectified image and low-level image-quality metrics were considered accordingly. The synthetic distorted KITTI dataset using [26,27] was employed to evaluate the rectified image with respect to GT (distortion-free KITTI sample). The accuracy of the distortion-rectified image can be evaluated in two different ways such as image quality metrics, peak signal-to-noise ratio (PSNR); structural similarity index (SSIM); spectral, spatial, and sharpness metric (S_3); local phase coherence sharpness index (LPC-SI); and pixel consistency metrics such as pixel-point error (PPE). The subsections below illustrate the individual significance of each evaluation method present in both strategies.

5.1.1. Image Quality Evaluations

The image quality of the distortion-rectified image must be preserved, and it can be validated using comparative measures with respect to original distortion-free samples in terms of similarly and noise aspects.

- Peak Signal-to-Noise Ratio (PSNR): The pixel consistency of the output (undistorted image) with respect to the original distortion-free image can be assessed using PSNR value. The mathematical measure is directly proportional to the quality of the output, i.e., if the PSNR value is high, the signal information in the output image corresponding to that of the distortion-free image is high and vice versa.
- Structural Similarity Index (SSIM): SSIM is one of the most prominent metrics, which is analogous to human visual perception. The fundamental blocks in the estimation of SSIM are luminance (L), contrast (C), and structural difference (S), which are calculated using the combinations of mean, standard deviation, and covariance [28].
- Spectral spatial sharpness (S_3): The S_3 metric was proposed by [29] and is best suited to examine the sharpness of an image without the reference ground truth. This metric can be retrieved from the pixel properties of the image in terms of spectral and spatial attributes. First, the color image is converted to grayscale and then S_1 and S_2 are extracted from the grayscale image. The metric S_1 represents the spectral sharpness map which is the local magnitude spectrum slope; and the metric S_2 represents the spatial sharpness map which is the local total variation. The geometric mean of these S_1 and S_2 is termed as final sharpness map S_3, which is the overall perceived sharpness of the entire image.
- Local phase coherence sharpness index (LPC-SI): This metric was introduced by [30] to evaluate the sharpness of an image from a different perspective rather than using edge, gradient, and frequency content. This sharpness metric quantifies the sharpness of an image with strong local phase coherence.

5.1.2. Pixel-Point Error Evaluation

The pixel-point error was calculated by estimating the distance between the ground truth pixel point location and the refined image pixel point. For this experiment, the synthetic distortion center benchmark dataset [4] was utilized as shown in the Figure 7 below:
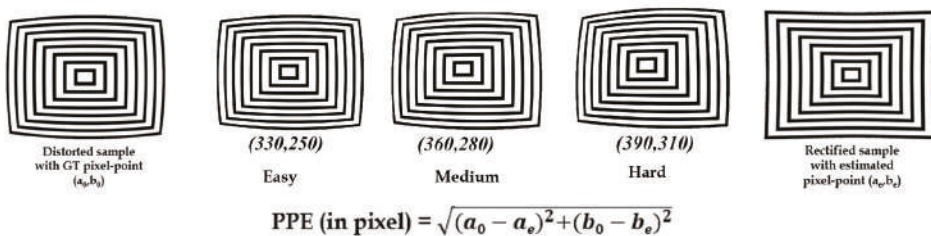

$$PPE \text{ (in pixel)} = \sqrt{(a_0 - a_e)^2 + (b_0 - b_e)^2}$$

Figure 7. Pixel-point error calculation on distortion center synthetic dataset [4].

5.2. High-Level Metrics: ADAS and Video-Surveillance Experiments

This subsection elaborates on the essential usage of wide-angle and fish-eye lens models with proposed automatic distortion rectification techniques to yield better performance in the ADAS, video-surveillance-based vision tasks. In the ADAS context, the state-of-the-art (SOTA) pretrained models were employed to evaluate the proposed algorithm in terms of object detection on real and synthetic data. In the video-surveillance tasks, the height estimation using fixed camera intrinsics from [31] was employed to evaluate the proposed algorithm. The datasets used in this study

were collected at Computer Vision Laboratory, Inha University, among which some are publicly available [31] and few were stated in our previous works [19].

5.2.1. Datasets Used

The datasets utilized in the experiments were of three types:

- **Public-Synthetic dataset:** The publicly available KITTI dataset was synthetically modified using open-sourced distortion induction codes [26]. This dataset can be used to quantitatively measure the performance of distortion rectification algorithms and high-level metrics.
- **Private-Real dataset:** This dataset has been collected using various cameras with diverse lens models such as fish-eye (190°) and wide-angle (120°). This real dataset tests the robustness of the rectification algorithms with respect to the object detection scenarios.
- **Public- and Private-Real dataset:** This dataset has been collected using various cameras with diverse lens models such as super wide-angle (150°) and wide-angle (120°). This real dataset tests the robustness of the rectification algorithms with respect to the height estimation and metric-level information.

5.2.2. Object Detection Using Pretrained Models

Various pretrained models were employed, such as YOLOv3 (pretrained on PASCAL VOC) and SSD (pretrained on MS COCO), as object detectors. These experiments were carried out on diverse lens models such as fish-eye (190°) and wide-angle (120°). The qualitative comparisons were made between various automatic rectification algorithms with respect to detection along the edges. Additionally, for the quantitative measure, the distorted KITTI data samples are rectified using various algorithms alongside the proposed method, and the detection mean average precision (mAP) scores were recorded. The major intent of investigating the proposed algorithm against various algorithms on SOTA pretrained object detectors is to validate the improved performance on rectified frames in streamlining (deploying) object detection tasks. In normal raw samples, the detection accuracy drops due to the distortions along the edges and using SOTA object detectors on those frames would not help, as shown in Figure 8:

Figure 8. Performance of pretrained state-of-the-art (SOTA) models on different larger FOV raw samples: (**a**) Pretrained YOLOv3 on 190° fish-eye sample (car undetected along the edge); (**b**) Pretrained SSD on 120° wide-angle sample (person undetected along the edge).

5.2.3. Height Estimation on Fixed Monocamera Sensor

The height estimation is considered a metric-based task, as the pixel distribution in the image plays a vital role in deciding the metric information. For a fixed camera setup, the experiments were designed on the basis of estimating the intrinsic using walking humans metrology, proposed by Li, Shengzhe et al. [31], employing the Computer Vision Lab's video-surveillance dataset collected at Inha University.

During this study, we modified the previous height estimation method [31] such that the rectified pixel points are retrieved and used to initiate the pixel locations of the walking human (top and bottom) for intrinsic-based height estimation. The modified phenomenon is illustrated in Figure 9, where the objects are not deformed as they are in the raw distortion samples. The camera sensors used in evaluating the algorithm under this portfolio are wide-angle lens cameras. They are employed to capture all the data, as specified in [31], and the subjects used in that study were used in our study as well to maintain the consistency in the ground truth. The height estimation errors in cm is used as a metric for better comparison.
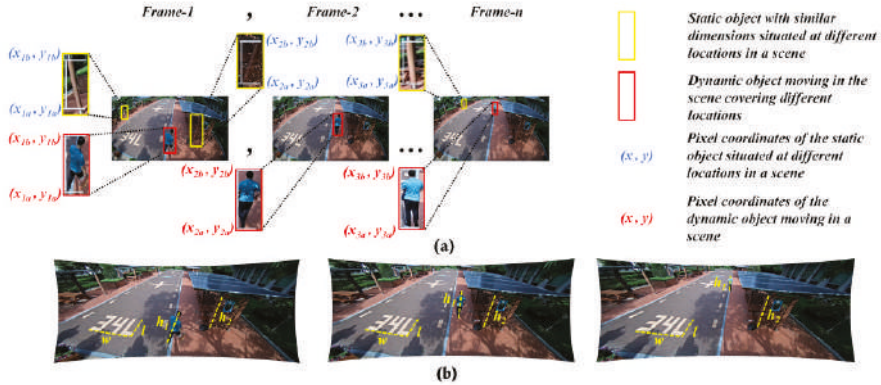
Figure 9. Retrieval of distortion-rectified reference pixel correspondences for better accuracy: (**a**) Top and bottom reference points in distorted case; (**b**) Corresponding top and bottom rectified reference points in rectified case.

6. Results and Discussions

6.1. Pixel Quality and Consistency

The consistency in the pixel information, especially regarding the stretching issue, was clearly investigated, as shown in Figure 10 below. The stretching along the edges caused the inconsistency in the case of traditional OpenCV and Santana et al. [5]. Due to the refinement of outliers, the stretching was significantly reduced in the proposed method.
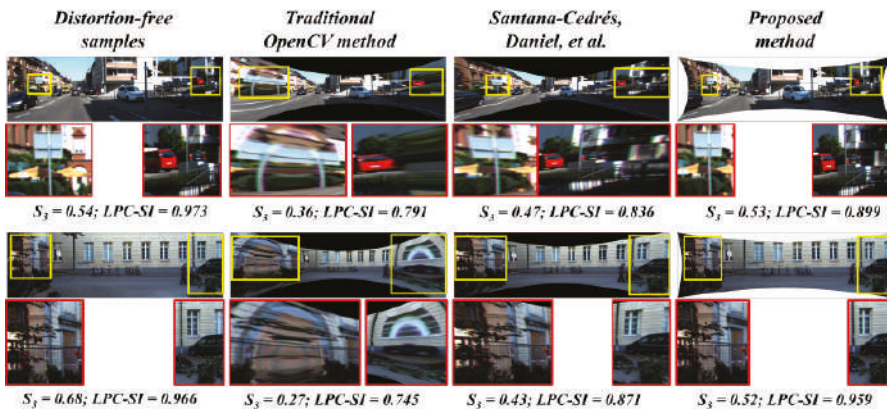
Figure 10. Qualitative analysis: pixel quality and consistency.

6.1.1. Quantitative Analysis: Image Quality

The proposed method was able to rectify the random synthetic distortions, and the average image quality scores in terms of similar metrics and spectral context seem to be high compared to that of the manual and automatic methods. The corresponding results are illustrated in Table 2.

Table 2. Qualitative analysis: image quality metrics on synthetic distorted dataset.

Image Quality Metrics	Distortion Rectification Algorithm			
	Traditional OpenCV	**Bukhari et al. [4]**	**Santana et al. [5]**	**Proposed Method**
PSNR [in dB.]	8.75	13.61	17.5	19
SSIM [in %]	22.9	30.3	43.2	47.2
S3 [↓ 0∼1 ↑]	0.44	0.34	0.41	0.51
LPC-SI [↓ 0∼1 ↑]	0.78	0.82	0.86	0.92

6.1.2. Quantitative Analysis: Pixel-Point Error

The pixel-point error calculations were made using difference of distances from two pixel points in the rectified image distortion center and given GT distortion center on difference samples. The average pixel-point errors were calculated against [5,18] algorithms and the results are stated in Table 3 below. The average pixel-point error in the case of Alvarez et al. [18] and Santana et al. [5] appears to be higher for the examples that have higher variations in the distortion center. The filtering of line-member set for robust line candidate selection influences the proposed method to attain lower average pixel-point error. For the better understanding of quantitative analysis, the average pixel-point errors of all the three methods are indicated in bold.

Table 3. Quantitative analysis: pixel-point error metrics on synthetic distorted dataset.

Synthetic Distortion Pixel-Point (GT)	Pixel-Point Errors on Distortion-Rectified Samples [in px.]		
	Alvarez et al. [18]	**Santana et al. [5]**	**Proposed Method**
Easy (330,250)	21.3	14.1	10.1
Medium (360,280)	17.0	18.4	15.9
Hard (390,310)	49.5	39.9	28.8
Average point error [in px.]	**29.2**	**24.1**	**18.3**

6.2. High-Level Metrics: ADAS Use-Case

The data samples utilized in the experiments were mainly ADAS-centered and are heavily distorted in terms of field-of-view and real-time challenges. The performance analysis was carried out both qualitatively and quantitatively against various automatic distortion rectification methodologies.

6.2.1. Qualitative Performance Analysis

The performance comparisons were carried out between original samples, Aleman et al. [3], Santana et al. [5], and the proposed method with respect to two pretrained models on 3 different cameras. The results were depicted in Figures 11–15 to illustrate the case-by-case scenario robustness of object detection. The objects such as person, car, truck, motorbike, and bus were successfully detected in the case of rectified samples using the proposed method. Although the same pretrained detector was employed on all the SOTA-rectified frames, the proposed method frame yields best performance.

Figure 11. Pretrained YOLOv3 object detection on various rectified 190° fish-eye frames: car detected along the edge in the proposed rectified algorithm.
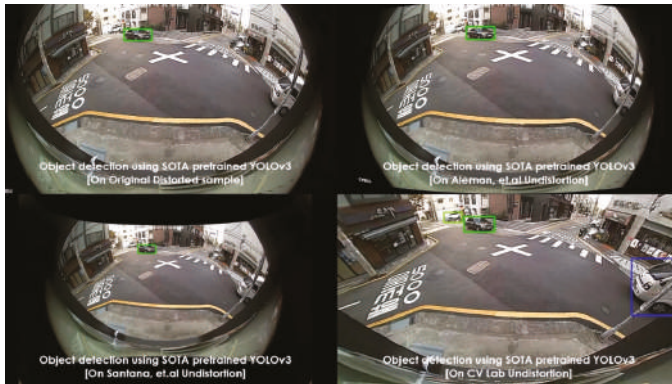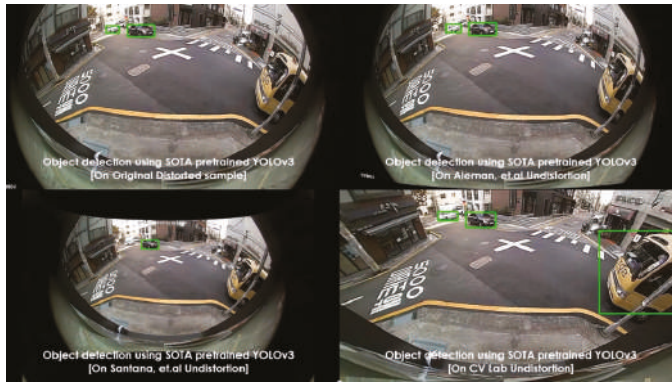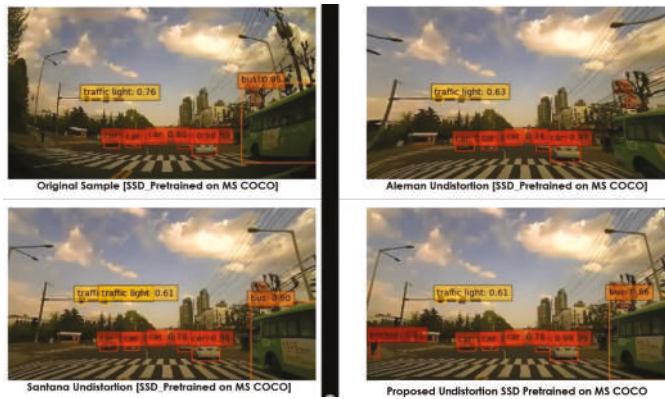

Figure 12. Pretrained YOLOv3 object detection on various rectified 190° fish-eye frames: van detected along the edge in the proposed rectified algorithm.

Figure 13. Pretrained YOLOv3 object detection on various rectified 190° fish-eye frames: motorbike detected along the edge in the proposed rectified algorithm.

Figure 14. Pretrained YOLOv3 object detection on various rectified 190° fish-eye frames: bus detected along the edge in the proposed rectified algorithm.

Figure 15. Pretrained SSD object detection on various rectified 120° wide-angle frames: person detected in the proposed rectified algorithm frame.

6.2.2. Quantitative Performance Analysis

The quantitative analysis has been carried out using the synthetic distorted KITTI dataset on various rectified algorithms—Aleman et al. [3], Santana et al. [5], and the proposed method—alongside distortion-free and randomly distorted samples. The SOTA pretrained YOLOv3 and SSD were employed to detect the objects in the scene, and comparisons were done with respect to various cases. The corresponding quantitative analysis in terms of mAP is depicted in Figure 16. The pretrained SSD achieved 72.4 mAP on rectified samples using the proposed method, which is higher than the distorted an other rectified samples. Similarly, pretrained YOLOv3 achieved 79.8 mAP on proposed method rectified samples, which is greater than the distorted and other rectified samples. The rectified samples used in the streamlining of trained detectors must perform well in order to improve the detection accuracy, and this must be validated using distortion-free samples for proper analysis. The original samples are considered as a ground-truth benchmark such that the algorithm which can produces better rectified samples can therefore be streamlined on to pretrained detectors for better accuracy. This phenomenon proves that the rectified samples using the proposed method are more pixel-consistent and preserved the object characteristics through stretch-free rectification compared to the other rectification algorithms.
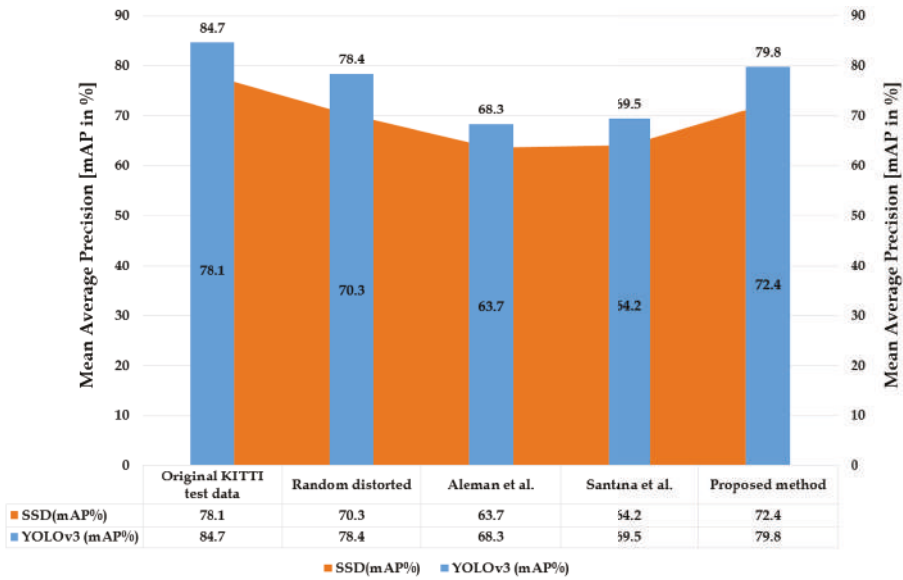
Figure 16. SOTA pretrained YOLOv3 and SSD were employed to detect the objects in the scene on distorted KITTI samples rectified with various algorithms.

6.3. High-Level Metrics: Video-Surveillance Use-Case

The quantitative and qualitative analysis was carried out on various samples retrieved from different camera systems. Primarily, the comparisons were carried out between the use cases where the inevitability of distortion is high. Both the quantitative and qualitative analyses were dealt with using experiments where the distortions were rectified and thereby the intrinsic estimation and height calculations were performed. This process was done for both cases—the distortion rectification process proposed in this study as well as the manual rectification following the approach of Li, Shengzhe et al. [32]. The accuracy in height measurements was estimated with a straightforward method of retrieving errors between the estimated and available ground truth.

The results corresponding to the camera IDs 03, 04, and 08 are depicted in Figures 17–19, respectively, as they spread-over the samples retrieved from both indoor and outdoor. The distortion effect was nullified using both the rectification methods, and the rectified pixel points were used for the further process of estimating the heights of all 11 subjects recorded using a similar camera ID. The red plot line represents the height error values in the case of manual rectification, where the distortions are not completely rectified and that resembles a concave effect due to inappropriate estimation of distortion parameters. The blue plot line represents the error in height estimations in case of the rectification using proposed method.

The results clearly state that the method used in Li, Shengzhe et al. [32] is manual in a manner with the intrinsic-based height estimation, which can be termed as manual distortion-rectification-guided intrinsic-based height estimation (DR-IE) has an effect due to pixel irregularities. This inconsistency in pixel locations and corresponding error in metric information increases with the increase in the distortion levels. The method proposed by Li, Shengzhe et al. [32] is unable to handle such irregularities through manual rectification. In contrast, the proposed method uses the rectified frames to get the pixel location which has relatively low pixel inconsistency resulting in the low height estimation error in cm. This can be clearly shown in the error plots where the height estimation errors are relatively larger in Li, Shengzhe et al. [32] than the proposed method.
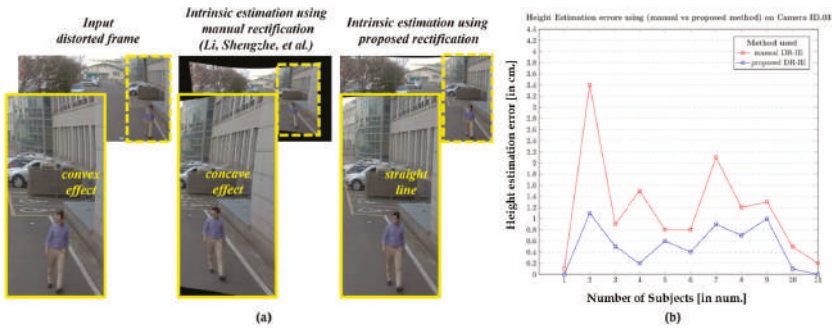
Figure 17. Height Estimation errors using (Li, Shengzhe et al. [31] vs. proposed method) on Outdoor camera ID.03: (**a**) Qualitative pixel-consistency. (**b**) Height estimation error plot corresponding to all the 11 subjects.
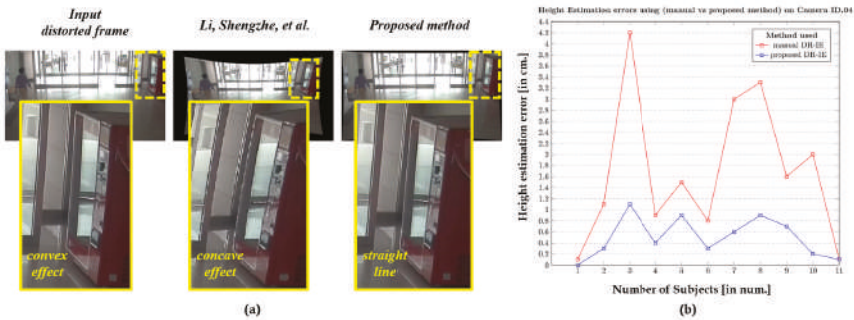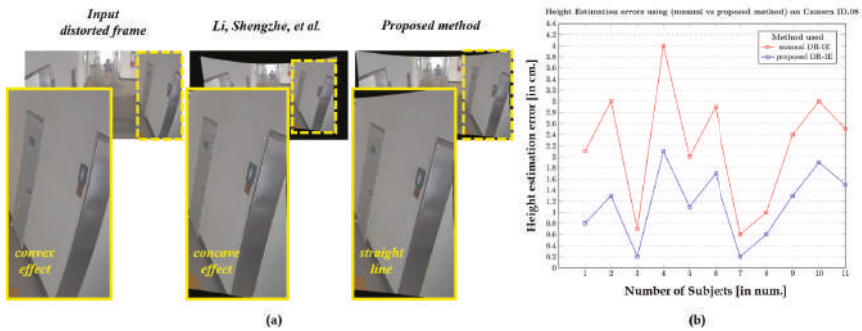

Figure 18. Height Estimation errors using (Li, Shengzhe et al. [31] vs. proposed method) on Indoor camera ID.04: (**a**) Qualitative pixel-consistency. (**b**) Height estimation error plot corresponding to all the 11 subjects.

Figure 19. Height Estimation errors using (Li, Shengzhe et al. [31] vs. proposed method) on Indoor camera ID.08: (**a**) Qualitative pixel-consistency. (**b**) Height estimation error plot corresponding to all the 11 subjects.

The effect of the distortion-rectification-guided height estimation can be observed clearly in the context of the wide-angle camera scenario. The below Figure 20 illustrates the robustness of the proposed system in the presence of darkness and severe illumination changes.
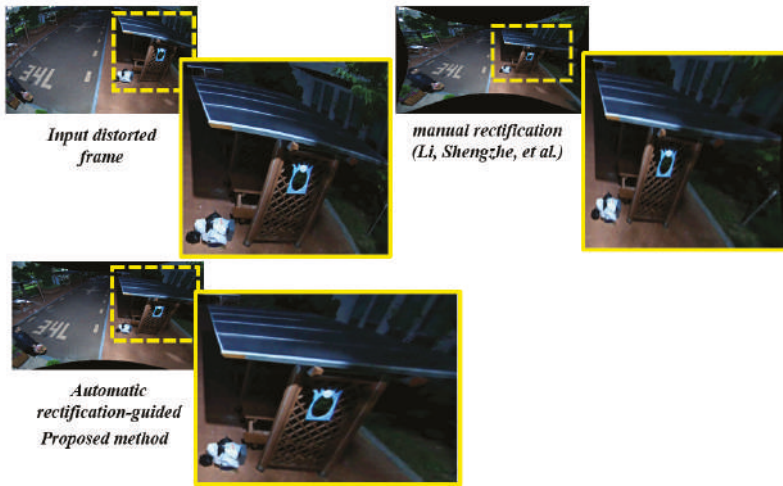
Figure 20. Robustness of proposed distortion-rectification-guided height estimation on wide-angle camera at night time.

The overall height estimation errors with respect to various camera sensors in the context of 11 subjects have been extensively tested with the Li, Shengzhe et al. [31] result as a baseline. The proposed method preserved the pixel consistency in the distortion-rectified image, thereby when those rectified pixels are used for the height estimations, the errors seem to decline. These quantitative comparisons are clearly illustrated in Table 4 below. The camera IDs 1, 2, 6, 7 were used to compare the distortion effects on the metric height estimation because these camera sensors posses a slightly higher amount of distortions compared to the other camera sensors used in the study. The average height estimation errors are indicated in bold in the below table which clearly explains the effectiveness of height estimation via the proposed automatic distortion rectification method.

Table 4. Quantitative comparison: Average height estimation errors with respect to various cameras on 11 subjects.

Subject ID	Height Estimation Errors with Respect to Various Cameras on 11 Subjects [in cm]							
	Cam1		Cam2		Cam6		Cam7	
	Manual	Automatic	Manual	Automatic	Manual	Automatic	Manual	Automatic
S1	0.1	0	0.1	0.2	0.1	0	0.1	0.1
S2	1	0.4	2	0.7	0.5	0.2	0.1	0.2
S3	0.1	0	0.2	0.1	1.2	0.7	0.6	0.5
S4	0.1	0	0.8	0.4	1.3	0.9	2.2	0.4
S5	4.2	1.5	0.2	0.3	3	0.6	3	1.2
S6	0.5	0.3	1.4	0.5	0.4	0.2	2.4	0.8
S7	2.6	0.7	3	0.8	0.3	0.2	0	0.1
S8	1.1	0.9	0.9	0.6	1.2	0.7	1.1	0.7
S9	2	0.6	0.3	0.2	0.4	0.3	0.9	0.8
S10	4.1	1.1	0.9	0.3	0.8	0.1	2.1	0.6
S11	2.8	1.3	1.3	0.2	1.4	0.4	0.8	0.5
Average Errors [in cm]	**1.69**	**0.61**	**1.01**	**0.39**	**0.96**	**0.39**	**1.20**	**0.53**

7. Conclusions

An outlier refinement methodology for automatic distortion rectification of wide-angle and fish-eye lens camera models was proposed. The novel cumulative plumbline angular loss over

line-member set aggregation exhibits better performance in conjunction with the outlier refinement optimization scheme. The design elements were evaluated using various metrics on real datasets (wide-angle: 120° < FOV < 150°; fish-eye: 165° < FOV < 190°) and synthetic distortions on distorted KITTI comprising of several real-time challenges and diverse distortion variations. The practical significance of the proposed novel elements was investigated using an ablation study in accordance with public and private datasets on image quality and pixel consistency metrics. The novel cumulative plumbline angular loss in conjunction with outlier refinement optimization scheme exhibited better performance in rectifying severe distortions compared to other rectification options in the ablation study. A diverse range of experiments were conducted in relevance to the low-level metrics such as image quality, stretching, and pixel-point error on various metrics such as PSNR, SSIM, S3, and LPC-SI. Besides, most of the experiments were carried out in the context of streamlining vision tasks on the rectified frames. The high-level scenarios, such as object detection in ADAS and metric height estimation in video surveillance, were extensively exploited on the distortion-rectified frames to validate the proposed method. Application-oriented metrics such as mean average precision (mAP) and height estimation errors (in cm) were employed to investigate the adaptability of the proposed method in both learning-based appearance tasks and metric-based tasks. Both the quantitative and qualitative metrics were employed in all the streamlined experiments to examine the practical usage of the proposed method. The rectification algorithm proposed using the outlier refinement optimization scheme guided the streamlining vision-based tasks to achieve better accuracy.

Author Contributions: Conceptualization, V.K., H.K.; methodology, V.K.; validation, V.K., H.K.; formal analysis, V.K., M.K. and H.K.; writing—original draft preparation, V.K.; writing—review and editing, V.K., M.K. and H.K.; visualization, V.K., J.L., C.R.; supervision, H.K.; project administration, H.K. All authors have read and agreed to the published version of the manuscript.

References

1. Shah, S.; Aggarwal, J. A simple calibration procedure for fish-eye (high distortion) lens camera. In Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, CA, USA, 8–13 May 1994; pp. 3422–3427.

2. Nayar, S.K. Catadioptric omnidirectional camera. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 482–488.

3. Alemán-Flores, M.; Alvarez, L.; Gomez, L.; Santana-Cedrés, D. Automatic lens distortion correction using one-parameter division models. *Image Process. Line* **2014**, *4*, 327–343. [CrossRef]

4. Bukhari, F.; Dailey, M.N. Automatic radial distortion estimation from a single image. *J. Math. Imaging Vis.* **2013**, *45*, 31–45. [CrossRef]

5. Santana-Cedrés, D.; Gomez, L.; Alemán-Flores, M.; Salgado, A.; Esclarín, J.; Mazorra, L.; Alvarez, L. An iterative optimization algorithm for lens distortion correction using two-parameter models. *Image Process. Line* **2016**, *6*, 326–364. [CrossRef]

6. Schwalbe, E. *Geometric Modelling and Calibration of Fisheye Lens Camera Systems*; Institute of Photogrammetry and Remote Sensing-Dresden University of Technology: Dresden, Germany, 2005.

7. Ho, T.H.; Davis, C.C.; Milner, S.D. Using geometric constraints for fisheye camera calibration. In Proceedings of the IEEE OMNIVIS Workshop, Beijing, China, 17–20 October 2005.

8. Thirthala, S.; Pollefeys, M. The radial trifocal tensor: A tool for calibrating the radial distortion of wide-angle cameras. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 321–328.

9. Tardif, J.P.; Sturm, P.; Roy, S. Self-calibration of a general radially symmetric distortion model. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 186–199.

10. Podbreznik, P.; Potocnik, B. Influence of temperature variations on calibrated cameras. *Int. J. Comput. Inf. Sci. Eng.* **2008**, *2*, 261–267.

11. Handel, H. Analyzing the influence of camera temperature on the image acquisition process. In *Three-Dimensional Image Capture and Applications 2008*; International Society for Optics and Photonics: Bellingham, WA USA, 2008; Volume 6805, p. 68050X.

12. Zhang, Z.; Matsushita, Y.; Ma, Y. Camera calibration with lens distortion from low-rank textures. In Proceedings of the CVPR 2011, Providence, RI, USA, 20–25 June 2011; pp. 2321–2328.

13. Barreto, J.; Roquette, J.; Sturm, P.; Fonseca, F. Automatic camera calibration applied to medical endoscopy. In Proceedings of the BMVC 2009—20th British Machine Vision Conference, London, UK, 7–10 September 2009; pp. 1–10.

14. Duane, C.B. Close-range camera calibration. *Photogramm. Eng.* **1971**, *37*, 855–866.

15. Brauer-Burchardt, C.; Voss, K. A new algorithm to correct fish-eye-and strong wide-angle-lens-distortion from single images. In Proceedings of the 2001 International Conference on Image Processing (Cat. No. 01CH37205), Thessaloniki, Greece, 7–10 October 2001; Volume 1, pp. 225–228.

16. Fitzgibbon, A.W. Simultaneous linear estimation of multiple view geometry and lens distortion. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Kauai, HI, USA, 8–14 December 2001; Volume 1, pp. 125–132.

17. Cipolla, R.; Drummond, T.; Robertson, D.P. Camera Calibration from Vanishing Points in Image of Architectural Scenes. In Proceedings of the BMVC 1999, Nottingham, UK, 13–16 September 1999; Volume 99, pp. 382–391.

18. Alvarez, L.; Gomez, L.; Sendra, J.R. Algebraic lens distortion model estimation. *Image Process. Line* **2010**, *1*, 1–10. [CrossRef]

19. Kakani, V.; Kim, H.; Kumbham, M.; Park, D.; Jin, C.B.; Nguyen, V.H. Feasible Self-Calibration of Larger Field-of-View (FOV) Camera Sensors for the Advanced Driver-Assistance System (ADAS). *Sensors* **2019**, *19*, 3369. [CrossRef] [PubMed]

20. Kakani, V.; Kim, H. Adaptive Self-Calibration of Fisheye and Wide-Angle Cameras. In Proceedings of the TENCON 2019—2019 IEEE Region 10 Conference (TENCON), Kochi, India, 17–20 October 2019; pp. 976–981.

21. Bogdan, O.; Eckstein, V.; Rameau, F.; Bazin, J.C. DeepCalib: A deep learning approach for automatic intrinsic calibration of wide field-of-view cameras. In Proceedings of the 15th ACM SIGGRAPH European Conference on Visual Media Production, London, UK, 13–14 December 2018; ACM: New York, NY, USA, 2018; p. 6.

22. Lopez, M.; Mari, R.; Gargallo, P.; Kuang, Y.; Gonzalez-Jimenez, J.; Haro, G. Deep Single Image Camera Calibration With Radial Distortion. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–21 June 2019; pp. 11817–11825.

23. Liao, K.; Lin, C.; Zhao, Y.; Gabbouj, M. DR-GAN: Automatic Radial Distortion Rectification Using Conditional GAN in Real-Time. *IEEE Trans. Circuits Syst. Video Technol.* **2019**. . [CrossRef]

24. Park, D.H.; Kakani, V.; Kim, H.I. Automatic Radial Un-distortion using Conditional Generative Adversarial Network. *J. Inst. Control Robot. Syst.* **2019**, *25*, 1007–1013. [CrossRef]

25. Akinlar, C.; Topal, C. EDPF: A real-time parameter-free edge segment detector with a false detection control. *Int. J. Pattern Recognit. Artif. Intell.* **2012**, *26*, 1255002. [CrossRef]

26. Vass, G.; Perlaki, T. Applying and removing lens distortion in post production. In Proceedings of the 2nd Hungarian Conference on Computer Graphics and Geometry, Budapest, Hungary, 3 June 2003; pp. 9–16.

27. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.* **2013**, *32*, 1231–1237. [CrossRef]

28. Hore, A.; Ziou, D. Image quality metrics: PSNR vs. SSIM. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 2366–2369.

29. Vu, C.T.; Chandler, D.M. S3: A spectral and spatial sharpness measure. In Proceedings of the 2009 First International Conference on Advances in Multimedia, Colmar, France, 20–25 July 2009; pp. 37–43.

30. Hassen, R.; Wang, Z.; Salama, M.M. Image sharpness assessment based on local phase coherence. *IEEE Trans. Image Process.* **2013**, *22*, 2798–2810. [CrossRef] [PubMed]

31. Liu, J.; Collins, R.T.; Liu, Y. Surveillance camera autocalibration based on pedestrian height distributions. In Proceedings of the British Machine Vision Conference (BMVC 2011), Dundee, UK, 29 August–2 September 2011; Volume 2.

32. Li, S.; Nguyen, V.H.; Ma, M.; Jin, C.B.; Do, T.D.; Kim, H. A simplified nonlinear regression method for human height estimation in video surveillance. *EURASIP J. Image Video Process.* **2015**, *2015*, 32. [CrossRef]

Article

Analyzing Passive BCI Signals to Control Adaptive Automation Devices

Ghada Al-Hudhud [1,*,†], Layla Alqahtani [2,†], Heyam Albaity [1,†], Duaa Alsaeed [1,†] and Isra Al-Turaiki [1,†]

1   Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh 12371, Saudi Arabia
2   King Abdulaziz City for Science and Technology, National Satellite Technology Center, Riyadh 12354, Saudi Arabia
*   Correspondence: galhudhud@ksu.edu.sa
†   These authors contributed equally to this work.

Abstract: Brain computer interfaces are currently considered to greatly enhance assistive technologies and improve the experiences of people with special needs in the workplace. The proposed adaptive control model for smart offices provides a complete prototype that senses an environment's temperature and lighting and responds to users' feelings in terms of their comfort and engagement levels. The model comprises the following components: (a) sensors to sense the environment, including temperature and brightness sensors, and a headset that collects *electroencephalogram* (EEG) signals, which represent workers' comfort levels; (b) an application that analyzes workers' feelings regarding their willingness to adjust to a space based on an analysis of collected data and that determines workers' attention levels and, thus, engagement; and (c) actuators to adjust the temperature and/or lighting. This research implemented independent component analysis to remove eye movement artifacts from the EEG signals and used an engagement index to calculate engagement levels. This research is expected to add value to research on smart city infrastructures and on assistive technologies to increase productivity in smart offices.

Keywords: Passive Brain Signals; adaptive automation and controller; EOG artifact; independent component analysis; engagement index

1. Introduction

Worker engagement and concentration are essential to ensure productivity in the workplace. However, busy workers may find it hard to concentrate since their focus can be easily broken by many factors, and this may affect their engagement at work. The environment surrounding the employee is one of these factors [1]. A room's temperature [2], brightness level, window size [3], and noise level [4] can affect focus at work, especially when employees have a busy schedule. For instance, small changes in room temperature may directly affect engagement, which influences the productivity of employees; this may sometimes occur without anyone noticing the causes for it. Therefore, providing a control system to maintain an environment that helps increase user engagement can improve productivity at work. Furthermore, the control system's level of interaction with users in maintaining an appropriate environment is critical, as most busy workers find it difficult or time consuming to track their environment in order to continually adjust it. Although such a control system sounds promising as an assistive technology to accommodate workers with movement disabilities, it would be impractical in offices with more than one worker. Hence, the proposed research assumed the environment of a small office with one or two workers that is equipped with sufficient infrastructure to assist workers with disabilities.

1.1. Passive Brain Computer Interaction

The original goal of *Brain Computer Interaction* (BCI) is to provide a communication and a control channel for people with severe disabilities, especially those who are completely paralyzed. Most BCIs are used for direct or *explicit* control, which might involve users controlling a cursor or selecting letters on a computer screen using mental activity. The channel transfer rate of these applications remains under 25 bits per minute [5] Such explicit BCIs often require a long training period but remain a solution for patients.

Non-command user interfaces [6] have been proposed to use a BCI as an implicit communication channel between a user and a computer. Implicit or passive BCIs refer to BCIs in which the user does not try to control their brain activity. Passive BCIs have been deployed in recent research on adaptive automation. In the field of adaptive automation, the first brain-based system was developed by Pope et al. [7]. In this system, tracking tasks were allocated between a human and a machine based on an engagement index, which was calculated using users' brain activity. More recently, Kohlmorgen et al. presented the use of implicit BCI in the context of a real driving environment [8]. In this study, the user was engaged in a task mimicking an interaction with the vehicle's electronic warning and information systems. This task was interrupted when a high mental workload was detected. This experiment showed good reaction times on average using BCI based on implicit interaction.

1.2. BCIs in the Office

The environment around a person working in an office has a direct effect on their engagement and productivity [2,4]. In the last few years, BCI researchers have studied BCI technology and its uses, both for disabled and healthy users [9]. Finding an easy and smart way to automatically detect the best environmental conditions and then adjust them accordingly would make the work environment a more enjoyable and proactive place. Recent research has been conducted to develop smart offices using various technologies and techniques, such as smartphones [10], speech commands, gestures [11], and even active brain signals [12]. While some systems do not require direct interaction with the user, most do. Such an interaction involves using passive BCI signals and concentrating on only one environmental factor, such as a window [13]. However, to our knowledge, no research has been conducted that includes adjusting multiple factors in the office environment to enhance worker engagement and concentration [13]. Thus, an intelligent system that passively observes workers' mental status, automatically and passively acquires their brain signals, analyzes these signals alongside environmental measurements to find the perfect state to enhance workers' concentration, and adjusts environmental factors to meet the required state is needed.
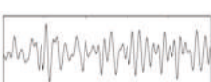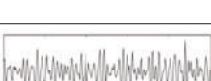
2. Background and Literature Review

Using BCIs as a tool in smart offices [12] and homes [14,15] varies in terms of purpose, methodology, and environment. Hence, in this section, a background on brain wave types and the use of BCI technologies and methodologies in smart offices is discussed.

2.1. Brainwaves

Brain waves are classified based on bandwidth, and each type serves a different function. Low-frequency waves dominate when a person is tired or daydreaming. High-frequency waves appear more often when a person is active. The bandwidths of brain waves are shown in Table 1.

Table 1. Brainwave bandwidths and functions.

Name	Speed	Description	Wave Sample
Infra-low	<0.5 HZ	This is the basic cortical tune that underlies higher brain functions. This type of wave is very slow, which makes it hard to detect; therefore, limited knowledge exists surrounding it.	N/A
Delta	0.5 to 3 HZ	This wave is usually associated with deep stages of sleep and meditation. In addition, it has the highest amplitude and the slowest rate.	
Theta	3 to 8 HZ	This is a low-frequency and low-altitude wave that occurs in sleep, daydreaming, and meditation.	
Alpha	8 to 12 HZ	This wave is generated by the occipital lobe when closing the eyes or relaxing. It is most visible over the parietal and occipital lobes.	
Beta	12 to 38 HZ	This wave dominates most of the human waking state. This wave becomes small and fast when performing hard mental work, such as problem-solving, decision-making, etc. It is most prominent in the frontal cortex during intense and focused mental activity.	
Gamma	38 to 42 HZ	This is the fastest brain wave and occurs when a person is facing a sudden situation.	

2.2. BCI

Electroencephalogram (EEG) electrodes can be used to measure the voltage resulting from brain activities. In the signal-processing stage, several steps are taken to obtain control signal, including preprocessing, feature extraction, and classification. In the last stage, the processed signal is interpreted into the desired action. Figure 1 describes these stages.
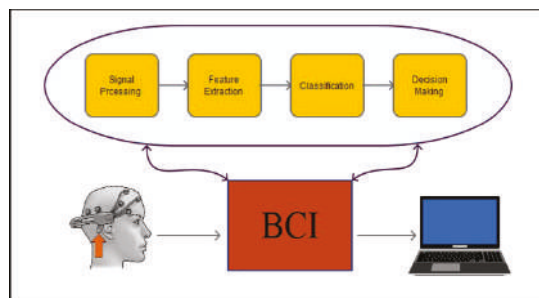

Figure 1. Typical Brain Computer Interaction structure, including data acquisition and signal processing; finally, the interpreted action is shown as a result.

Advancements in the development of BCI systems in recent years have helped to make them more appealing to a wider range of user groups. The cost of such systems has dramatically dropped, and they have become more convenient to use; the electrodes are now wireless, dry, and easy to move during wear. Currently, many commercial BCI devices are available, including NeuroSky [16] and EMOTIV EPOC [17]. EMOTIV EPOC is a BCI device that was developed for research and development

applications. It contains 14 sensors to acquire brain signals at the following locations: AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, and AF4, as shown in Figure 2 [18].
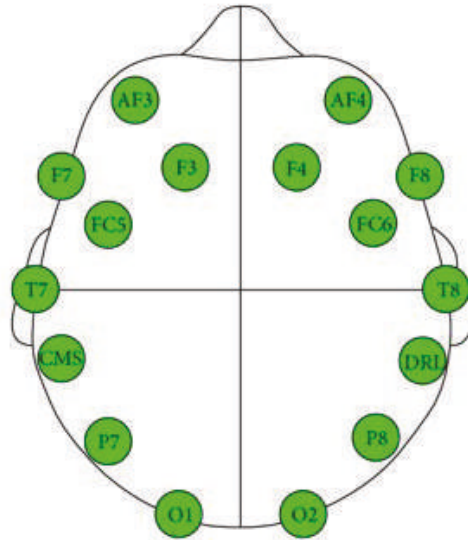

Figure 2. Emotiv EEG neuroheadset sensor position [18].

BCIs vary in terms of their properties and the ways they acquire, analyze, and translate signals. Typical BCIs involve brain signal acquisition, processing, and interpretation. Brain signals can be acquired using various methods, such as EEG.

2.3. Electrooculography/Electromyogram Artifact Removal

Artifacts are the undesirable signals and noise that can interfere with acquired brain signals. Artifacts have a much stronger amplitude than EEG signals and may affect the acquired brain signals, thereby reducing the performance of BCIs. There are two types of artifacts: physiological and non-physiological. Common physiological artifacts include eye movements, which are detected by *electrooculography*(EOG), and muscle movements, which are detected by *electromyography* (EMG). They usually appear as large-amplitude, high-frequency distortions within brain signals. Non-physiological artifacts are usually technical and caused by the environment; they include power-line noises and disturbances caused by recording equipment (e.g., changes in electrode impedances). Non-physiological artifacts are easy to handle and prevent (by applying filtering and the proper recording procedure, respectively). However, physiological artifacts are challenging to eliminate from brain signals and are a significant problem in designing BCIs [19]. Recently, researchers have published many methods to remove eye movement and blinking artifacts from EEG data. Among these methods is rejecting contaminated EEG epochs; however, this method results in a significant loss of collected information. Another method is performing regression on simultaneous EEG recordings, including EMG and EOG recordings in the time or frequency domain. This method aims to derive the parameters that characterize the appearance and spread of EOG artifacts in EEG channels. EOG records may also contain brain signals; hence, removing EOG activity would result in the loss of relevant EEG signals. As there is no clear reference channel for the artifacts, regression methods cannot be used to removed them. A recent method was proposed by Hsu et al. 2016 [20] that involves applying *Independent Component Analysis* (ICA) to eliminate artifacts from EEG sensors. In comparison

with results obtained using regression-based methods and principal component analysis, Hsu et al.'s published results show that ICA can effectively detect, separate, and remove artifacts from EEG records.

2.4. Processing EEG to Measure Engagement Levels

The growth of the BCI technology has attracted many researchers, who often use this technology to measure the engagement of a user. The purpose of measuring user engagement differs from that of enhancing user experience, interfaces, games, and online learning systems. For example, [21] developed an attention-aware system that monitors a student's attention to an online education system and alerts their teacher when attention decreases. The system uses NeuroSky and achieved an 89.52% accuracy rate on average. Ref. [22] developed a prototype system to enhance user experience in museums. This system uses real-time feedback regarding user engagement, using a BCI to provide a tailored museum experience based on a user's taste. The system provides guided tours and suggests exhibits based on a user's engagement level.

A smart office was presented for the first time in [11], in which a user was observed and their intentions anticipated to augment their environment and communicate useful information. At first, the system was controlled using voice commands and gestures [11]. In recent years, much research has adapted BCIs for use in smart offices in order to enhance worker experience and productivity. The system in [13] was designed to improve user engagement by blocking outside distractions; this was done by controlling the opacity of an office's glass wall. The system uses BCI to passively measure a user's level of engagement through NeuroSky's ThinkGear device. The user's level of engagement was used to determine the opacity of an electrochromic smart glass tile, which could change from being fully opaque to being fully transparent. As the user focused, the system increased the opacity of the window as a signal for others to not to disturb the user. However, this system ignores other surrounding factors, such as lighting and room temperature. Ref. [12] used an EMOTIV headset to actively acquire a worker's brain signals and translate them to control the office environment. The system allows users to control the temperature and brightness using their thoughts. However, the system does not control the environment passively; it requires user intervention. Therefore, using BCIs to develop smart offices is a growing research area that still has room for improvement.

3. Methodology

3.1. Proposed Structure

The basic structure of the proposed system is divided into the following phases: brain and environment signal acquisition, signal processing, user engagement calculation, and decision making, as shown in Figure 3. These phases continue working in a cycle to ensure the continuing functionality of the system in order to provide a suitable environment for the worker.
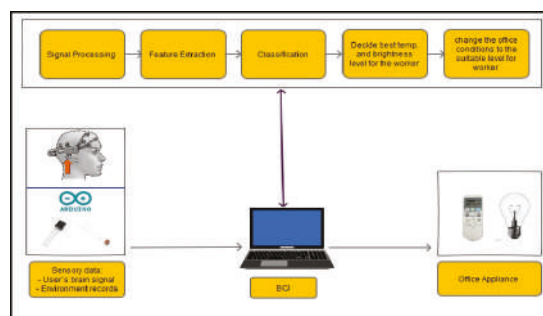

Figure 3. Basic structure of the proposed smart office controller.

3.2. Development Environments and Tools

LabVIEW (also called G) is a dataflow programming language that uses virtual instruments to represent a program. In the field of BCI research, the system in [23,24] used LabVIEW as the development environment to build a system for a smart house. The system's purpose is to monitor the temperature, humidity, lighting, fire and burglar alarms, and gas density in the house in order to ensure safety. Ref. [25] built a smart home system based on a wireless sensor network designed to ensure the safety of elderly people living alone. However, neither of these studies used BCIs. Although many researchers have used LabVIEW to develop BCIs, none were used to create smart offices.

3.3. Signal Acquisition

The proposed system controls the office environment automatically as it detects the user's engagement and intensity levels and the temperature of the environment, which it then maps onto the user's comfort level. Hence, the system adjusts temperature and light intensity as needed. An EMOTIV headset is placed on a user's scalp to collect real-time data (brain signals) in various situations (comforted, stressed, engaged, and distracted), while environmental sensors acquire the temperature and brightness level. The sampling frequency of the headset is fixed at 128 samples per second. Since electrodes placed on the frontal and occipital lobes perform better in obtaining cognitive EEG data than in other locations, the electrodes are placed at the following locations: F3, F4, FC5, FC6, P7, P8, O1, and O2. These channels were chosen because they are closest to those used in other engagement research [22,26–28].

To acquire environmental signals, a temperature sensor was used, and a photoresistor was used to acquire the light intensity. The sensors were set on an electronic circuit connected to an Arduino UNO board, which functions as an interface between the sensors and the computer.

3.4. Signal Processing

After the brain and environmental signals are acquired, they are imported for processing. The processing of the temperature and light signals is done by conditioning them. The temperature and light signals are analog signals; signal conditioning involves converting these signals into digital signals for the next stage. This conversion is done using an existing function in LabVIEW.

The signal processing of the acquired EEG signals is considered difficult due to noise and artifacts. As such, this process is divided into the following stages in order to extract the frequencies required for the task:

- A high-pass filter with a cut-off frequency of 0.6 Hz and a low-pass filter with a cutoff frequency of 50 Hz are used to remove the DC offset.
- Linear finite impulse response filtering is used to remove 50- or 60-Hz line noise.
- Since the reading is collected in real time, a buffer is used to read and then remove the mean of n samples.
- A band-pass filter is used to remove frequencies that are not related to the task. Since only alpha, beta, and theta waves are required, frequencies outside the ranges of 13–22 Hz, 8–12 Hz, and 5–7 Hz are eliminated. An *Infinite Impulse Response* filter of the second order is used as a band-pass filter to filter alpha and beta waves.
- EOG artifacts represent the comfort level of a worker trying to relax by closing the eyes. Such artifacts include eye movements. Blinks are separated using ICA in order to tangibly improve the EEG data interpretation and analysis. ICA is used to identify, separate, and remove these artifacts with a minimal loss of brain activity data. The algorithm follows the steps described below:

    1. Figure 4 shows all the components, including those that contain eye artifacts and spatial mixtures of brain and artifact activities.

2. The rows of the input matrix, X, are EEG signals recorded with different electrodes, and the columns are measurements recorded at different time points, where participants were instructed regarding their eye movement. Hence, the input data are weighted as follows: C = MX.

3. ICA-based artifact separation from EEG data is performed using linear decomposition. The inverse matrix of M represents the projection strength of the components at each of the brain sensors. After linear separation, the multi-channel data result is a sum of independent and spatially fixed components.

4. Components that represent artifacts based on spatial filters are extracted.

5. Artifactual components, including blink artifacts, are removed. Eye movement artifacts are isolated as a linear subtraction of the components representing the artifacts, as shown in Figure 5.
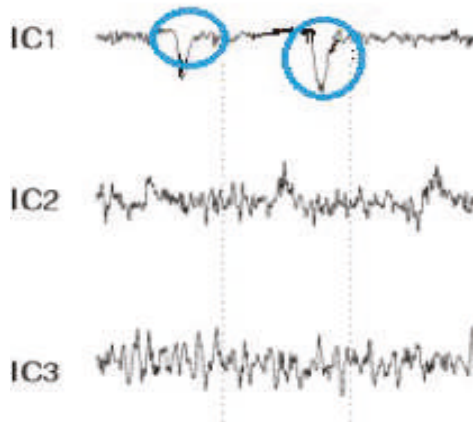
Figure 4. Mixed electroencephalogram/electrooculography (EEG/EOG) data. Note the pulses in the independent components.
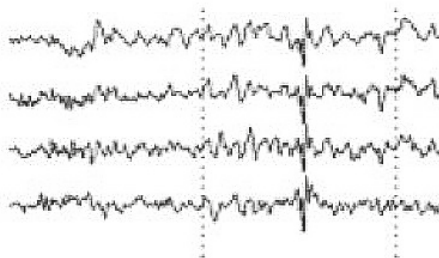
Figure 5. EEG data after eye blink removal.

3.5. Engagement Index Computations and Environmental Control

This system uses the engagement value to determine the best action. If a new high engagement value is recorded, the system saves its related temperature and light intensity values to use when controlling the system. Since temperature and light intensity do not change suddenly in smart offices, the overload on the system is reduced, making it more efficient. The system logs all environmental data and their related engagement levels as references to find the best condition for the user.

- Monitor the surrounding office environment (room/body temperature, brightness level, curtain state) using special sensors.
- Analyze these data to predict the best possible environment to help the user stay focused.
- Associate the brain signal data with the conditions in the second step above.
- If the statement (current engagement score > maximum engagement) is true, then the system will first set a new maximum engagement value and then acquire the temperature and light intensity using the acquisition function. The system only acquires the temperature and light intensity here.
- If the statement (current engagement score > maximum engagement) is false, the system has detected a low engagement level and will find the most suitable environment to raise engagement; then, it adjusts the office environment to this environment. First, the system checks whether current engagement is low; if so, then it sets the temperature and light intensity values to the saved ones. After, the system logs the changes.
- If the engagement value is normal, the system does nothing.
- The system automatically adjusts the office environment to maintain a user's engagement when it is high.

3.6. Feature Selection and Classification of Actions

The purpose of feature selection is to convert digitized brain signals that are recorded at various locations into features [20]. For the feature extraction, a fast Fourier transform (FFT) is used. A Hanning window is first applied; then, the FFT is performed for each epoch. After, the average power spectral density value is extracted for the alpha, beta, and theta frequency bands in order to extract the features of each band. The FFT reduces the computation complexity by obtaining faster results than the discrete Fourier transform. In addition, the FFT is a commonly used algorithm for signal processing; in this project, it is used to capture the frequency components of the EEG signals recorded by the EMOTIV headset. Figure 6 shows a real-time EEG data spectrum. The system acquires new EEG data and processes these data in each iteration. Then, it identifies the engagement level of a user by analyzing the extracted features associated with their mental status using a suitable classification. The alpha, beta, and theta powers collected previously are used to calculate the engagement score. The engagement index is calculated using the following function:

$$\text{Engagement level} = \frac{\beta}{\alpha + \theta} \tag{1}$$
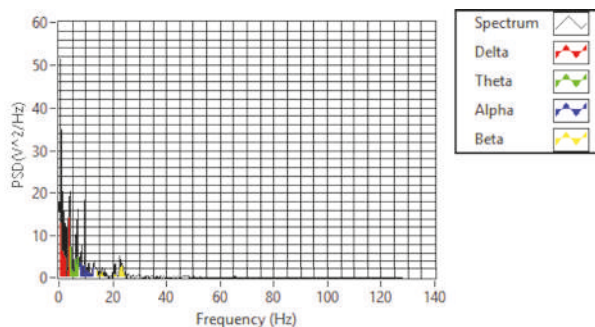

Figure 6. EEG data spectrum.

Then, the engagement values are scaled from zero to one; the higher the value is, the higher engagement level is. In this stage, the system makes a decision and applies changes depending on the results of the previous stage. All the environmental data and engagement data are logged. In a case where the engagement level is below the threshold, the system takes action. First, the system checks

whether the environmental data are in the normal range to ensure that the cause of this loss is the environment. If so, the system adjusts the environmental data to a suitable range (from the log data) and then notifies the user about the changes. The decision algorithm is shown in Algorithm 1.

Algorithm 1 Decision algorithm.

---

if engagement value $\leq$ Threshold **then**

1: Find temperature and light intensity values associated with Max Eng. Value.
2: Adjust the temperature to it;
3: Adjust the light intensity to it;
4: Notify the user about changes; **else** Do Nothing **end if**=0

---

4. Experimental Setup

The raw EEG data were collected from two participants, both male and aged 20–29 years of age. The mental status and health conditions of both participants were normal. For this experiment, there was no pre-knowledge required from the participant about how to use a computer or how to operate an office appliance. The experiment took place in a room where the participants were asked to perform specific tasks that required their attention. The tasks included sending emails, editing a Word document, and reading a document. The participants performed the tasks wearing a headset connected to a computer containing the system software. The headset sampling frequency was set to 256 samples per second. In addition, an Arduino board with two circuits containing a temperature sensor and light sensor were connected to the same computer.

5. Discussion

In this section, the results are presented in more detail and analyzed in three phases. The first phase records the values of a user's engagement level at a neutral status in order to learn and record the engagement level of the user. The second phase analyzes the impact of changing the temperature and fixing the light intensity value on the user's engagement level. The third phase analyzes the effect of changing the light intensity value while fixing the temperature on the user's engagement level.

Figure 7 shows the engagement across one full session. Figure 8 presents the maximum engagement, temperature, and light intensity values over time to show changes to the system over time and to allow for a rapid analysis of the system's performance and the user's engagement.
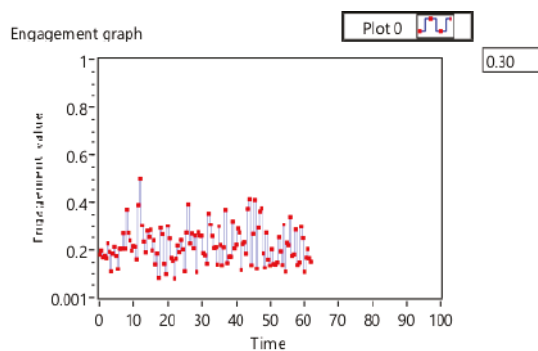

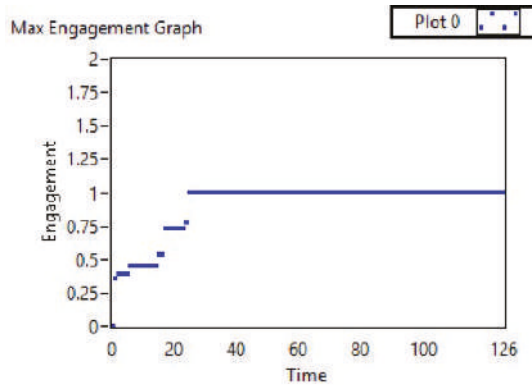**Figure 7.** Engagement data for one session.

Figure 8. Engagement and sensor results for one session.

Table 2 shows the results of calculating the maximum engagement value and its related environment sensory readings. The system accurately saved the high engagement value. The highest score in this session was achieved at a temperature of 24 and a light intensity of 86. By observing the engagement recorded after changing the office environment status, an improvement in engagement was seen for a reasonable time (Table 2). The recording, at first, was below the threshold (0.342805); after setting the temperature to 24.9 and the light intensity to 86.14, the engagement score slightly increased.

Table 2. Calculating the maximum engagement results for one session.

Maximum Engagement Score	Temperature Value	Light Intensity Value
0.225049	25.459999	86.044
0.302148	25.459999	86.142
0.32732	25.459999	85.848
0.369682	25.459999	85.946
0.406814	25.459999	85.946
0.684875	25.459999	85.652
0.936263	24.969999	86.142

5.1. Maximum Engagement Records and Associated Temperature and Light Intensity

Table 3 shows the temperature and light intensity values related to the maximum engagement value from each session. The engagement values changed over 15 min across the three sessions. The first session was run under a low temperature (18–20), as shown in Figure 9. The second session was run under a high temperature (24–26), and the third session was run in the middle of the low and high temperatures used (21–23). The system calculated the engagement values and saved the maximum engagement values. The light intensity remained in the same range. Figure 10 shows the changes to the saved maximum engagement values over 15 min for each session.

Table 3. Related temperature and light intensity values.

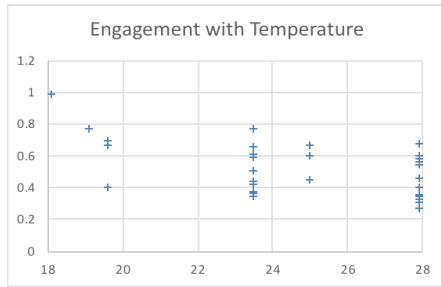| Session	Temperature	Light Intensity
Session 1	19.000	90.944
Session 2	25.400	75.600
Session 3	23.500	77.028

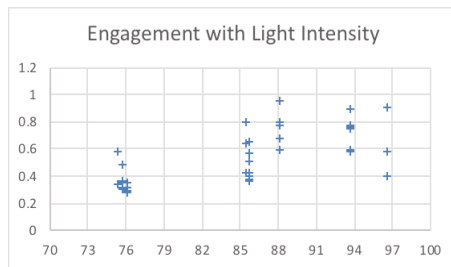**Figure 9.** Engagement values for different temperatures.

Figure 10. Engagement values for different light intensity values.

5.2. Engagement versus Synthesized Changing Temperature Values

In this phase, the temperature values were changed to record the associated engagement values. The results show the influence of changes in temperature on the engagement level of the user; our results conform to those reported in [7,29].

By changing the temperature and light intensity (Figures 9 and 10), at first, all sessions started with low engagement values, at 0.25056, 0.266476, and 0.225049. Since these values are below the threshold (0.4), the system set the temperature and light intensity values to the saved optimal values for each session. After, as shown in Figure 11, the engagement values slightly increased over time until they reached high values (above 0.6) at the end of the sessions.
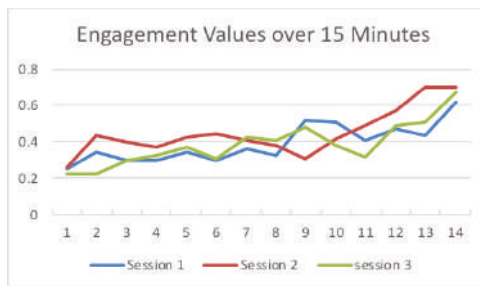

Figure 11. Engagement values over 15 min.

6. Conclusions

EEG is expected to be a future user input technology. This research provides a prototype and a first step to relate EEG data with the office environment in order to enhance and develop smart offices. The environmental sensory data remained the same from sensors connected to Arduino board. Alpha,

beta, and theta powers were extracted from the EEG data and used to calculate user engagement. Based on the engagement value, the system sets the temperature and light density values. The system saved 22.3 C as the optimal temperature and 76 lux as the optimal light intensity.

The experimental results show efficient control in terms of the focus level of users by correctly adjusting the office temperature and light intensity.

In this study, the best possible environment was determined based on the engagement value. There are two factors considered in this study: room temperature and illumination. The value for each was set based on the highest engagement value obtained. In the future, Artificial Intelligence algorithms may be utilized to determine the best environment. The temperature and lumen values in this study are assumed to be fixed. However, during the testing phase we changed the values within a limited range (range of normal situation in offices).

Many factors affect EEG, including: emotional state, fatigue, sleepiness, age, body temperature, and blood oxygen saturation. All these factors are important to consider. Thus, it is suggested to collect more feedback from the subjects and also to vary the duration and frequency of training sessions.

Due to time and device limitations, our system only deals with room temperature and lighting. Future enhancements should count more than two parameters and investigate the use of other mechanisms to track eye blinking as a sign of discomfort. In addition, low engagement should be eliminated using linear discriminant analysis for the classification of the feature vectors extracted from the ICA components. Further enhancements could use a more convenient EEG headset—this may generate effective results since the user would not feel a difference in their daily routine at the office.

References

1. Timm, S.; Gray, W.A.; Curtis, T.; Chung, S.S.E. Designing for Health: How the Physical Environment Plays a Role in Workplace Wellness. *Am. J. Health Promot.* **2018**, *32*, 1468–1473. [CrossRef] [PubMed]
2. Seppanen, O.; Fisk, W.J.; Lei, Q.H. *Effect of Temperature on Task Performance in Officeenvironment*; Technical Report LBNL-60946; Ernest Orlando Lawrence Berkeley National Laboratory: Berkeley, CA, USA, 2006.
3. Ergan, S.; Shi, Z.; Yu, X. Towards quantifying human experience in the built environment: A crowdsourcing based experiment to identify influential architectural design features. *J. Build. Eng.* **2018**, *20*, 51–59. [CrossRef]
4. Bergs, J. The Effect of Healthy Workplaces on the Well-Being and Productivity of Office Workers. In Proceedings of International Plants for People Symposium, Amsterdam, The Netherlands, 2002. Available online: http://seedengr.com/The%20Effect%20of%20Healthy%20Workplaces%20on%20the%20Well-being%20and%20Productivity%20of%20Office%20Workers.pdf (accessed on 1 January 2019).
5. Wolpaw, J.R.; Birbaumer, N.; McFarland, D.J.; Pfurtscheller, G.; Vaughan, T.M. Brain-computer interfaces for communication and control. *Clin. Neurophysiol.* **2002**, *113*, 767–791. [CrossRef]
6. Nielsen, J. Noncommand User Interfaces. *Commun. ACM* **1993**, *36*, 83–99. [CrossRef]
7. Pope, A.T.; Bogart, E.H.; Bartolome, D.S. Biocybernetic system evaluates indices of operator engagement in automated task. *Biol. Psychol.* **1995**, *40*, 187–195. [CrossRef]
8. Dornhege, G.; Mill, J.d.R.; Hinterberger, T.; McFarland, D.J.; Mler, K. Improving Human Performance in a Real Operating Environment through Real-Time Mental Workload Detection. In *Toward Brain-Computer Interfacing*; MIT Press: Cambridge, MA, USA, 2007.

9. Allison, B.Z.; Dunne, S.; Leeb, R.; Mill, J.R.; Nijholt, A. (Eds.) *Towards Practical Brain-Computer Interfaces: Bridging the Gap from Research to Real-World Applications*; Biological and Medical Physics, Biomedical Engineering; Springer: Berlin/Heidelberg, Germany, 2013.

10. Nichols, J.; Myers, B.A. Controlling Home and Office Appliances with Smart Phones. *IEEE Pervasive Comput.* **2006**, *5*, 60–67. [CrossRef]

11. Gal, C.L.; Martin, J.; Lux, A.; Crowley, J.L. SmartOffice: design of an intelligent environment. *IEEE Intell. Syst.* **2001**, *16*, 60–66. [CrossRef]

12. Al-Hudhud, G.; Alrajhi, N.; Alonaizy, N.; Al-Mahmoud, A.; Almazrou, L.; Muribah, D.B. Brain Signal for Smart Offices. In *Proceedings of the Third International Conference on Distributed, Ambient, and Pervasive Interactions-Volume 9189*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 131–140.

13. Ducao, A.; Tseng, T.; von Kapri, A. Transparent: Brain computer interface and social architecture. In Proceedings of the International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 2012), Los Angeles, CA, USA, 5–9 August 2012; ISBN 978-1-4503-1682-8.

14. Edlinger, G.; Holzner, C.; Guger, C.; Groenegress, C.; Slater, M. Brain-computer interfaces for goal orientated control of a virtual smart home environment. In Proceedings of the 2009 4th International IEEE/EMBS Conference on Neural Engineering, Antalya, Turkey, 29 April–2 May 2009, pp. 463–465.

15. Jafri, S.R.A.; Hamid, T.; Mahmood, R.; Alam, M.A.; Rafi, T.; Ul Haque, M.Z.; Munir, M.W. Wireless Brain Computer Interface for Smart Home and Medical System. *Wirel. Person. Commun.* **2019**. *106*, 2163–2177 [CrossRef]

16. EEG—Electroencephalography—BCI | NeuroSky. Available online: http://neurosky.com/biosensors/eeg-sensor/ (accessed on 1 January 2019).

17. EMOTIV EPOC+ - 14 Channel Wireless EEG Headset. Available online: https://www.emotiv.com/epoc/ (accessed on 1 January 2019).

18. Wang, T.; Guan, S.U.; Man, K.L.; Ting, T.O. EEG Eye State Identification Using Incremental Attribute Learning with Time-Series Classification. *Math. Probl. Eng.* **2014**, *2014*, 365101. [CrossRef]

19. Fatourechi, M.; Bashashati, A.; Ward, R.K.; Birch, G.E. EMG and EOG artifacts in brain computer interface systems: A survey. *Clin. Neurophysiol.* **2007**, *118*, 480–494. [CrossRef] [PubMed]

20. Hsu, S.; Mullen, T.R.; Jung, T.; Cauwenberghs, G. Real-Time Adaptive EEG Source Separation Using Online Recursive Independent Component Analysis. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2016**, *24*, 309–319. [CrossRef] [PubMed]

21. Chen, C.M.; Wang, J.Y.; Yu, C.M. Assessing the attention levels of students by using a novel attention aware system based on brainwave signals. *Br. J. Educ. Technol.* **2017**, *48*, 348–369. [CrossRef]

22. Abdelrahman, Y.; Hassib, M.; Marquez, M.G.; Funk, M.; Schmidt, A. Implicit Engagement Detection for Interactive Museums Using Brain-Computer Interfaces. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*; ACM: New York, NY, USA, 2015; MobileHCI '15, pp. 838–845.

23. Degenhart, A.D.; Kelly, J.W.; Ashmore, R.C.; Collinger, J.L.; Tyler-Kabara, E.C.; Weber, D.J.; Wang, W. Craniux: A LabVIEW-based modular software framework for brain-machine interface research. *Comput. Intell. Neurosci.* **2011**, *2011*, 363565. [CrossRef] [PubMed]

24. Hamed, B. Design and implementation of smart house control using LabVIEW. *Int. J. Soft Comput. Eng.* **2012**, *1*, 2231–2307.

25. Ransing, R.S.; Rajput, M. Smart home for elderly care, based on Wireless Sensor Network. In Proceedings of the 2015 International Conference on Nascent Technologies in the Engineering Field (ICNTE), Navi Mumbai, India, 9–10 January 2015; pp. 1–5.

26. Davies, S.R.; Mistry, U.; Millen, L.; Skrypchuk, L.; Braithwaite, J. Evaluation of an EEG/Electro-dermal Hybrid Device to Ascertain a User's Attentional State. *Proc. Hum. Factors Ergon. Soc. Ann. Meet.* **2016**, *60*, 26–30. [CrossRef]

27. 8 Next-Generation User Interface That Are (Almost) Here—Hongkiat. Available online: https://www.hongkiat.com/blog/next-gen-user-interface/ (accessed on 1 January 2019).

28. Cutrell, E.; Tan, D. BCI for passive input in HCI. In Proceedings of the ACM CHI Conference on Human Factors in Computing Systems Workshop on Brain-Computer Interfaces for HCI and Games, San Jose, CA, USA, 27 April–3 May 2007.
29. Huang, J.; Yu, C.; Wang, Y.; Zhao, Y.; Liu, S.; Mo, C.; Liu, J.; Zhang, L.; Shi, Y. FOCUS: Enhancing Children's Engagement in Reading by Using Contextual BCI Training Sessions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*; ACM: New York, NY, USA, 2014; CHI '14, pp. 1905–1908.