

Springer Proceedings in Advanced Robotics 32
Series Editors: Bruno Siciliano · Oussama Khatib

Cristian Secchi
Lorenzo Marconi *Editors*



European Robotics Forum 2024

15th ERF, Volume 1



Series Editors

Bruno Siciliano, *Dipartimento di Ingegneria Elettrica e Tecnologie dell'Informazione,
Università degli Studi di Napoli Federico II, Napoli, Italy*

Oussama Khatib, *Robotics Laboratory Department of Computer Science, Stanford
University, Stanford, CA, USA*

Advisory Editors

Gianluca Antonelli, *Department of Electrical and Information Engineering, University
of Cassino and Southern Lazio, Cassino, Italy*

Dieter Fox, *Department of Computer Science and Engineering, University of
Washington, Seattle, WA, USA*

Kensuke Harada, *Engineering Science, Osaka University Engineering Science,
Toyonaka, Japan*

M. Ani Hsieh, *GRASP Laboratory, University of Pennsylvania, Philadelphia, PA, USA*

Torsten Kröger, *Karlsruhe Institute of Technology, Karlsruhe, Germany*

Dana Kulic, *University of Waterloo, Waterloo, ON, Canada*

Jaeheung Park, *Department of Transdisciplinary Studies, Seoul National University,
Suwon, Korea (Republic of)*

The Springer Proceedings in Advanced Robotics (SPAR) publishes new developments and advances in the fields of robotics research, rapidly and informally but with a high quality.

The intent is to cover all the technical contents, applications, and multidisciplinary aspects of robotics, embedded in the fields of Mechanical Engineering, Computer Science, Electrical Engineering, Mechatronics, Control, and Life Sciences, as well as the methodologies behind them.

The publications within the “Springer Proceedings in Advanced Robotics” are primarily proceedings and post-proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. Also considered for publication are edited monographs, contributed volumes and lecture notes of exceptionally high quality and interest.

An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

Indexed by SCOPUS, SCIMAGO, WTI Frankfurt eG, zbMATH.

All books published in the series are submitted for consideration in Web of Science.



Cristian Secchi · Lorenzo Marconi
Editors

European Robotics Forum 2024

15th ERF, Volume 1

Editors

Cristian Secchi 

University of Modena and Reggio Emilia
Reggio Emilia, Italy

Lorenzo Marconi 

University of Bologna
Bologna, Italy

ISSN 2511-1256

ISSN 2511-1264 (electronic)

Springer Proceedings in Advanced Robotics

ISBN 978-3-031-76423-3

ISBN 978-3-031-76424-0 (eBook)

<https://doi.org/10.1007/978-3-031-76424-0>

© The Editor(s) (if applicable) and The Author(s), under exclusive license
to Springer Nature Switzerland AG 2024

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

If disposing of this product, please recycle the paper.

Foreword

These volumes collects a series of scientific contributions presented at the European Robotics Forum (ERF) held in Rimini from 13 to 15 March 2024. The forum, which has been held on an annual basis since 2010 around Europe, is the reference event of the EuRobotics association. The latter is an association created with the aim of providing a common home for the main European stakeholders, both industrial, academic and institutional, with interests in robotics. One of EuRobotics' missions is precisely to facilitate interaction between the industrial and research worlds, promoting collaboration with the aim of providing answers to societal problems and identifying strategic areas useful for defining future research roadmaps.

In this context, the ERF is a strategic annual event whose programme aims to be of interest to both industrial and academic players and whose structure is specifically designed to facilitate interaction. ERF is not a trade fair, although there is regularly an exhibition area where industries and research centres can showcase the latest technical and technological advances, and it is not a conference, although there is always a rich programme of workshops and scientific sessions where current project activities and latest research results are presented. The event places itself ‘in the middle’ by providing a regular event that can be of interest to both industry and pure research with the aim, among other things, of promoting the field of robotics and related disciplines to an ever-wider audience.

The edition of ERF 2024 in Rimini was characterised by a particularly rich scientific programme, specifically designed to complement the spirit of the forum mentioned above. For the first time in the history of ERF, sessions with the presentation of scientific papers were promoted and organised, in the style of those characterising ‘regular’ conferences, called ‘Industrial Scientific Sessions’ and ‘Insight Scientific Sessions’. The former gathered contributions, mainly from research facilities and universities, on macro-areas explicitly stimulated by industry. In the months leading up to the forum, in fact, a direct call was launched to the many industrial players who are members of EuRobotics and who were asked to specify particularly important areas of development according to their roadmap. The outcome of this survey was then used to calibrate a ‘call for papers’ on specific topics, thus forming “Industrial-driven” Scientific Sessions. The Insight Scientific Sessions, on the other hand, were an addition to the scientific workshops that have characterised the ERF programme since the first edition. ERF workshops have always been an opportunity to present ongoing projects (European and otherwise) and stimulate debate on topics of particular relevance in the field of robotics. With the ‘insights’, the aim is to provide a follow-up to these workshops, in which very often the more technical aspect is kept under wraps to the benefit of dissemination and inclusion of the broader community, by presenting more technical scientific insights on the topics that characterise the workshop.

These volumes collects the 144 contributions presented in the Industrial and Insight Scientific Sessions of ERF 2024. Based on the topics covered, the papers have been

divided into four chapters: the first chapter brings together contributions in the field of ‘Mechatronics and Algorithms for Robotics’, touching, among other things, on autonomous navigation, control for robotics and designing aware robots. The second chapter is instead more focused on aspects of “Artificial Intelligence and Robotics” where the emphasis is on the development of efficient and portable robot learning algorithms for real-world settings and also on aspects of computational hardware that enable the implementation of AI algorithms in real applications. The third chapter, on the other hand, contains contributions that fall under the umbrella of ‘Human robot interaction and collaboration’, touching on, among others, issues of trustability, dependability and acceptability of intelligent robots in smart societies, extended reality and safe adaptation for long-term autonomy in human-populated areas. Many applications in robotics are finally presented in the fourth chapter.

Like all large-scale events, ERF2024 saw the decisive contribution of many people and organisations to whom we extend our sincere thanks. We would particularly like to highlight the close and fruitful cooperation with the Eurobotics team represented by President Bernd Liepert, secretary general Reinhard Lafrenz, administrative team Inge Rehorst, Marie Fortems and Javier Lunque and all board members, co-organisers of ERF2024. Special thanks to Antonio Bicchi and Bruno Siciliano, co-general chairs of ERF2024, whose experience and vision contributed significantly to making the event scientifically relevant and internationally visible. Decisive was the work of Giovanni Berselli, Michele Focchi and Nicola Mimmo, to whom great thanks are due for the time and quality put into the organisation of the Robotics Challenges that distinguished ERF2024 and attracted the participation of many young researchers and students from all over Europe. We also acknowledge the patronage of the associations A.N.I.P.L.A. (Associazione Nazionale Italiana Per L’Automazione), I-RIM (Istituto di Robotica e Macchine Intelligenti) and SIRI (Associazione Italiana di Robotica e Automazione), which gave ERF significant visibility within their spheres. Decisive, finally, was the contribution of the PCO AIM group, with the great work done by Marienza Marguglio, Serena Lamarucciola and Federica Russo, and the local team at the Rimini Palacongressi, which saw Lara Sandre and Giulia Viscusi at the forefront, for all logistical and organisational secretarial aspects.

Lorenzo Marconi
Cristian Secchi

Contents

Mechatronics and Algorithms for Robotics

| | |
|--|----|
| A Multi-purpose Robotic Platform for Precision Agriculture | 3 |
| Francesco Visentin, Simone Cremasco, Edoardo Fiorini, Fabio Castellini, and Riccardo Muradore | |
| Design and Radiation Resistance Analysis of a Robotic Bolting Tool Applied to IFMIF-DONES Maintenance | 8 |
| Manuel Ferre, Violeta Redondo, Nancy Barbosa, Paul Espinosa, and Miguel Á. Sánchez-Urán | |
| Estimation of the Orientation of the Connector for Dual-Arm Robotic Cable Manipulation | 13 |
| Andrea Monguzzi, Donato Malascorta, Andrea Maria Zanchettin, and Paolo Rocco | |
| Homography-Based Industrial Image Stitching | 19 |
| Johann J. Mitteramskogler, Fabian Widmoser, Markus Ikeda, and Andreas Pichler | |
| Design Principles for Upper Limb Robotic Rehabilitation: Bridging the Gap from Research Prototype to Clinical Tool | 25 |
| Irene Pippo, Marco Guazzotti, Giulia A. Albanese, Maddalena Mugnocco, Amel Cherif, Maura Casadio, Giovanni Berselli, and Jacopo Zenzeri | |
| Vision-Based Path Planning for Autonomous Robotic Fabric Layering with a Sewing Machine | 31 |
| Marcel Lahoud, Gabriele Marchello, Eleonora Fontana, Amal Meddahi, Khelifa Baizid, Farshad Nozad Heravi, Haider Abidi, Mariapaola D'Imperio, and Ferdinando Cannella | |
| Electrolyzer Control Cabinet Wiring as a Holistic Approach | 37 |
| Elías Milloch, Stefanie Bartelt, Milan Brisse, Robert Egel, and Bernd Kuhlenkötter | |
| Design and Fabrication of a Phantom Head for Robotic Neurosurgery Simulation | 43 |
| Federico Mariano, Carola Abello, Nabeel Kamal, Giovanni Berselli, Elena De Momi, Gianluca Piatelli, and Leonardo S. Mattos | |

| | |
|--|----|
| SoftGrip: Towards a Soft Robotic Platform for Automatized Mushroom Harvesting | 48 |
| <i>Niccoló Pagliarani, Costas Tzafestas, Evangelos Papadopoulos, Petros Maragos, Athanasios Mastrogeorgiou, Antonis Porichis, Helen Grogan, Robin Ehrhardt, and Matteo Cianchetti</i> | |
| A Modular Architecture for IMU-Based Data Gloves | 53 |
| <i>Alessandro Carfi, Mohamad Alameh, Valerio Belcamino, and Fulvio Mastrogiovanni</i> | |
| Mechanical Design of an Agile Quadruped Robot | 58 |
| <i>Edoardo Del Bianco, Navvab Kashiri, Marco Roveri, and Nikos G. Tsagarakis</i> | |
| Design of a Novel Tomato Harvesting Gripper | 64 |
| <i>Boi Okken, Floris Kaper, Rick Muller, and Jan de Jong</i> | |
| Preliminary Performance Assessment of Hardware Architecture for Affordable Haptics | 70 |
| <i>Francesco Rocchi and Alberto Parmiggiani</i> | |
| Variable Stiffness Mechanism Using a Cam Profile | 76 |
| <i>Floris van Ruitenbeek, Boi Okken, and Wesley Roozing</i> | |
| A Reinforcement Learning Method to Minimize the Damage on a Falling Ballbot | 81 |
| <i>Giulia Buzzetti, Davide Zappetti, and Giovanni Iacca</i> | |
| EMERGE - Emergent Awareness from Minimal Collectives | 87 |
| <i>Davide Bacciu, Vincenzo Ambriola, Bahador Bahrami, Andrea Ceni, Andrea Cossu, Valerio De Caro, Cosimo Della Santina, Ophelia Deroy, Claudio Gallicchio, Riccardo Guidotti, Sabine Hauert, Simon Jones, Jurgis Karpus, Suet Lee, Jingyue Liu, Vincenzo Lomonaco, Nadine Meertens, Emma Milner, Anna Monreale, Renan Picoreti Nakahara, Mariano Ramirez, Ebrahim Shahabi, and Maximilian Stözlé</i> | |
| The Landscape of Collective Awareness in Multi-robot Systems | 92 |
| <i>Miguel Fernandez-Cortizas, David Perez-Saura, Ricardo Sanz, Martin Molina, and Pascual Campoy</i> | |
| Awareness in Robots | 98 |
| <i>Burak Sisman and Tim Djedilbaev</i> | |

| | |
|--|-----|
| Metacognitive Control of Linear Dynamic Systems with Self-confidence Adaptation | 103 |
| <i>Ajith Anil Meera and Pablo Lanillos</i> | |
| Awareness in Robotics: An Early Perspective from the Viewpoint of the EIC Pathfinder Challenge “Awareness Inside” | 108 |
| <i>Cosimo Della Santina, Carlos Hernandez Corbato, Burak Sisman, Luis A. Leiva, Ioannis Arapakis, Michalis Vakalellis, Jean Vanderdonckt, Luis Fernando D’Haro, Guido Manzi, Cristina Becchio, Aïda Elamrani, Mohsen Alirezaei, Ginevra Castellano, Dimos V. Dimarogonas, Arabinda Ghosh, Sofie Haesaert, Sadegh Soudjani, Sybert Stroeve, Paul Verschure, Davide Bacci, Ophelia Deroy, Bahador Bahrami, Claudio Gallicchio, Sabine Hauert, Ricardo Sanz, Pablo Lanillos, Giovanni Iacca, Stephan Sigg, Manel Gasulla, Luc Steels, and Carles Sierra</i> | |
| In-Context Interference In Chat-Based Large Language Models | 114 |
| <i>Eric Nuertey Coleman, Julio Hurtado, and Vincenzo Lomonaco</i> | |
| MIDGARD: A Robot Navigation Simulator for Outdoor Unstructured Environments | 120 |
| <i>Giuseppe Vecchio, Riccardo E. Sarpietro, Francesco Cancelliere, Simone Palazzo, Dario C. Guastella, Alessandro Strano, Ignacio Carlucho, Giovanni Muscato, Stefano V. Albrecht, and Concetto Spampinato</i> | |
| Beluga: A Modern Monte Carlo Localization Package for ROS and ROS 2 | 126 |
| <i>Gerardo Puga, Nahuel Espinosa, Michel Hidalgo, Olmer García, and Ivan Paunovic</i> | |
| Distributed Coverage Control for Robotic Systems Employing On-Board Sensors | 131 |
| <i>Federico Pratissoli, Chiara Buono, and Lorenzo Sabattini</i> | |
| Towards Robust Autonomous Robots Using Statistical Model Checking | 137 |
| <i>Michaela Klauck, Ralph Lange, Christian Henkel, Selma Kchir, and Matteo Palmas</i> | |
| Simulating Aerial Event-Based Environment: Application to Car Detection | 143 |
| <i>Ismail Amessegher, Hajar Fradi, Clémence Liard, Jean-Philippe Diguet, Panagiotis Papadakis, and Matthieu Arzel</i> | |

| | |
|---|-----|
| Enhanced Localization of ArUco Markers for Autonomous Robotics: A Comparative Study | 148 |
| <i>Alessandro Minervini, Jean Carlos Quito Casas, Davide Fassio, Davide Buoso, Claudio Giuseppe Messina, Francesco Marino, and Giorgio Guglieri</i> | |
| Adaptive Distributed Coverage Control for Learning Spatial Phenomena in Unknown Environments | 153 |
| <i>Mattia Mantovani, Federico Pratissoli, and Lorenzo Sabattini</i> | |
| Insights on Control Barrier Functions Application for Complex Robotic Systems | 158 |
| <i>Filippo Bertonecelli and Lorenzo Sabattini</i> | |
| Virtual Forward Dynamics Models Applied to Orbital Robotics Scenarios | 164 |
| <i>Mohatahem Reyaz Makhdoomi, Vivek Muralidharan, Juan Sandoval, Miguel Olivares-Mendez, and Carol Martinez</i> | |
| Inverse Kinematics for Robotic Manipulators Using Iterative Preconditioned Optimization Algorithms | 170 |
| <i>Tianchen Liu, Lasitha Weerakoon, and Nikhil Chopra</i> | |
| Accurate Registration of Transparent Objects in 2D LiDAR SLAM | 175 |
| <i>Gurtajbir Singh Herr, Lasitha Weerakoon, and Nikhil Chopra</i> | |
| Generic Nuclear Robotics Architecture (GNRA) A Standard for Nuclear Robotics Electronic Architectures and Interoperability | 180 |
| <i>Ipek Caliskanelli, Periklis Charchalakis, Matthew Goodliffe, Tomoki Sakue, Fumiaki Abe, Elias Stipidis, and Robert Skilton</i> | |
| Robots and Social Sustainability | 185 |
| <i>Bipin Indurkhy and Barbara Sienkiewicz</i> | |
| Planning and Inverse Kinematics of Hyper-Redundant Manipulators with VO-FABRIK | 195 |
| <i>Cristian Morasso, Daniele Meli, Yann Divet, Salvatore Sessa, and Alessandro Farinelli</i> | |
| A Scalable Multi-robot System for Cooperative Exploration | 200 |
| <i>Vincenzo Scognamiglio, Riccardo Caccavale, Alberto Finzi, and Vincenzo Lippiello</i> | |

AI and Robotics

| | |
|--|-----|
| Learning-Based Ground Vehicle Navigation in Outdoor Unstructured Environments | 207 |
| <i>Simone Palazzo, Dario C. Guastella, Giuseppe Vecchio, Riccardo E. Sarpietro, Giuseppe Sutera, Francesco Cancelliere, Giovanni Muscato, and Concetto Spampinato</i> | |
| Enhancing Robotic Demonstration-Based Learning Method with Preliminary Visual Target Localization | 212 |
| <i>Pasquale Foggia, Francesco Rosa, and Mario Vento</i> | |
| Towards Robotic 3D Surface Processing with Global Local Neural Region Descriptor Fields | 218 |
| <i>Anish Pratheeepkumar, Markus Ikeda, Andreas Pichler, and Markus Vincze</i> | |
| Enabling Cognitive Robotics Through Autonomous Motion Planning and Model-Driven Intuitive Programming | 224 |
| <i>Antonio Venezia, Ileana Romata, Giorgio Nicola, Roberto Fausti, Massimiliano Nitti, Adriano Liso, Vito Renò, Nicola Pedrocchi, Manual Beschi, Nicola Longo, Giovanni Di Stefano, and Simone Panicucci</i> | |
| Acroba Gym: A Unity-Based ROS Compliant Simulator for Robotics | 230 |
| <i>Marco Ojer, Xiao Lin, Iñigo Mendizabal-Arrieta, and Antonio Tammaro</i> | |
| Exploitation of Similarities in Point Clouds for Simplified Robot Programming by Demonstration | 236 |
| <i>Philipp Möhl, Markus Ikeda, Michael Hofmann, and Andreas Pichler</i> | |
| Intuitive Cobot Programming for Small-Medium Enterprises | 241 |
| <i>Niccolò Lucci, Elias Montini, Isacco Zappa, Andrea Maria Zanchettin, and Paolo Rocco</i> | |
| Real-Time 3D Reconstruction Adapted for Robotic Applications in Construction Sites | 247 |
| <i>Dimitrios Katsatos, Dimitrios Alexiou, Theodora Kontodina, Ioannis Kostavelis, Dimitrios Giakoumis, and Dimitrios Tzovaras</i> | |
| An Intelligent Robotic Platform for Fruit Selective Harvesting | 252 |
| <i>Clemente Lauretti, Christian Tamantini, Alessandro Zompanti, Sara Cimini, Laura De Gara, Marco Santonico, and Loredana Zollo</i> | |

| | |
|---|-----|
| AI-Enabled Disaster Response Planning for Multi-robot and Autonomous Systems via Task Scheduling and Path-Finding | 258 |
| <i>Matteo Lanzarini and Lorenzo Marconi</i> | |
| Road Pavement Inspection with UAVs Beyond Visual Line of Sight in Long-Range Operations | 263 |
| <i>Javier Curado-Soriano, Francisco J. Pérez-Grau, and Antidio Viguria</i> | |
| SandRo, a Robot for Beach Waste Cleanup | 269 |
| <i>Francesco Cancelliere, Luca Reitano, Dario Calogero Guastella, Giuseppe Sutera, Concetto Spampinato, and Giovanni Muscato</i> | |
| Domain-Specific Fine-Tuning of Large Language Models for Interactive Robot Programming | 274 |
| <i>Benjamin Alt, Urs Keßner, Aleksandar Taranovic, Darko Katic, Andreas Hermann, Rainer Jäkel, and Gerhard Neumann</i> | |
| Exploiting Roadside Sensor Data for Vehicle Manoeuvring Assistance | 280 |
| <i>Federico Princiotto, Edoardo Bonetto, Guido Gavilanes, Matteo Minotti, Daniele Brevi, Claudio Pastrone, Fabrizio Gatti, and Ezio Chiocchetti</i> | |
| Combining Local and Global Perception for Autonomous Navigation on Nano-UAVs | 286 |
| <i>Lorenzo Lamberti, Georg Rutishauser, Francesco Conti, and Luca Benini</i> | |
| GAP9Shield: A 150GOPS AI-Capable Ultra-low Power Module for Vision and Ranging Applications on Nano-drones | 292 |
| <i>Hanna Müller, Victor Kartsch, and Luca Benini</i> | |
| Fusing Multi-sensor Input with State Information on TinyML Brains for Autonomous Nano-drones | 298 |
| <i>Luca Crupi, Elia Cereda, and Daniele Palossi</i> | |
| Optimized Deployment of Deep Neural Networks for Visual Pose Estimation on Nano-drones | 304 |
| <i>Matteo Rizzo, Francesco Daghero, Beatrice Alessandra Motetti, Daniele Jahier Pagliari, Enrico Macii, Massimo Poncino, and Alessio Burrello</i> | |
| Real-Time Fully Convolutional Networks for Peer-to-Peer Nano-drone Visual Localization and LED State Estimation | 310 |
| <i>Nicholas Carlotti, Mirko Nava, and Alessandro Giusti</i> | |

| | |
|---|------------|
| Effect of Optimizer, Initializer, and Architecture of Hypernetworks on Continual Learning from Demonstration | 315 |
| <i>Sayantan Auddy, Sebastian Bergner, and Justus Piater</i> | |
| Exploiting Foundation Models for Efficient Labeling of Deformable Linear Objects | 321 |
| <i>Alessio Caporali, Kevin Galassi, Matteo Pantano, and Gianluca Palli</i> | |
| Approaches for Exploiting Neural Networks for Semi-supervised Myoelectric Control of Robot Hands | 327 |
| <i>Roberto Meattini, Alessandra Bernardini, Alessio Caporali, Gianluca Palli, and Claudio Melchiorri</i> | |
| Closing the Sim-to-Real Gap for Dynamics-Static Friction and Inertial Parameters: A Franka Robot Case Study | 333 |
| <i>Davide Bargellini, Andrea Govoni, Riccardo Zanella, and Gianluca Palli</i> | |
| Learning Passive Policies | 338 |
| <i>Riccardo Zanella, Federico Califano, Cristian Secchi, and Stefano Stramigioli</i> | |
| Sim-to-Real Gap in RL: Use Case with TIAGo and Isaac Sim/Gym | 344 |
| <i>Jaume Albardaner, Alberto San Miguel, Néstor García, and Magí Dalmau</i> | |
| Author Index | 349 |

Mechatronics and Algorithms for Robotics



A Multi-purpose Robotic Platform for Precision Agriculture

Francesco Visentin^(✉), Simone Cremasco, Edoardo Fiorini, Fabio Castellini,
and Riccardo Muradore

Department of Engineering for Innovation Medicine, Section of Engineering and
Physics, University of Verona, Verona, Italy

francesco.visentin@univr.it

Abstract. This paper introduces a modular robotic platform based on tracks for versatile terrain adaptability. The system integrates a vision system and sensors for autonomous navigation, enabling semi-autonomous and fully autonomous operations. Additionally, the platform features a multi-spectral camera mounted on a robotic arm for plant monitoring. The evaluation of the systems demonstrates the robot's capability to navigate toward target positions, avoid obstacles, and detect and scan crops. This modular approach offers a foundation for advancing sustainable farming practices and addressing the challenges the agricultural industry faces.

Keywords: Precision Agriculture · Agricultural Robotics · Mobile Robotics

1 Introduction

As the global demand for agricultural productivity intensifies to meet the needs of an ever-expanding population, the farming field needs to reinvent itself to face a multitude of challenges. These range from the imperative for heightened productivity and efficiency to apprehensions regarding labor shortages, the conservation of resources, and the promotion of environmental sustainability. Fortunately, the integration of robotics holds great promise in addressing these pressing issues. By harnessing the power of automation and precision, robotics in agriculture offers innovative solutions to enhance crop yields, reduce resource wastage, and ensure a sustainable and resilient future for the farming industry.

Different are the agricultural activities where robotics provides semi- or fully-autonomous solutions, each requiring a dedicated approach. Here, we only briefly introduce some of the state-of-the-art platforms, focusing on the commercially available ones; a more detailed overview can be found in [6]. As a generic-use platform, the Cäsar robot from Raussendorf [8] offers two fertilization methods: remote control or autonomous operation. For autonomous tasks, it leverages Real-Time Kinematic (RTK) technology in conjunction with the Global

Navigation Satellite System (GNSS), which allows it to obtain a location accuracy of up to 3 cm. Another commercially available solution is the Greenbot robot [5], which excels at performing fertilization, plowing, and seeding tasks and can transport high loads, helping the farmers in their work. Both solutions require minimal intelligence to accomplish their task; however, applications such as weeding and harvesting need to increase the number of sensors and the complexity of the algorithms used to navigate and identify weeds, parasites, and crops. The French company Naio Technologies has a series of robots [11–13] equipped with advanced technology, such as RTK/GPS sensors, RGB cameras, and Light Detection and Ranging (LiDAR), that can operate autonomously eliminating the need for human supervision. The robots are designed for various agricultural sectors, serving market farmers (vegetables, nurseries, horticulture), large-scale vegetable farms (row and bed crops), and vine growers (row width larger than 150 cm). They rely on mechanical implements to remove weeds and run solely on Lithium batteries, providing an autonomy of 8 h. A different approach for weed removal is the VITIROVER [10] and Franklin Robotics' Tertill robots [9]. These robots are compact and lightweight, incorporate photovoltaic panels into their structures for operation in varying weather conditions, and specialize in weed removal. They offer the convenience of controlling and monitoring robot information through a mobile application, embracing IoT concepts. Another robotic platform is the Agrointelli Robotti [2], which is a multi-purpose tool carrier designed for precision weeding, harrowing, seeding, and spraying.

Even if different in size and their applications, all the solutions presented above share a single-task driven design based on wheels that may not be suitable due to adverse weather conditions, such as heavy rain or muddy terrain. For this reason, in this paper, we propose a new modular, track-based system that can be used in different applications and accommodate any terrain. The robotic system can operate autonomously or be controlled by an operator (either locally or remotely). In the following sections, we will detail one of the possible configurations, discuss its extension, and present the results of a plant monitoring task.

2 The Robotic Platform

Tracks have always shown a higher adaptability to different terrain conditions, either dry or muddy; for this reason, we decided to use those in the locomotion system. Differently from other non-articulated body, robotic, track-based systems [3] we opted for a different shape of the tracks, increasing its height and adding a passively, deformable profile (Fig. 1A). The two tracks (width: 100 cm, height: 80 cm, depth: 30 cm, weight: 100 kg each) are independently controlled by two brushless DC motors (Wuxi Sulong, China), which are reduced to a 40:1 ratio to increase the torque. Thus, it permits overcoming slopes with an angle up to 45°, with a step up to 50 cm of height. To allow for self-navigation, the rover is equipped with two RGBD cameras (Realsense D345i and Realsense 405, Intel) in front and back. In addition to the vision system, the robotic platform is equipped

with an RTK/GPS (GNSS Compass, Advanced Navigation) and two laser range finders (URG-04LX-UG01, Hokuyo) positioned at the side of the robot. With all the components, the robot's mass reaches a total weight of 320 kg. In its current implementation, the system is powered by an external generator, which can be easily replaced with a battery pack for the green transition.

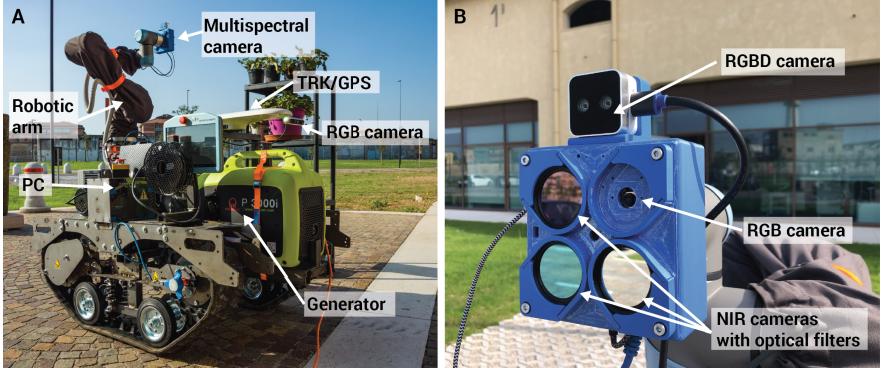


Fig. 1. (A) The proposed modular, track-based, robotic platform. (B) A highlight on the multi-spectral camera connected to a robotic arm.

In the current configuration, an anthropomorphic arm is mounted on the cart (UR5e, Universal Robotics). Still, it can be equipped with other robotic solutions, such as a three-axial stage for weed removal as in [15]. Since the main application of this configuration is monitoring hydroponic crops (strawberries), an internally developed multi-spectral camera is mounted on the robotic arm. This is composed of an RGB camera and three NIR (Near InfraRed) cameras (IMX708, Sony), an RGBD sensor (Realsense D345i, Intel), and a thermal camera (Lepton, Flir), all registered with respect to the same image frame. In addition, the camera embeds a set of ambient sensors (luminosity, pressure, temperature) to monitor the environment. The same robot can also be used for other applications, such as harvesting fruits as presented in [14].

3 Software Architecture

The system is controlled at a high level by a computer running the Robot Operating System (ROS 2) [4], allowing for an extended and modified modular structure. ROS is in charge of i) acquiring and processing the data from the cameras and the sensors, ii) controlling the rover's motion, and iii) operating the robotic arm. At a low level, a single board computer (Raspberry Pi4) and a dedicated board are used to acquire and transfer the data from the camera and the motor to the main computer via ROS.

The system is programmed to perform the required task semi- or fully autonomously. Specifically, the system integrates navigation algorithms that fuse the RGB cameras and the RTK/GPS sensor to move autonomously in confined spaces such as vineyards, greenhouses (hydroponic), and open fields. The latter implements a semantic segmentation algorithm as in [7], where the field is semantically segmented to identify the soil, the crop, and the sky. This information is then combined with the system odometry and the GPS sensor to move in the field correctly. Moving in a confined environment simplifies the task by adding geometrical information of the space so approaches such as [1] can be easily exploited to move towards the end of the row correctly.

4 System Evaluation

The system was tested in a controlled, open-range scenario where we evaluated the robot's capability to move towards a target position to collide with obstacles on its way and to detect and scan for crops (strawberries). The robot aligned with the setup, with a soilless bag of strawberries positioned 1.5 cm from the ground. The robot was instructed to move towards the hydroponic structure and to stop when a ripe strawberry was identified [14]. Once in position, the system was programmed to acquire images with the multi-spectral camera to better recognize the fruit and its ripe status. Figure 2 shows a sequence of the scanning sequence and a sample of the acquired images.

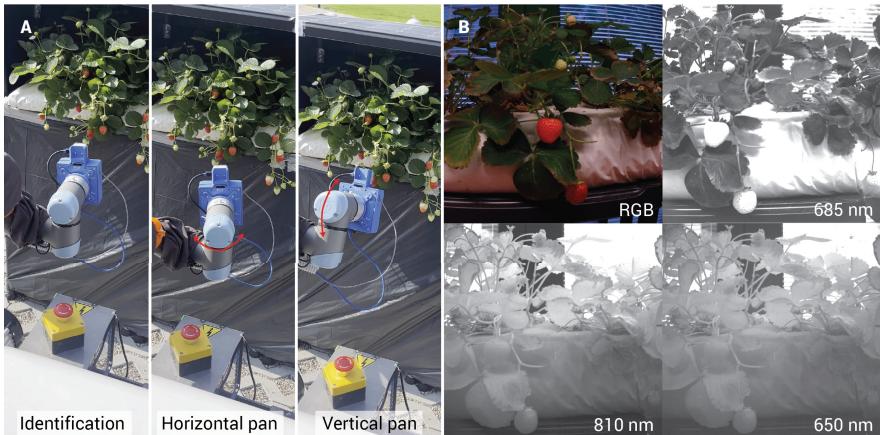


Fig. 2. (A) Scanning sequence of the fruit (B) Results of the multi-spectral camera imaging.

5 Discussion and Conclusions

This paper presented a modular approach to developing robotic platforms for precision agriculture. The robotic system comprises a track-based rover on top of which a series of manipulators and sensors can be mounted to perform autonomous tasks. In the current configuration, it mounts a UR5e and a multi-spectral camera for plant monitoring tasks. Thanks to the equipped vision system, RTK/GPS, and laser finder sensors, the system can navigate autonomously in confined and open spaces. The prototype establishes a solid foundation for further research and development to enhance agricultural operations' efficiency, safety, and profitability. This technology promises to advance sustainable farming practices and emphasizes the pivotal role of innovation in this field.

References

1. Aghi, D., Mazzia, V., Chiaberge, M.: Local motion planner for autonomous navigation in vineyards with a RGB-D camera-based algorithm and deep learning synergy. *Machines*, **8**(2) (2020)
2. AGROINTELLI: Robotti lr.<https://agrointelli.com/robotti/lr/>
3. Bruzzone, L., Nodehi, S.E., Fanghella, P.: Tracked locomotion systems for ground mobile robots: a review. *Machines*, **10**(8) (2022)
4. Macenski, S., Foote, T., Gerkey, B., Lalancette, C., Woodall, W.: Robot operating system 2: design, architecture, and uses in the wild. *Sci. Robot.* **7**(66), eabm6074 (2022)
5. Makers, P.: Greenbot. <https://www.precisionmakers.com/en/greenbot-2/>
6. Oliveira, L.F.P., Moreira, A.P., Silva, M.F.: Advances in agriculture robotics: a state-of-the-art review and challenges ahead. *Robotics*, **10**(2) (2021)
7. Panda, S.K., Lee, Y., Jawed, M.K.: Agronav: autonomous navigation framework for agricultural robots and vehicles using semantic segmentation and semantic line detection. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE (2023)
8. Raussendorf: Fruit robot. <https://www.raussendorf.de/en/fruit-robot.html>
9. Robotics, F.: Meet tertill-a better way to weed. <https://tertill.com>
10. Solutions, V.: Vitirover-a revolution in soil grassing management. <https://www.vitirover.fr/en-home>
11. Tecnologies, N.: Dino-autonomous mechanical weeding robot. <https://www.naio-technologies.com/wp-content/uploads/2019/04/brochure-DINO-ENGLISH-HD.pdf>
12. Tecnologies, N.: Oz-weeding, transportation and harvest assistance robot. <https://www.naio-technologies.com/wp-content/uploads/2019/04/brochure-OZ-ENGLISH-HD.pdf>
13. Tecnologies, N.: Ted—multifunctional straddling vineyard robo. <https://www.naio-technologies.com/wp-content/uploads/2019/04/brochure-TED-ENGLISH-3.pdf>
14. Visentin, F., Castellini, F., Muradore, R.: A soft, sensorized gripper for delicate harvesting of small fruits. *Comput. Electron. Agric.* **213**, 108202 (2023)
15. Visentin, F., et al.: A mixed-autonomous robotic platform for intra-row and inter-row weed removal for precision agriculture. *Comput. Electron. Agric.* **214**, 108270 (2023)



Design and Radiation Resistance Analysis of a Robotic Bolting Tool Applied to IFMIF-DONES Maintenance

Manuel Ferre^(✉) , Violeta Redondo , Nancy Barbosa , Paul Espinosa ,
and Miguel Á. Sánchez-Urán

Centre for Automation and Robotics (CAR) UPM-CSIC, Universidad Politécnica de Madrid, C/
José Gutiérrez Abascal 6, 28006 Madrid, Spain

m.ferre@upm.es

Abstract. This work proposes a design for a robotic bolting tool to be used in the maintenance procedures for the installation of IFMIF-DONES (International Fusion Materials Irradiation Facility - DEMO Oriented Neutron Source). As it will have to work under radiation this paper presents an analysis of its radiation resistance. Specifically, the goal of the bolting tool developed in this contribution is the installation of a Quick Disconnection System (QDS). This is a device which firmly joins pipe flanges, which avoid cutting and welding operations.

Keywords: Bolting tool · Robotic End-Effector · Radiation Resistance · IFMIF-DONES · Remote Handling · Piping · Quick Disconnection System (QDS)

1 Introduction

Operating bolts is a key task in several industries, such as maintenance and assembly. Depending on the environment it can be dangerous for human operators [1], this is why, important efforts have been made to automate it [2]. This work will focus on tightening at high torques, this means torques that could be greater than what the robot can withstand [3]. To protect it, a solution is to acquire specialized equipment to resist those reaction torques [4] or to design tools that protect the robot by firmly holding the base of the target and closing the force loop without affecting the robot. This last option is the one used in this work. Specifically, the bolting tool presented here will connect and disconnect sections of pipes in the maintenance procedures in IFMIF-DONES. One of the key aspects of this tool is that it will work in radiation environments, which means that it will have to resist those harsh conditions. For this reason, this contribution will focus not only on the design of the tool but also on estimating the radiation dose it can be expected to endure without failing.

There are two different types of failures for tools working under radiation. Single effect failures, where the electronics are capable of detecting the error and resetting going back to the state before [5], and permanent failures where the tool would remain blocked and need manual intervention.

1.1 IFMIF-DONES

IFMIF-DONES (International Fusion Materials Irradiation Facility - DEMO Oriented Neutron Source) is a facility being built in Granada (Spain) whose objective is to characterize and qualify structural materials under severe irradiation conditions of a neutron field with an energy spectrum similar to the one present in a fusion power reactor [6]. In Fig. 1 left, can be seen a sketch of the main building.



Fig. 1. (Left) View of the main building of the IFMIF-DONES facility. (Right) View of the High Energy Beam Transport Line (HEBT) of IFMIF-DONES.

Several components in the facility have foreseen maintenance as they are critical for the functioning of the system. Due to the hazardous radiation conditions these procedures will take place by remote handling [7]. In Fig. 1 right, is displayed the High Energy Beam Transport (HEBT) Line, this is where the tool presented in this paper will work. One of these components is the Beam Dump (BD), where the beam ends during the commissioning and decommissioning phases. This module has cooling pipes connected to its rear part using QDS, as shown in Fig. 2 left.

1.2 Quick Disconnection System (QDS). An Example of a Complex Bolting Task

A QDS is a mechanical device designed to join sections of pipes. The main advantage of using this mechanism is that it makes the tasks quicker and avoids the cutting and welding procedures. As can be seen in Fig. 2 right, a QDS consists in several links with a nut and a screw. When tightening the screw, the clamp section adjusts to the flanges

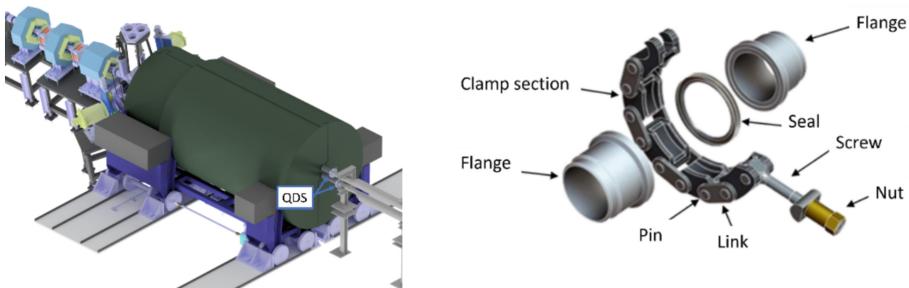


Fig. 2. (Left) Rear cooling pipes of the BD. (Right) Quick Disconnection System (QDS)

of the pipes. The seal between the two sections allows the isolation of the interior of the pipe enabling even the vacuum inside.

2 Proposed Design

The proposed design can be seen in Fig. 3. It consists of a camera that gives feedback about the position and orientation of the bolt, a servomotor with a harmonic gear box which is connected to the socket wrench by custom torque transmission elements and aluminum casing parts for structural integrity. The casing parts include the fingers below the socket wrench which are the contact points with the QDS and redistribute the reactions forces protecting the robot joints. Also, among the torque transmission elements there are a load cell and a spring to give feedback about the attachment of the bolt. More details of the design and information about the validation experiments done in the laboratory and its integration in the remote handling procedure can be found in [8].

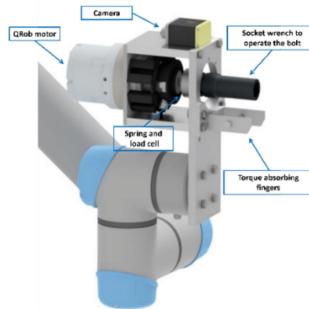


Fig. 3. Parts of the bolting tool for QDS installation

3 Radiation Resistance Estimation

As was done in [9] for the finger of a robotic hand, here it has been analyzed the radiation resistance of the tool taking into consideration the materials and electronic components selected. The parts of the bolting tool have been divided into two groups: electronic components, and metallic components. The estimation for the radiation resistance for each can be seen in Tables 1 and 2. The significant cumulative radiation damage threshold means a different type of failure for each type of component, for example in the metallic parts it corresponds to the dose when defects appear causing an increase in yield strength and a decrease in ductility [10]. However, for the camera it corresponds to the dose when the received image has no longer enough quality to give the operator proper feedback of the ongoing task.

According to literature, in motors the most sensible parts to radiation are the sealing and the lubrication [10]. As stated in [11], independently of the elastomers used for the sealing and the lubrication there are no significant damages for cumulative doses under

Table 1. Significant cumulative radiation damage threshold for the metallic components [11]

| Component | Material | Significative cumulative radiation damage threshold |
|------------------------------|---------------------|---|
| Torque transmission elements | Stainless Steel 304 | 1011Gy |
| Socket wrench | Stainless Steel 304 | 10^{11} Gy |
| Fingers | Stainless Steel 304 | 10^{11} Gy |
| Spring | Stainless Steel 301 | 10^{11} Gy |
| Support elements and casing | Aluminum 6060 | $5 \cdot 10^{11}$ Gy |

10kGy. Besides that, the components of the motor can increase their temperature due to the radiation accelerating the damage.

It is also important to mention that in the case of the camera, the radiation has two main effects. One of them is the dose rate effect that appears only when the camera is under radiation, it consists in the presence of snow in the image, that gets more intense when the radiation rate increases; in [13] the damage threshold was at 155Gy/hr. And the other one is the total dose effect, which remains even when the radiation ceases, it can appear in different forms, such as the loss of the picture [12]. The references found for this component relate to studies done several years ago, so it is expected for those thresholds to have got a bit higher due to the general improvement of the viewing systems.

Table 2. Significant cumulative radiation damage threshold for the electronic components

| Component | Significant cumulative radiation damage threshold | Ref |
|-----------|---|------|
| Load cell | 100k Gy | [10] |
| Camera | 400 Gy (Dose rate effect) | [12] |
| Motor | 10 kGy | [11] |

According to the most recent radiation maps of the accelerator vault in IFMIF-DONES [14], one day after shutting down the beam the radiation in the working area of the bolting tool will be between 0,05 and 0,1 mSv/h. The installation of the QDS is expected to take no more than 1 h, this means that each time the task is done the bolting tool will get maximum 0,1 mGy, which is well under the threshold mentioned in the analysis. As a result, the bolting tool can be expected to last for several maintenance cycles, and it can also be shielded to make it last even longer.

4 Conclusion

The tool presented in this contribution is aimed to take part in the maintenance procedures of IFMIF-DONES. The novelty in this work is the update of the design and the analysis of its radiation resistance with a positive outcome once again. This means that the design

can keep on progressing as it meets the requirements and presents an innovative way to install QDS by remote handling.

Thanks to the study of the radiation resistance of the components and materials, it is now possible to change the most sensible ones or protect them to increase the life expectancy of the device.

Acknowledgements. This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Program (Grant Agreement No 101052200 — EUROfusion). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

References

1. Jian, X., et al.: The robot maintenance method for fitting bolt in high-altitude settlement environment. In: 2021 China Automation Congress (CAC). <https://doi.org/10.1109/CAC53003.2021.9728609>
2. Jia, Z., et al.: A survey of automated threaded fastening. IEEE Trans. Autom. Sci. Eng. <https://doi.org/10.1109/TASE.2018.2835382>
3. Tao, R., et al.: A robotic end-effector for screwing and unscrewing bolts from the side. IEEE Robot. Autom. Lett. <https://doi.org/10.1109/LRA.2022.3192773>
4. Nam, H.: Design of a bolting robot for constructing steel structure. In: 2007 International Conference on Control, Automation and Systems, Seoul, Korea (South) (2007). <https://doi.org/10.1109/ICCAS.2007.4406667>
5. Coloma, S., et al.: The effect of ionizing radiation on robotic trajectory movement and electronic components. In: 2023 Nuclear Engineering and Technology. <https://doi.org/10.1016/j.net.2023.07.041>
6. Ibarra, A., et al.: The IFMIF-DONES project: preliminary engineering design. Nucl. Fus. **58**(10), 105002 (2018). <https://doi.org/10.1088/1741-4326/aad91f>
7. Miccichè et al.: The remote handling system of IFMIF-DONES. Fus. Eng. Des. <https://doi.org/10.1016/j.fusengdes.2019.01.112>
8. Ferre, M., et al.: Forces analysis on robotics screwing tasks. In: 6th Iberian Robotics Conference, ROBOT (2023). Not published yet
9. French, R., et al.: Usability study to qualify a dexterous robotic manipulator for high radiation environments. In: 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA). <https://doi.org/10.1109/ETFA.2016.7733521>
10. Houssay, L.P.: ROBOTICS AND RADIATION HARDENING IN THE NUCLEAR INDUSTRY. Master Thesis, 2009, University of Florida
11. Vandergriff, K.U.: Designing Equipment for use in gamma radiation environments (1990)
12. Sharp, R.E., Gamma radiation effects on CCTV cameras — an update. In: 2009 European Conference on Radiation and Its Effects on Components and Systems <https://doi.org/10.1109/RADECS.2009.5994688>
13. Sharp, R.E., et al.: Radiation tolerance of current CCD-based CCTV cameras. In: Proceedings of the Third European Conference on Radiation and its Effects on Components and Systems. <https://doi.org/10.1109/RADECS.1995.509785>
14. López, A.J.: Update of the radiation dose maps of the accelerator rooms v1.0. Report EFDA_D_2QBYLP (2023). Available upon request to the corresponding author



Estimation of the Orientation of the Connector for Dual-Arm Robotic Cable Manipulation

Andrea Monguzzi^(✉), Donato Malascorta, Andrea Maria Zanchettin, and Paolo Rocco

Politecnico di Milano, DEIB, Piazza Leonardo da Vinci 32, 20133 Milan, Italy
{andrea.monguzzi, andreamaria.zanchettin, paolo.rocco}@polimi.it,
donato.malascorta@mail.polimi.it

Abstract. The robotic manipulation of semi-deformable linear objects (SDLOs), such as cables featuring a connector, is currently a challenge. This work addresses the problem of inserting different kinds of non-symmetric connectors once the corresponding SDLO is grasped in proximity to its end, exploiting a dual-arm robot and an RGB in-hand camera. A vision algorithm is proposed to estimate the connector orientation, and a regrasping routine is introduced if a considerable reorientation has to be performed. The method has been experimentally validated by manipulating and inserting six different types of connectors.

1 Introduction

Semi-deformable linear objects (SDLOs), such as cables featuring rigid parts (i.e. connectors), play a crucial role in the assembly processes of electromechanical systems such as vehicles, aircraft or 3C products. However, due to the challenges characterizing the automation of SDLOs manipulation, operations involving SDLOs are today completely executed by human operators, often with poor efficiency, which can create bottlenecks in production.

In this work, we propose a strategy to automate the connection operation involving several kinds of non-symmetric connectors exploiting a dual-arm robot and an RGB in-hand camera. In particular, once the robot grasps the SDLO near the end, the actual orientation of the connector is estimated by a vision algorithm, and the connector is reoriented to properly perform the connection operation. A regrasping routine is also implemented in case a considerable reorientation of the connector has to be performed. The only input required is an image of the connector grasped in the correct pose for executing the connection operation. Figure 1 shows the kinds of connectors considered in this work. Note that the connectors exhibit asymmetries in the portion involved in the connection operation. Therefore it is essential to orient them correctly before performing the insertion. In particular, [1] analyses the mating tolerances of various cable connectors, showing that the position control method is suitable for connection operation if the connector is properly oriented. [2] proposes instead a method to predict symbolic states during the connector-socket insertion task and perform hence the insertion operation. However, in this case, both a vision sensor and a force one are necessary, and the cycle time is not deterministic since the connector is not reoriented before the insertion. In theory, one could utilize modern object recognition techniques [3] to identify

connectors and estimate their position by leveraging methods like shape-based matching [4] exploiting the CAD model. Such methods, however, rely on specific features of the connector such as colour, texture, dimension and shape and are, thus, difficult to generalize. In this work, no particular feature in terms of colour or shape is required to properly estimate the orientation of the connectors and a simple RGB camera is used.



Fig. 1. Kinds of connectors considered (from left to right): HDMI male, VGA male, rj45, IEC 320 C13, 3 pins Italian plug, IP67 automotive connector.

2 Methodology

We consider a dual-arm robot equipped with an in-hand camera and two pairs of custom fingertips (Fig. 2a): one to firmly grasp the cable, and the other, shown in Fig. 2b, featuring a layer of soft rubber so to properly grasp all the different kinds of connector considered. The dual-arm robot first grasps the SDLO in proximity to its connector, in a point assumed known. Then the other arm, featuring the in-hand camera, approaches, and an image is acquired aiming to estimate the connector orientation (Fig. 2a). A frame $\{\mathbf{h}_C\}$, describing the connector pose with respect to the base frame $\{\mathbf{h}_B\}$, is considered. The y axis of $\{\mathbf{h}_C\}$, denoted as y_C , is placed along the insertion direction, while x_C along the longer flat face of the connector. In the detailed configuration, the position of O_C (origin of $\{\mathbf{h}_C\}$) can be assumed known with a bounded uncertainty, since the robot is grasping the cable near the connector. Moreover, the image plane can be assumed to be almost parallel to the connector face. It follows that the orientation of the connector to be estimated and compensated before the insertion can be expressed by the angle θ , defined as the smallest angle between x_C and a user-defined axis direction, selected to align the connector with the socket where it has to be inserted. This definition allows to compensate for the orientation misalignment defined by θ , minimizing the torsion applied to the cable and bringing x_C in the same direction as the user-defined axis. Assume from now on, without loss of generality, that θ is defined as the smallest angle between x_C and the direction expressed by the x_B , as shown in Fig. 2a. After the compensation of the misalignment expressed by θ , the connector is repositioned, as explained in the next paragraph, and another image is acquired. A matching score H is computed to compare the current orientation of the connector with the desired one, specified via an image previously acquired by the user, showing the connector in the proper pose before the insertion. After the reorientation to compensate θ , the connector may indeed be in the correct configuration for insertion or upside down, and this is revealed by comparing the value of H to a threshold. In the latter case, a dual-arm regrasping strategy is executed to rotate the connector by 180°, moving the connector into the proper configuration for insertion.

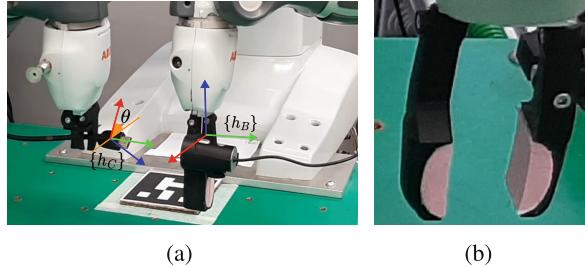


Fig. 2. (a) Connector orientation estimation. Relevant frames and θ are highlighted. x , y , z axes are depicted in red, green and blue, respectively. (b) Custom fingertips.

Our methodology involves calculating θ and subsequently H and may seem less effective than a solution that immediately computes the rotation to bring the connector directly into the insertion orientation. However, generalizing such a strategy for different types of connectors, as the ones shown in Fig. 1, is complex and may lack robustness. The proposed strategy is instead more general. Finally, the used vision algorithms are computationally efficient. The following paragraphs present the vision algorithm to compute θ and H , and the regrasping strategy, respectively.

Since connectors lack general and peculiar features and are usually monochromatic, edges are exploited to estimate θ as shown in Fig. 3. In the following, the bold letters refer to the phases depicted in Fig. 3. After the image is acquired (**a**), the brightness is increased (**b**), and the resulting image is converted in grayscale and cropped, exploiting the rough knowledge of O_c . Since edge detection results are sensitive to image noise, Gaussian blur is applied to smooth it (**c**). Edges are then detected using the Canny Edge detector (**d**), whose parameters are tuned considering tests involving the connectors in Fig. 1. Some morphological transformations based on square kernels, namely erosion, dilation and closing, are then applied (**e**). Subsequently, the contours in the image are extracted, and the biggest contour is then selected (**f**). Due to the previous cropping, this contour corresponds to the one of the connector face. Finally, the convex hull of the selected contour is computed (**g**), and its minimum oriented bounding box (OBB) is considered (**h**). θ is computed considering the longer side of the OBB (aligned with x_C) and the user-specified direction defined by x_B . Once the misalignment expressed by θ is compensated, two situations can arise and are shown in the central images of Figs. 4a and 4b. The yellow contour represents the desired pose of the connector specified by the user to correctly perform the insertion. The robot moves to make the centroid of the actual convex hull (in red) coincide with the centroid of the yellow one (right image in Fig. 4a and 4b). To classify if the connector has the desired orientation or is upside-down, a matching score H comparing the red (R) and yellow (Y) convex hulls is defined by exploiting the Hausdorff distance:

$$H(Y, R) = \max_{y \in Y} \{h(Y, R), h(R, Y)\} \quad \text{where} \quad h(Y, R) = \max_{y \in Y} \{ \min_{r \in R} d(y, r) \} \quad (1)$$

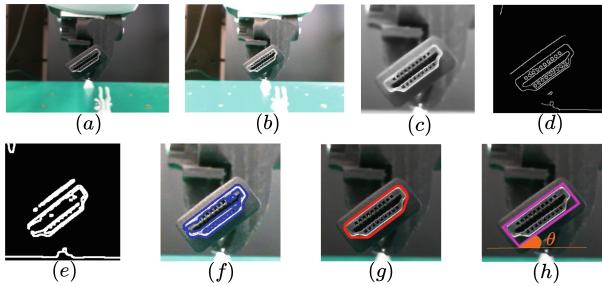


Fig. 3. Pipeline summarizing the main steps of the procedure to compute θ .

and $d(y, r)$ is set as the Euclidean distance between the two element y, r belonging to Y and R , respectively. A second image is hence acquired, and H is computed: if $H \leq T$, where T is a user-defined threshold, then the connector can be grasped by the other arm and inserted. Instead, if $H > T$, a regrasping is necessary before performing the connection operation and is executed as explained in the following.



Fig. 4. θ computation (left); rotation of θ (middle); connector centred and comparison with the desired orientation (in yellow) (right). (a) $H \leq T$, (b) $H > T$.

The dual-arm regrasping strategy is defined to rotate the connector of 180° : Fig. 5 shows the performed manipulation. In particular, the arm holding the SDLO rotates the connector of 90° around y_c , bringing it into a vertical configuration (Fig. 5c). This enables the other arm to grasp the connector, without the risk of collisions (Fig. 5d). Finally, the second arm rotates the grasped connector of 90° around y_c (Fig. 5e), moving it in the proper configuration for insertion (Fig. 5f).

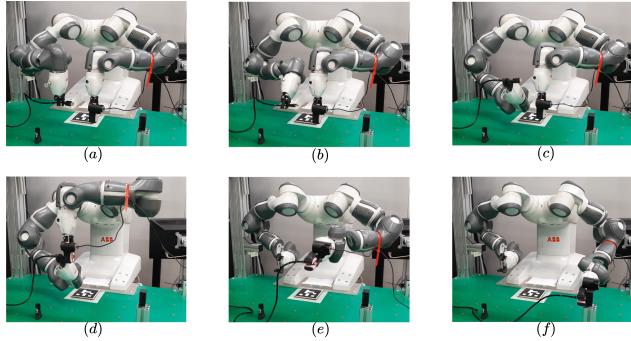


Fig. 5. (a) θ computation. (b) Rotation, centering of the connector and H computation. (c-e) regrasping strategy. (f) insertion operation.

3 Experimental Validation

Ten experimental tests are carried out for each connector shown in Fig. 1 to validate the method. The initial poses of the connectors are randomly selected. YuMi by ABB and a Microsoft LifeCam Cinema camera are used in the experiments. The accompanying video (https://youtu.be/kMEql_o38Is) shows some of the performed tests. Figure 6 depicts some convex hulls computed in the experiments. Table 1 shows the success rates linked to the correct reorientation of the connector and to its insertion in a 3D-printed socket, indicating satisfactory results. Note that changes in light conditions can lead to issues in the reorientation phase, while the insertion success rates might be improved by exploiting force sensing, especially for connectors with complex structures as VGA.

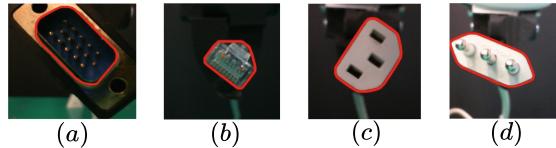


Fig. 6. Convex hulls obtained in some of the executed tests.

Table 1. Success rates of the reorientation phase and the insertion one.

| Task | HDMI | VGA | rj45 | C13 | Plug | IP67 connector |
|---------------|------|------|------|------|-------|----------------|
| Reorientation | 9/10 | 8/10 | 9/10 | 9/10 | 10/10 | 8/10 |
| Insertion | 8/10 | 7/10 | 9/10 | 9/10 | 8/10 | 8/10 |

Acknowledgment. This work was supported by Progetto Prin 2020 “Co-Mir”, prot. 2020CMEFPK.

References

1. Yumbla, F., et al.: Tolerance dataset: mating process of plug-in cable connectors for wire harness assembly tasks. *Intel. Serv. Robot.* **13**, 159–168 (2020)
2. Migimatsu, T., et al.: Symbolic state estimation with predicates for contact-rich manipulation tasks. In: 2022 International Conference on Robotics and Automation (ICRA), pp. pp. 1702–1709. IEEE (2022)
3. Loncomilla, P., Ruiz-del-Solar, J., Martínez, L.: Object recognition using local invariant features for robotic applications: a survey. *Pattern Recogn.* **60**, 499–514 (2016)
4. Zhou, H., et al.: A practical solution to deformable linear object manipulation: a case study on cable harness connection. In: 2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM), pp. 329–333. IEEE (2020)



Homography-Based Industrial Image Stitching

Johann J. Mitteramskogler^(✉), Fabian Widmoser, Markus Ikeda,
and Andreas Pichler

PROFACTOR GmbH, Im Stadtgut D1, 4407 Steyr-Gleink, Austria
johann.mitteramskogler@profactor.at

Abstract. Global robotic vision on large-scale workspaces frequently requires the composition of data from different or moving sensors. We propose a method for constructing accurate panoramic images of a robotic workspace from standard 2D images by estimating the trajectory of a moving camera. Utilizing the camera’s trajectory directly yields the positional homographies into the panoramic image, without the need to extract features and works even in the presence of zero overlap. Finally, we evaluate the overall performance of the panoramic vision approach in combination with a robotic setup on a flat work plane.

Keywords: robotic vision · homography · image stitching

1 Introduction

A major challenge in (industrial) image stitching is to estimate the relative alignment of partial images and, subsequently, how they are mapped into an overall panoramic image [4, Ch. 9]. The standard approach is to extract distinct features in each image and robustly match these features to obtain pairwise image correspondences. A reasonable amount of overlap in adjacent images is required for the success of this method, further complicated by the problem of finding a globally consistent correspondence model in the presence of many images. With only few distinct features in the overlaps, or in the presence of repeated patterns, this approach fails to produce consistent results.

These problems can be overcome if the images are captured by a camera which moves along a known trajectory in space. The geometry of the viewpoints directly translates into a relation between the pixel coordinates of the individual images, eliminating the strict requirement of overlapping regions and the problem of global consistency. In this paper we discuss the case where the camera moves along a straight line in space and evaluate the efficiency of the approach for an industrial application, where a camera/robot combination is mounted on a linear axis to cover a large workspace. A detailed description of the workstation

This work was accomplished within the Lighthouse project supported by the Austrian Institute of Technology (AIT).

together with the visual result is shown in Fig. 1. Furthermore, we estimate the overall positional accuracy of the axis-mounted robot, based on the reconstructed panoramic image, by using a motion tracking system.

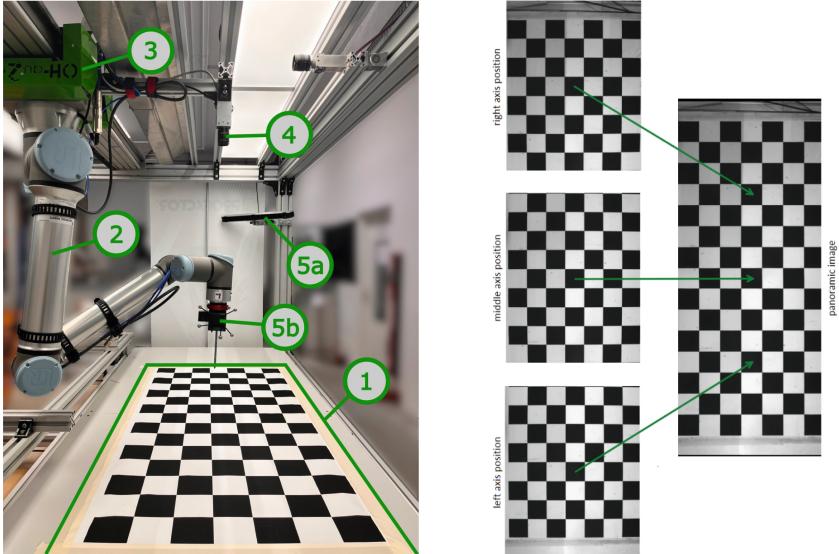


Fig. 1. Workstation setup: a pendent-mounted robot (Universal Robots UR10e (<https://www.universal-robots.com/products/ur10-robot/>)). (2) and a grey scale camera (Teledyne DALSA Genie Nano M2590 <https://www.teledynedalsa.com/en/products/imaging/cameras/genie-nano-1gige/>). (4) are mounted on a linear axis (OH-au2mate slideONE <https://oh-au2mate.de/en/shop/slideoone/>). (3) to cover the entire work plane (1). An accurate panoramic image of the work plane is reconstructed from three partial camera images at different axis positions. The positional accuracy of the robot in combination with the panoramic vision is evaluated on a chessboard pattern on the work plane by tracking the measuring tip (5b) with a motion tracking system (OptiTrack V120 Trio <https://optitrack.com/cameras/v120-trio/>). (5a).

2 Methodology

Before reconstructing the workspace from a number of partial images it is necessary to perform a one-time camera trajectory estimation wrt. the work plane.

2.1 Estimating the Camera Trajectory

The camera trajectory is estimated from at least two camera poses at different axis positions. A calibration target is placed on the work plane and the camera

pose is computed at various axis positions, cf. Fig. 2. Let $p_1, \dots, p_n \in \mathbb{R}$ be distinct axis positions and $(\mathbf{R}, \mathbf{t}_1), \dots, (\mathbf{R}, \mathbf{t}_n) \in \mathbb{R}^{3 \times 3} \times \mathbb{R}^3$ be the corresponding camera poses, where \mathbf{R} is the common rotation matrix and each \mathbf{t}_i denotes a translation vector. Estimating the axis-dependent camera translation is done by fitting a multivariate linear regression model (MLR) [3, Ch. 10], with observational data $X = (p_1, \dots, p_n)$ and $Y = (\mathbf{t}_1, \dots, \mathbf{t}_n)$. Let $\mathbf{B} = \text{MLR}(Y | X) \in \mathbb{R}^{3 \times 2}$ be the parameters of the resulting general linear model, where the columns specify the inclination and offset of the straight-line trajectory, respectively. The general extrinsic camera matrix (camera pose) at axis position p is given by:

$$\mathbf{M}_c : \mathbb{R} \rightarrow \mathbb{R}^{4 \times 4}, \quad p \mapsto \begin{bmatrix} \mathbf{R} & \mathbf{B}[p \ 1]^T \\ \mathbf{0} & 1 \end{bmatrix}. \quad (1)$$

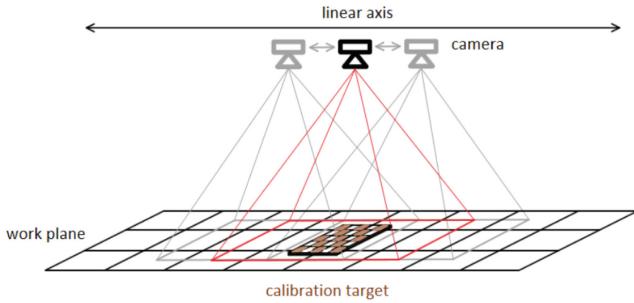


Fig. 2. Camera trajectory estimation: the straight-line trajectory is estimated from a number of camera poses with respect to a static calibration target. While the rotational part is constant, the axis-dependent translation is determined by fitting a multivariate linear regression model.

2.2 Homography-Based Stitching

As soon as the camera pose is known for arbitrary axis positions it is possible to define how individual camera images are warped into the final panoramic image. We propose a homography-based approach [4, Sec. 2.1] which defines this mapping through a common plane (the work plane). First, it is necessary to define the panoramic viewpoint on the work plane, which is done by introducing a virtual panorama camera as illustrated in Fig. 3. In this setting, the virtual panorama camera is constructed such that it corresponds to a wide angle variant of the (physical) camera at the axis' centre position. Let $\mathbf{K}_c \in \mathbb{R}^{3 \times 4}$ be the camera's intrinsic matrix, defined by its focal length and principal point, and let $\mathbf{M}_c(p)$ be the extrinsic camera matrix at axis position p , cf. Eq. 1. We choose the intrinsic virtual panorama camera matrix \mathbf{K}_{vpc} such that its focal length is identical to the camera's and principal point such that the whole work

plane can be covered. Furthermore, define the extrinsic virtual panorama camera matrix $\mathbf{M}_{vpc} = \mathbf{M}_c(p_{ac})$, where p_{ac} is the axis centre position. Since we are interested predominantly in points on the work plane, i.e. image stitching should happen along that plane, the third column of the extrinsic matrices is effectively cancelled. This yields the following 3×3 homographies:

$$\mathbf{H}_{wp \rightarrow c}(p) = \mathbf{K}_c \mathbf{M}_c(p) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^\top \quad \text{and} \quad \mathbf{H}_{wp \rightarrow vpc} = \mathbf{K}_{vpc} \mathbf{M}_{vpc} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^\top, \quad (2)$$

which define the projective transformations from the work plane to the corresponding camera image plane. Finally, the axis-dependent camera-to-panorama homography is given by:

$$\mathbf{H}(p) = \mathbf{H}_{wp \rightarrow vpc} \mathbf{H}_{wp \rightarrow c}(p)^{-1}. \quad (3)$$

After capturing parts of the workstation at specific axis positions, the rectified (undistorted) camera images are warped consecutively into the overall panoramic image using the camera-to-panorama homography from Eq. 3.

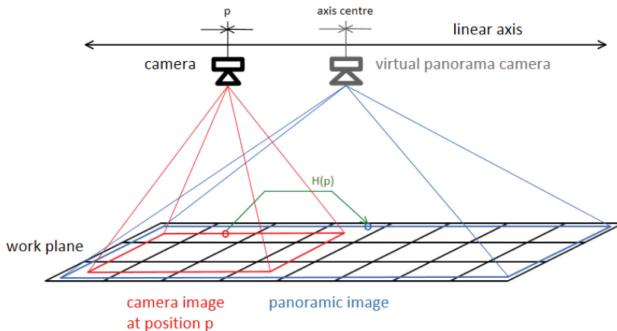


Fig. 3. Axis-dependent homography: the axis-dependent camera image is warped into the final panoramic image by constructing the homography $\mathbf{H}(p)$ from the camera image plane to a virtual panorama image plane. The latter is defined through a virtual panorama camera which extends the perspective of the physical camera located at the centre of the linear axis.

3 Results and Discussion

The attainable robot accuracy on the work plane is evaluated on a 15×7 chessboard pattern with tiles of size 0.1×0.1 m, cf. Fig. 1. For this purpose we calibrated the camera to the robot (hand-to-eye calibration), the robot itself is mounted on the linear axis such that its x -axis is parallel to the linear axis' movement direction. A physical resolution of 0.4 mm/pixel is achieved on the pattern by the camera, the latter being located at a distance of 0.9 m from the

work plane. The workspace panorama, captured and stitched as illustrated in Fig. 1 (overlaps blended linearly), is passed to a corner detector which yields the coordinates of the 14×6 inner chessboard corners. These corners, after converting to robot coordinates via the calibration, were tapped by the robot with an externally tracked measuring tip, cf. Fig. 1. Corner detection and calibrations were performed using OpenCV [1], the MLR model was computed with scikit-learn [2]. A comparison of the measuring tip positions on the (physical) chessboard inner corner grid is shown in Fig. 4.

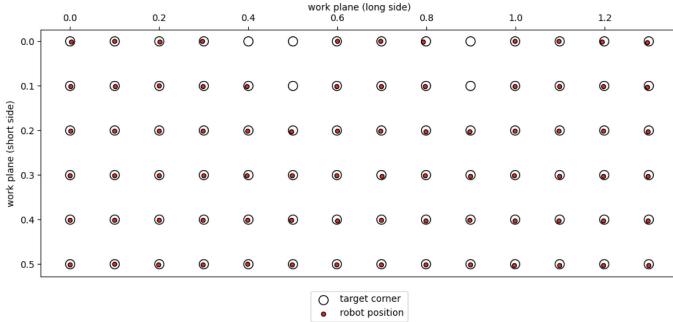


Fig. 4. Experimental results (axis labels in metre): the robot accuracy is evaluated on a sparse 14×6 grid on the work plane. A corner detector is used to find the inner chessboard corners of the workspace panorama from Fig. 1, which are translated to robot coordinates through a robot-camera calibration. The diagram shows the actual robot positions on the work plane grid (red dots), measured by an optical tracking system, and the physical corner locations (circles). Missing measurements are due to occlusions of the tracking markers by the robot arm.

On average, the robot positions deviate 1.4 mm from the ideal location on the work plane, which is sufficient for surface processes such as sanding and bonding on smaller and medium-sized objects. It is difficult to assess the precise impact of each source of inaccuracy. However, a critical factor in this pipeline constitutes the camera (resolution, lens, etc.) and the quality of its intrinsic calibration, including distortion parameters. This has a direct influence on the precision of robot-camera calibration. Furthermore, it should be noted that the resulting panorama is only valid near the work plane due to the projection onto said plane. Effects such as parallax and visual artefacts at overlaps may occur when stitching objects of varying altitudes.

References

1. Bradski, G.: The OpenCV Library. Dr. Dobb's J. Softw. Tools (2000)
2. Pedregosa, F., et al.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
3. Rencher, A.C., Christensen, W.F.: Methods of Multivariate Analysis. Wiley Series in Probability and Statistics (2012)
4. Szeliski, R.: Computer Vision: Algorithms and Applications. Springer (2010)



Design Principles for Upper Limb Robotic Rehabilitation: Bridging the Gap from Research Prototype to Clinical Tool

Irene Pippo^{1,2(✉)}, Marco Guazzotti^{1,3(✉)}, Giulia A. Albanese^{1(✉)},
Maddalena Mugnocco^{1(✉)}, Amel Cherif^{1(✉)}, Maura Casadio^{3(✉)},
Giovanni Berselli^{2(✉)}, and Jacopo Zenzeri^{1(✉)}

¹ ReWing s.r.l., Galleria del Corso 1, 20122 Milano, Italy

² Department of Mechanics, Energetics, Management and Transportation (DIME),
University of Genova, Via all'Opera Pia 15/A, 16145 Genova, Italy

³ Department of Informatics, Bioengineering, Robotics and Systems Engineering
(DIBRIS), University of Genoa, Via all'Opera Pia 13, 16145 Genova, Italy
irene.pippo@rewingtech.com, marco.guazzotti@rewingtech.com,
giulia.albanese@rewingtech.com, maddalena.mugnocco@rewingtech.com,
amel.cherif@rewingtech.com, maura.casadio@unige.it,
giovanni.berselli@unige.it, jacopo.zenzeri@rewingtech.com

Abstract. Robotic rehabilitation is transforming traditional practices, offering precise, intensive training, and quantitative assessment. While these devices have great potential, integrating them into clinical settings presents challenges. This paper outlines design principles for industrializing rehabilitation robots, focusing on hardware, human-machine interface, and clinical features. Hardware considerations include the choice of robust materials and the need for customization. The interface design prioritizes usability and accessibility, considering the diverse user base made of both therapists and patients. Clinical features offer evidence-based clinical protocols and outcome reports for therapists and patients. This work aims to bridge the gap between potential and practical applications on the market, enhancing the impact of rehabilitation robots in clinical settings.

Keywords: Robotic Rehabilitation · Industrialization · Clinical Impact · User Friendly

1 Introduction

Robotic rehabilitation is rapidly evolving, transforming the landscape of traditional rehabilitation tools and methodologies. Its role is to support and complement conventional rehabilitation practice, considering both rehabilitative training and evaluations. Rehabilitation robots are able to deliver highly intensive

I. Pippo, M. Guazzotti and G. A. Albanese—These authors contributed equally to the work.

and repetitive training, providing high-quality treatment at low effort [1]. These devices incorporate sensors that allow spatial and temporal accurate measurements of movement kinematics and kinetics, which can be used to derive information related to joint function. Additionally, by synchronizing the robot with other devices (e.g. electromyography, electroencephalography, Inertial Measurement Units), it is possible to broaden the scope of what can be measured. A further advantage of these devices with respect to traditional therapy lies in the possibility of developing intelligent and AI-based algorithms, to enhance therapy effectiveness. Despite these remarkable benefits, the integration of robotic rehabilitation devices into clinical practice has encountered numerous challenges [2]. Even when hospitals and clinics invest in these technologies, there is often a reluctance or difficulty in their actual use. It becomes evident that the barriers are different from mere financial considerations. Various factors play a significant role, such as difficult acceptance by therapists on the effectiveness of these technologies, logistic issues including the need for time to set up the device, or the need for assistance from specifically skilled staff [3]. These challenges raise several questions on how to bridge the gap between the potential and practical application of robotic rehabilitation in clinical settings. In this work, we present a set of fundamental design principles to enhance the impact of rehabilitative robots as they transition from research prototypes to market-ready solutions. Our focus goes beyond the essential certification requirements that robots typically need to adhere to. Instead, we aim to spotlight additional features that can substantially augment the application and acceptance of these devices within clinical environments. To present these principles and make their application easier to be understood by the reader, we will exploit a specific use case. Our use case will be an end-effector robotic device [4], designed specifically for upper limb rehabilitation that can provide varying levels of assistive forces and measure the kinematics of upper limb joints (Fig. 1).



Fig. 1. End-effector robotic device for upper limb rehabilitation

2 Product Design Principles

The design and development of advanced robotic systems requires a very specific set of skills shared across different application domains. The design of a new system is guided by fundamental principles and common steps that underpin the development process: i) identification of the expected functional requirements, ii) analysis of the design constraints, iii) definition of safety requirements, iv) evaluation of technical and economic feasibility, and v) conceptualization and ideation in a framework that comprehends mechanical, electronic, software, clinical, and biological aspects. Translating a prototype to a market-ready medtech product requires processes such as risk and quality management to ensure the safety of all the users involved, evaluation of the clinical effectiveness of the device on patients, and evaluation of the usability for clinical operators. Guidelines for the development of these aspects are provided in several specific technical standards. Compliance with these standards and regulations for medical devices is crucial to obtain the certification. In addition to these mandatory procedures, the following sections introduce further design guidelines to increase the devices' acceptance and impact in clinical practice.

2.1 Mechanical Design Strategies

From an industrial perspective, a few technical details in the mechanical design strategy make the industrialization process more effective: i) the incorporation of wear-resistant materials ensures longevity and durability; ii) the use of commercially long available components not only streamlines the manufacturing process but also facilitates replacements; iii) easier maintenance is ensured by modularity if each part can be easily accessed, removed, and replaced without requiring extensive disassembly of the entire system, thus minimizing the time the device is out of service; iv) cost-effectiveness and accessibility without compromising on quality is crucial for optimizing the device's market competitiveness. From an end-user perspective, considerations are mainly on anthropometrics, ergonomics, and comfort: i) designing the rehabilitation robot to be modular and scalable allows easy customization based on different rehabilitation needs and patients of different ages, and enables future upgrades to meet evolving healthcare requirements; ii) designing a device that can be used at different stages of the disease and by people who may also have other needs (e.g., bedridden or wheelchair users) contribute to its versatility; iii) being easily transportable and adaptable to different environments contribute to the device's widespread applicability and is especially important given the diverse settings in which healthcare professionals work; iv) fast and easy patient's placement and easily accessible adjustments streamline customized settings; v) comfort and ergonomics (e.g., the use of soft materials next to the patient's skin) foster a more relaxed environment conducive to therapy.

2.2 User-Centric GUI Strategies

The Graphical User Interface (GUI) is the primary means of interaction between the user and the device and the scope of a well-designed GUI is to prioritize the enhancement of the user experience. This could be achieved through a careful design of the interface that aims to maximize usability [5] and accessibility [6], considering the specific needs of the two main users: therapists and patients. Considering our use case, in a clinical environment therapists have different technological expertise, so a GUI should have clear sections to group related controls together and visual cues (i.e. shadings, borders) that will help therapists identify and understand all the elements without the need for extensive training processes. To further enhance the accessibility for users with specific needs, a set of customizable options needs to be implemented, allowing users to change font sizes and color schemes in order to take account of potential color blindness or other visual impairments. Other important aspects are interactivity and responsiveness. The interface should respond quickly to user inputs, providing real-time feedback and updates; this can be achieved with animations, progress bars, or status indicators. It is also important to design the interface in a way that minimizes the potential for errors or accidental inputs (e.g., by incorporating confirmation dialogs or lock mechanisms that can prevent unintended actions that could result in harm to the user or the robot). Additionally, the GUI should be useful to integrate advanced features such as data analysis capabilities, that can provide valuable insights into the therapy's effectiveness and feedback to the patients (see Sect. 2.3). As for patients, it is crucial to consider that they could have very different motor skills and cognitive abilities. Therefore, proper widgets to customize rehabilitation exercise parameters are needed. Considering cognition, visual aesthetic aspects could impact the effectiveness of the exercises: having too many objects on the screen or too elaborate backgrounds could be detrimental to patients with cognitive disabilities, making the focus on the task more challenging.

2.3 Clinical Features

Protocols can play a pivotal role in customizing the rehabilitation process to the patient's specific needs. A protocol is a set of tasks that are presented to the user in a specific order. The therapist should have the flexibility to curate the set of exercises, arrange them in a specific order, and customize the preferences based on the condition of each patient. These personalized exercise regimens could serve as the standard rehabilitation/assessment routine for the patient, keeping this information integrated into the system despite therapist changes or takeover. Additionally, the industrial player can provide pre-defined, clinically validated protocols for specific patient populations. Considering our use case, a robotic training protocol designed for stroke patients can focus on addressing upper-limb both sensory and motor deficits, or on targeting the most affected aspect in the specific patients under treatment. In this context, the therapist may choose to apply a protocol targeting specifically proprioceptive deficits [7]. Such

pre-configured protocols serve as validated starting points for therapists and can be modified as needed. These protocols not only expedite the therapy planning process but also ensure that evidence-based practices are readily available for therapists, enhancing the overall quality of care. Another essential feature for both therapists and patients is the generation of outcome reports [8], which provide insights into the progress and effectiveness of the rehabilitation program. These reports must be designed with the clinical audience in mind, ensuring that the outcomes and units of measure used are familiar to therapists, facilitating a seamless interpretation of the data. To accommodate both therapists and patients, it is advantageous to provide two separate reports. The therapist's report can offer a detailed, data-rich overview of the patient's progress, while the patient's report can be simplified and designed to be motivating, using graphical elements like smiling emoticons or motivational sentences that encourage and remind of their achievements [9]. This not only keeps patients motivated but also fosters a sense of involvement and ownership in their rehabilitation journey.

3 Conclusions

The journey from a research prototype to a market-ready rehabilitation robot is fraught with complexities and challenges. However, by adhering to the design principles outlined in this paper, we believe that rehabilitation robots can not only meet the rigorous demands of clinical settings but also contribute significantly to the motivation of patients and therapists to employ these technologies. The path forward lies in a continued collaboration between engineers, researchers, healthcare professionals, and patients to harness the full potential of these cutting-edge technologies.

References

1. Díaz, I., Gil, J., Sánchez, E.: Lower-limb robotic rehabilitation: literature review and challenges. *J. Robot.* **408–418** (2011)
2. Mehrholz ,J., Pohl, M., Platz, T., Kugler, J., Elsner, B.: Electromechanical and robot-assisted arm training for improving activities of daily living, arm function, and arm muscle strength after stroke. *Cochrane Database Syst. Rev.* **9** (2018)
3. Chen, C.C., Bode, R.K.: Factors influencing therapists' decision-making in the acceptance of new technology devices in stroke rehabilitation. *Am. J. Phys. Med. Rehabil.* **90**(5), 415–425 (2011)
4. Iandolo, R., et al.: Perspectives and challenges in robotic neurorehabilitation. *Appl. Sci.* **9**(15), 3183 (2019)
5. Abran, A., Khelifi, A., Suryn, W., Seffah, A.: Usability meanings and interpretations in ISO standards. *Softw. Qual. J.* **11**(4), 325–338 (2003)
6. Kavcic, A.: Software accessibility: recommendations and guidelines. In: The International Conference on Computer as a Tool, Belgrade, Serbia, pp. 1024–1027 (2005)
7. Albanese, G.A., Basile, E., De Momi, E., Zenzeri, J.: A new robot-based proprioceptive training algorithm to induce sensorimotor enhancement in the human wrist. In: International Conference on Rehabilitation Robotics (ICORR), Rotterdam, Netherlands, pp. 1–6 (2022)

8. Luciani, B., Braghin, F., Pedrocchi, A.L.G., Gandolla, M.: Technology acceptance model for exoskeletons for rehabilitation of the upper limbs from therapists' perspectives. *Sensors* **23**, 1721 (2023)
9. Popović, M.D., Kostić, M.D., Rodić, S.Z., Konstantinović, L.M.: Feedback-mediated upper extremities exercise: increasing patient motivation in poststroke rehabilitation. *BioMed Res. Int.* 520374 (2014)



Vision-Based Path Planning for Autonomous Robotic Fabric Layering with a Sewing Machine

Marcel Lahoud^{1,2(✉)}, Gabriele Marchello¹, Eleonora Fontana¹,
Amal Meddahi¹, Khelifa Baizid¹, Farshad Nozad Heravi^{1,2}, Haider Abidi¹,
Mariapaola D'Imperio¹, and Ferdinando Cannella¹

¹ Industrial Robotics Facility, Italian Institute of Technology, Genoa, Italy

{marcel.lahoud,gabriele.marchello,eleonora.fontana,amal.meddahi,
khelifa.baizid,farshad.nozadheravi,haider.abidi,mariapaola.dimperio,
ferdinando.cannella}@iit.it

² DIBRIS, University of Genoa, Genoa, Italy

<https://inbot.iit.it>

Abstract. This paper introduces an innovative robotic solution to autonomously plan stitching paths for two fabric layers. The developed system is inspired by the conventional manufacturing process and aims at assisting the human effort while ensuring high-quality results. Extensive experiments reveal comparable performance to skilled human operators in terms of path accuracy and stitching quality, accommodating various fabric types and patterns. This research advances robotic fabric processing offering a modular and flexible solution. It holds promise for the textile industry and precision-dependent applications, simplifying automation and enhancing efficiency.

1 Introduction

In recent years, the clothing industry has made substantial investments in automating the sewing process to reduce manufacturing costs and time [1–3]. Major players in the industry, such as Adidas and TY Garments, have adopted partial or complete automation [4]. However, working with soft materials like fabrics presents several challenges, primarily due to the material nonlinear behavior [5], resulting in issues such as fabric fold-overs and localization uncertainty. As a result, human operators have remained pivotal in handling such materials, limiting full automation.

This paper introduces an autonomous robotic system designed to automate the assembly of textile components used in cycling clothing production, in collaboration with Decathlon, the sportswear company. Our system keeps human operators in the process, improving working conditions and production efficiency. The robotic solution addresses fabric manipulation and control of an industrial sewing machine and was tested in a real manufacturing scenario. As a result, it

significantly reduces operator stress, yet guarantees comparable results to manual production in terms of both time and quality. This work is part of SOFT-MANBOT, a project funded by the European Union Horizon 2020 program, aiming to automate the manipulation and assembly of deformable materials while collaborating with human operators.

2 State of Art

Efforts to automate soft material production, like clothing, began decades ago. Gershon’s early work decomposed robotic sewing tasks to address fabric flexibility-related uncertainty. Tasks included vision-based contour tracking, fabric tension sensing, feeding fabric, and sewing [1]. Zhu *et al.* categorized challenges, focusing on hardware design (*e.g.*, soft-object grippers), state sensing, reception-action modeling, and manipulator planning/control [6]. Kudo *et al.* introduced a sewing system with two fabric-handling robots mimicking human behavior, aided by a task-oriented robot language and graphical interface [2]. Shungo *et al.* proposed a dual-arm manipulator with rollers as end effectors, utilizing vision feedback for stitching trajectory control [3]. Robust control during fabric manipulation is a key challenge. Koustoumpardis and Aspragathos employed an intelligent hierarchical controller with neural networks and fuzzy logic for fabric tension control [7]. Kumar and Ashok used neural networks to predict 2R manipulator position errors due to fabric mechanics, with feedforward correction [8]. Winck *et al.* addressed fabric position and tension using a servo-controlled manipulator, combining machine vision for real-time position control [9]. These strategies highlight the evolution of the automated solutions designed to manage fabric uncertainty in robotic sewing.

3 Methodology

A multi-robot system (Fig. 1) was developed to automate the sewing process of cyclist garments, made of a foam pad and an elastic cloth [10]. This robotic cell is composed of two robotic arms, one equipped with a soft fingers gripper and the other with a needle gripper, a vision system, and a Cartesian robot equipped with a clamping mechanism. The Cartesian robot is responsible for holding the cloth during the wrinkle removal task [11] and subsequently positioning it under the sewing machine needle to ensure the proper stitching of the pad while maintaining a flat fabric and avoiding creases.

The vision system acquires a picture of the pad and Principal Component Analysis (PCA) is hence applied to derive the center of mass, principal axes, and multiple waypoints relative to the center of the pad. Additionally, the information extracted from the picture is first converted from pixels into the dimensions of the real pad, then shrunk in order to ensure the application of the stitches over the pad and then interpolated through a Spline function.

The so-obtained parametric trajectory ($[x(t), y(t), \text{atan}2(y(t), x(t))]$) has to be adapted to each geometric model of the robots. The model of the Cartesian robot

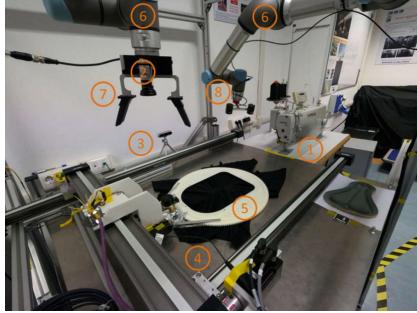


Fig. 1. An overview of the designed robotic cell. It includes an industrial sewing machine (1), an RGB (2) and depth camera (3), for pad and fabric wrinkle detection, a Cartesian robot (4) with a clamping mechanism (5) for fabric manipulation, two UR5e robotic arms (6) with electric grippers featuring soft fingers (7), and two pneumatic needle grippers (8) for wrinkle flattening and pad handling.

is straightforward, with decoupled degrees of freedom. Conversely, the model of the robotic arm is complex, but solved by using ROS controller libraries for UR5e. In order to ensure the pad alignment with the sewing needle, the pose of the needle is imposed on the trajectory. Figure 2a depicts the schematic representation of the geometry of the foam pad, with P as the center of mass of the pad and N as the current trajectory position, *i.e.* the needle position. Therefore, the solution for the Cartesian motion is derived by Eq. 1.

$$\begin{bmatrix} X \\ Y \\ \theta \end{bmatrix} = \begin{bmatrix} X_N - x_{off} - (x(t)\text{Cos}(\theta) - y(t)\text{Sin}(\theta)) \\ Y_N - y_{off} - (x(t)\text{Sin}(\theta) + y(t)\text{Cos}(\theta)) \\ \phi - \pi/2 \end{bmatrix} \quad (1)$$

where x_{off} and y_{off} are the offsets from the joint axes of the Cartesian robot to the center of mass of the foam pad (P), while θ and ϕ are the pad rotation wrt \mathcal{F}_0 and the stitch direction wrt \mathcal{F}_P .

Similarly, it is possible to derive the transformations of the tool frame of the UR5e. Since ROS solves the inverse kinematics of the UR5e robot and it is already well represented in their corresponding packages, we focused on the computation of the poses of the end-effector flange (\mathcal{F}_{ee}) of the robot (Fig. 2b). Consequently, the solution for the motion of the UR5e robot is obtained by solving Eq. 2.

$${}^0H_{ee} = \begin{bmatrix} \text{Sin}(\theta) & -\text{Cos}(\theta) & 0 & X_N - ((-A - y(t))\text{Cos}(\theta) - x(t)\text{Sin}(\theta)) \\ 0 & 0 & -1 & Y_N + B \\ \text{Cos}(\theta) & \text{Sin}(\theta) & 0 & Z_N - ((A + y(t))\text{Sin}(\theta) + x(t)\text{Cos}(\theta)) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

Once the stitching trajectory is derived, the robotic arm with the needle gripper places the pad over the flattened fabric held in the clamping mechanism of the Cartesian robot. Both robots move in sync along the trajectory under the

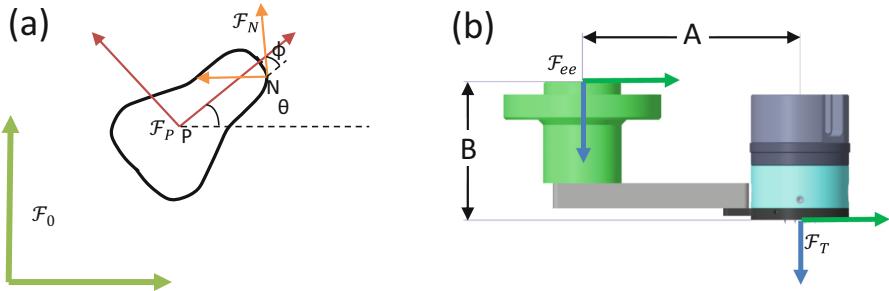


Fig. 2. (a) On the left, a schematic of the geometry of the planar foam pad during stitching. (b) On the right, the schematic of the needle gripper depicts straightforward vertical **B** and horizontal **A** translation, sourced from the CAD model.

sewing needle through collision-free waypoints, controlled by a PD controller in ROS. After completing the seam, the product is removed, and the system is re-initialized for a next cycle¹.

4 Results and Discussion

The pivotal step of the automated sewing process relies on the synchronisation of the motion of both the Cartesian robot and the robotic arm equipped with the needle gripper that holds the pad. A misalignment between the robots would lead into poor quality of the final product. Therefore, the performance of the system in autonomously planning stitching paths for fabric layers was evaluated through extensive experiments, in which the stitches had to be applied at a constant distance of 8 mm from the border, as suggested by Decathlon staff. It was possible to observe over dozens of tests that the distance between the stitches and the border of the pad varies between a minimum of 3 mm and a maximum of 10 mm. Given the desired distance of 8 mm from the border of the pad, these values result in an uncertainty of about 3.5 mm, a value acceptable for the manufacturer standard (Fig. 3).



Fig. 3. An overview of the automatically sewn cyclist garments.

¹ [video](#).

The time required to run the full stitching trajectory resulted to be constant over the different tests and equals to 131 s. Despite this value results to be slightly higher than the human performance measuring about 100 s, it is important to notice that in the conventional manufacturing process, the operator is involved full-time. Conversely, in the automated scenario, the workload of the operator is reduced, consequently opening to the possibility of having them work on a multitude of sewing processes at the same time.

5 Conclusion

This research introduces an innovative robotic fabric manufacturing approach that efficiently handles soft materials, ensuring high-quality performance with minimal human intervention. By seamlessly integrating robotics and vision systems, the system achieves autonomous real-time processing and precise execution of fabric stitching paths. Extensive experimentation highlights its exceptional capability to autonomously plan stitching paths, closely matching the proficiency of skilled human operators.

The proposed system pivots around the capability of deriving the stitching trajectory from a picture, guaranteeing great flexibility and capability to adapt to soft materials of different shapes and sizes. Consequently, the proposed approach holds considerable potential for the textile industry and other precision-dependent domains, marking a significant step forward for efficient and high-quality fabric layering processing.

References

1. Gershon, D.: Parallel process decomposition of a dynamic manipulation task: robotic sewing. *IEEE Trans. Robot. Autom.* (1990)
2. Kudo, M., Nasu, Y., Mitobe, K., Borovac, B.: Multi-arm robot control system for manipulation of flexible materials in sewing operation. *Mechatronics* (2000)
3. Shungo, T., et al.: Development of fabric feed mechanism using horizontal articulated dual manipulator for automated sewing. In: 2021 IEEE 17th CASE, IEEE (2021)
4. Lee, S., Rho, S.H., Lee, S., et al.: Implementation of an automated manufacturing process for smart clothing: the case study of a smart sports bra. *Processes* (2021)
5. Koustoumpardis, P.N., Smyrnis, S., Aspragathos, N.A.: A 3-finger robotic gripper for grasping fabrics based on cams-followers mechanism. In: *Advances in Service and Industrial Robotics: Proceedings of the 26th International Conference on RAAD 2017*, Springer (2018)
6. Zhu, J., Cherubini, A., Dune, C.: et al.: Challenges and outlook in robotic manipulation of deformable objects. *IEEE Robot. Autom. Mag.* (2022)
7. Koustoumpardis, P.N., Aspragathos, N.A.: Intelligent hierarchical robot control for sewing fabrics. *Robot. Comput. Integrat. Manuf.* (2014)
8. Kumar, P., et al.: Artificial neural network based geometric error correction model for enhancing positioning accuracy of a robotic sewing manipulator. *Proc. Comput. Sci.* (2018)

9. Winck, R.C., Dickerson, S., Book, W.J., Huggins, J.D.: A novel approach to fabric control for automated sewing. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, IEEE (2009)
10. Lahoud, M., Marchello, G., Abidi, H., D'Imperio, M., Cannella, F.: Robotic manipulation system for multi-layer fabric stitching. In: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, ASME (2021)
11. Fontana, E., et al.: Wrinkle-free sewing with robotics: the future of soft material manufacturing, p. 2023. Biomimetic and Biohybrid Systems, Living Machines (2023)



Electrolyzer Control Cabinet Wiring as a Holistic Approach

Elías Milloch^(✉) , Stefanie Bartelt , Milan Brisse , Robert Egel , and Bernd Kuhlenkötter

Chair of Production Systems, Ruhr-Universität Bochum, Industriestr. 38c, 44894 Bochum,
Germany
millock@lps.rub.de

Abstract. The demand for green hydrogen is expected to increase in the next years. As a result, the market for suitable production means such as electrolyzers will also increase. An essential step in meeting this growth is automating production steps, which have so far been mainly manual. A crucial part of electrolyzer production is the control cabinet assembly and wiring. Automating this production step promises high economic potential and an increased output. In this context, this paper proposes a concept for automated control cabinet wiring using industrial robots. The control cabinet of an alkaline electrolyzer is taken as a reference object and analyzed regarding its hurdles for automation. Based on this analysis, a cabinet redesign is proposed yielding a higher automation potential. Subsequently, a concept for automated wire assembly and wiring is presented. The concept includes two industrial robots, enabling maximum flexibility regarding wire supply and control cabinet wiring.

Keywords: Electrolyzer production · Control cabinet wiring · Automated wiring

1 Introduction

Green hydrogen is considered an important enabler for reducing greenhouse gas emissions. Hence the German government passed a national hydrogen strategy to increase its production [1]. An established method to produce green hydrogen is by using electrolyzers, which split water into hydrogen and oxygen using electricity. Electrolysis is done using different technologies; however, whichever technology is used, several components are in common such as gas separators, pumps, or safety devices [2]. These components are wired inside a control cabinet. Nevertheless, there is a high variance in the design and the cabinets are currently manufactured manually [3]. Due to the expected increase in the demand for electrolyzers, automated production of these control cabinets promises a high economic potential [4]. This paper presents an approach for an appropriate automated production of these cabinets. The concept focuses on the wiring of the components, as this process is about half of the overall control cabinet production time [5]. While automated wiring is generally a highly complex task, the similar structure of the electrolyzer cabinets yields several benefits. The following section presents a brief

overview of related work. Section three analyzes the structure of the control cabinets and identifies requirements for a design suitable for automation. The robot-based automation concept is shown in section four.

2 Related Work

Electrical control cabinets are predominantly fabricated manually, whereby mechanical processing, assembly, wiring, testing, and logistics consume about 57% of the production time [5]. For mechanical processing and cabinet assembly, there are some solutions to automate certain production steps [6]. However, available solutions usually can only handle a small number of product variants.

Various approaches in science and industry aim at automatic wiring as it is the most time-consuming step in control cabinet production. Industrial solutions, such as the SYNDY wiring system, are very restricted in terms of wires and components and can therefore only be used for a limited range of automation tasks. Academic approaches are more general, but they focus only on specific aspects of the wiring, such as sensors, models, or grasping technology [6]. Other work focuses on the handling of the wires (e.g., [7]), or simulation and path generation (e.g., [8]). In the context of control cabinet wiring, computer vision plays a significant role not only in the detection of the correct insertion position of the wire (e.g. [9]) but also in the recognition of the wire gripping position [10]. Furthermore, a comprehensive data model of the control cabinet is an essential prerequisite for efficient automation. The information required for the model is provided by different sources, e.g., the planning tool, ERP systems, or the different manufacturers of the integrated components. To combine all these sources, system-neutral and open-exchange formats are beneficial [11].

Related work shows approaches for specific process steps but a flexible approach covering the complete process chain from wire supply to cabinet wiring is not existent.

3 Analysis and Optimization Towards Automated Wiring

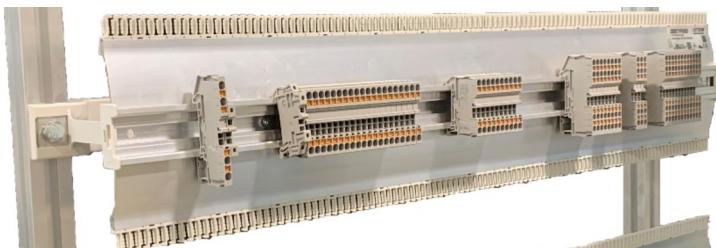
For the automated wiring, a control cabinet of an alkaline electrolyzer with a capacity of 14 kW has been chosen as a reference object for the wiring process, since this electrolyzer size promises high automation potential [4]. The cabinet was analyzed regarding its challenges and difficulties for automated wiring. The electrical components, their connection types, the number of ports, and their mounting position were considered. Furthermore, the number of utilized wires was analyzed since the more wires, the more interfering contours there are. Also, the diameters were considered, for they are crucial regarding wire handling. The results of the analysis are summarized in Table 1.

Aside from that, a qualitative analysis was carried out on the arrangement of the components and the reachability of the component's ports, to detect risks of collisions during the wiring. As depicted in Fig. 1, the ports are very near to the cable duct, which leads to a limited space for wire insertion. The terminal blocks on the right image contain ports positioned tightly to one another. This leads to limited space for the wiring and collision risks due to the number of wires in a relatively small area.

Table 1. Summarized results of the control cabinet analysis

| Criteria | Number | Types |
|-----------------|--------|---|
| El. Components | 165 | 19 (2 of them spring type; 17 screw type) |
| Number of wires | 540 | 6 (diameters [mm^2]: 0.25, 0.5, 0.75, 1, 1.5, 2) |

Based on the analysis, three modifications to the control cabinet are proposed. Firstly, the distance between the mounting rails should be extended to reduce collision risks when wiring and to ensure enough space between the wire ports and any obstacles. Secondly, an adjusted arrangement of components in the control cabinet is proposed, using a wiring frame as displayed in Fig. 2. Instead of mounting cable ducts, the wires can be routed behind the mounting rails, thus reducing obstacles during the wiring process. Lastly, components with spring terminals should be used when possible since they do not require an additional tool for screwing.

**Fig. 1.** Connecting ports of circuit breakers (left) and PLC terminals (right)**Fig. 2.** Detail of the redesigned control cabinet using a wiring frame and spring terminals

4 Cell Layout for Automated Wiring

The mounting of the wires is performed in two steps. In the first step, the wire is assembled, i.e., cut to length and crimped. The second step is the mounting into the cabinet. The interface between assembly and wiring is particularly interesting here, as the wire can be supplied in various forms after assembly. Figure 3 shows different supply variants, which will be investigated and compared experimentally in future trials.

To test and compare different concepts for automated control cabinet wiring, a flexible test cell has been designed. To achieve this, both the assembly and the wiring will be implemented in one automation cell. An automatic cutting machine, a crimping machine, a camera for checking the cable ends, and a labeling device are used for the wire assembly. A robot simulation was carried out for an initial validation of the concept.

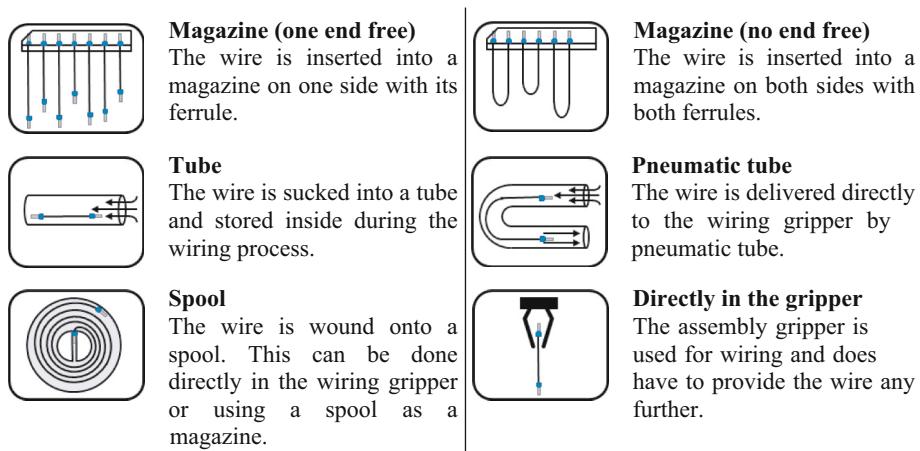


Fig. 3. Determined variants for wire assembly and wire supply

The test setup should further offer the possibility of realizing the process with one or two robots. While the workspace of the first robot must cover the wire assembly and the mounting locations, it is sufficient for the second robot to cover the mounting locations. Figure 4 illustrates the setup, which allows different concepts to be investigated and compared regarding flexibility, cycle time, and process stability.

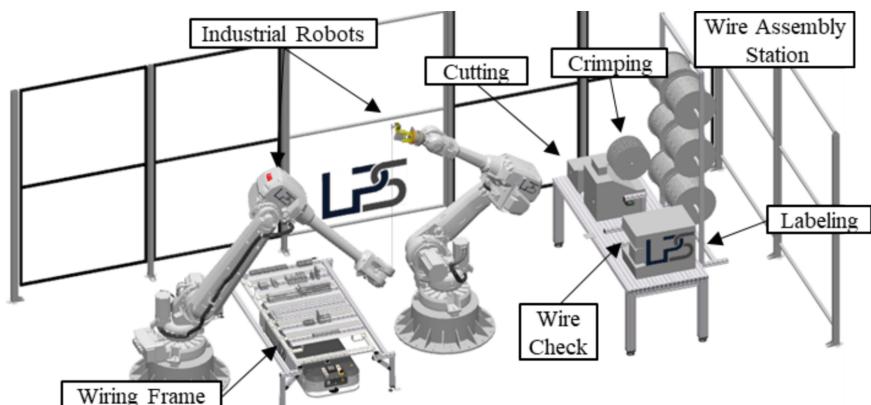


Fig. 4. Layout for automatic wire assembly

5 Conclusion

To meet the rising demand for electrolyzers, its production needs to be optimized and scaled up. For this purpose, this paper presented a concept for automated wiring of electrolyzer control cabinets, one of the most complex tasks in automation. Based on the analysis of a representative control cabinet, impediments regarding the automated wiring are identified and a redesign proposal for the control cabinet is presented. Finally, a concept for a robot cell was introduced, which combines automated wire assembly and wiring.

Redesigning the control cabinet promises great potential for automated wiring since collision risks are reduced by increasing the distance between the mounting rails. Additionally, the reachability of the terminal blocks is improved since a wiring frame is used, and no extra tool for wire connection is needed. The presented concept is suited to explore the potential of automated wiring further. Since both wire assembly and wiring are done using robots, the concept enables high flexibility regarding wire supply and wiring strategies. Future work will focus on testing end-effectors for wire assembly and wiring in the real test cell. Another focus will lie in path planning methods for the robots and the recognition of wire gripping positions using novel vision technology.

Acknowledgement. This work was supported by the Federal Ministry of Education and Research (BMBF) under the grant number 03HY113A within the research project H₂Giga –FertiRob.

References

1. Federal Ministry for Economic Affairs and Climate Action (2023) Fortschreibung der Nationalen Wasserstoffstrategie. <https://www.bmbf.de/>. Accessed 13 Oct 2023
2. Sood, S., Prakash, O., Boukerdja, M., et al.: Generic dynamical model of PEM electrolyser under intermittent sources. *Energies* **13**, 6556 (2020)
3. Smolinka, T., Wiebe, N., Sterchele, P., et al.: Study IndWEDe – brief overview: industrialisation of water electrolysis in Germany: opportunities and challenges for sustainable hydrogen for transport, electricity and heat, Berlin (2018)
4. Prior, J., Bartelt, M., Sinnemann, J., et al.: Investigation of the automation capability of electrolyzers production. *Procedia CIRP* **107**, 718–723 (2022)
5. Tempel, P., Eger, F., Lechler, A., et al.: Schaltschrankbau 4.0: Eine Studie über die Automatisierungs- und Digitalisierungspotentiale in der Fertigung von Schaltschränken und Schaltanlagen im klassischen Maschinen- und Anlagenbau, Stuttgart (2017)
6. Gebauer, D., Dirr, J., Reinhart, G.: Concept for robot-based cable assembly regarding industrial production. In: Schüppstuhl (Ed.) 2022 – Annals of Scientific Society for Assembly, Handling and Industrial Robotics, pp. 113–124 (2021)
7. Chang, P., Padir, T.: Sim2Real2Sim: bridging the gap between simulation and real-world in flexible object manipulation. In: Hu, R.-M. (ed.) The Fourth IEEE International Conference on Robotic Computing: IRC 2020: Proceedings, 9–11 November 2020, virtual conference. IEEE, Piscataway, NJ, pp. 56–62 (2020)
8. Zhao, C., Liu, Y., Lu, Z.: Wiring simulation of electric control cabinet based on industrial robot. In: Yang, H., Liu, H., Zou, J., et al. (eds.) Intelligent Robotics and Applications: 16th International Conference, ICIRA 2023, Hangzhou, China, July 5–7, 2023, Proceedings, Part VIII, 1st ed. 2023, vol. 14274, pp. 454–462. Springer Nature Singapore; Imprint Springer, Singapore (2023)

9. Hefner, F., Schmidbauer, S., Franke, J.: Pose error correction of a robot end-effector using a 3D visual sensor for control cabinet wiring. *Procedia CIRP* **93**, 1133–1138 (2020)
10. Sardelis, A., Zacharaki, N.-C., Arkouli, Z., et al.: 2-Stage vision system for robotic handling of flexible objects. *Procedia CIRP* **97**, 491–496 (2021)
11. Bartelt, S., Bartelt, M., Kuhlenkötter, B.: Method for creating a control cabinet model with realistic wires. In: Herberger, D., Hübner, M. (eds.) *Proceedings of the Conference on Production Systems and Logistics: CPSL 2023-2*, pp. 486–495 (2023)



Design and Fabrication of a Phantom Head for Robotic Neurosurgery Simulation

Federico Mariano^{1,2(✉)}, Carola Abello^{1,3}, Nabeel Kamal¹, Giovanni Berselli^{1,4}, Elena De Momi², Gianluca Piatelli³, and Leonardo S. Mattos¹

¹ Biomedical Robotics Lab, ADVR, Istituto Italiano di Tecnologia, Genoa, Italy
{federico.mariano,carola.abello,leonardo.demattos}@iit.it

² DEIB, Politecnico di Milano, Milan, Italy
elenam.demomi@polimi.it

³ Istituto Giannina Gaslini, Genoa, Italy
gianluca.piatelli@gaslini.org

⁴ DIME, Università di Genova, Genoa, Italy
giovanni.berselli@unige.it

Abstract. Phantoms play a crucial role in planning and simulating surgical procedures, evaluating new medical devices, and training purposes. Current simulators, such as human cadavers, animal models, and virtual reality systems, are quite expensive and not easily accessible. This research aims to design and develop an affordable 3D replica of a child's head for paediatric neurosurgery. The process began with a Magnetic Resonance Imaging (MRI) scan of a child's head. Afterward, a phantom was fabricated using nylon 3D printing and incorporating agar to replicate the skull and soft tissue. Moreover, Vaseline balls simulating tumors and a silicone skin were added to the phantom, enabling realistic MRI scans for surgical navigation. The functionality of a novel paediatric neurosurgery robot was then evaluated using the phantom as a realistic representation of a child's head. The cost-effectiveness of this approach makes it an ideal and reproducible method for surgical simulation and training objectives.

Keywords: MRI-visible phantom head · Agar · 3D printing · Neurosurgery

1 Introduction

Phantoms serve as invaluable tools in surgical practice, faithfully replicating human anatomy and enabling a realistic simulation of diverse clinical scenarios. In experimental and educational applications, they have shown immense value and have proven to be irreplaceable resources [1]. Our primary goal is to develop a realistic phantom that allows neurosurgeons to, safely and in a controlled manner, evaluate an innovative robot designed for paediatric neuroendoscopy [2].

F. Mariano and C. Abello—The authors contributed equally.

Among many phantom production techniques, gelatin has emerged as one of the most commonly used materials for soft tissue simulation due to its availability and visibility in Magnetic Resonance Imaging (MRI) scans [3]. However, gelatin presents drawbacks, including decomposition, the need for meticulous refrigeration, and limited reusability. This study aims to overcome these limitations by developing a new phantom head using agar, a natural gelling agent derived from red seaweed [4]. Agar enables the creation of a gel that accurately mimics the characteristics of human brain tissue, while eliminating concerns about decay [3,5]. This development opens up new opportunities for long-term use and reproducibility in various medical procedures. In our specific case, to simulate the presence of tumors, we incorporated Vaseline balls into an agar-based phantom. In addition, we added a silicone skin to the skull to make it visible under MRI.

2 Materials and Methods

2.1 Phantom Fabrication

The fabrication of the phantom involved a combination of materials carefully selected to replicate a range of anatomical structures accurately. Therefore, agar was employed to simulate the characteristics of internal tissues, while a nylon structure allowed the reproduction of the skull. The choice of these materials was guided by their ability to be visualized in MRI scans, their capacity to faithfully reproduce tissue-like consistency and their potential for reuse without degradation of their properties. In particular, the phantom head realisation was divided into several phases, each with a well-defined purpose:

1) From MRI images to 3D mold design: The initial phase in the phantom modeling process involved the acquisition of MRI images of a child's head. Once the images were acquired, they were imported into the 3D Slicer software. The software allowed us to delineate the external contours of the skull in planes parallel to the sagittal plane. The resulting splines were then imported into CAD software (Creo Parametric) to create the corresponding solid structure. Subsequently, a support structure was designed within the phantom head to house three Vaseline balls that simulate tumors, positioning them on typical surgical targets according to a neurosurgeon's indication (Fig. 1). Finally, holes were add to the skull to allow the introduction of the surgical endoscope. Afterward, the solid was divided into two parts in relation to the sagittal plane to simplify the 3D printing of the parts and the introduction of the balls within the skull;

2) 3D model printing: The skull was then printed in nylon using selective laser sintering technology. It is important to note that nylon is not visible in MRI scans; however, its presence can be deduced by analyzing the surrounding silicone skin and agar in the MRI images;

3) Targets realization: For target realization, we opted to use Vaseline as it has been demonstrated to be well visible in MRI images. Small Vaseline balls were created using rubber balloons and placed in the internal support structure;

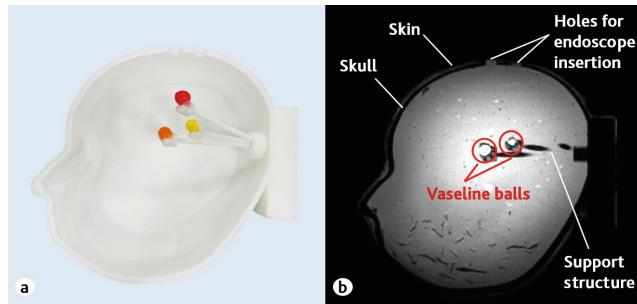


Fig. 1. (a) Nylon half mold with inner support structure to hold the Vaseline balls; (b) MRI of the phantom head illustrating the structures of interest.

4) Filling of the inner cavity: Before filling the cavity, the two half molds were joined and sealed with silicone glue. Subsequently, to mimic brain tissue, the agar gel was prepared stirring 750 ml of tap water and 38 g of agar powder. A teaspoon of flour was then added and the water-agar suspension was microwaved for 5 min, achieving a thick and gelatinous consistency [3]. It is crucial to swiftly insert the mixture into the phantom's cranial cavity due to rapid solidification;

5) Adding silicone: The last step in the process involved applying a thin layer of pink silicone (Pro-lastix 10 shore A, Prochima) to the outer surface of the phantom to add greater realism to the anatomical model, and to make the surface of the skull visible in MRI. This step is of critical importance to allow the use of the phantom head with neurosurgical navigation systems. Neuronavigation integrates medical imaging, such as MRI, with a real-time 3D localization of instruments to provide detailed surgical guidance. Without the skin, it is very difficult to perform the initial system registration that allows the use of such a system.

2.2 Agar Deterioration Experiment

Since agar conservation is a highly critical and at the same time fundamental aspect to ensure the stability and integrity of the material over time, we wanted to investigate this particular aspect. Therefore, experimental tests were carried out both under refrigerated conditions and at room temperature for one month to analyze material changes, such as weight reduction, colour changes, condensation or mould formation on the surface. To obtain a more detailed and in-depth representation of the agar's response to different environmental conditions, the samples, with an initial weight of 10g, were appropriately divided into subgroups, distinguishing those in closed containers from the ones open.

2.3 Phantom Assessment Procedure

The fabricated phantom head was tested using a new robotic system for neuroendoscopy developed at Istituto Italiano di Tecnologia [2]. This innovative robot

offers safe and precise support to position and insert the endoscope into the skull, reducing potential risks for the patient. The core of the robot's mechanism consists of a semi-circular track motion system and a translation mechanism, which collectively provide it with a total of 6 controlled degrees of freedom (DOFs). The experimental procedure for conducting tests with the robot involves:

- i) MRI scan:** Initially, an MRI scan of the phantom head is performed to provide a visual representation of cranial and internal structures (Fig. 1b);
- ii) Neuronavigation system:** To ensure precise guidance during subsequent steps, the registration with the neuronavigation system is performed;
- iii) Positioning and insertion of the endoscope:** Once the specific target is chosen, the robot, in conjunction with the neuronavigation system, achieves precise positioning and orientation of the endoscope, allowing its safe insertion towards the surgical target.

3 Results

The MRI scans confirmed the realism of the phantom, allowing a precise visualization of tumors, skin, and internal tissues. These MRI images aided surgeons in determining the optimal insertion point and the correct orientation of the endoscope for reaching each target tumor. The neuronavigation system enabled and confirmed the correct attainment of the Vaseline balls using the paediatric neurosurgery robot. This was verified also visually by observing the endoscope images. This experiment also confirmed that agar effectively simulate the consistency of internal tissues, providing tactile feedback that can enhance surgeon training and skill development.

Regarding the agar tests, daily observation showed that the degradation of the material is strongly influenced by exposure to air and temperature. Results are presented in Fig. 2. In particular, it was noted that the use of an open container leads to a significant reduction in volume due to the evaporation, change in colour, and the formation of mould on the surface within a few days. The effect was more evident in the case of storage at room temperature than in the fridge. Analysis of the data showed a significant reduction in weight for these samples; in particular, a daily reduction of 1 g for the sample in the fridge and 2 g for the one at room temperature. On the other hand, samples with the cap closed had little or no degradation, going from 10 g to 8 g after one month, both in the refrigerator and room temperature, maintaining a gelatinous consistency.

4 Discussion

Optimal preservation requires the isolation of the material from external agents, such as light, humidity and fluctuating temperature, to prevent any chemical or biological changes leading to the degradation of the material itself. The choice of storage environment therefore plays a crucial role in this context. In particular, long-term preservation is preferable in a refrigerated environment to ensure a constant temperature and reduce the risk of material deterioration due to

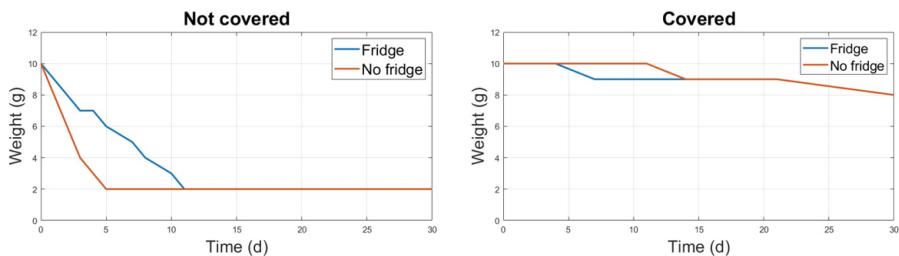


Fig. 2. The diagrams show the change in agar weight with the passage of days in the open (left) and closed (right) samples.

unwanted chemical reactions or the growth of microorganisms. On the other hand, storage at room temperature can only be a viable option if laboratory conditions allow it, i.e. a dry and cool place out of direct sunlight is required. Furthermore, the temperature must be constant, controlled and possibly below 25 °C.

5 Conclusion

The proposed phantom, although simple, allows surgeons effectively simulate neurosurgical procedures and test new surgical devices in a safe and controlled environment. This is possible thanks to MRI-visible properties of the phantom combined with the possibility of modelling it according to the specific features of the patient's anatomy, in a cost-effective and fast way. The phantom fabrication is estimated to cost around 720 euros (790 USD) with a manufacturing time of twelve hours. This is about 90% cheaper than other options available on the market [6]. The main cost regards the nylon structure. Reuse of the phantom may require the replacement of the Vaseline balls. This involves opening the phantom and replacing the agar, but this is a simple and very low cost procedure.

References

1. Müns, A., Meixensberger, J., Lindner, D.: Evaluation of a novel phantom-based neurosurgical training system. *Surg. Neurol. Int.* (2014)
2. Kamal, N.: Design and development of a robotic system for pediatric neurosurgery, Istituto Italiano di Tecnologia, unpublished
3. Earle, M., De Portu, G., DeVos, E.: Agar ultrasound phantoms for low-cost training without refrigeration. *Afr. J. Emerg. Med.* 18–23 (2016)
4. Pandya, Y.H., Bakshi, M., Sharma, A.: Agar-agar extraction, structural properties and applications: a review. *Pharma Innov. J.* 1151–1157 (2022)
5. Katiyar, V., et al.: A high-fidelity agar-based phantom for ultrasonography-guided brain biopsy simulation: a novel training prototype with visual feedback. *World Neurosurg.* 333–343 (2022)
6. Anthropomorphic Phantoms Human Like Phantoms - Universal Medical. <https://www.universalmedicalinc.com/all-products/diagnostic-imaging/imaging-quality-control/phantoms/anthropomorphic-phantoms.html>. Accessed December 2023



SoftGrip: Towards a Soft Robotic Platform for Automatized Mushroom Harvesting

Niccolò Pagliarani^{1,2(✉)}, Costas Tzafestas³, Evangelos Papadopoulos³, Petros Maragos³, Athanasios Mastrogeorgiou⁴, Antonis Porichis⁵, Helen Grogan⁶, Robin Ehrhardt⁷, and Matteo Cianchetti^{1,2}

¹ The BioRobotics Institute, Scuola Superiore Sant'Anna, Pontedera, Pisa, Italy
{n.pagliarani,m.cianchetti}@santannapisa.it

² Department of Excellence in Robotics and AI, Scuola Superiore Sant'Anna, Pisa, Italy

³ School of Electrical, and Computer Engineering, National Technical University of Athens, Athens, Greece

⁴ TWI-Hellas Center, Halandri, Greece

⁵ AI Innovation Centre, University of Essex, Essex, Colchester, UK

⁶ Horticulture Development Department, Teagasc Food Research Centre, Ashtown, Ireland

⁷ Mitsui Chemicals, Düsseldorf, Germany

Abstract. The fresh food industry significantly depends on manual labor, which can make up to 40% of total production costs. Until now, implementing safe robotic automation for gently harvesting fresh produce has been difficult due to the complex and delicate nature of these tasks. The EU-funded SoftGrip Project aims to revolutionize the fresh food sector with technological advancements. By integrating artificial intelligence (AI) and robotic automation, it is possible to achieve gentle harvesting, enhance productivity, and lower labor costs for small and medium-sized European mushroom farms. The innovative smart soft gripper, designed to learn skills from expert harvesters through imitation learning, seeks to provide an economically feasible, scalable, and environmentally friendly solution, transforming the mushroom cultivation industry and the wider fruit market.

Keywords: agricultural robotics · soft robotics · AI-enabled robotics

1 Motivation and Background

Robotic systems delicately handling items can be applied across various production sectors, offering substantial economic advantages. For example, deploying robots to manage pressure-sensitive products in the agri-food industry can lower labor expenses, enhance productivity, and improve working conditions [1]. In particular, the mushroom industry is particularly experiencing increasing pressure due to high labor costs, which can make up 40% of the total production

expenses [2]. The white button mushroom (*Agaricus bisporus*) is the fifth most widely cultivated mushroom globally, comprising 11–15% of worldwide production, amounting to 4.4–4.7 million tonnes between 2013 and 2018–19 [3]. In Europe, *A. bisporus* production in 2020 was 1.24 million tonnes of which approx 65% was harvested by hand for the fresh market and 35% for the processed market [4]. While mushrooms can be mechanically harvested and processed for canning and freezing, more advanced automation is needed for picking and processing mushrooms for the fresh market due to stringent quality standards. Harvesting fresh mushrooms is a demanding task that requires the dexterity, precision, and sensitivity of human hands to avoid damaging the mushrooms. The conditions for harvesting can be challenging, often involving work in confined spaces with high humidity, and these conditions can vary significantly from one country to another. Despite some recent innovations, finding and retaining labor for this demanding work has remained a significant challenge in Europe and globally in recent years. As a result, the horticulture sector is increasingly turning to automation and robotics to address labor shortages. The EU-funded SoftGrip project, with its innovative approach to mushroom harvesting, offers a reassuring solution to this pressing issue. Robotic harvesting systems for fresh mushrooms have been developed in the past; however, none have yet met the precise quality requirements of the market. Mushrooms have delicate structures that are easily damaged or bruised by external forces. Conventional gripper designs often struggle with this delicacy and face challenges due to the high variability in orientation and attachment strength. Bruising and discoloration of mushrooms can happen at multiple stages throughout the crop and supply chain. Conventional rigid end-effectors are not well-suited for handling delicate organic objects in dense environments, as they often damage both the target mushroom and those nearby. Moreover, rigid end-effectors need high-resolution position and force sensors and precise transmission systems to prevent harm to the fragile mushrooms [5]. Previous efforts to replace rigid end-effectors with robotic vacuum end-effectors [6] have shown only partial success. The primary issue is that the gripping forces applied by suction cups can still be too strong because of their limited contact area. Given the high variability in size, orientation, and cluster density, improving traditional suction cup designs is unlikely to prevent damage to mushrooms completely.

2 Objectives

To address the challenges associated with gently grasping delicate items, the EU-funded SoftGrip project suggests using soft robotic structures made from food-safe and recyclable elastomeric materials. The mechanical compliance of these structures is a crucial benefit, as they can passively adapt and mold around the object, thereby distributing contact forces more evenly and minimizing damage [5]. Finger-based soft grippers, which mimic the high dexterity of the human hand by allowing significant joint deformations and a broad range of motion, have proven effective for harvesting vegetables and fruits [5]. The

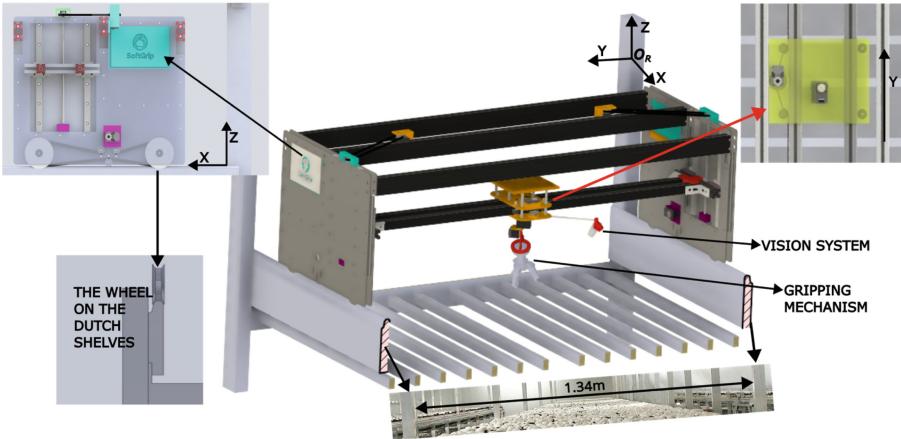


Fig. 1. Soft robotic platform installed at the level of cultivation shelves

SoftGrip project enhances these grippers with advanced modeling algorithms to improve real-time control and learning capabilities. Additionally, SoftGrip is developing a learning-by-demonstration framework that enables robots to learn mushroom-picking techniques from human workers, potentially applying these skills to similar tasks. This approach ensures safe and precise handling of delicate and high-value agri-food products.

3 Soft Robotic Platform Overview

3.1 Architecture

The overall architecture of the SoftGrip system is depicted in Fig. 1. The robot comprises two devices: (1) a Cartesian robot mounted over the shelf that moves in the x-y-z axis and (2) the soft gripper which is attached to the end-effector of the Cartesian robot and faces the mushroom cultivation. The Cartesian with the gripper will be installed over the cultivation and will be able to reach any position in the workspace. A central computer hosts the SoftGrip supervision module, which generates the sequence of grasping tasks and supervises their execution. The input to the supervision system is the estimation of mushroom size, position, and orientation. This estimation is generated by the vision module, which processes the information captured by low-cost environment cameras installed on the shelves. The supervisor generates a sequence of grasping tasks, which is fed to the grasp planner module, which in turn computes the trajectories of the robot and the grasping primitives of the soft-gripper. Then, the low-level closed-loop controllers generate the actuation commands, which are fed to the drivers of the robotic devices to execute the grasping primitives. The commands are adjusted based on feedback signals generated by the sensors, both proprioceptive and exteroceptive, embedded into the soft gripper and the encoders of the Cartesian robot actuators.

3.2 Grasping Control Strategy for Outrooting

The grasping strategy is outlined through the following sequence of steps:

1. Mushroom detection, identification, part segmentation (stem, cap), localization, and pose estimation of the cap. The information on position, orientation, and characteristic lengths of the cap of the mushroom is sent to the grasp planner.
2. The Cartesian robot positions itself above the mushroom target. The soft gripper assumes the angle of attack of the mushroom, i.e., its orientation is aligned with the orientation of the axis of the cap.
3. The fingers of the gripper are preshaped to fit the cap. The Cartesian robot moves in x-y-z until the mushroom cap is within the grasping space of the gripper. Closure of the preshaped fingers is actuated to grasp the mushroom cap.
4. The soft gripper is driven along a small curve in space to provide a combination of bending and twisting. The set of predefined movements may depend on the specific mushroom, indeed some adjustment may be required to adapt the current strategy to induce a final result that a portion of the roots is broken while the cap and the stem remain connected.
5. The soft gripper applies a small torsion on the cap and stem. This torsion results in the transfer of the tensile forces (generated in the previous step) to the rest of the roots and breaks them. Hence, the mushroom is outrooted.
6. The Cartesian robot executes a fast transfer of the mushroom in the allocated bin. The gripper releases the mushroom in the bin and returns to the workspace. The steps are repeated for the next mushrooms to be picked.

3.3 Skill Transfer Through Imitation Learning Framework

The complexity of mushroom picking, as demonstrated by the fact that it takes about 12 weeks for an adult human to master makes it impossible to pre-program grasping and force control strategies that can carry out the task reliably. This challenge is common in various other tasks involving the handling of delicate deformable objects. Thus, within SoftGrip, we aim to develop a learning-by-demonstration framework that will allow the robot to capture the mushroom-picking skill in a way that is extensible to other similar tasks. The control layer will be able to cope with the variations presented in the environment or even in the object's configuration, reinforcing the adaptability and improving the learning speed of our implementation. It will be based on the concept of probabilistic movement primitives, which constitutes a probabilistic framework that allows the exploitation of the properties of trajectory distributions for representing and learning movement primitives.

4 Conclusion and Perspectives

The Robotics 2020 Strategic Research Agenda by the European Commission underscores the strategic importance of Europe's robotics market. SoftGrip's

introduction of a soft robotic system to automate mushroom harvesting is a crucial step in enhancing Europe's AI and Robotics capabilities. This innovation represents a shift in the industrial use of soft robotics, positioning Europe as a leader in this promising field with wide-ranging industry applications. Focusing on agriculture, SoftGrip addresses critical challenges in this sector through a soft robotic platform that performs sensitive tasks with a learn-by-demonstration method. This approach will accelerate robotic adoption in mushroom picking and similar sectors, marking a new era in robotic automation in the agriculture and food industries.

Acknowledgement. The authors thank all the members of the SoftGrip consortium for their valuable contribution. This work was supported by the European Union's Horizon 2020 through the SoftGrip project (contract 101017054).

References

1. Sanchez, J., Corrales, J.-A., Bouzgarrou, B.-C., Mezouar, Y.: Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey. *Int. J. Robot. Res.* **37**(7), 688–716 (2018)
2. Teagasc. Horticulture input costs 2021 - impact assessment, 2021. <https://www.teagasc.ie/publications/2021/Horticulture-Input-Costs-2021---Impact-Assessment.php>
3. Singh, M., Kamal, S., Sharma, V.P.: Status and trends in world mushroom production-iii-world production of different mushroom species in 21st century (2021)
4. EMGG. European mushroom growers group - production figures, 2020. <http://www.infochampi.eu/wp-content/uploads/2023/02/Sans-titre.png>
5. Elfferich, J.F., Dodou, D., Della Santina, C.: Soft robotic grippers for crop handling or harvesting: a review. *IEEE Access* **10**, 75428–75443 (2022)
6. Rowley, J.H.: Developing flexible automation for mushroom harvesting (*agaricus bisporus*). University of Warwick, 2009



A Modular Architecture for IMU-Based Data Gloves

Alessandro Carfi^(✉), Mohamad Alameh, Valerio Belcamino, and Fulvio Mastrogiovanni

Department of Informatics, Bioengineering, Robotics, and Systems Engineering,
University of Genoa, Via Opera Pia 13, 16145 Genoa, Italy
alessandro.carfi@dibris.unige.it

Abstract. The flexibility and range of motion in human hands play a crucial role in human interaction with the environment and have been studied across different fields. Researchers explored various technological solutions for gathering information from the hands. These solutions include tracking hand motion through cameras or wearable sensors and using wearable sensors to measure the position and pressure of contact points. Data gloves can collect both types of information by utilizing inertial measurement units, flex sensors, magnetic trackers for motion tracking, and force resistors or touch sensors for contact measurement. Although there are commercially available data gloves, researchers often create custom data gloves to achieve the desired flexibility and control over the hardware. However, the existing literature lacks standardization and the reuse of previously designed data gloves. As a result, many gloves with unclear characteristics exist, which makes replication challenging and negatively impacts the reproducibility of studies. This work proposes a modular, open hardware and software architecture for creating customized data gloves based on IMU technology. We also provide an architecture implementation along with an experimental protocol to evaluate device performance.

Keywords: Data Glove · Hand Tracking · Inertial Measurement Unit

1 Introduction

Human hands' flexibility and range of motion are fundamental for how humans interact with each other and the world. Many research communities have studied human hand motion, focusing on the hand's motion and its interaction with the environment [1]. Hand studies typically include information about hand kinematics and sensory input. Hand kinematics are well-described, with minimal variation among individuals, except for bone proportions [3]. On the other hand, humans primarily rely on two senses when using their hands: proprioception and touch. Proprioception helps determine limb position, while touch provides information about forces and points of contact. Depending on the type of study,

researchers may require information from one or both of the hand senses. As a result, various technological solutions have been explored. In terms of proprioception, tracking the hand's motion has been achieved through the use of cameras or wearable sensors. Instead, for touch, although a few attempts have been made to estimate them using cameras [5], the position and pressure of contacts are primarily measured using wearable sensors. The data glove is a wearable device embedding sensors that can collect all the previously defined information [4]. The sensors embedded in a data glove can be adapted to meet the application requirements. Inertial Measurement Units (IMU) [7], Flex sensors [8], and magnetic trackers [2] are the most common choices for tracking motion, while force sensing resistors [10] or touch sensors can be used to measure contacts [9]. While various data gloves are commercially available [6], researchers often opt to construct custom data gloves to achieve the desired flexibility and control over the hardware. However, the existing literature lacks standardization and the re-use of previously designed devices. As a result, many gloves with unclear characteristics have been developed, making replication challenging and affecting the reproducibility of the studies for which they were created. This work aims to introduce a modular, open hardware and software architecture for creating customized data gloves based on IMU technology.

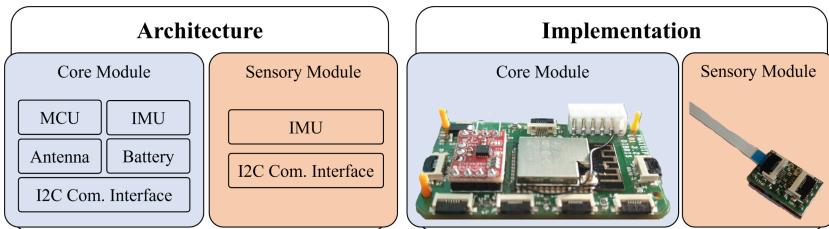


Fig. 1. The image compares the architecture on the left with the implementation on the right. The core module is in blue, while the sensory module is in red.

2 Hardware Architecture

To provide maximum flexibility, a data glove should support a variable IMUs number to adapt to the application's requirements, work without needing external equipment, and be easy to repair and reproduce. Our architecture addresses these requirements by defining two modules, displayed in Fig. 1: core and sensory modules. The *core module* includes the MCU for data collection and processing, an antenna to send data to a PC, interfaces for connecting other modules, a battery for power and an IMU. The *sensory module* contains an IMU and the communication interface with the core module. Each module has a structure that includes the basic functionalities of an IMU data glove. However, each component can be expanded to incorporate more functionalities while keeping the main

structure intact. The only technical constraint is the communication interface, which should remain fixed to ensure compatibility across different implementations of this architecture. The communication interface has two key components: the communication protocol and the physical medium. The two most commonly adopted communication protocols for digital sensors are Serial Peripheral Interface (SPI) and Inter Integrated Circuit (I2C). SPI is typically used for communication between components on the printed circuit board (PCB). It is more vulnerable to noise and requires more communication lanes. Instead, I2C allows the same physical lane sharing across multiple devices without additional selection lanes and is less affected by noise. Therefore, we have chosen I2C as the communication protocol for the modules in our hardware architecture. Instead, in the literature, solutions for the physical medium often involved soldering cables or using flexible PCBs. However, these options make it hard to modify the number of sensors and increase the complexity of reproducing and repairing the device. To solve these issues, we opted for using a Flexible Flat Cable (FFC) connector as the physical interface for the modules and FFC cables to connect them. This solution allows easy addition of extra sensory modules or replacement of faulty modules. Additionally, FFC cables are flexible and do not restrict hand motion.

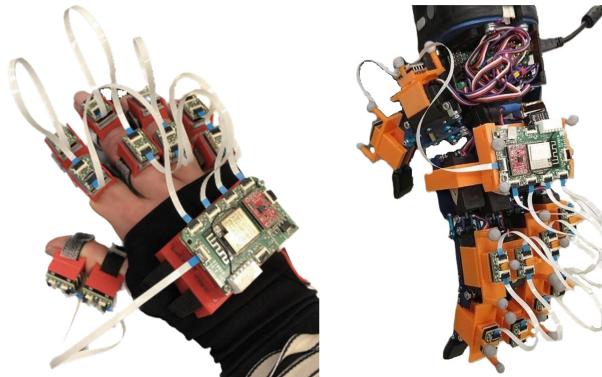


Fig. 2. On the left, we have the dataglove worn by a human, and on the right, it is mounted on the AR10 Robotic hand.

3 Implementation

The implementation of the two modules can be seen in Fig. 1 while the device is pictured in Fig. 2. The sensory module design includes a SparkFun IMU Breakout, which embeds the MPU-9250 from InvenSense, and a custom-designed shield to support FFC connection and daisy chaining. Although the I2C protocol only requires four lanes (ground, power, SCL, and SDA), we opted for a six-lane FFC connector and cable for future expansion. Each I2C lane can connect only two

sensors since the MPU-9250 only offers two selectable addresses. The two sensory modules of each I2C lane can be worn on the proximal and intermediate phalanges to monitor the motion of a single finger. Instead, the core module is placed on the back of the hand and consists of a custom PCB to ensure a compact design. This PCB includes an ESP32 with a WiFi antenna, 7 FFC connectors, an MPU-9250 IMU, and an I2C multiplexer. Each finger requires a separate I2C lane for motion tracking, so we allocated one FFC connector for each lane. For symmetry and compatibility with both right and left hands, we included two connectors in symmetric positions for the thumb. Since the ESP32 has only two I2C controllers, but the system requires 6 I2C lanes (one for each finger and one for the hand back), the design incorporates an I2C multiplexer to manage all the communication lanes. Finally, the core module also includes a connector for battery power. Our software consists of two modules. The first module runs on the ESP32 and collects data from connected sensors, sending them via UDP communication. The data includes the sensor's unique ID, accelerometer and gyroscope readings, and orientations in quaternions. The ESP32 uses the I2C multiplexer to collect data from each sensor module. The sensor orientation is estimated using a data fusion process run by the MPU-9250's digital motion processor. The second module runs on the PC and receives sensory data through UDP communication, acting as a driver.

4 Results

The purpose of the experimental setup is to demonstrate the general functionalities of the data glove. For maximum reproducibility and accuracy, we installed our device on an AR10 hand from Active8, mounted on the Baxter manipulator from Rethink Robotics, see Fig. 2. The first test assessed the autonomy and acquisition frequencies of the data glove under static conditions. Equipped with eleven IMUs and powered by a 220 mAh 3.7 V battery, the glove had an average autonomy of 62.89 min ($SD = 4.89$) and transmitted data with a frequency of 21.8 Hz ($SD = 9.47$) across six independent tests. In the same static conditions, we measured the drifting of the sensors' estimated orientations over time. The root mean square error (RMSE), averaged across all sensors, was 8.91° ($SD = 3.89$) after 30 min. We also conducted experiments involving random movements of the robot's hand and arm. Each experiment lasted 45 min and was repeated five times. The overall RMSE averaged across all sensors and trials was 9.17° ($STD = 9.30$).

5 Conclusions

This article presents a modular architecture for an IMU-based dataglove and its early implementation. The device, equipped with a small battery, can transmit data from eleven sensors at a frequency higher than 20 Hz for over an hour. Furthermore, tests conducted under unfavourable conditions, without proper calibration or drifting compensation, demonstrated a reasonably accurate tracking

of motions. The error in dynamic conditions is not significantly different from that in stationary conditions, as shown in the result sections. This result suggests that most tracking errors are due to sensor drifting, which can be compensated for with appropriate software solutions. The proposed device represents an initial attempt to provide an easily reproducible and modular platform for IMU-based hand tracking. Its extensibility offers opportunities for future research to propose new versions or develop more accurate tracking software solutions.

Acknowledgments. This work is supported by the CHIST-ERA (2014–2020) project InDex and received funding from the Italian Ministry of Education and Research (MIUR). This work has been also made with the Italian government support under the National Recovery and Resilience Plan (NRRP), Mission 4, Component 2 Investment 1.5, funded from the European Union NextGenerationEU.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Carfi, A., et al.: Hand-object interaction: from human demonstrations to robot manipulation. *Front. Robot. AI* **8**, 714023 (2021)
2. Cazacu, E., van der Grinten, C., Bax, J., Baeten, G., Holtkamp, F., Lee, C.: A position sensing glove to aid ankle-foot orthosis diagnosis and treatment. *Sensors* **21**(19), 6631 (2021)
3. Cobos, S., Ferre, M., Uran, M.S., Ortego, J., Pena, C.: Efficient human hand kinematics for manipulation tasks. In: 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2246–2251. IEEE, October 2008
4. Dipietro, L., Sabatini, A.M., Dario, P.: A survey of glove-based systems and their applications. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **38**(4), 461–482 (2008)
5. Grady, P., et al.: Pressurevision: estimating hand pressure from a single rgb image. In: European Conference on Computer Vision, pp. 328–345. Springer (2022)
6. He, K., Choosri, N.: Commercial data glove selection for vr-based hand rehabilitation gaming project. In: 2023 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON), pp. 177–182. IEEE (2023)
7. Huang, H., Liang, Z., Sun, F., Dong, M., et al.: Virtual interaction and manipulation control of a hexacopter through hand gesture recognition from a data glove. *Robotica* **40**(12), 4375–4387 (2022)
8. Luo, Y., et al.: Heterogeneous strain distribution based programmable gated microchannel for ultrasensitive and stable strain sensing. *Adv. Mater.* **2207141** (2022)
9. Maiolino, P., Mastrogiovanni, F., Cannata, G., et al.: Skinning a robot: design methodologies for large-scale robot skin. *IEEE Robot. Autom. Mag.* **23**(4), 150–159 (2016)
10. Wang, J., Li, B., Li, Z., Zubrycki, I., Granosik, G.: Grasping behavior of the human hand during tomato picking. *Comput. Electron. Agric.* **180**, 105901 (2021)



Mechanical Design of an Agile Quadruped Robot

Edoardo Del Bianco^{1,2,3(✉)}, Navvab Kashiri¹, Marco Roveri³,
and Nikos G. Tsagarakis²

¹ IAS, Leonardo Labs, Leonardo S.p.A. Genova, Genova, Italy

{edoardo.delbianco, navvab.kashiri}@leonardo.com

² HHCM, Istituto Italiano di Tecnologia, Genova, Italy

nikos.tsagarakis@iit.it

³ DISI, Università di Trento, Trento, Italy

marco.roveri@unitn.it

Abstract. Agile quadrupedal robotic platforms have become a commercial reality and are widespread in research laboratories. Nevertheless, the freedom to experiment with high performance on these commercial devices is impaired by restrictions to access and customise the low-level layers of their hardware and software. In this work, we present the design of Leon, an agile quadruped robot based on commercial components which we minimally customise. Following this approach, we benefit from the availability of partially integrated components on the market, but commercial restrictions do not impair the performance of the robot.

The robot cells are designed in a manner that only a few parts are required and can be fabricated without complex machining. This paper details the design and the strength analysis of the critical parts.

Keywords: Legged robot · Quadruped platform · Quasi-direct-drive

1 Introduction

Legged systems demonstrated their advantage in terms of mobility and versatility in complex missions [1–3]. Nevertheless, their state of development is far from wide-spread commercial exploitation. In 2008, the Boston Dynamics BigDog quadruped robot [4] proved capable of navigating over harsh terrains and overcoming obstacles while maintaining agility and carrying a payload. Such performance inspired the development of HyQ of IIT [5], ANYmal of ETH [6], Spot of Boston Dynamics and more. Mini Cheetah of MIT [7] exploited quasi-direct-drive actuators. Leveraging on this concept, the company Unitree commercialised AlienGO, Go1 and B1, providing an effective solution for higher-level developments on locomotion, navigation and manipulation. On the other hand, these commercial products blight the prospect of advancing the underlying layers including lower-level control and software architecture, or upgrading the mechatronics design due to restricted sources and limited commercial/legal practices.

Quasi-Direct-Drive actuators have been widely elaborated e.g. [8–10], and have accordingly become available on the market. In this work, we introduce the mechanical design of a quadruped robot, capable of agile motions, that relies majorly upon commercial components including the actuator. While this accelerates the development time, to avoid confining the overall performance to the specifics of the commercial unit used, we upgrade the necessary parts of the actuator. The robot design and the critical upgrades are presented in this paper.

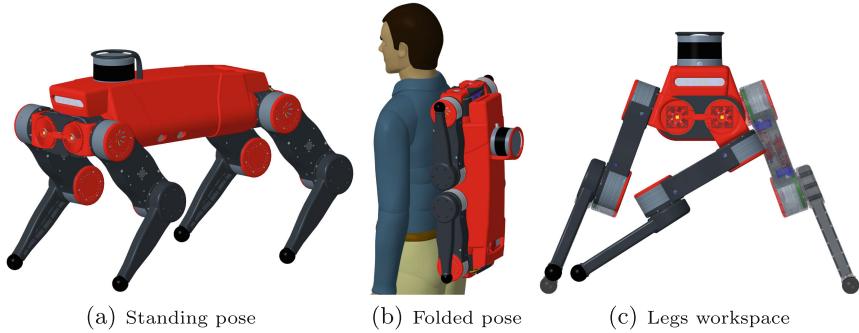


Fig. 1. The 3D views of the Leon robot.

2 Platform Overview

The envisioned use case of the robot is the exploration of unstructured environments, performing locomotion on uneven terrain and processing of sensory information from the field. To this end, we deploy a set of sensors including two RGB-D cameras, two tracking cameras, a lidar and a combined GPS and IMU device. We utilise three computers (allocated to the lower-level communications with EtherCAT motor controllers, middleware control and the processing of data from the sensors) in addition to the Ethernet and Wi-Fi communication routers, and the EtherCAT hub. Moreover, the robot embodies a customised power management board providing safety features when delivering power from a 48 V battery. This 720 Wh battery enables an untethered autonomy of 1.5 h approximately, considering similar systems [11]. We strove to keep the overall dimensions as compact as possible, so that the folded robot can be carried by an operator as a backpack, as demonstrated in Fig. 1b. Given a primary target of 50 cm for the leg length, the robot mass based on the main components noted above and rough approximation of body mass is about 30 Kg overall.

3 Commercial Actuator Selection and Enhancements

Given the primary dimension/mass estimation of the robot, we rely on the gravity compensation torque of joints in common poses, as discussed in [12], as the

measure for the continuous torque requirement of the actuators, while that in extreme poses e.g. legs fully extended, is to be managed by the peak torque capacity. It results in 18 Nm continuous torque requirement, and 36 Nm torque at extreme poses to be covered by peak torque capacity. For simplicity in development/maintenance, we exploit the same actuator for all the joints and select the T-motor AK10-9/V2.0-KV60 that meets our torque requirements with the rated and peak torques of 18 Nm and 48 Nm, respectively. It presents better power density than other commercial solutions available in the market, given 960 g mass and the output velocity of 226 RPM thanks to the low reduction ratio of 9:1. To employ the actuator in the robot, we implement a set of upgrades:

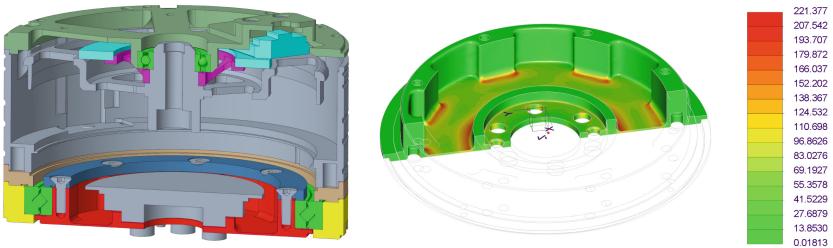


Fig. 2. On the left, the upgraded actuator (cross section): the default components are in grey and empty space, the bearings in light green, the position sensor in light blue, the encoder disk and head holders in purple and dark green, the output interface in red. On the right, FEM simulation of the output part (Von Mises stress in MPa).

- Output Bearing: the default output bearing of the actuator is replaced with a larger cross-roller bearing, i.e. slim type, sealed IKO CRBS 708 V UU, so that it can account for loads resulting from the robot mass. Specifically, we have considered radial and axial loads each equal to the robot mass force, as well as a momentum equivalent to the robot mass force on a lever with the length of the leg extended. This results in safety factors of 1.9 and 3.2 based on the dynamic and the static capacity of the bearing respectively. It is shown in the cross-section view of Fig. 2 with the corresponding implementation. While the correct implementation of the cross-roller bearing requires full clamping of both rings, the lack of information on the assembly tolerances, as a downside of utilizing an off-the-shelf actuator, can lead to over-constraining the moving parts connected to the centre of the output interface. To address this issue, we have propounded a design for the output interface rendering axial compliance while transmitting the torque. As the measurement of corresponding parts exhibits up to 0.4 mm distance/tolerance between the two mounting interfaces of the output part, the FEM analysis of this part is carried out when enforcing this constraint in addition to transmitting torque with a safety factor of 3, as shown in Fig. 2.

- Position Encoder: the actuator is equipped with two medium/low-resolution position encoders by default which are not supported by up-to-date motor divers, such as Synapticon Somanet Circulo 9 that we have used to meet industry-standard safety requirements. We have therefore integrated a multi-turn magnetic encoder with 20+18 bits of resolution, Bogen AKS16-MT. It communicates data via BiSS protocol with the motor driver and can hold the counts when is powered with a backup battery, thereby retaining the absolute output position without the need for a secondary encoder.

4 Leg Design

Leon's leg follows a roll-pitch-pitch arrangement [12] with each actuator located in-axis with its joint to allow for an extended workspace, as shown in Fig. 1. The hip-roll actuator is mounted on the quadruped pelvis. The hip-pitch and knee-pitch actuators, the corresponding electronics and drivers are located on the upper-leg link which, as shown in Fig. 3b, presents a symmetrical design, in such a way that the two half-cell structures are identical, and can be separately assembled and maintained. An L-shaped hip link connects the output of the hip roll to that of the hip pitch, and a lower leg is connected to the output of the knee pitch actuator. The FEM simulations of the aforesaid parts are presented in Fig. 3a and Fig. 3c respectively. In the former, one interface is fixed while the other is subject to the actuator peak torque and a force equal to the robot mass normal to the interface, and in the latter, the foot is fixed when the actuator interface is subject to the actuator peak torque. The upper parts of the leg are fabricated in 7075-T6 Aluminium. The lower leg is designed with a stringer and frame ribbing approach to minimise its weight and is 3D-printed in PA12/carbon fiber. A modified squash ball is incorporated as foot, in a similar fashion to that Mini Cheetah leg [9]. The upper-leg and lower-leg lengths are, in the final design, 215 mm and 264 mm.

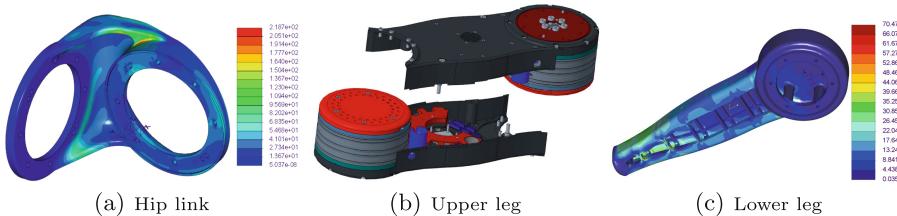


Fig. 3. Leg design and the FEM results of the critical parts (Von Mises stress in MPa).

5 Pelvis Design

The robot pelvis is shaped and sized in such a way that it incorporates all the components, allows for simultaneous inward folding of the legs, and maximises

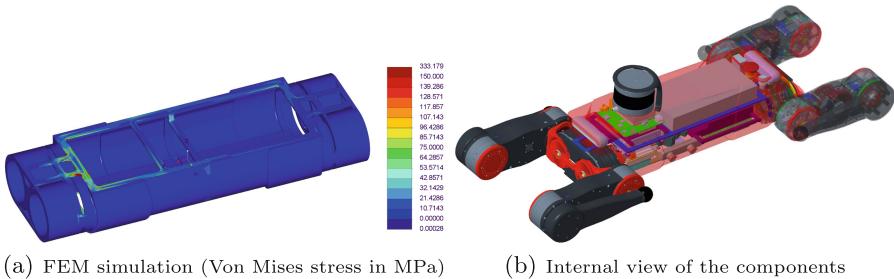


Fig. 4. Robot pelvis design.

the workspace of the legs. It is structured with an aluminium frame providing mounting to all the components as shown in Fig. 4b and, combined with the ABS support covers on the sides and bottom, it ensures structural rigidity. A FEM simulation of the structure is presented in Fig. 4a, where the robot is fixed/constrained at the two hip roll joints of one side, while a load equal to the robot weight is applied on one of its opposite joint, normal to the pelvis plane.

6 Conclusions and Future Work

This paper presents the mechanical design of the agile quadruped Leon, that exploits a commercial quasi-direct drive actuator with minimal necessary upgrades. The future work, which is currently ongoing, focuses on the assembly of the first prototype and the evaluation of the pros and cons of the proposed design.

References

1. Spenko, M., Buerger, S., Iagnemma, K.: The DARPA robotics challenge finals: humanoid robots to the rescue, Springer Tracts in Advanced Robotics, vol. 121 (2018)
2. Rouček, T., et al.: DARPA subterranean challenge: multi-robotic exploration of underground environments. In: Modelling and Simulation for Autonomous Systems, pp. 274–290. Springer, 2020
3. Kashiri, N., et al.: Centauro: a hybrid locomotion and high power resilient manipulation platform. IEEE Robot. Autom. Lett. **4**(2), 1595–1602 (2019)
4. Raibert, M., Blankespoor, K., Nelson, G., Playter, R.: BigDog, the Rough-terrain quadruped robot. IFAC Proc. Vol. **41**(2), 10822–10825 (2008)
5. Semini, C., Tsagarakis, N.G., Guglielmino, E., Focchi, M., Cannella, F., Caldwell, D.G.: Design of HyQ - a hydraulically and electrically actuated quadruped robot. Proc. Inst. Mech. Eng. Part I: J. Syst. Control Eng. **225**(6), 831–849 (2011)
6. Hutter, M., et al.: ANYmal - a highly mobile and dynamic quadrupedal robot. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 38–44. IEEE (2016)

7. Katz, B., Di Carlo, J., Kim, S.: Mini cheetah: a platform for pushing the limits of dynamic quadruped control. In: 2019 International Conference on Robotics and Automation, pp. 6295–6301. IEEE (2019)
8. Wensing, P.M., Wang, A., Seok, S., Otten, D., Lang, J., Kim, S.: Proprioceptive actuator design in the MIT cheetah: impact mitigation and high-bandwidth physical interaction for dynamic legged robots. *IEEE Trans. Rob.* **33**(3), 509–522 (2017)
9. Katz, B.: A low cost modular actuator for dynamic robots. PhD thesis, Massachusetts Institute of Technology, 2018
10. Singh, A., Kashiri, N., Tsagarakis, N.: Design of a quasi-direct-drive actuator for dynamic motions. In: Multidisciplinary Digital Publishing Institute Proceedings, vol. 64, p. 11 (2020)
11. Kashiri, N., et al.: An overview on principles for energy efficient robot locomotion. *Front. Robot. AI* **5**, 129 (2018)
12. Kashiri, N., Ajoudani, A., Caldwell, D.G., Tsagarakis, N.G.: Evaluation of hip kinematics influence on the performance of a quadrupedal robot leg. In: ICINCO, vol. 1, pp. 205–212 (2016)



Design of a Novel Tomato Harvesting Gripper

Boi Okken^{1,2(✉)}, Floris Kaper¹, Rick Muller¹, and Jan de Jong¹

¹ Precision Engineering, University of Twente, Enschede, The Netherlands

² Robotics and Mechatronics, University of Twente, Enschede, The Netherlands
{b.okken,j.j.dejong}@utwente.nl

Abstract. The robotic harvesting of tomatoes in a greenhouse setting presents many challenges in grasping, separation and manipulation of the produce. Presented is the design and development of a tomato harvesting tool, using a minimal amount of sensors and actuators to overcome the associated challenges.

Keywords: Gripper · Agri-food · Smart farming · Tomato harvesting

1 Introduction

The agri-food sector is currently experiencing its own renaissance, where the application of robotics takes the forefront. The FlexCRAFT project attempts to tackle many of the challenges in different use-case projects. One of these use-case projects is the robotic harvesting of tomatoes within a greenhouse.

Within the context of the manipulation of agricultural products, there are many unique requirements. These include, but are not limited to: the variation in physical produce dimensions and shape, complex plant structure, plant movement during interaction, occlusion (leading to positioning inaccuracies), and hygiene requirements (Fig. 1).



Fig. 1. The manufactured gripper manipulating produce in a greenhouse environment.

This research attempts to explore and address the unique issues presented in robotic tomato harvesting from a grasping perspective, through the use of mechanical guidance and combined grasping/cutting mechanism with minimal usage of sensors and actuators.

2 System Overview

2.1 Targeted Produce

Tomatoes grow clustered together as ‘trusses’, which are connected to the main stem of the vine-like tomato plant through side-vines called ‘peduncles’ [1].

By targeting the trusses, instead of the fruits directly [2], risk of produce damage [3] is reduced and fruit obstructions are avoided. Additionally, packaging complete trusses is desirable for aesthetic reasons.

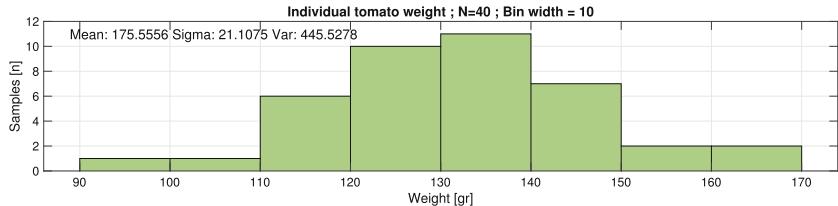


Fig. 2. Statistical measurements on individual tomato weights.

Of primary importance for a gripper is the mass of the targeted items. In order to better deduce the requirements of the gripper, the mass of $N = 40$ tomatoes have been measured (Fig. 2) using an electronic scale. The exact amount of tomatoes on a truss varied between 4 and 7. This provides a guideline for the mass that the gripper has to be able to cope with during harvesting.

2.2 Harvesting Procedure

To be able to deduce the required functions of the gripper, it is important to distinguish the different phases during harvesting.

1. Detection and scanning phase - Determine the exact locations and orientations of trusses, and select a target truss.
2. Approach - The robot arm moves towards the targeted truss.
3. Grasp - The truss is grasped by the robotic arm.
4. Cutting - The truss is cut as close as possible¹ to the the main stem.
5. Transport - The truss is carried from the plant, to the target container.
6. Release - The item is dropped in the container awaiting further processing.

¹ This reduces the chance of infection, which could damage the health of the host plant.

2.3 Subsystems

The manipulator has to perform three main functions in order to successfully perform the aforementioned harvesting tasks.

- Guidance - To successfully guide the robot towards the targeted peduncle.
- Grasping - To hold the truss during cutting and subsequent manipulation.
- Cutting - To separate the truss from the tomato plant.

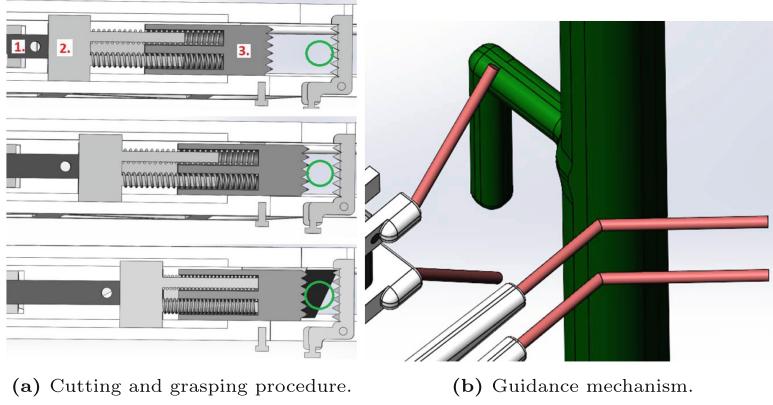


Fig. 3. Highlights of the structural design.

3 Structure Design

3.1 Grasping and Cutting Mechanism

In order to reduce the overall size and number of actuators, the grasping and cutting functionality can be integrated into a single mechanism, similar to Jia et al. [4]. A cutaway view showing the mechanical working is shown in Fig. 3a.

The mechanism works through two linear carriages. The first carriage (marked as 2 in Fig. 3a, from here on referred to as 'A') is driven by a linear actuator. Directly attached to A is a blade. Carriage B features an integrated jaw element, which is opposed by a jaw connected to the frame through a rotational joint (forming a gate). Each jaw has a spiked surface pattern to bite into the peduncle to help with retention. The carriages are connected together through a set of compression springs.

During grasping, the peduncle will pass through the gate, after which it is effectively locked in the gripper. Next by applying actuator force to A, both carriages move in unison closing the jaw. After the jaw contacts the peduncle, any additional actuator movement will result in spring compression. This in effect

increases the applied grasping force as described in Eq. 1, as well as moving the blade forward, facilitating cutting functionality.

$$F_{grip} = K_{spring}(\Delta x_{spring}) = K_{spring}(x_{rest} - (x_{car,3} - x_{car,2})) \quad (1)$$

Maximum spring compression thus implies full blade extension and maximum clamping force. This in effect determines the stroke of the blade through relative movement of the carriages. Assuming a spring resting position of x_{rest} , the spring rate can be determined with help of Hooke's law².

3.2 Guidance Mechanism

Try-outs in the greenhouse show that positioning accuracy of the robot is relatively poor due to tolerance stacking of various errors in the system. In order to increase robustness and compensate for this, we propose the implementation of mechanical guidance as shown in Fig. 3b. The mechanical guidance consists of 'prongs' which help guide the plant into the correct position. One pair of prongs guides the main stem, whilst the other set guides the peduncle into the jaws. Initially these were constructed of metal wire, later iterations feature a more rounded profile, constructed out of polymer.

4 Results

Figure 4 shows the developed (2nd generation) gripper in a payload testing scenario. It illustrates the ability to successfully manipulate twice the nominal payload.



Fig. 4. The second generation gripper successfully manipulating a total of 11 tomatoes.

² The constructed prototype features a stroke (Δx_{spring}) of 12[mm] combined with a spring rate of 2.5[N/mm], resulting in a clamping force 30[N] and cutting force of >40[N].

Experiments on the effect of both blade angle and peduncle diameter on cutting force are shown in Fig. 5a and Fig. 5b respectively.

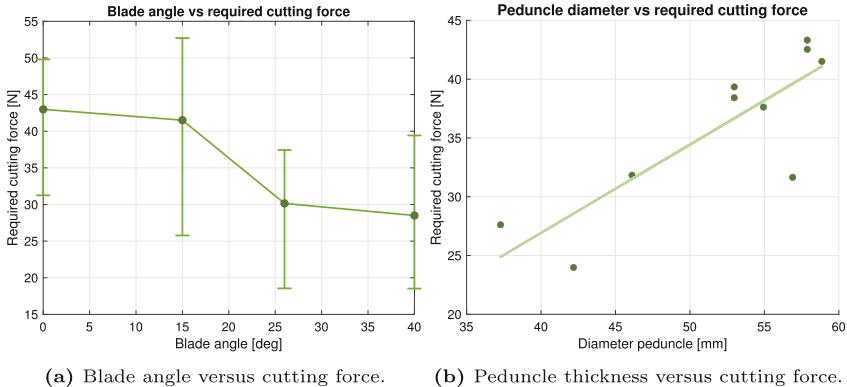


Fig. 5. Experimental results on parameters influencing required cutting force.

5 Discussion

Testing highlighted various qualitative results. Firstly, although guidance helps in compensating for positioning inaccuracies, over-constraint in complete system loop, can cause jamming and excessive friction. Control methods or mechanical changes could help solve this. Additionally, deposits and blade dulling significantly increase the required cutting force. This results in bending of the blade and peduncles which are not fully separated. Lastly, the current mechanism is physically quite long due to the linear actuator used, which can cause interference during the approach phase.

6 Conclusions and Future Work

Presented is the design and construction of a tomato truss harvesting tool. It features a single actuator combined with passive mechanisms to enable cutting, grasping and holding functionality required for produce manipulation. Additionally, it includes mechanical guidance to aid in increasing the success-rate during initial approach of the truss. Demonstrate the ability to effectively locate, grasp, cut and transport the trusses. Some practical issues with the guidance and cutting have been uncovered, which can be addressed in future iterations.

Acknowledgments and Declaration of Conflict of Interest. This work has been conducted under the FlexCRAFT project, funded by NWO/TTW (Under the grant “P17-01 Flexcraft - P4 Gripping and manipulation”). There are no known conflicts of interest in the production of this work.

References

1. Senden, J., et al.: Multi-hypothesis tracking in a graph-based world model for knowledge-driven active perception. *IEEE Robot. Autom. Lett.* **8**(9), 5934–5941 (2023)
2. Jun, J., Kim, J., Seol, J., Kim, J., Son, H.I.: Towards an efficient tomato harvesting robot: 3d perception, manipulation, and end-effector. *IEEE Access* **9**, 17631–17640 (2021)
3. Zhang, B., et al.: Comparative study of mechanical damage caused by a two-finger tomato gripper with different robotic grasping patterns for harvesting robots. *Biosyst. Eng.* (2018)
4. Jia, B., Zhu, A., Yang, S.X., Mittal, G.S.: Integrated gripper and cutter in a mobile robotic system for harvesting greenhouse products. In: 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 1778–1783 (2009)



Preliminary Performance Assessment of Hardware Architecture for Affordable Haptics

Francesco Rocchi^{1,2(✉)} and Alberto Parmiggiani¹

¹ Manufacturing and Design Facility, Center for Robotics and Intelligent Systems (CRIS), Istituto Italiano di Tecnologia, via S. Quirico 19/D, 16163 Genova, Italy
{francesco.rocchi,alberto.parmiggiani}@iit.it

² Department of Mechanical, Energy, Management and Transportation Engineering (DIME), University of Genova, Genova, Italy

Abstract. Most commercially available haptic devices come with high prices, often exceeding €10,000. The recent advent of the design paradigm employing BLDC motors with low ratio transmission and high-frequency current control for quasi-direct drive actuation has seen numerous implementations in robots requiring torque control. Building on this concept, this paper reports on preliminary tests assessing the feasibility of utilizing *mjbots* hardware for creating affordable haptic devices, with a focus on affordability (total cost below €200), ease of use, and customization.

Keywords: affordable haptics hardware · benchmark performance tests

1 Introduction

Haptic devices have important applications in various fields, including medicine, rehabilitation, gaming, human-computer interaction, teleoperation, augmented reality, and virtual reality [1–3]. However, their widespread adoption is still limited by their cost, with most commercially available devices exceeding 10,000€. Despite this limitation, there have been several attempts over the years to develop affordable hardware, primarily focusing on educational applications [4, 5], such as the Hapkit [6], and the ETHZ haptic paddle [7]. A parallel design paradigm that has emerged in recent years, which might also be relevant for haptics, relies on the use of BLDC motors (generally outrunners) coupled with low-ratio transmission and motor drivers capable of controlling motor torque through a high-frequency current control loop. Recent implementations include

This work was developed within the RAISE (Robotics and AI for socio-economic empowerment) Next-Generation EU project.

the Stanford Doggo quadruped [8], the Upkie balancing robot¹. This work reports preliminary tests aimed at assessing the feasibility of using hardware by the *mjbots* company as a possible solution for creating affordable haptic devices. The focus of the work was on affordability (the total cost of the setup was below 200€), ease of use, and ease of customization.

2 Methodology

2.1 System Architecture

The three essential components of the architecture are the actuator, its controller, and a computer for higher-level control functions. The motor controller implements the high-frequency torque control loop while carrying out the communication with the central node. The computer serves as the communication hub with the motors, executing higher-level computation and diagnostics. For this specific project implementation, we selected the MJ5208 brushless motor², the moteus r4.11 controller³, and the Raspberry Pi 4 Model B single-board computer (SBC)⁴. The moteus r4.11 controller incorporates torque feedback, features position control, and is programmable. This architecture can be controlled with a Python script running on the Raspberry Pi SBC, making it very convenient for prototyping and experimentation. The system architecture is represented in Fig. 1.

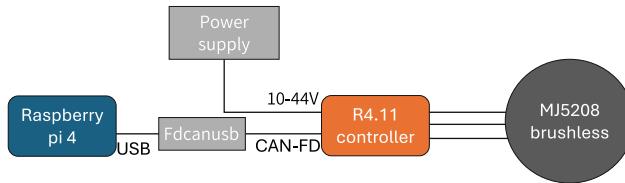


Fig. 1. Block diagram. The figure represents a block diagram of the main subsystems of the experimental setup.

2.2 Raspberry Pi Cycle Frequency Test

The Raspberry Pi SBC lies at the kernel of the architecture and is crucial for precise motor control and haptic feedback calculations. Initial tests were carried out to assess its computational capacity and communication with motor drivers. All programs were written in Python (avoiding advanced approaches relying

¹ <https://github.com/upkie/upkie>.

² <https://mjbots.com/products/mj5208>.

³ <https://mjbots.com/products/moteus-r4-11>.

⁴ <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>.

on C++) to maintain simplicity and ease of programming. The baseline control loop frequency was tested with a “minimal” controller sending simple commands through USB/CAN interface. The test, repeated 1000 times, estimated cycle frequency by recording total execution time. Subsequent tests introduced matrix inversion for simulating the computational load for haptic feedback, with 10×10 , 50×50 , and 100×100 matrices. Tests were repeated 100 times in two conditions: with X (the Unix graphical environment) running and with X deactivated. Results are shown in Fig. 2. As can be seen, the achievable cycle frequencies when the X environment is deactivated are higher, albeit only slightly. This suggests it is better to run the controller without X , although activating it for debug purposes should not cause a relevant impact on the overall system performance. As expected, the more computationally intensive the task becomes, the more time is required to complete each cycle, the lower the cycle frequency. Interestingly, even in the most demanding conditions (although running the test without X), cycle frequencies up to 200 [Hz] can be maintained.

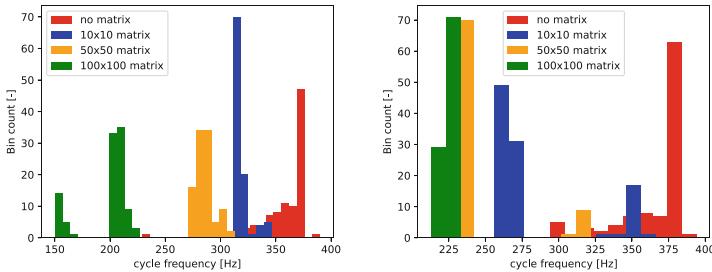


Fig. 2. Raspberry Pi cycle frequency. The figure show the comparison of frequency distributions with X (left) and without (right), among the baseline (red), the 10×10 matrix (blue), the 50×50 matrix (orange), and the 100×100 matrix (green) conditions.

2.3 Test Setup Description

Testing of the BLDC motor/driver combination required the development of a dedicated setup, represented in Fig. 3. The system relied on a cable routed through a set of pulleys to apply a torque at the motor axis. The resisting torque was generated by applying weights at the end of the cable. The torque was measured with a PCE-DFG N 500 dynamometer⁵ connected to the intermediate pulley, with a force resolution of 0.1[N].

⁵ https://www.pce-instruments.com/eu/measuring-instruments/test-meters/force-gage-pce-instruments-force-gage-pce-dfg-n-500-det_5972230.htm.

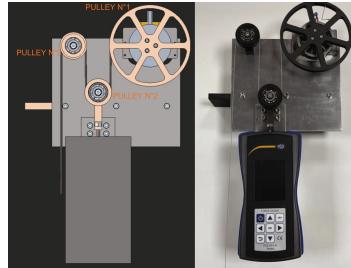


Fig. 3. The figure represents a CAD diagram and a photograph of the torque test setup.

2.4 Starting Torque Test

This test aimed to investigate the amount of torque needed to initiate motion, hence to provide an indication of the minimum backdriving force for the resulting haptic device. The motor position was first set to a random position; starting from this configuration the torque applied by the controller was gradually increased, with 0.01[Nm] until the motor was set in motion. The test was repeated 480 times. The results, plotted in Fig. 4, indicate that in most cases a value of 0.03 [Nm] was sufficient to obtain the movement of the motor, whereas a value of 0.04 [Nm] always set the motor in motion.

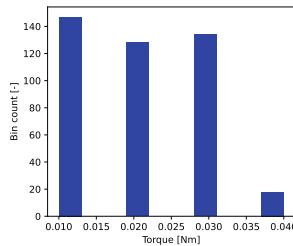


Fig. 4. Starting torque test. The figure show the histogram of the starting torque values for the test described in the main text.

2.5 Torque Precision Test

This test was designed to analyze the precision with which the BLDC motor/driver combination applied the desired torques. Calibrated weights were attached to the free end of the cable and the controller was instructed to apply the torque to balance the weight, bringing the system into a state of equilibrium. The force sensor was then used to measure the motor's output torque and thus to evaluate the difference between the requested torque and the one delivered. The results of this test are shown in the table below.

| | | | | | | | | | | | | |
|-----------------|------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| applied weight | [kg] | 0.05 | 0.10 | 0.15 | 0.20 | 0.25 | 0.30 | 0.35 | 0.40 | 0.45 | 0.50 | 0.55 |
| applied torque | [Nm] | 0.029 | 0.058 | 0.087 | 0.115 | 0.144 | 0.173 | 0.202 | 0.231 | 0.260 | 0.288 | 0.317 |
| measured torque | [Nm] | 0.024 | 0.053 | 0.079 | 0.109 | 0.138 | 0.165 | 0.197 | 0.226 | 0.256 | 0.285 | 0.312 |
| torque error | [Nm] | -0.005 | -0.005 | -0.007 | -0.007 | -0.006 | -0.008 | -0.005 | -0.004 | -0.004 | -0.003 | -0.006 |

The table highlights a slight error in the applied torque, with a maximum deviation of 0.008[Nm]. In addition to the results presented in Sect. 2.5, these data underline that the precision of the controller is little influenced by the presence or absence of a load connected to the motor.

2.6 Cogging Torque Test

The last test aimed at assessing the level of torque cogging that could be obtained with the basic BLDC motor/driver combination, and without the use of sophisticated compensation strategies.

As in the previous test, calibrated weights were attached to the free end of the cable to provide resisting torques, and the controller was instructed to set the motor in motion at a constant angular velocity; torque fluctuations were measured with the force sensor.

The experimental protocol involved the application of four different speeds, and three different weights. The results are represented in Fig. 5. In almost all cases the cogging torque remained low, oscillating between ± 0.02 [Nm] the nominal value.

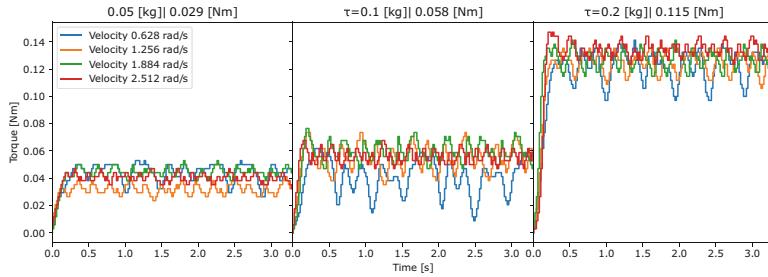


Fig. 5. Charts of torque at constant velocity. The figure show three comparisons of four velocities required to lift identical weights.

3 Conclusions

The BLDC motor/driver combination proved to be effective in controlling low torques at low angular velocities. Furthermore, it exhibited minimal levels of cogging torque. Tests demonstrated that the high-level SBC computing unit could implement a control loop with a frequency of up to 300 [Hz]. Therefore, the hardware combination tested in this work appears promising for the development of cost-effective haptic devices.

References

1. Laycock, S.D., Day, A.M.: Recent developments and applications of haptic devices. In: Computer Graphics Forum, vol. 22, pp. 117–132. Wiley Online Library (2003)
2. Giri, G.S., Maddahi, Y., Zareinia, K.: An application-based review of haptics technology. *Robotics* **10**(1), 29 (2021)
3. Hannaford, B., Okamura, A.M.: Haptics, pp. 1063–1084. Springer International Publishing, Cham, 2016
4. Gorlewicz, J.L., Webster, R.J.: A formal assessment of the haptic paddle laboratories in teaching system dynamics. In: 2012 ASEE Annual Conference & Exposition, pp. 25–49, 2012
5. Okamura, A.M., Richard, C., Cutkosky, M.R.: Feeling is believing: using a force-feedback joystick to teach dynamic systems. *J. Eng. Educ.* **91**(3), 345–349 (2002)
6. Martinez, M.O., Nunez, C.M., Liao, T., Morimoto, T.K., Okamura, A.M.: Evolution and analysis of hapkit: an open-source haptic device for educational applications. *IEEE Trans. Haptics* **13**(2), 354–367 (2019)
7. Gassert, R., et al.: Physical student–robot interaction with the ethz haptic paddle. *IEEE Trans. Educ.* **56**(1), 9–17 (2012)
8. Kau, N., Schultz, A., Ferrante, N., Slade, P.: Stanford doggo: an open-source, quasi-direct-drive quadruped. In: Proceedings of the ICRA, pp. 6309–6315. IEEE, 2019



Variable Stiffness Mechanism Using a Cam Profile

Floris van Ruitenbeek, Boi Okken, and Wesley Roozing^(✉)

Robotics and Mechatronics, University of Twente, Enschede, The Netherlands
{b.okken,w.roozing}@utwente.nl

Abstract. We present a variable stiffness mechanism based on a cam profile which nonlinearly deflects a linear spring. An iterative, feedback-based method for generating the cam profile following a desired force-deflection curve is proposed. Experimental results demonstrate strong correlation between desired and measured force-deflection curves.

Keywords: variable stiffness · compliance · VSA · cam profile

1 Introduction

Variable stiffness mechanisms are important elements for robots that need to robustly interact with a wide array of environments and objects. Many concepts for variable stiffness mechanisms have been proposed [1]. Importantly, variable stiffness mechanisms based on the agonist-antagonist approach require nonlinear elastic elements to vary mechanism stiffness. Proposed mechanisms include basic cam-follower [2], variable-radius pulleys [3,4], and rolling flexures [5]. Other approaches include material- or magnetism-based approaches [6–8].

In this paper we propose a flexible approach to generating variable-radius pulleys that combined with a linear spring produce a desired force–deflection profile. The end goal is to develop compact nonlinear stiffness elements for integration in a recently proposed variable-stiffness gripper that relies on the agonist-antagonist approach [9]. We demonstrate the approach on different desired force–deflection profiles, both in simulation and with experimental data.

2 Concept and Analysis

The basic operating principle of the mechanism, shown in Fig. 1a, is that of deflecting a linear stiffness element through a nonlinear transmission, in this case a variable-radius pulley. In Fig. 1a, (x, F) denotes the tendon deflection and tensile force, (θ, τ) the pulley rotation and torque, $R(\theta)$ the pulley radius at deflection angle θ , and k_{rot} the torsion spring constant. The angle θ is measured as the rotation of the pulley w.r.t. the x-axis. Furthermore, we assume the tendon detaches vertically from the pulley, on the x-axis¹. Therefore, the tendon tension is given by

$$F(\theta) = R(\theta) \tau, \quad (1)$$

¹ Note that this is an approximation that holds only for limited nonlinearity, as with increasing nonlinearity the detachment point will move off the x-axis.

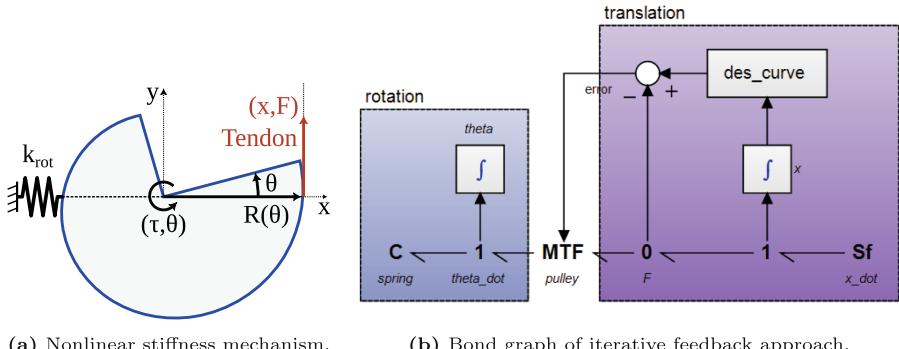


Fig. 1. Mechanism concept and pulley profile design approach.

which given the linear rotational spring with $\tau = k_{\text{rot}} \theta$ leads to

$$F(\theta) = R(\theta) k_{\text{rot}} \theta. \quad (2)$$

With the pulley acting as a nonlinear transformer between \dot{x} and $\dot{\theta}$ as $\dot{x} = R(\theta) \dot{\theta}$, shaping the angle–radius relationship $R(\theta)$ implies directly shaping the tendon stiffness–deflection profile, and therefore indirectly the force–deflection profile. However, the tendon *displacement* and the pulley rotation are related through a line integral, which is impossible to invert uniquely for a given deflection. Therefore, in order to obtain an expression for $R(\theta)$ that produces the desired force–deflection profile, we propose an iterative feedback method.

2.1 Variable-Radius Pulley Profile Synthesis Through Iterative Feedback

Figure 1b shows a bond graph of the proposed approach, in which the linear elastic element (C-element) is (slowly) deflected by the imposed tendon motion (S_f), through a modulated transformer (MTF) representing the variable-radius pulley. Given the current tendon deflection x , the error between desired and measured force forms a feedback loop into the modulated transformer. This signal adjusts the pulley radius, minimising the error:

$$R = \gamma [F^*(x) - F(x)] + R_0, \quad (3)$$

where F^* denotes the desired tendon force, $\gamma > 0$ denotes the feedback gain, and R_0 denotes the initial radius for $\theta = x = 0$, which is given by the initially desired stiffness $k * (0)$:

$$k(0) = \left. \frac{dF}{dx} \right|_{x=0} = k_{\text{rot}} R_0^{-2} \rightarrow R_0 = \sqrt{\frac{k_{\text{rot}}}{k^*(0)}} \quad (4)$$

The resulting angle–radius profile $R(\theta)$ is recovered from the simulation as the generated pulley profile design. This method is flexible as it does not

rely on a predefined pulley profile function, allowing to fit to a wide range of desired force–deflection profiles. For the desired force profile we use a polynomial $F^*(x) = ax^3 + bx$, which leads to the stiffness $k^*(x) = 3ax^2 + b$, i.e., a quadratic stiffness curve with nonzero initial stiffness (which, again, implies non-zero initial pulley radius R_0). Importantly, this choice yields $\frac{dk^*(x)}{dx} \Big|_{x=0} = 0$, which ensures $\frac{dR(x)}{dx} \Big|_{x=0} = 0$, i.e. the pulley surface is tangent to the tendon at $x = 0$.

3 Experimental Validation

We experimentally validate the proposed approach with a simple test setup, shown in Fig. 2. We utilise a Dynamixel MX-64AR servo with a circular pulley to displace the tendon. A TAL220 load-cell with HX711 amplifier measures the tendon tensile force through the servo base. Both connect to a Arduino Uno with Dynamixel Shield. We utilise Spiderwire Dura 4 0.25 mm as tendons. A large circular pulley with extension spring is placed on a shared axle with the variable-radius pulley (Fig. 2). Due to the difficult installation of torsion springs and limited availability, an equivalent extension spring is used of 0.23 N/mm, which in combination with the circular pulley is equivalent to a torsional spring constant of 0.15 Nm/rad.

We generate four pulley profiles (Fig. 3), with decreasing exponential factor $a \in \{0.20, 0.12, 0.04, 0\}$ N/cm³ and constant $b = 2.5$ N/cm, to produce force–deflection profiles ranging from high nonlinearity to the final one being linear. The latter linear spring profile allows to verify the system, without influence of several of the approximations taken. The array of profiles should show an incremental decrease in error between profiles originating from drift of the detachment

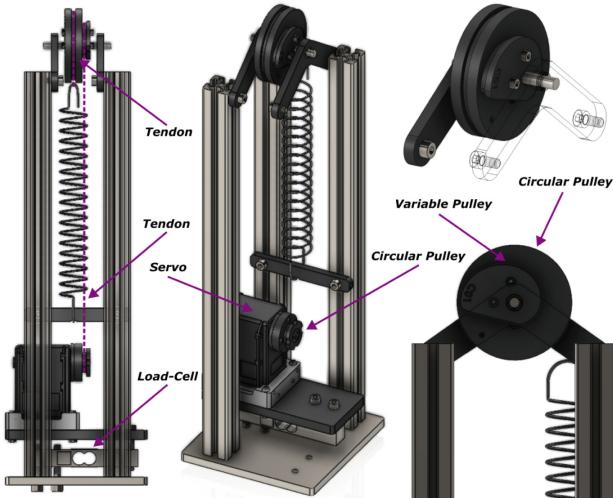


Fig. 2. 3D CAD model of testing setup.

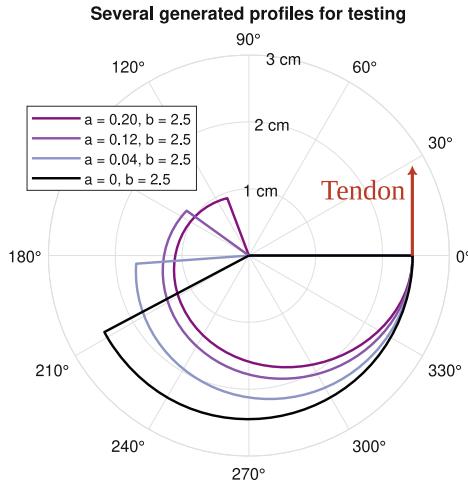


Fig. 3. Pulley profiles generated for experimental validation.

point (Sect. 2). Initial non-linear behaviour of the physical extension spring is mitigated by preloading it at approximately 5 N.

Figure 4 shows the resulting experimental results. Although the simulated and measured results are in good agreement for the cases with limited nonlinearity, for the two high nonlinearity cases errors become significant.

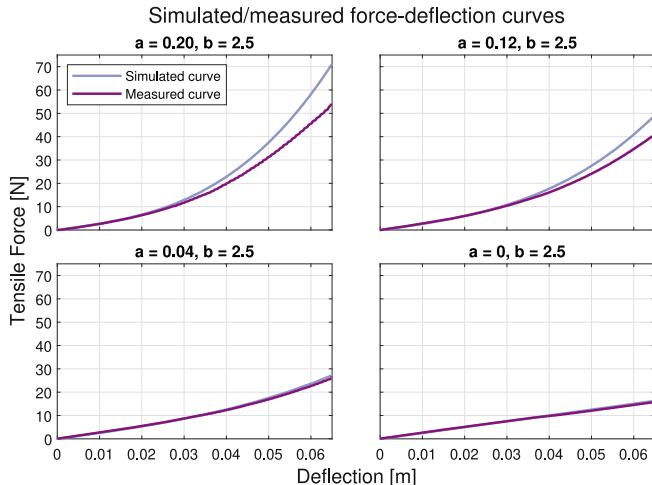


Fig. 4. Simulated and measured force-deflection curves

4 Discussion and Conclusions

In this paper, we propose a nonlinear stiffness mechanism using a variable-radius pulley and a linear spring. We propose a straightforward method to generate pulley profiles that provide desired non-linear force–deflection profiles. Due to its numerical iterative nature without parameterised functions, realising a larger range of different desired curves is possible. Experimental data demonstrates the method is able to generate nonlinear force–deflection profiles. However, for large nonlinearity, deviations arise that are mainly attributed to the shift in the tendon detachment point, which was approximated. Furthermore, we observed practical non-idealities such as material creep and knot tightening. Accurately accounting for the detachment point of the tendon from the pulley profile would likely improve accuracy. The low-friction and compact nature of the mechanism, together with mechanical simplicity, make the concept a promising choice for integration into variable-stiffness mechanisms that rely on the agonist/antagonist principle.

References

1. Ham, R., Sugar, T., Vanderborght, B., Hollander, K., Lefeber, D.: Compliant actuator designs. *IEEE Robot. Autom. Mag.* **3**(16), 81–94 (2009)
2. Migliore, S.A., Brown, E.A., DeWeerth, S.P.: Biologically inspired joint stiffness control. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation, pp. 4508–4513, IEEE (2005)
3. Schmit, N., Okada, M.: Synthesis of a non-circular cable spool to realize a nonlinear rotational spring. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 762–767. IEEE (2011)
4. Yigit, C.B., Bayraktar, E., Boyraz, P.: Low-cost variable stiffness joint design using translational variable radius pulleys. *Mech. Mach. Theory* **130**, 203–219 (2018)
5. Malzahn, J., Barrett, E., Tsagarakis, N.: A rolling flexure mechanism for progressive stiffness actuators. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 8415–8421 (2019)
6. Kim, D.-H., Oh, J.-H.: Hysteresis modeling for torque control of an elastomer series elastic actuator. *IEEE/ASME Trans. Mechatron.* **24**(3), 1316–1324 (2019)
7. Jarrett, C., McDaid, A.: Modeling and feasibility of an elastomer-based series elastic actuator as a haptic interaction sensor for exoskeleton robotics. *IEEE/ASME Trans. Mechatron.* **24**(3), 1325–1333 (2019)
8. Okken, B., Stramigioli, S., Roozing, W.: Progressive series-elastic actuation with magnet-based non-linear elastic elements. In: 2022 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 166–173 (2022)
9. Bluiminck, M. Roozing, W.: An underactuated tendon-driven gripper with variable stiffness for deformable agri-food objects,. In: 2023 IEEE 19th International Conference on Automation Science and Engineering (CASE), pp. 1–8. IEEE (2023)



A Reinforcement Learning Method to Minimize the Damage on a Falling Ballbot

Giulia Buzzetti¹(✉), Davide Zappetti², and Giovanni Iacca¹

¹ University of Trento, Trento, Italy

{giulia.buzzetti,giovanni.iacca}@unitn.it

² Enchanted Tools, Paris, France

davide@enchanted.tools

Abstract. As robots become ever more popular in various fields and require flexibility in unexpected situations, reducing damage to their body becomes ever more relevant. This paper presents our preliminary results on fall damage reduction on a commercial Ballbot, Miroki. While various approaches have been proposed for damage reduction for humanoid and animal-like robots, no previous works have addressed this challenge on Ballbots. To this aim, we implemented a deep RL framework and tested it on the Isaac Gym simulator. Compared to the free fall case, the learned control strategy manages to reduce the damage on the Ballbot by 52%, by controlling the arms of the robot during the fall.

Keywords: Reinforcement Learning · Damage Minimization · Ballbot

1 Introduction

Robotic systems play a pivotal role in contemporary society, with applications spanning various sectors such as industry, healthcare, logistics, and catering. Their ubiquity is attributed to the reliability of robotic structures, surpassing human labor in diverse scenarios. The evolution of robotic systems has yielded robust and precise robots that increasingly operate in close proximity to people, including customers and operators.

Despite the sophisticated hardware and software constituting these robots, occasional failures, such as *falls*, can still happen. When falls occur, mitigating any possible damage (to the robot and the environment) is a critical challenge. This challenge is particularly pronounced for robots capable of locomotion tasks, such as humanoid, animal-like, or wheeled robots, which lack a fixed base [5].

Addressing this challenge involves optimizing control systems, where *awareness*, namely a precise understanding of the current situation based on the available information or experience, emerges as a crucial factor in enhancing control strategies. Previous research [2] delineates various levels of awareness, including self-monitoring and experiential awareness. In this regard, minimizing fall damages in robotic systems can be seen as a form of *C0 awareness*, i.e., unconscious

processing (a sort of “instinctive behavior”), which in robotic systems can be obtained in various ways, such as through Reinforcement Learning (RL) [9].

Several strategies have been proposed to minimize damage during falls, especially for humanoid and animal robots. Fujiwara et al. [3] introduced a movement sequence, known as the UKEMI strategy, to reduce impact. RL strategies for hand protection during falls are explored in [4], while [11] presents a fall sequence minimizing joint and robot part damage. Recent work by Ma et al. [6] focuses on a legged robot with a robotic arm, implementing an RL strategy for damage reduction and recovery after a fall. To the best of our knowledge, no previous work has addressed the problem of minimizing damages on Ballbots, which is the main focus of our work.

A Ballbot, distinguished by a single large wheel or ball for locomotion, exhibits dynamic stability, making it suitable for agile movement in cluttered and narrow environments [1,8]. However, maintaining balance during movement requires constant adjustments to avoid falls. Given their intended interaction in social environments, finding an *aware* (of the fall itself, and the environment) control strategy for reducing damages derived from falls in Ballbots is paramount.

This paper presents our preliminary findings on a control strategy aimed at minimizing damage when a Ballbot falls. We developed a deep RL framework to train the robot based on its experiences. Our methodology was implemented and tested on Miroki, a commercial robot developed by Enchanted Tools¹, featuring a rolling globe for agile movement, co-manipulating arms, AI-based speech recognition, and expressive facial features (see Fig. 1). With the policy developed, we assess a 52% reduction of the peak contact force on all body parts.

The remainder of this paper is organized as follows: in Sect. 2 we outline the proposed framework; in Sect. 3 we present the results of our approach; finally, the conclusions are presented in Sect. 4.

2 Methods

This section describes the implementation of the deep RL framework built to minimize the damage on the Miroki robot. The robot is 115 cm tall and weighs 30 kg. It stands on a ball that is controlled through 3 wheels. This enables swift movement and it is easy both to move and to transport. Due to its configuration, it has an unstable equilibrium. The body is attached to the ball and has two hands with 7 Degrees of Freedom (DOF) each. Three additional DOF are present in the neck, plus one additional DOF for the each ear. In the experiments we present here, we decided to control only the arms of the robot, keeping all the other DOF as fixed joints. As we will see in Sect. 3, the specific configuration of Miroki allows indeed to reduce the damage on the body using only the arms.

¹ <https://enchanted.tools>.

Simulation Environment. The robot underwent training through simulating falls in 8192 parallel (vectorized) environments² using Isaac Gym [7]. Due to the complexity of employing triangular meshes and force sensors, the entire pipeline could not be trained on the GPU. Consequently, the simulation ran on the CPU, while the RL part was trained on the GPU. The RL algorithm employed was the Proximal Policy Optimization (PPO) method [10]. The neural network incorporates as input the actuated joint position and velocity, the body orientation, the linear and angular velocities, and the previous actions. Further details on the reward function definition are presented in the subsequent section, while Table 1 outlines the hyperparameters governing the training process.

Table 1. Weight configuration for the loss function and parameters used in the experimentation.



Fig. 1. Front view of the Ballbot developed by Enchanted Tools.

| Reward component | Weight |
|--|----------------------|
| End-episode: Body contact force (ω_b) | -0.005 |
| End-episode: Joint jank (ω_j) | -0.05 |
| Mid-episode: Velocity (ω_v) | -0.005 |
| Parameter | Value |
| Max. episode length | 5 s |
| Controller frequency | 50 Hz |
| Mini-batch size | 131072 |
| KL threshold | 1.6×10^{-2} |
| Horizon length | 32 |
| # environments | 8192 |
| # total training timesteps | 786×10^6 |
| # epochs | 4 |
| # hidden units | 256, 128, 64 |

Reward Design. The policy was trained using two different rewards: one given *during* the episode, and another one given just *at the end* of the episode. We refer to these rewards as *end-episode* (r_e) and *mid-episode* (r_m) rewards, respectively.

Before the episode ends, the *mid-episode* reward is computed at each timestep of the simulation. The goal of this reward is to achieve the final goal in the most efficient way possible, and it is related to the *minimization of the energy* spent while doing the movement, to ensure that the movement is smooth and unnecessary movements are not performed. This reward has the following form:

$$r_m = \omega_v ||\dot{q}|| \quad (1)$$

² See: <https://github.com/NVIDIA-Omniverse/IsaacGymEnvs>.

where $\|\dot{q}\|$ denotes the Euclidean norm of the actuated joint angular velocities, ω_v is a negative weight needed to minimize the actuated joints angular velocity and therefore the energy. This reward resets at each timestep and it is not cumulative through the steps of each episode. It is important to notice that the mid-episode reward should be negligible when compared to the end-episode reward, since the maximization of the end-episode reward allows to accomplish the actual minimization of the damage on the robot. This is possible thanks to the weight parameter ω_v .

At the end of the episode, the *end-episode* reward r_e measures the *damage that the robot suffers from the fall*. Following previous works [4, 11], this reward takes into account both the damage on each body part and on each controlled joint. To do so, the end-episode reward is given by the sum of the quantities r_{eb} and r_{ej} , where:

$$r_{eb} = \omega_b \max_{\mathcal{T}} \sum_{b \in \mathcal{B}} \|cf_b(t)\|^2 \quad (2)$$

$$r_{ej} = \omega_j \max_{\mathcal{T}} \sum_{i=1}^N \|df_i(t) - df_i(t-1)\|^2 \quad (3)$$

where ω_b and ω_j are negative weights; \mathcal{T} represents the set comprising all the timestamps that constitute an episode; \mathcal{B} is the set of all the body parts; cf_b is the contact force acting on the b -th body part; N is the number of actuated DOF; and, df_i represents the internal joint torque on the i -th DOF. We prioritize the minimization of the contact forces acting on the body w.r.t. the joint jank (i.e., given their relative scales, $|\omega_b| < |\omega_j|$) as, in general, it is more onerous to replace body parts rather than dealing with the damage occurred on joints.

3 Results

We evaluated the worst-case (out of 4 test runs) results obtained by the policy found by comparing them with the setting where the robot falls without performing any action (free fall). Following [4], we evaluated the damage by considering the *peak contact force* along the z direction over all the body parts during the fall. While in the free fall the robot falls heavily on its back, the policy allows to mitigate the impact on the other body parts. The robot uses the right arm to do it, and the contact sequence is as follows: palm, forearm, shoulder transversal, and shoulder frontal. It is worth mentioning that the strategy found by the policy is similar to the policy implemented with sequential planning, as done in [11]. Observing Fig. 2, we can see that the maximum impact with the policy is on the shoulder frontal, and occurs about 0.1 s later than in the free fall setting (1.3 s vs. 1.2 s). This behavior is likely due to a lower velocity during the episode that leads to a lower velocity in the moment of impact, a desired behavior since the impact velocity and the contact forces are proportional. By comparing the peak contact force on the z axes over all the body parts, we observe that our policy allows us to reduce the damage on the robot by 52%: the policy exhibits

a peak contact force of 1261.8556N on the frontal shoulder link, while the free fall registers a peak force of 2626.8125N on the trunk. A video comparison of the simulation of the two settings can be found at the following link: https://github.com/giu950/Ballbott_Fall.git.

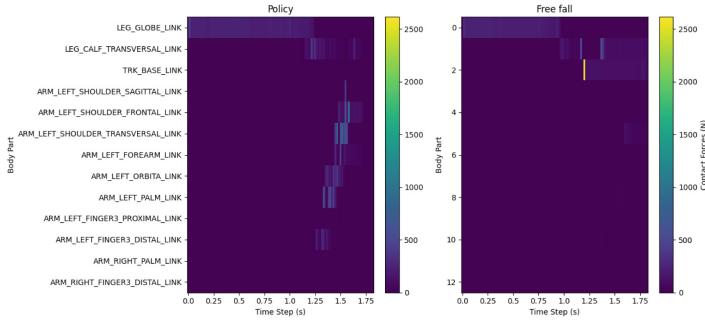


Fig. 2. 95-th percentile (over 4 test runs) of the contact forces along the z direction acting on each body part during the fall with (left) and without (right) policy. For conciseness, the plot presents only the body parts experiencing a contact force $>0.5\text{N}$.

4 Conclusion

Our initial findings demonstrate the efficacy of the proposed deep RL framework in minimizing damage to a Ballbot during falls. Through the control of the two arms, the proposed policy achieved a remarkable 52% reduction in robot damage, with a strategy akin to human-inspired approaches. Future efforts will focus on optimizing the control framework to facilitate sim-to-sim and sim-to-real transfer. Additionally, we aim to explore a novel metric that simultaneously mitigates damage to both the robot and its surrounding environment. These directions aim to further refine our approach, contributing to more robust and adaptive strategies for fall damage reduction in Ballbots.

References

1. Cai, C., Lu, J., Li, Z.: Kinematic analysis and control algorithm for the ballbot. *IEEE Access* **7**, 38314–38321 (2019)
2. Dehaene, S., Lau, H., Kouider, S.: What is consciousness, and could machines have it? *Science* **358**(6362), 486–492 (2017)
3. Fujiwara, K., Kanehiro, F., Kajita, S., Kaneko, K., Yokoi, K., Hirukawa, H.: UKEMI: falling motion control to minimize damage to biped humanoid robot. In: International Conference on Intelligent Robots and Systems, vol. 3, pp. 2521–2526. IEEE (2002)

4. Kumar, V.C.V., Ha, S., Liu, C.K.: Learning a unified control policy for safe falling. In: International Conference on Intelligent Robots and Systems, pp. 3940–3947. IEEE (2017)
5. Li, Q., Chen, X., Zhou, Y., Yu, Z., Zhang, W., Huang, Q.: A minimized falling damage method for humanoid robots. *Int. J. Adv. Robot. Syst.* **14**(5), 1729881417728,016 (2017)
6. Ma, Y., Farshidian, F., Hutter, M.: Learning arm-assisted fall damage reduction and recovery for legged mobile manipulators. In: International Conference on Robotics and Automation, pp. 12,149–12,155. IEEE (2023)
7. Makoviychuk, V., et al.: Isaac Gym: High Performance GPU-Based Physics Simulation For Robot Learning (2021)
8. Nagarajan, U., Kantor, G., Hollis, R.: The ballbot: an omnidirectional balancing mobile robot. *Int. J. Robot. Res.* **33**(6), 917–930 (2014)
9. Pessiglione, M., Petrovic, P., Daunizeau, J., Palminteri, S., Dolan, R.J., Frith, C.D.: Subliminal instrumental conditioning demonstrated in the human brain. *Neuron* **59**(4), 561–567 (2008)
10. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint [arXiv:1707.06347](https://arxiv.org/abs/1707.06347) (2017)
11. Ruiz-del Solar, J., Palma-Amestoy, R., Marchant, R., Parra-Tsunekawa, I., Zegers, P.: Learning to fall: designing low damage fall sequences for humanoid soccer robots. *Robot. Auton. Syst.* **57**(8), 796–807 (2009)



EMERGE - Emergent Awareness from Minimal Collectives

Davide Bacciu¹(✉), Vincenzo Ambriola¹, Bahador Bahrami³, Andrea Ceni¹, Andrea Cossu¹, Valerio De Caro¹, Cosimo Della Santina², Ophelia Deroy³, Claudio Gallicchio¹, Riccardo Guidotti¹, Sabine Hauert⁵, Simon Jones⁵, Jurgis Karpus³, Suet Lee⁵, Jingyue Liu², Vincenzo Lomonaco¹, Nadine Meertens³, Emma Milner⁵, Anna Monreale¹, Renan Picoreti Nakahara⁴, Mariano Ramirez², Ebrahim Shahabi², and Maximilian Stölzle²

¹ University of Pisa, Pisa, Italy

davide.bacciu@unipi.it

² Delft University of Technology, Delft, Netherlands

³ Ludwig Maximilian University, Munich, Germany

⁴ Da Vinci Labs, Tours, France

⁵ University of Bristol, Bristol, UK

Abstract. We introduce the concept of collaborative awareness as a means to enhance interoperability, resilience and self regulation in synthetic agent collectives. We discuss the theoretical, computational and engineering framework of collaborative awareness built by the EU project EMERGE, and its application to distributed robotic systems.

Keywords: awareness · distributed robotic systems · robot swarms · neural networks · dynamical systems

1 Introduction

Awareness is a fundamental aspect of any agent capable of information integration into complex and rich internal representations, across a multitude of dimensions including perceptual, introspective and social facets. In this respect, it characterizes as a fundamental concept to engineer efficient and effective distributed loosely-coupled systems of synthetic agents [6], as it caters for the ability to communicate and interoperate by exchanging bits of semantically rich information, which are the awareness states possessed by the single agents. Awareness in biological agents has converging definitions when considering local states describing content-related awareness from an agent-specific perspective. However, it becomes highly debated when it comes to global states. The issue magnifies when considering collectives of artificial agents, such as in robotic swarms. Several frameworks exist, all unsatisfactory in the limitations posed to agents' heterogeneity and disappearance of the local self into an integrated state. Ultimately, existing awareness frameworks are ineffective in explaining, facilitating,

and supporting cooperative behaviours in artificial agents. The lack of a compelling theory of global awareness in AI is currently a significant barrier to the effective deployment of artificial agents in the real world. The EIC Pathfinder Challenge project EMERGE (grant n. 101070918,2022–2026)¹ is a recent 4-years research endeavour tackling this grand challenge by introducing and characterizing a novel concept of global awareness that we call collaborative awareness. EMERGE builds on a scenario comprising collectives of artificial beings with no shared language and constrained individual capabilities, which nevertheless leads to high-complexity behaviours at the collective level. Collaborative awareness becomes an emergent process supporting complex, distributed, and loosely coupled systems capable of high degrees of collaboration, self-regulation, and interoperability without pre-defined protocols. EMERGE designs a philosophical, mathematical, and technological framework that enables to know how and where to allocate awareness to optimally achieve a goal through the collective, and whose characterizing aspects are overviewed in Sect. 2. Collaborative awareness can have a transformative effect on those application domains involving delivering a service over a loosely coupled collective of virtual/physical entities, by providing enhanced resilience to high heterogeneity and compelling constraints on computation and energy. As such, we envision IoT, smart cities and transportation, microservice-based ICT systems, biomedical nanodevices, and robotics as immediate impact areas. Among them we believe that robotics provides the perfect setting to test the fundamental concepts of our framework and, in Sect. 3, we provide a brief account of how EMERGE is feeding into robotic applications involving soft robots, robotic swarms and cobots.

2 A Theoretical, Computational and Engineering Framework of Collaborative Awareness

Theoretical Framework. EMERGE defines collaborative awareness using an emergent approach where locally aware artificial agents, of possibly heterogeneous capabilities and without a shared language, develop a form of collective awareness through interacting with each other and the environment. We build on the idea that this form of awareness emerges when coordination or cooperation across a collective of task-aware agents is needed. We formalise collaborative awareness following a dimensional framework [1] representing the variations across individuals on a multidimensional space of awareness capabilities and corresponding scales. Our framework acknowledges that there are different domains where awareness needs to be defined and measured, e.g. bodily, spatial, temporal, situational, etc. It acknowledges for possible dependencies between dimensions, without imposing a unified model throughout the collective: single agents can be comparable in terms of quantity of awareness while differing on the dimensions on which they fare higher. This framework allows us to consider collaborative awareness as an addition of new dimensions and changes the way awareness is measured across dimensions, when confronted to purely local awareness states.

¹ EMERGE project web page: <https://eic-emerge.eu/>.

Ethical Framework. EMERGE assesses the awareness framework also from an ethics perspective, evaluating the attribution of agentive and moral responsibility to collectives having collective awareness, and characterising transparency, trust, and risks of such systems. In particular, we are exploring key ethical challenges and opportunities that could arise with the introduction of collective aware systems powered by AI into human society. Some of our early results in this sense, highlight that humans are keen to exploit benevolent algorithms with significant differential treatment of humans and machines under the same experimental scenario. Another direction of study [10] explores the key differences between human attitude about responsibility against other humans, AI agents (single and collective), and conventional, non intelligent, tools. The third dimension of our ethical framework studies transparency of AI systems and, in particular, defining and measuring awareness through explainability approaches [3]. To this end, we are building a framework that relates the different awareness types of our dimensional model with specific explainability techniques and methodologies.

Computational Framework. Collaborative awareness emerges as goal-promoted interactions unfold following transformations of perceptual states into richer awareness states. To this end, it is important to be able to identify and characterize template dynamical transformations, and their compositions, that favour emergence of collaborative awareness from local perceptions. We use nonlinear dynamical system theory, for its ability to integrate rich temporal information across scales, to establish, analyse, implement, and test a new AI framework for collaborative awareness. The underlying mathematical framework comprises a dictionary of low-dimensional template dynamical systems and ways of connecting them (*Archetypes*) so that they can be combined (in *Archetypes Networks*) to generate awareness. We complement this by mechanisms capable of inferring a map from a real-world system to its archetype representation, and viceversa. The composed Archetype Networks are not abstract and static items. Rather, they are meant to be executed and deployed in the physical world, i.e. to provide agents with the ability of developing awareness about specific aspects, from their interaction with the environment and other agents. This calls for a computational engine (the *Archetype Computing System* (ACS)) with the level of adaptivity needed to process and represent dynamically changing information. To this end, we provide the ACS with life-long learning mechanisms operating on Archetype units parameters and evolutionary mechanisms to adapt structural aspects of the Archetype Networks. The ACS provides a computing engine to perform dynamic distributed information processing by executing the Archetype Networks, seeking emergence of awareness in the attractors of their non-linear systems. Our computational framework is general enough to be able to describe both the information transformation *computed* by the body of a robot, as well as neural layer transformations. The ultimate goal of EMERGE, in this sense, is going towards the realization of a universal dynamical systems computing engine [7], strengthening the bridge between physics and computation, creating a substrate for the execution on non-conventional architectures, e.g. neuromor-

phic and morphological computing [11]. In our early results, we have shown how the Archetype framework can be used to define new recurrent neural networks designs based on oscillator-like neural units [2]. From a mapping perspective, we have looked into how to learn physical-consistent dynamics of Lagrangian systems from data [9] and how to compress high dimensional mechanical systems akin to soft robots into low dimensional neural representations [8].

Engineering Framework. To realize an actionable implementation of collaborative awareness for collectives of intelligent synthetic agents, we need an additional ingredient, that is an engineering framework that can control emergence of (the different dimensions of) awareness. Understanding and engineering awareness requires to have impartial metrics that can measure changes in behaviour based on different dimensions of awareness. However, awareness cannot be measured directly. Hence, our engineering framework builds first and foremost a scheme that will allow to evaluate each dimension of awareness in a given context for any system. We put forward a framework in which an awareness dimension is measured by evaluating the performance of the systems in *proxy tasks*, associated with specific *capacities* for the given awareness dimension. Performance is measured with respect to defined *metrics* (e.g. robustness, adaptability, speed) and tasks are chosen to allow differential evaluation with respect to capacities. Under such a setting we measure awareness through the change of task performance between systems with and without awareness capacities. Such awareness metrics will then serve two purposes: (i) they will be used by evolutionary methods to guide the evolution of agent collectives with emergent collaborative awareness; (ii) they will serve to provide human-readable insight and control over the resulting collective.

3 Collaborative Awareness in Distributed Robotics

EMERGE results find perfect match in distributed robotic systems. We consider three scenarios each corresponding to a representative areas of robotics: (i) modular soft robots, as an example of a physically distributed collective where the body needs to self-organize to account for the dynamic addition of components; (ii) robotic swarms, as an example of large scale minimal collective where agents need coordination to achieve a collaborative goal; and (iii) cobots, as a closer-to-market use case where interoperability is currently a significant barrier.

Soft robots [4] are characterized by continuously deformable bodies inspired by invertebrates. We target a local minimal collective under the form of a modular soft squid-like robot entirely made of silicon. Its main body will contain some central intelligence, where any number of soft tentacles can be added in a modular fashion (along with their local intelligence). Minimal or no explicit information is shared between the various parts, which will autonomously learn to swim in an unknown environment. This will allow us to investigate the capability of physically distributed systems to create awareness in a scenario of contained complexity. Our *robot swarm* deployment [5] is based on an autonomous

warehouse with users taking robots out-of-the-box and needing the collective to gain collaborative awareness in order to achieve the task and communicate their awareness to the user. Swarm members will be artificially made minimal, their capabilities varied by evolutionary learning to understand the impact of complexity and heterogeneity on the emergence of collaborative awareness. *Cobots* have potential for large-scale industrial use and to allow cooperation with human workers. However, they still suffer from two significant issues: (i) limited individual intelligence and inability to autonomously adapt to situations and, (ii) lack of interoperability between systems by different providers. We will investigate if emergent awareness can solve these two challenges by putting collaborative robots in a realistic retail environment, and comprising cobots from different companies. These will interact without direct information transfer, assessing whether collaborative awareness can emerge to a sufficient extent to perform standard tasks like filling shelves with products.

Acknowledgements. This work was funded by the EU under GA 101070918, EMERGE project. UK participants in EMERGE are supported by UKRI grant number 10038942.

References

1. Birch, J., Schnell, A.K., Clayton, N.S.: Dimensions of animal consciousness. *Trends Cogn. Sci.* **24**(10), 789–801 (2020)
2. Ceni, A., et al.: Randomly coupled oscillators for time series processing. In: ICML Workshop on New Frontiers in Learning, Control, and Dynamical Systems (2023)
3. Cossu, A., Spinnato, F., Guidotti, R., Bacciu, D.: A protocol for continual explanation of shap. In: Verleysen, M. (ed.) *Proceedings of ESANN 2023* (2023)
4. Della Santina, C., Duriez, C., Rus, D.: Model-based control of soft robots: a survey of the state of the art and open challenges. *IEEE Control Syst. Mag.* **43**(3), 30–65 (2023)
5. Jones, S., Hauert, S.: Frappe: fast fiducial detection on low cost hardware. *J. Real-Time Image Process.* **20**(6), 119
6. Jones, S., Milner, E., Sooriyabandara, M., Hauert, S.: Distributed situational awareness in robot swarms. *Adv. Intell. Syst.* **2**(11) (2020)
7. Kia, B., Lindner, J.F., Ditto, W.L.: Nonlinear dynamics as an engine of computation. *Philos. Trans. R. Soc. A: Math. Phys. Eng. Sci.* **375** (2017). <https://api.semanticscholar.org/CorpusID:3711548>
8. Lepri, M., Bacciu, D., Della Santina, C.: Neural autoencoder-based structure-preserving model order reduction and control design for high-dimensional physical systems. *IEEE Control Syst. Lett.* (2023)
9. Liu, J., Borja, P., Santina, C.D.: Physics-informed neural networks to model and control robots: a theoretical and experimental investigation (2023)
10. Longin, L., Bahrami, B., Deroy, O.: Intelligence brings responsibility - even smart AI assistants are held responsible. *iScience* **26**(8) (2023)
11. Marković, D., Mizrahi, A., Querlioz, D., Grollier, J.: Physics for neuromorphic computing. *Nat. Rev. Phys.* **2**(9) (2020)



The Landscape of Collective Awareness in Multi-robot Systems

Miguel Fernandez-Cortizas^{1(✉)}, David Perez-Saura¹, Ricardo Sanz²,
Martin Molina^{1,3}, and Pascual Campoy¹

¹ Computer Vision and Aerial Robotics Group (CVAR), Universidad Politécnica de Madrid, Madrid, Spain

miguel.fernandez.cortizas@upm.es

² Autonomous Systems Laboratory (ASL), Universidad Politécnica de Madrid, Madrid, Spain

³ Department of Artificial Intelligence, Universidad Politécnica de Madrid, Madrid, Spain

Abstract. The development of collective-aware multi-robot systems is crucial for enhancing the efficiency and robustness of robotic applications in multiple fields. These systems enable collaboration, coordination, and resource sharing among robots, leading to improved scalability, adaptability to dynamic environments, and increased overall system robustness. In this work, we want to provide a brief overview of this research topic and identify open challenges.

Keywords: Collective Awareness · Distributed Awareness · Multi-Robot Systems

1 Introduction

Multi-robot systems (MRS) are becoming more and more capable day after day. However, there is still a lack of awareness about what is happening around them, that may lead to errors and failures when there are unexpected changes in their operational conditions. Trying to solve this issue, roboticists work on improving the situational awareness of these robotic systems so that they become more robust and resilient, and they can “know” how to act in order to fulfill their objective coordinately in the best way. In these systems, it is necessary to have not only an “individual awareness” per agent, but also an awareness about the other agents of the system and their interactions, which can be integrated to generate a higher level of awareness, a “Collective Awareness”.

In this mapping work, we analyze existing research and literature to identify patterns and knowledge gaps in the field. By examining the current state of the art, the work hopes to provide useful insights into what collective awareness is and how it can be effectively integrated into multi-robot systems to enhance their performance and functionality. In this work, we want to respond to the following questions:

1. What *Collective Awareness* is in multi-robot systems?
2. How is collective awareness achieved in the literature?
3. What are the open challenges in the field?

2 Literature Review

In this work, we did a qualitative analysis to evaluate the literature involving the use of collective awareness in multi-robot systems. To this end, a series of searches were conducted in the Scopus database¹. We have searched for “Awareness” AND “robot” adding different keywords such as “collaborative”, “distributed”, or “collective” to increase the scope of the search up to 517 publications between 1992 and 2024. The number of publications has been increasing since 2007 showing an emerging interest in this research field.

1. Collective Awareness Definition

During the review, we found that there is no consensus on what *collective awareness* is. Some authors [16] use the term “distributed” or “collective” situational awareness as an extension of the former definition of situational awareness [6] as “the perception of elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future” but applied to multi-agent systems that perform this comprehension and projection in a collaborative way. Elaborating on that, Kube et al. [12] define Collective Awareness in robotics as “the ability of multiple robots to work together in a coordinated manner, often using decentralized control”. This approach is inspired by the collective behavior of social insects, which can work together effectively without a centralized leader. Following these ideas, works like [1], define “Swarm Intelligence” as an approach to the coordination of multiple robots as a system consisting of large numbers of mostly simple physical robots. In this approach, systems operate following the principles of Swarm Robotics, in which the self-organization of a large number of simple (and normally cheap) robots, with only local sensing, communication, and actuation capabilities, generates an emerging behavior that fulfills a concrete task.

As most of the works that use the term “collective awareness” do not define it, we try to extract what this means for them based on the structure and capabilities of the systems described by them.

2. Collaboration in Multi-robot Systems

Generally, to achieve some kind of collaboration, each team agent collects information that contributes to its local knowledge of the surrounding environment. To provide a globally consistent view of the entire environment, agents need to share their information with other team members [2].

In this section, we have summarized some of the most relevant works in this field, trying to identify commonalities and differences between them, leading to the different categorizations that appear in the following. Table 1 shows a comparison between the most significant works that recapitulate their main characteristics:

¹ <https://www.elsevier.com/products/scopus>.

Data Shared: Depending on the information that is exchanged between agents, three categories can be defined [4]: **Low-level sensory data:** raw data output from sensors. **Intermediate processed data:** raw data that have been processed to generate some data with a minimal to moderate structure, such as feature/value pairs or the output of sensor fusion algorithms. **Semantic data:** the most abstract layer, in which data are structured around objects with properties and relations, often semantically grounded, including ontological structures, logical structures, or graph-structured knowledge.

Type of MRS: Describes the type of multi-robot system, which determines the behavior and coordination between the agents. There are three different types: **Centralized Systems:** A central unit collecting information from multiple sources. This information is integrated for generating system-level knowledge, which can be used for deciding which is the best actions for each robot. It is effective in complex environments with robots having limited perception, but it lacks robustness, as failure in the central unit results in the entire system failing [3,8]. **Decentralized Systems:** Multiple robots or agents collaborate without central control. Each agent processes information locally, making decisions based on its observations and interactions with the environment and other agents. Decentralized systems are fault-tolerant, if one agent fails, the system can still operate [7,9,17]. **Swarm Intelligence Systems:** Distributed system composed of self-organizing groups of simple and cost-effective robots. Operating with local sensing, communication, and actuation, they collectively generate emergent behavior[15] to accomplish specific tasks [10].

System State Awareness: A binary property of the MRS that describes if the system generates a global view of the state of the system.

Mission Awareness: A binary property of the MRS that describes if the cognitive agents of the system know what is the mission that they are performing so that they can generate plans according to it.

Table 1. Comparison between some of the most relevant works. Data shared is categorized as S: Semantic data, I: Intermediate data, L: Low-level sensory data. SSA: System State Awareness. MA: Mission Awareness.

| Work | Task | Data Shared | Type of MRS | SSA | MA |
|---------|---------------|-------------|---------------|-----|----|
| [14] | Construction | L | Swarm | ✗ | ✓ |
| [5] | Exploration | L | Swarm | ✗ | ✓ |
| [3, 8] | SLAM | L,S | Centralized | ✓ | ✗ |
| [13] | Area coverage | L | Centralized | ✓ | ✗ |
| [10] | Manipulation | S | Decentralized | ✓ | ✓ |
| [9, 17] | Exploration | L | Decentralized | ✓ | ✓ |
| [4] | Exploration | I, L, S | Decentralized | ✓ | ✓ |
| [7] | SLAM | L,S | Decentralized | ✓ | ✗ |

3 Conclusions

In this study we explore the field of collective awareness in multi-robot applications, finding that although there is not a lot of work in this field, there are promising results that can improve the capabilities of MRS. Below we collect our insights of this mini-review when trying to answer the former questions that were formulated at the start of this work.

What is Collective Awareness? From the previous categorization we have found that there are multiple ways of achieving collaboration in Multi-Robot Systems; however, we consider that not all the Multi-agent systems in which there is some kind of collaboration have this “Collective Awareness”. Based on the above, we propose the following definition.

Definition 1. *A multi-agent system has Collective Awareness when:*

1. *Each agent has some degree of awareness about itself, about the environment that surrounds it, and about what is its goal.*
2. *Different agents are able to communicate between them.*
3. *The system as a whole is able to integrate all the information provided by the agents to extract knowledge that can be used to improve the knowledge of the system or the capabilities of the system.*

We consider that the Collective Awareness extends the sum of the individual Situational Awareness of each agent, since new observations between agents can be taken into account to extend the Individual Awareness of each robot or to generate emergent knowledge about the status of the system, which can lead into improved emergent behavior.

Some of the systems previously described do not fit in this definition, like the swarm systems, because each agent only has a very restricted knowledge of the whole system. In that case, the system only has a partial collective awareness, since it does not have awareness of the complete collective.

How is Collective Awareness Achieved in the Literature? In the literature, we can observe that the awareness of the robots is achieved by generating models of interest of the environment or the robot itself and sharing them with the rest of the system. There is a trend to take advantage of the use of high-level semantic knowledge, what can improve the performance and robustness of multi-robot systems as is demonstrated in [3,4,7].

What are the Open Challenges in the Field? One of the main limitations of these approaches is the lack of generality; most systems are tailored specifically for a very concrete use case, which limits the expansion or reutilization of these systems [11]. Some works provide tools for extending the knowledge of collective-aware systems with the use of formalizations such as ontologies. However, these

systems does not take into account relevant aspects such as knowledge uncertainty, risk associated with one decision, or failure tolerance [4]. Moreover, in the majority of systems, a secure and stable communication link between robots is taken as guaranteed, which may not be present in multiple applications.

Acknowledgments. This work was funded by: European Union's Horizon Europe Project No. 101070254 CORESENSE, project COPILOT ref. 2020/EMT6368, funded by the Madrid Government under the R&D Synergic Projects Program, project INSERTION ref. ID2021-127648OBC32 funded by the Spanish Ministry of Science and Innovation. The work of the second author is supported by the Spanish Ministry of Science and Innovation under its Program for Technical Assistants PTA2021-020671.

References

1. Beni, G., Wang, J.: Theoretical problems for the realization of distributed robotic systems. In: Proceedings of the 1991 IEEE International Conference on Robotics and Automation, vol. 3, pp. 1914–1920 (1991). <https://doi.org/10.1109/ROBOT.1991.131906>
2. Berger, C., Doherty, P., Rudol, P., Wzorek, M.: Auton. Agent. Multi-Agent Syst. **37**(2), 47 (2023)
3. Chang, Y., Hughes, N., Ray, A., Carlone, L.: Hydra-multi: collaborative online construction of 3D scene graphs with multi-robot teams (2023)
4. Doherty, P., Berger, C., Rudol, P., Wzorek, M.: Hastily formed knowledge networks and distributed situation awareness for collaborative robotics. Auton. Intell. Syst. **1**, 1–29 (2021)
5. Ebert, J.T., Berlinger, F., Haghigat, B., Nagpal, R.: A hybrid PSO algorithm for multi-robot target search and decision awareness. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 11,520–11,527. IEEE (2022)
6. Endsley, M.R.: Toward a theory of situation awareness in dynamic systems. Hum. Factors **37**(1), 32–64 (1995)
7. Fernandez-Cortizas, M., Bavle, H., Perez-Saura, D., Sanchez-Lopez, J.L., Campoy, P., Voos, H.: Multi s-graphs: an efficient real-time distributed semantic-relational collaborative slam. arXiv preprint [arXiv:2401.05152](https://arxiv.org/abs/2401.05152) (2024)
8. Greve, E., Büchner, M., Vödisch, N., Burgard, W., Valada, A.: Collaborative dynamic 3D scene graphs for automated driving (2023)
9. Hui, Y., Zhang, X., Shen, H., Lu, H., Tian, B.: DPPM: decentralized exploration planning for multi-UAV systems using lightweight information structure. IEEE Trans. Intell. Veh. 1–13 (2023). <https://doi.org/10.1109/TIV.2023.3322705>
10. Jurt, M., Milner, E., Sooriyabandara, M., Hauert, S.: Collective transport of arbitrarily shaped objects using robot swarms. Artif. Life Robot. **27**(2), 365–372 (2022)
11. Kernbach, S.: Awareness and self-awareness for multi-robot organisms. arXiv preprint [arXiv:1111.5219](https://arxiv.org/abs/1111.5219) (2011)
12. Kube, C.R., Zhang, H.: Collective robotics: from social insects to robots. Adapt. Behav. **2**(2), 189–218 (1993)
13. Luna, M.A., et al.: Spiral coverage path planning for multi-UAV photovoltaic panel inspection applications. In: 2023 International Conference on Unmanned Aircraft Systems (ICUAS), pp. 679–686. IEEE (2023)

14. Meng, Y., Gan, J.: A distributed swarm intelligence based algorithm for a cooperative multi-robot construction task. In: 2008 IEEE Swarm Intelligence Symposium, pp. 1–6. IEEE (2008)
15. Potts, M.W., Sartor, P., Johnson, A., Bullock, S.: A network perspective on assessing system architectures: foundations and challenges. *Syst. Eng.* **22**, 485–501 (2019)
16. Salmon, P.M., Stanton, N.A., Jenkins, D.P.: Distributed Situation Awareness: Theory, Measurement and Application to Teamwork (2017)
17. Zhou, B., Xu, H., Shen, S.: Racer: Rapid collaborative exploration with a decentralized multi-UAV system. *IEEE Trans. Rob.* **39**(3), 1816–1835 (2023). <https://doi.org/10.1109/TRO.2023.3236945>



Awareness in Robots

Burak Sisman¹(✉) and Tim Djedilbaev²

¹ Department of Cognitive Robotics, Delft University of Technology, 2628CD Delft, The Netherlands

b.sisman@tudelft.nl

² Department of Materials Science and Engineering, Delft University of Technology, 2628CD Delft, The Netherlands

t.e.djedilbaev@tudelft.nl

Abstract. This study explores the concept of awareness in robots. Human awareness, rooted in psychological and cognitive sciences, encompasses conscious perception of self and the environment. In contrast, robotic awareness is engineered, focusing on programmed perception, autonomy, and interaction with humans and the environment. This research highlights the converging functionalities of cognitive capabilities in robots.

Keywords: awareness · human cognition · metacognition · robotics

1 Introduction

The concept of awareness transcends the simple act of knowing, encompassing a spectrum that ranges from basic sensory perception to complex cognitive functions. This broad notion of awareness serves as a point of convergence for disciplines such as psychology, neurobiology, cognitive neuroscience, and robotics. Each field contributes unique perspectives and understandings, yet they are united in their exploration of awareness as a fundamental aspect of both living beings and artificial systems. It is important to recognize that awareness extends beyond the individual level. It also entails awareness of the environment.

In the literature, the term ‘awareness’ is broadly employed, encapsulating aspects such as consciousness, intelligence, cognition, and metacognition across both human and nonhuman agents. This general application allows its usage in diverse research contexts, though it sometimes leads to ambiguity. Clinically, the concept of awareness, relates to responsiveness. It might denote the state of consciousness where a person is able or unable to recall an episode [1]. In metacognitive studies, awareness is often linked to the capacity for monitoring and control. Some researchers do not call it metacognition but executive function [2, 12] as it is largely modulated by working memory [13]. In the fields of neurobiology and cognitive neuroscience, awareness signifies the embodied and brain-based human consciousness [11]. In robotics, awareness is defined as the capability of a non-biological system to perceive its environment or itself and

respond based on this input, akin to human or animal awareness. This gives rise to the notion of ‘synthetic awareness’, a term used to describe robots’ ability to predict the outcomes of their actions, mirroring the human brain’s capacity to anticipate behavioral consequences. ‘Synthetic awareness’ in robotics aligns with the notion of human metacognition but remains distinctly non-anthropomorphic.

Given these perspectives, any extensive examination of awareness should start with an understanding that, in biological systems, neuronal activity underpins states of awareness. This introductory exploration aims to identify the layers of awareness, dissecting its various forms and functions. By doing so, it seeks to provide a understanding of how awareness operates within robotics.

2 Literature Review

2.1 Awareness in Robots

Embodied robots, unlike human or animal agents, are non-biological cognitive systems with artificial origins. As such, traditional paradigms from human and animal cognitive research may not be entirely applicable when investigating a technical system engineered to cognize [3]. The term ‘cognize’ from the Oxford dictionary encompasses ‘to perceive, know, or become aware of’ [4]. Notably, the entry integrates the three dimensions of cognition: perception, knowledge acquisition, and awareness. The definitions of cognition differ slightly between broad reference sources, but none distinguish between ‘natural’ and ‘synthetic’ versions. In humans, cognition is inherent; in robots, it is designed and regulated, including the possibility for (synthetic) awareness.

In the fields of cognition, artificial intelligence, and related areas, there are two primary perspectives. One maintains that awareness and consciousness are distinct, while the other views them as the same phenomenon [5,6]. The debate is complicated as some researchers define one concept through the other, leading to circularity. Roboticists frequently do not explicitly include either construct in their models [9]. Nonetheless, several cognitive processes targeted and successfully recreated in robots bear functional similarities to the fundamental qualities of conscious cognition or awareness exhibited in organic cognitive systems. Robots now exhibit cognitive capacities that allow them to navigate uncertainties, which can be recognized as enablers of synthetic awareness. Such capabilities encompass the environment and self-awareness.

Environment Awareness in Robots. Environment awareness in robots is an expanding field that draws from the study of human perceptual awareness, particularly vision and attention. This area of research has progressed from static saliency maps to dynamic, context-aware models that mimic the evolutionary processes of cognitive abilities. Notable advancements in robotic simulations have been influenced by human brain mechanisms, as evidenced by the evolution from basic attention models to sophisticated Bayesian models that guide the

exploratory behaviors of modern unmanned aerial vehicles (UAVs) [14]. Essentially, environment awareness refers to the ability of robots to react to their environments. Robots have the cognitive ability to perceive data and recognize its importance and causal relationships. It allows them to not only perceive the presence of objects and entities in their visual area, but also perceive the relationships that exist among these elements. This form of awareness serves as a substitute for the robot's grasp on the world.

Social Awareness in Robots. The exploration of social awareness in robots, a branch of human-robot interaction and human-autonomy teams, delves into cognitive architectures that enable robots to interact socially. These architectures draw from human and animal social behaviors, facilitating a robot's ability to navigate social contexts, collaborate, and ensure personal assistance and safety [19, 22]. Central to these frameworks is the robot's capacity for social cue recognition and response adaptation, underpinned by sensor technology and contextual behavior modification [18]. Robot social engagement relies on situational and feedback awareness, guided by shared mental models and forward models [20, 21].

Self-awareness in Robots. The main reason for adding self-awareness in robots is that self-aware systems can handle novel situations with significantly higher flexibility and efficiency than non-self-aware counterparts [7, 8]. Self-awareness in robots refers to a robot's ability to identify its own condition while also situating itself within an environment and delimiting itself from it. In cognitive robotics and autonomous systems research, self-awareness states are often postulated and implemented as internal models. These models' lower-level organization may vary; for example, they could be knowledge-based [15]. However, they all have a conceptual congruence. This congruence is manifested through robot skills such as self-assessment, self-monitoring, self-regulation, self-localization, self-adaptation, self-preservation, and others that fall under the category of self-derivation. Some publications contend that the literature contains a "lack of the concept of self" [10, p. 9]. However, self-concept is one of the most intensively explored subjects in psychology and philosophy, with a broadly accepted understanding of its meaning [16]. In some ways, self-concept and self-awareness are synonymous: both refer to identifying oneself as a distinct agent independent from (1) the environment and (2) others. The latter is intimately tied to the ability of humans and animals to recognize their own physical selves, as demonstrated by the mirror test paradigm [16], but it is also applicable to robotic agents that pass the mirror test [17].

3 Conclusions

Our pursuit to understand awareness in robots shows that cognitive capabilities, though fundamentally different in origin, indicates converging functionalities.

This research has underscored that robotic awareness is engineered, aiming to replicate the functional aspects of human conscious cognition and adaptability. In humans, awareness is deeply rooted in conscious processing of self and the environment. Awareness in robots is an engineered construct, rooted in modulated perception, autonomy, and interaction with humans and the environment. Robots, equipped with ‘synthetic awareness’ are able to predict outcomes of actions and improve their performance, somewhat akin to human metacognition. This synthetic awareness relies on computational, not biological, processes.

Moreover, both humans and robots possess the ability to process sensory information from their surroundings, though the underlying mechanisms differ significantly. Humans rely on a biologically evolved cognition, enabling them to perceive, interpret, and adapt to environmental changes intuitively. This involves complex processes like visual recognition, spatial awareness, and cognitive flexibility, essential for survival and decision-making. Robots’ awareness is based on data perception, recognition of objects and entities, and understanding relationships within their environment. However, unlike humans, robots still lack the innate, adaptive responses and cognitive flexibility that come from biological evolution.

Additionally, self-awareness in robots is primarily a programmed state, focusing on self-assessment, monitoring, and adaptation within an environment. Unlike humans, robot self-concept does not necessarily include agency. As a result, the awareness in robots is limited by the level of complexity of their sensory and processing systems.

Acknowledgement. This work was funded by European Union’s Horizon Europe (EIC Pathfinder challenge Awareness Inside) Project No. 101070940 METATOOL.

References

1. Blumenfeld, H.: Impaired consciousness in epilepsy. *Lancet Neurol.* **11**, 814–826 (2012). [https://doi.org/10.1016/S1474-4422\(12\)70188-6](https://doi.org/10.1016/S1474-4422(12)70188-6)
2. Brooks, M.: Bridging metacognition and executive function: enhancing metacognition via development of the dorsolateral prefrontal cortex (2022)
3. Asada, M.: Rethinking autonomy of humans and robots. *J. Artif. Intell. Consciousness* **7**(02), 141–153 (2020)
4. Stevenson A., Lindberg C. A.: New Oxford American Dictionary, 3rd edn. Oxford (2010)
5. Deshmukh, V.D.: Consciousness, awareness, and presence: a neurobiological perspective. *Int. J. Yoga* **15**(2), 144 (2022)
6. Damasio, A., Meyer, K.: Consciousness: an overview of the phenomenon and of its possible neural basis. In: The Neurology of Consciousness: Cognitive Neuroscience and Neuropathology, pp. 3–14 (2009)
7. Gorbenko, A., Popov, V., Sheka, A.: Robot self-awareness: exploration of internal states. *Appl. Math. Sci.* **6**(14), 675–688 (2012)
8. Hernández, C., Bermejo-Alonso, J., Sanz, R.: A self-adaptation framework based on functional knowledge for augmented autonomy in robots. *Integr. Comput.-Aided Eng.* **25**(2), 157–172 (2018)

9. Vaughan, R., Zuluaga, M.: Use your illusion: sensorimotor self-simulation allows complex agents to plan with incomplete self-knowledge. In: International Conference on Simulation of Adaptive Behavior, 298–309. Springer, Heidelberg (2006)
10. Anshar, M., Williams, M.A.: Evolving synthetic pain into an adaptive self-awareness framework for robots. *Biol. Inspir. Cogn. Archit.* **16**, 8–18 (2016)
11. Arsiwalla, X.D., Verschure, P.: Measuring the complexity of consciousness. *Front. Neurosci.* **12**, 424 (2018)
12. Goldberg, L. R.: International personality item pool (2001). <http://bit.ly/1AfXuFco>
13. Baddeley, A.: Exploring the central executive. *Q. J. Exp. Psychol. Sect. A* **49**(1), 5–28 (1996)
14. Lanillos, P., Cheng, G.: Adaptive robot body learning and estimation through predictive coding. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4083–4090 (2018)
15. Silva, G.R., et al.: MROS: a framework for robot self-adaptation. In: 2023 IEEE/ACM 45th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), pp. 151–155 (2023)
16. Gallagher, S.: Philosophical conceptions of the self: implications for cognitive science. *Trends Cogn. Sci.* **4**(1), 14–21 (2000)
17. Lanillos, P., Pagès, J., Cheng, G.: Robot self/other distinction: active inference meets neural networks learning in a mirror. In: European Conference on Artificial Intelligence (2020). <https://doi.org/10.48550/arXiv.2004.05473>
18. Adams, B., Breazeal, C., Brooks, R.A., Scassellati, B.: Humanoid robots: a new kind of tool. *IEEE Intell. Syst. Their Appl.* **15**(4), 25–31 (2000)
19. Breazeal, C.: Designing Sociable Robots. MIT Press, Cambridge (2004)
20. Endsley, M.R.: Supporting human-AI teams: transparency, explainability, and situation awareness. *Comput. Hum. Behav.* **140**, 107574 (2023)
21. Ho, M.K., Griffiths, T.L.: Cognitive science as a source of forward and inverse models of human decisions for robotics and control. *Annu. Rev. Control Robot. Auton. Syst.* **5**, 33–53 (2022)
22. O’Neil, A., Russell, J.D., Thompson, K., Martinson, M.L., Peters, S.A.: The impact of socioeconomic position (SEP) on women’s health over the lifetime. *Maturitas* **140**, 1–7 (2020)



Metacognitive Control of Linear Dynamic Systems with Self-confidence Adaptation

Ajith Anil Meera^{1(✉)} and Pablo Lanillos^{1,2}

¹ Donders Institute, Department of Artificial Intelligence, Radboud University Nijmegen, Nijmegen, The Netherlands
ajith.anilmeera@donders.ru.nl

² Cajal International Center for Neuroscience, Spanish National Research Council, Madrid, Spain

Abstract. Metacognitive control is a salient capability of the human cognitive processes that allows them to regulate their executive functions and improve their performance and learning. Here, we contribute towards a mathematical model of metacognitive control for linear dynamic systems with three key features i) control the system, ii) evaluate a second order judgement about the decisions made (confidence in control) and, iii) self-monitor the control confidence and adapt it to improve the controller. We show that adapting the control confidence can improve the controller performance, particularly when the agent is biased with over confidence on its prior beliefs, i.e., it is too confident about its control.

Keywords: Robotic Awareness · Control Systems · Active Inference · Metacognition

1 Introduction

Metacognitive control is an integral part of self-awareness that allows humans to regulate their executive functions, such as learning [9]. For instance, they can regulate the time response to not make mistakes, depending on their confidence to perform a task. Analogously, in robotics, a controller that can evaluate and adapt its self-confidence online can help to optimally act in changing environments. Several attempts have been made to model metacognition in artificial agents [1, 4, 7]. The recent works have modelled metacognition by optimizing the agent's confidence [6] for a POMDP problem. However, solving it in the continuous time state action space still remains a challenge. In this paper, we present an adaptive control algorithm that optimizes for task performance, control cost and control self-confidence. And, importantly, it can estimate and adapt its self-confidence online. We construct the proposed controller from first principles under the brain-inspired Bayesian framework of active inference [11]. This approach has been shown to have interesting characteristics for robotics [8, 12], such as the uncertainty monitoring, which is essential for metacognitive approaches [2]. Our controller provides a closed-form expression for the control confidence (2nd order

judgement on its control action) and adapts it online to clear off any prior biases in its self-confidence. Consequently, it adapts a weighting term in the controller's objective function that we show is analogous to the weight on control cost of the optimal linear controller, thereby contributing to the optimal control theory.

2 Methods

Consider a linear time invariant system of the form:

$$\dot{x} = Ax + Bu, \quad y = Cx, \quad (1)$$

where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times r}$ and $C \in \mathbb{R}^{m \times n}$ are the matrices defining the system dynamics, $u \in \mathbb{R}^{r \times 1}$ is the control input to the system, $x \in \mathbb{R}^{n \times 1}$ is the hidden state, and $y \in \mathbb{R}^{m \times 1}$ is the measured output. We consider the problem of evaluating the control action u , to reach the goal x^g , within a desired level of uncertainty or prior covariance $\Sigma^{x^g} = (P^{x^g})^{-1}$.

We design a Bayes optimal controller from first principles by following the methodology of Variational Inference [3]. This is, for instance, to estimate the state of the system by approximating a known density to the posterior probability of the state given the observation. According to Bayes rule, the posterior distribution $p(\theta|y)$ of parameter θ , when the measurement y is known, is written as $p(\theta|y) = p(\theta, y)/p(y)$. Since the computation of $p(y) = \int p(y, \theta)d\theta$ is intractable for large search spaces of θ , variational methods use a recognition density $q(\theta)$ to closely approximate $p(\theta|y)$ by minimizing the Kullback-Leibler (KL) divergence between both the distributions. This procedure results in the minimization of an objective function called free energy, given by [5]:

$$F = \int q(\theta) \ln p(y|\theta)p(\theta)d\theta - \int q(\theta) \ln q(\theta)d\theta. \quad (2)$$

Under the free energy principle, brain's perception and control follows the minimization of its free energy and active inference agents optimize F to choose the control commands via gradient descent on the free energy objective. We proceed to evaluate this objective. We assume a Gaussian distribution of the form $q(u) = \mathcal{N}(u : \mu^u, (\Pi^u)^{-1})$ for the recognition density. The notation P is used for the prior precision (or inverse covariance) and Π is used for the conditional precision. We assume a Gaussian prior distribution on u , written as $p(u) = \mathcal{N}(u : \eta^u, (P^u)^{-1})$. To adapt P^u online, we parametrize it using λ , and place a Gaussian prior over it as $p(\lambda) = \mathcal{N}(\lambda : \eta^\lambda, (P^\lambda)^{-1})$. The distribution $p(x/u)$ is assumed to be Gaussian distributed as $p(x/u) = \mathcal{N}(x : x^g, (P^x)^{-1})$. Using x as the direct measurement ($y = x$) and $\theta = \{u, \lambda\}$ as the unknown parameter, the free energy in Eq. 2 (assuming $\eta^u = 0$, $\eta^\lambda = 0$, and dropping the constants) reduces to:

$$\begin{aligned}
F = & \underbrace{\frac{1}{2}(x - x^g)^T P^x (x - x^g)}_{\text{performance error } U^g} + \underbrace{\frac{1}{2} u^T P^u u}_{\text{control effort } U^c} + \underbrace{\frac{1}{2} \lambda^T P^\lambda \lambda}_{\text{penalty on } \lambda} \\
& - \underbrace{\frac{1}{2} \ln |\Pi^u|}_{\text{control-confidence } H} + \underbrace{\frac{1}{2} \ln |P^u|}_{\text{prior entropy}}.
\end{aligned} \tag{3}$$

The active inference controller rely on taking actions that minimises this free energy. The discrete time update rule of the active inference controller to control the states x can be written using the first two gradients of F as:

$$u(t + dt) = u(t) + \left(e^{-k^l \frac{\partial^2 F}{\partial u^2} dt} - I \right) \left(\frac{\partial^2 F}{\partial u^2} \right)^{-1} \frac{\partial F}{\partial u}, \tag{4}$$

where the gradients can be computed by differentiating Eq. 3 as:

$$\frac{\partial F}{\partial u} = (x - x^g)^T P^{x^g} \frac{\partial x}{\partial u} + u^T P^u, \quad \frac{\partial^2 F}{\partial u^2} = \frac{\partial x^T}{\partial u} P^{x^g} \frac{\partial x}{\partial u} + P^u. \tag{5}$$

Differentiating Eq. 1 with u and substituting $\frac{\partial \dot{x}}{\partial u} = 0$ in it yields $\frac{\partial x}{\partial u} = -(A^{-1}B)^T$. Further substituting it in Eq. 5 yields the gradients as:

$$\frac{\partial F}{\partial u} = -(x - x^g)^T P^{x^g} (A^{-1}B) + u^T P^u, \quad \frac{\partial^2 F}{\partial u^2} = (A^{-1}B)^T P^{x^g} A^{-1}B + P^u. \tag{6}$$

Following our formulations from the previous work [2], the controller's confidence (second order judgement) on its decisions (Π^u) can be written as:

$$\Pi^u = \frac{\partial^2 (U^g + U^c)}{\partial u^2} = (A^{-1}B)^T P^{x^g} A^{-1}B + P^u, \tag{7}$$

where U^g is the performance error and U^c is the control effort from Eq. 3. The control confidence is dependent on the system matrices A and B , the goal precision P^{x^g} and the agent's prior precision on action P^u . Since Π^u is directly proportional to P^u , an incorrect assignment of P^u could bias the agent into being highly self confident about its actions. We propose a scheme to change P^u such that the agent's self confidence on its actions Π^u can be adaptively tuned in real time, based on the system performance. Intuitively, this means that if the agent is over confident about its actions, it can correct its confidence from data in real time based on its own performance. The online adaptation of P^u enables the agent to self monitor its actions (via F), and adapt its next actions ($u(t+1)$ in Eq. 4) and its confidence on those actions (Π^u in Eq. 7), all aimed at the task completion.

The key contribution of this paper is the online adaptation of Π^u such that it minimises F . We propose an exponential parametrization for P^u of the form $P^u = e^\lambda I_{n^u}$, where $\lambda = [\lambda^1 \ \lambda^2 \ \dots \ \lambda^{n^u}]$, followed by an update rule for λ such that F in Eq. 3 is minimised by:

$$\lambda(t + dt) = \lambda(t) + \left(e^{-k^\lambda \frac{\partial^2 F}{\partial \lambda^2} dt} - I \right) \left(\frac{\partial^2 F}{\partial \lambda^2} \right)^{-1} \frac{\partial F}{\partial \lambda}. \tag{8}$$

The gradients can be computed by differentiating Eq. 3 with λ as:

$$\frac{\partial F}{\partial \lambda} = \frac{1}{2} u^T P^u u + \frac{1}{2} n^u + \lambda^T P^\lambda, \quad \frac{\partial^2 F}{\partial \lambda^2} = \frac{1}{2} u^T P^u u + P^\lambda. \quad (9)$$

Since P^λ and P^u are positive definite matrices, $\frac{\partial^2 F}{\partial \lambda^2}$ is always positive definite, resulting in F being a convex function with respect to λ . Therefore, our update law is guaranteed to provide a global optimal solution. Equations 4, 7 and 8 together represent our metacognitive control algorithm. As a result, the agent performs three metacognitive control functions: i) compute the optimal control action, ii) compute the control confidence and iii) adapt the control confidence online. When $F = U^g + U^c$ in Eq. 3, F becomes the same objective as that of J from optimal linear control ($J = \frac{1}{2} x^T Q x + \frac{1}{2} u^T R u$, where Q and R are weights) [10]. Here R is the same as P^u . Our online adaptation of P^u within Eq. 3 is then equivalent to adapting the control cost R from the quadratic objective J , thereby contributing to the control system domain.

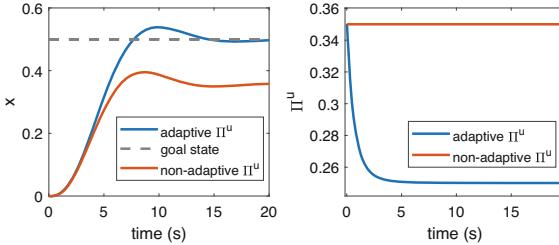


Fig. 1. The controller with an adaptive control confidence (in blue) outperforms the one without it (in red) by reaching the goal state x^g (in dotted black). It drops its high initial control confidence Π^u (right figure). However, the controller with non-adaptive Π^u (in red) has a steady state error with the goal state (left figure) because the high P^u biased the controller towards being over confident on its actions.

3 Simulation Results

This section provides a numerical simulation that shows the advantage of our adaptive control algorithm. We consider a spring mass damper system as a test example for the linear system in Eq. 1, with mass 0.4 kg, spring constant $k = 0.2$ N/m and damping constant $b = 0.4$ Ns/m. The system matrices are $A = \begin{bmatrix} 0 & 1 \\ -k & -b \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ m \end{bmatrix}$, $C = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$. We consider two variants of the controller, one with and one without the online adaptation of Π^u . We subject both the controllers to a case where P^u is highly biased ($P^u = 0.1$) towards the prior mean on the control action ($\eta^u = 0$). Intuitively, this prior belief forces the agent to penalise for higher control actions. Figure 1 shows the advantage of adapting Π^u online to drop the initial over confidence of the agent on its control priors. The non adaptive controller has a steady state error with the goal, while the adaptive controller does not.

4 Conclusion

We proposed a brain-inspired metacognitive controller for linear dynamic systems, with the capability to compute and adapt its control confidence online. Through a simulation experiment, we showed that the controller with an adaptive control confidence outperformed the one without it. Within awareness robotic technologies, this contributes to a mathematical description of metacognitive control, that can make 2nd order judgements on its decisions for self adaptation. The future work can focus on extensively benchmarking our controller against randomised systems, noises, model uncertainties, controllers and tasks.

Acknowledgments. This work has been funded by the EU METATOOL project funded by the EIC Pathfinder challenge Awareness Inside.

References

1. Anderson, M.L., Oates, T., Chong, W., Perlis, D.: The metacognitive loop I: enhancing reinforcement learning with metacognitive monitoring and control for improved perturbation tolerance. *J. Exp. Theor. Artif. Intell.* **18**(3), 387–411 (2006)
2. Anil Meera, A., Lanillos, P.: Towards metacognitive robot decision making for tool selection. In: International Workshop on Active Inference, pp. 31–42. Springer (2023)
3. Blei, D.M., Kucukelbir, A., McAuliffe, J.D.: Variational inference: a review for statisticians. *J. Am. Stat. Assoc.* **112**(518), 859–877 (2017)
4. Daglarli, E.: Computational modeling of prefrontal cortex for meta-cognition of a humanoid robot. *IEEE Access* **8**, 98491–98507 (2020)
5. Friston, K.: The free-energy principle: a unified brain theory? *Nat. Rev. Neurosci.* **11**(2), 127–138 (2010)
6. Hesp, C., Smith, R., Parr, T., Allen, M., Friston, K.J., Ramstead, M.J.: Deeply felt affect: the emergence of valence in deep active inference. *Neural Comput.* **33**(2), 398–446 (2021)
7. Kawato, M., Cortese, A.: From internal models toward metacognitive AI. *Biol. Cybern.* **115**, 415–430 (2021)
8. Lanillos, P., et al.: Active inference in robotics and artificial agents: survey and challenges. arXiv preprint [arXiv:2112.01871](https://arxiv.org/abs/2112.01871) (2021)
9. Lyons, K.E., Zelazo, P.D.: Monitoring, metacognition, and executive function: elucidating the role of self-reflection in the development of self-regulation. *Adv. Child Dev. Behav.* **40**, 379–412 (2011)
10. Ogata, K.: Modern control engineering fifth edition (2010)
11. Parr, T., Pezzulo, G., Friston, K.J.: Active Inference: The Free Energy Principle in Mind, Brain, and Behavior. MIT Press, Cambridge (2022)
12. Taniguchi, T., et al.: World models and predictive coding for cognitive and developmental robotics: frontiers and challenges. *Adv. Robot.* 1–27 (2023)



Awareness in Robotics: An Early Perspective from the Viewpoint of the EIC Pathfinder Challenge

“Awareness Inside”

Cosimo Della Santina^{1,2(✉)}, Carlos Hernandez Corbato¹, Burak Sisman¹, Luis A. Leiva³, Ioannis Arapakis⁴, Michalis Vakalellis⁵, Jean Vanderdonckt⁶, Luis Fernando D’Haro⁷, Guido Manzi⁸, Cristina Becchio⁹, Aïda Elamrani¹⁰, Mohsen Alirezaei¹¹, Ginevra Castellano¹², Dimos V. Dimarogonas¹³, Arabinda Ghosh¹⁴, Sofie Haesaert¹⁵, Sadegh Soudjani¹⁴, Sybert Stroeve¹⁶, Paul Verschure¹⁷, Davide Bacciu¹⁸, Ophelia Deroy¹⁹, Bahador Bahrami¹⁹, Claudio Gallicchio¹⁸, Sabine Hauert²⁰, Ricardo Sanz²¹, Pablo Lanillos^{17,22}, Giovanni Iacca²³, Stephan Sigg²⁴, Manel Gasulla²⁵, Luc Steels²⁶, and Carles Sierra²⁷

¹ Department of Cognitive Robotics, Delft University of Technology, Delft, The Netherlands

² Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Oberpfaffenhofen, Germany
c.dellasantina@tudelft.nl

³ Department of Computer Science, University of Luxembourg, Esch-sur-Alzette, Luxembourg

⁴ Telefónica I+D, Madrid, Spain

⁵ AEGIS IT Research GmbH, Braunschweig, Germany

⁶ Université catholique de Louvain, Ottignies-Louvain-la-Neuve, Belgium

⁷ Speech Technology and Machine Learning Group, ETSI de Telecomunicación, Universidad Politécnica de Madrid, Madrid, Spain

⁸ Alien Venture Studio, Alien Technology Transfer, Rome, Italy

⁹ Department of Neurology, University Medical Center Hamburg-Eppendorf (UKE), Hamburg, Germany

¹⁰ Institut Jean Nicod, École Normale Supérieure, Paris, France

¹¹ RTD Department, Siemens Digital Industry Software, Helmond, The Netherlands

¹² Department of Information Technology, Uppsala University, Uppsala, Sweden

¹³ Division of Decision and Control Systems, School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden

¹⁴ Max Planck Institute for Software Systems, Kaiserslautern, Germany

¹⁵ Department of Electrical Engineering, Control Systems Group, Eindhoven University of Technology, Eindhoven, The Netherlands

¹⁶ Netherlands Aerospace Centre (NLR), Amsterdam, The Netherlands

¹⁷ Donders Centre for Neuroscience, Radboud University, Nijmegen, The Netherlands

¹⁸ Computer Science Department, University of Pisa, Pisa, Italy

¹⁹ Faculty of Philosophy, Philosophy of Science and the Study of Religion, Ludwig Maximilian University of Munich, Munich, Germany

²⁰ School of Engineering Mathematics and Technology, University of Bristol, Bristol, UK

²¹ Autonomous Systems Laboratory, Universidad Politécnica de Madrid, Madrid, Spain

²² Cajal International Neuroscience Center (CINC), Spanish National Research Council (CSIC), Madrid, Spain

²³ Department of Information Engineering and Computer Science, University of Trento, Trento, Italy

²⁴ Department of Information and Communications Engineering, Aalto University, Espoo, Finland

²⁵ Department of Electronic Engineering, Universitat Politècnica de Catalunya, Barcelona, Spain

²⁶ Studio Stelluti, Brussels, Belgium

²⁷ Artificial Intelligence Research Institute (IIIA), Spanish National Research Council (CSIC), Barcelona, Spain

Abstract. While consciousness has been historically a heavily debated topic, awareness had less success in raising the interest of scholars. However, more and more researchers are getting interested in answering questions concerning what awareness is and how it can be artificially generated. The landscape is rapidly evolving, with multiple voices and interpretations of the concept being conceived and techniques being developed. The goal of this paper is to summarize and discuss the ones among these voices connected with projects funded by the EIC Pathfinder Challenge “Awareness Inside” call within Horizon Europe, designed specifically for fostering research on natural and synthetic awareness. In this perspective, we dedicate special attention to challenges and promises of applying synthetic awareness in robotics, as the development of mature techniques in this new field is expected to have a special impact on generating more capable and trustworthy embodied systems.

Keywords: Awareness · Robotics · Autonomous Systems · Living Machines · Artificial Consciousness

1 Introduction

According to the Cambridge English Dictionary [1], awareness is the “*knowledge that something exists, or understanding of a situation or subject at the present time based on information or experience*”. It may seem that, according to this broad definition, awareness is already widespread in robotics. Think, for example, of a drone creating an internal representation of its environment using a SLAM algorithm [10], a manipulator detecting an interaction with its environment through an observer [15], or a social robot acquiring knowledge on the logical structure of its environment [3] or its body [9]. If we accept this broad definition, it is evident that a common framework that enables the discussion

of levels of awareness across domains is missing. Even more, the concept may be loaded with more complex significance than this interpretation suggests. For example, *awareness* is sometimes semantically associated with the term *consciousness*, which is a much more debated concept. Researchers have proposed several definitions of the latter, including philosophical [14], psychological [8], architectural [5], neural correlates [4], and computer science theory [2].

As Artificial Intelligence (AI) progresses in reproducing a growing number of human capabilities [7, 11, 13], it is thus natural to ask if and how synthetic awareness can be produced and embodied into physical agents. Ultimately, questions that need answering include: What is awareness exactly? Can this concept be formalized philosophically or technically? What distinguishes an aware agent from a non-aware one? Does awareness improve the performance of robots? Is it even ethical to endow artificial agents with awareness?

In this vein, in June 2021, the European Innovation Council (EIC) opened a Pathfinder Challenge call for proposals with the explicit aim of pushing the boundaries of AI [6]. The call challenged the scientific community to define and achieve true awareness and understand awareness beyond the perception of surroundings or self-awareness. The call also pointed towards the role that awareness could play in both clinical settings and technology, also given the importance that trustworthy AI has in this field for the EU roadmap. The call suggested that robots or decision support systems that appear ‘aware’ could play a role in gaining human trust.

Eight projects were selected among the applicants, covering a large portion of the European research landscape. This paper aims to introduce this multi-faceted view to awareness research and provide some preliminary discussion on how these activities will impact the robotics field in the future. In the rest of the manuscript, we will review how each project looks at the challenge. In doing that, we observe that the views on awareness vary from project to project. At the same time, we show that—although not all of them are intended to generate direct applications for robotics—they all have the potential to generate disruptive innovation in this field.

2 A Bird View on the Projects’ Goals¹

SYMBIOTIK introduces a user-centric approach to information visualization, leveraging awareness and emotion-sensing to enhance human-machine cooperation for improved decision-making. The project aims to augment cognitive functions and create more natural, user-friendly interfaces.

ASTOUND is developing an AI architecture for artificial consciousness based on the Attention Schema Theory. The goal is to enhance natural language understanding in virtual agents by creating a model miming human awareness of attention states - with an expected impact in collaborative robots (cobots).

SymAware develops a conceptual framework for situational awareness in multi-agent systems, focusing on formal modeling and logical specifications, social

¹ More thorough descriptions are available in the full-length manuscript [12].

learning, risk quantification, and task negotiation. SymAware aims to improve coordination and functionality in dynamic environments, targeting autonomous vehicles and traffic management for unmanned aircraft systems.

CAVAA explores the concept that awareness enhances survival by enabling entities to interact with the unseen aspects of their environment, proposing a model where awareness is defined as a detailed experience of a consciously perceived virtual world. CAVAA aims to replicate this model of awareness in robots and artificial agents.

EMERGE seeks to establish collaborative awareness among minimal artificial agents, focusing on how simple agents can collectively develop an awareness of their existence, environment, and cooperation for shared goals. The project will enable collaboration and self-regulation in cyber-physical systems - especially focusing on modular soft robots, robotic swarms, and cobots.

METATOOL combines archaeology, neuroscience, and robotics to explore how robots can mimic ancient human cognitive advancements in tool invention. The project seeks to equip robots with an advanced understanding and creation of tools through self-awareness and operationalizable abstraction.

SUST(AI)N aims to enhance smart buildings with distributed, precision-sensing AI for energy autonomy and intelligent system integration. This approach will leverage probabilistic reasoning, optimizing the interaction between sensors, data, and node configurations - with potential applications in swarm robotics, multi-agent systems, and energy-efficient autonomous robots.

VALAWAI focuses on developing AI tools that adhere to human values, aiming to create value-aware AI applications capable of making or recommending explainable decisions based on ethical guidelines. This effort is crucial for ensuring the ethical deployment of AI, with applications ranging from medical protocols to social robots.

3 Discussion and Conclusions

This paper summarized ongoing and diverse activities around synthetic awareness, driven by the EIC Pathfinder Challenge “Awareness Inside” call funded by the European Union through the Horizon Europe program. This diversity enriches our understanding and poses significant challenges in defining and achieving a unifying concept of awareness applicable across different robotic applications. For example, the very definition of awareness varies strongly from project to project, with its position towards consciousness ranging from direct opposition (EMERGE) to almost a synonym of hard consciousness (ASTOUND). While in the context of SymAware, awareness is emphasized as the ability of agents in a multi-agent setting to recognize and comprehend external stimuli, especially in scenarios involving the presence of human beings, METATOOL awareness focuses on the neuroscience-derived idea of uncertainty monitoring. In both projects, awareness involves perceiving, adapting, coordinating, communicating, and making informed decisions while ensuring safety and resilience in a dynamic environment.

Of the eight projects, five are specifically focused on robotics-related experiments. Namely, METATOOL investigates robots capable of self-evaluation and tool invention; SymAware holds relevance for various applications, encompassing both industrial and domestic robotic systems, with a specific emphasis on autonomous vehicles and air-traffic control systems; CAVAA utilizes social robots like MiRo-e for its tests; EMERGE explores various robotic applications; and VALAWAI showcases its ideas through domestic social robots. Interestingly, the authors of this paper who are involved in the remaining three projects also believe in applying their innovation to robotics. More specifically, the SYMBIOTIK and ASTOUND innovation could transform social robotics. The SYMBIOTIK project could enhance human-robot interactions by integrating awareness and emotion-sensing capabilities into robotics, making these interactions more natural and user-friendly. Developing artificial consciousness based on the Attention Schema Theory, the ASTOUND project will enhance the performance of natural language understanding and provide collaborative robots with the capability to effectively and safely interact with human partners by predicting their intentions and objectives. Finally, in line with the goals of EMERGE, the SUST(AI)N project could advance collaborative and swarm robotics, as well as multi-agent systems, by developing a distributed form of self-aware intelligence, enhancing system performance, reliability, and adaptation through a hierarchical learning model and energy-efficient, self-sufficient technologies.

Ultimately, all the authors of this paper believe that incorporating awareness components in technology could make systems more resilient, adaptable, and human-centric. We believe that understanding awareness will allow AI systems to better grasp and respond to various situations. Synthetic awareness will advance AI towards more coherent, adaptive, and self-evolving behavior.

Acknowledgements. This work is supported by the EU EIC projects: SYMBIOTIK (grant 101071147), ASTOUND (grant 101071191), SymAware (grant 101070802), CAVAA (grant 10039052), EMERGE (grant 101070918), METATOOL (grant 101070940), SUST(AI)N (grant 101071179), and VALAWAI (grant 101070930).

References

1. Awareness - Cambridge dictionary (2024). <https://dictionary.cambridge.org/dictionary/english/awareness>. Accessed 12 Jan 2024
2. Blum, M., Blum, L.: A theoretical computer science perspective on consciousness. *J. Artif. Intell. Conscious.* **8**(01), 1–42 (2021)
3. Bruno, B., et al.: Knowledge representation for culturally competent personal robots: requirements, design principles, implementation, and assessment. *Int. J. Soc. Robot.* **11**, 515–538 (2019)
4. Crick, F., Koch, C.: A framework for consciousness. *Nat. Neurosci.* **6**(2), 119–126 (2003)
5. Dehaene, S., Lau, H., Kouider, S.: What is consciousness, and could machines have it? *Science* **358**(6362) (2017)
6. EIC pathfinder challenge: Awareness inside (2024). <https://eic.ec.europa.eu/eic-funding-opportunities/calls-proposals/eic-pathfinder-challenge-awareness-inside-en>. Accessed 12 Jan 2024

7. Epstein, Z., et al.: Art and the science of generative AI. *Science* **380**(6650), 1110–1111 (2023)
8. James, W.: *The Principles of Psychology*, vol. 1. Cosimo, Inc. (2007)
9. Lanillos, P., Pages, J., Cheng, G.: Robot self/other distinction: active inference meets neural networks learning in a mirror. In: ECAI (2020)
10. Placed, J.A., et al.: A survey on active simultaneous localization and mapping: state of the art and new frontiers. *IEEE Trans. Robot.* (2023)
11. Salvagno, M., Taccone, F.S., Gerli, A.G., et al.: Can artificial intelligence help for scientific writing? *J. Crit. Care* **27**(1) (2023)
12. Santina, C.D., et al.: Awareness in robotics: an early perspective from the viewpoint of the EIC pathfinder challenge “awareness inside”. [arXiv:2402.09030](https://arxiv.org/abs/2402.09030) (2024)
13. Silver, D., et al.: A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science* **362**(6419) (2018)
14. Tononi, G., Koch, C.: Consciousness: here, there and everywhere? *Philos. Trans. R. Soc. Lond. B Biol. Sci.* **370**(1668) (2015)
15. Zacharaki, A., Kostavelis, I., Gasteratos, A., Dokas, I.: Safety bounds in human robot interaction: a survey. *Saf. Sci.* **127**, 104667 (2020)



In-Context Interference In Chat-Based Large Language Models

Eric Nuertey Coleman¹, Julio Hurtado^{2(✉)}, and Vincenzo Lomonaco^{1(✉)}

¹ University of Pisa, Pisa, Italy

eric.coleman@phd.unipi.it

² University of Warwick, Coventry, UK

Abstract. Large Language Models (LLMs) have transformed society, but modifying their internal knowledge remains challenging. Here, we focus on interference in in-context learning, examining how new knowledge affects performance in self-aware robots. We propose an evaluation benchmark based on the bAbI dataset to assess the robot’s ability to manage interference, maintain stability, ensure flexible information routing, and facilitate task performance. Addressing these challenges is crucial for improving LLMs’ effectiveness in developing self-aware robots.

1 Introduction

Chat-based applications based on LLMs have gained attention for their versatile capabilities across various tasks, including medical question answering and patient-trial matching in the medical field [18]. In robotics, LLMs are increasingly integrated into autonomous industrial robotics and task planning, playing a crucial role in decision-making, task execution, and autonomous design within manufacturing contexts [7, 16].

While LLMs excel at understanding and engaging in conversations, modifying their internal knowledge is still challenging. Most approaches rely on adding new information via prompt engineering due to the inability to modify private models’ weight (e.g. ChatGPT). This approach is known as In-context learning, and this paper extends the discussion to a comprehensive understanding of their limitations and strengths. We focus on studying the model’s behavior by presenting a new benchmark based on the bAbI dataset, shedding light on potential limitations and enhancing overall performance. The problems presented in this work will facilitate the development of more robust and efficient language models, especially for a reliable interaction between users and self-aware robots, particularly in chat-based applications.

Our main contributions can be summarized as follows: (1) Proposing a benchmark to assess the accumulation and retention of information in LLMs, addressing their capabilities and limitations in in-context learning. (2) This benchmark reveals evidence of interference within the same chat session, impacting performance as new information is added, which is critical when relying on a continual stream of information. (3) We provide insights into the current ability of these models to learn, retain, and reason on in-context scenarios.

2 Related Works

In-Context Learning. Recent research highlights the significance of in-context learning in chat-based LLMs [8]. Meta-in-context learning is explored for recursive improvement across various tasks through meta-learning [4]. Innovative methods like LAMOC are used for knowledge-based visual question answering [6]. MCL-KBQA framework utilizes in-context learning for knowledge base question answering in low-resource settings [13]. Long-term reasoning for chat-based LLMs is challenging [15] and TempReason dataset is used to enhance LLMs' temporal reasoning capability [14].

Continual Learning in LLMs. Most methods in Continual Learning (CL) that works with LLMs, focus on training soft prompts [12] or adapters [10], modifying model weights for specific tasks. In contrast, our approach examines the influence CL in a in-context approach.

LLMs In Self-aware Robots. LLMs can enhance autonomous robots by providing natural language understanding, reasoning, and generation. Researchers have proposed a unified system where a large language model can summarize and answer questions about a robot's activities [5]. Researchers have also introduced CODEBOTLER [9], an open-source tool for programming robots through natural language. A benchmark called ROBOEVAL can assess LLMs in generating accurate sequences of robot actions and provide insights into common pitfalls and failures.

3 In-Session Evaluation

Typically, users engage with chat-based LLMs in closed sessions, treating the model as a black-box. Interaction is limited to prompts via the chat interface, expecting flawless recall of prior exchanges. We propose testing knowledge retrieval by gradually increasing stored information. Starting from knowledge s_0 , we ask the LLM questions q_0 associated with input s_0 , obtaining performance per_0 . At time step i , we will add input s_i to the context and ask all questions $q_{\leq i}$, meaning questions from input s_0 to s_i , to obtain performance per_i . If the model does not suffer from interference and flawlessly recalls prior knowledge, one should expect that performance should stay the same as we add new knowledge.

3.1 Benchmark

To evaluate knowledge retrieval and study the interference, we propose a benchmark based on a subset of the bAbI dataset [17]. This sequence of stories provided a stream of knowledge that the model must remember to answer the questions without depending on prior information correctly. Here, we evaluate only the retention capacity of the information delivered in the context.

We simplified the dataset by renaming entities and retained only the last two statements, along with a question about the final entity. This approach helped us evaluate the model’s context retention and question-answering accuracy. One limitation lies in the token limit, considering both computational capacity and the context size for optimal model performance. We selected 50 stories as a hyper-parameter, but this can be adjusted as the token limit of LLMs increases.

4 Experiments

We use the Vicuna model [3] within the LangChain [2] framework to evaluate the previous scenario. Due to computational limitations, we set the maximum number of tokens to 2048. Because the answer we are looking at is only one word long, to reduce the probability of longer answers, we empirically found that a temperature of 0.7 provides the best performance overall. Following previous work of in-context learning, we teach the model with prompt engineering.

4.1 One Context-One Story

As a baseline, we verify the model’s ability to correctly answer questions in a one-story-per-context scenario, removing possible interference within the context. The accuracy when using the whole story is 58%, as observed in our experiments. This low accuracy and the high number of tokens per story encourage us to propose the simplified version of the known dataset mentioned in the previous section. When testing the performance of the simplified version, the model obtained a 100%, probing that the complexity of the original dataset could interfere with the conclusions drawn.

4.2 Incremental Stories

Typically, users assume that LLMs accumulate information without interference with the previous context. We hypothesise that introducing new information can lead to interference, which can cause a decrease in performance. Recognizing this limitation allows us to study and propose methods to minimize training and token limitations.

As shown in Fig. 1a, as we add new knowledge to the prompts, the performance of the model decreases from 100% with only one story to around 75% when we have 8 in the same context. A similar effect appears when using the original stories, where the model shows a decrease in performance.

Due to the limitations of LLMs, we cannot significantly increase the number of stories since the number of tokens in the model is truncated during training. Some studies have shown that it is possible to increase the number of tokens [1] with little performance loss; however, this increases the computational cost. We also observed in our experiments that as we increase the size of the prompt (# of stories and length), the cost of delivering a response increases.

4.3 Summarizing

The interference between old and new information is not something new. The problem known as Catastrophic Forgetting (CF) [11] is precisely caused by the continual weight modification when training a Deep Learning model. However, it is a different process than the one presented here, where the model's weights are not modified. In this case, the interference is at the information level, and the model is unable to identify the relevant parts of all the information delivered correctly.

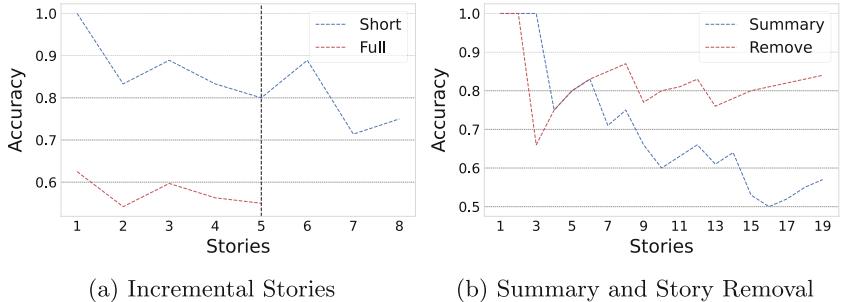


Fig. 1. Figure (a) shows the accuracy obtained when adding new knowledge to the context using short and full stories. Figure (b) shows the accuracy when removing or summarizing short stories stored in the context.

Like the CF problem, we need to minimize interference in the model's accumulated context. One approach is using built-in summarizing tools to reduce stored tokens while retaining crucial knowledge in the buffer. This allows the model to compress information, diminishing interference with unnecessary details and enhancing performance. However, performance may decline during information summarization, as illustrated in the Fig. 1b.

One theory of the above is that the model may interfere with the responses it has delivered (which are part of the context). For this reason, instead of summarizing the previous information, we deleted old stories and only kept 6 in the context. By removing old stories, we remove the option for the model to change its response (for better or worse), but more importantly, we are reducing the amount of tokens that cause interference. As shown in Fig. 1b, the model is able to preserve most of the performance even with a high number of stories.

5 Conclusion

This paper introduces a novel benchmark to assess how chat-based LLMs are impacted by interference in in-context learning. We proposed a bAbI dataset-based benchmark to evaluate the models' ability to accumulate information in

the context. Our findings reveal that introducing new information causes interference, negatively affecting model performance in user-uncontrolled scenarios. While this study addresses the limitations of in-context learning, it lays the groundwork for further exploration, including issues like interference between context-based and pre-trained knowledge in models.

Acknowledgements. This research was supported by Leonardo Labs and the EMERGE Project (Grant Agreement ID: 101070918). We extend our gratitude to both entities for their generous support, feedback, and guidance, significantly contributing to the success of this project.

References

1. Bulatov, A., Kuratov, Y., Burtsev, M.S.: Scaling transformer to 1m tokens and beyond with RMT. arXiv preprint [arXiv:2304.11062](https://arxiv.org/abs/2304.11062) (2023)
2. Chase, H.: Langchain (2022). <https://github.com/hwchase17/langchain>
3. Chiang, W.L., et al.: Vicuna: an open-source chatbot impressing GPT-4 with 90%* ChatGPT quality (2023). <https://lmsys.org/blog/2023-03-30-vicuna/>
4. Coda-Forno, J., Binz, M., Akata, Z., Botvinick, M., Wang, J.X., Schulz, E.: Meta-in-context learning in large language models (2023)
5. DeChant, C., Akinola, I., Bauer, D.: Learning to summarize and answer questions about a virtual robot's past actions (2023)
6. Du, Y., Li, J., Tang, T., Zhao, W.X., Wen, J.R.: Zero-shot visual question answering with language model feedback (2023)
7. Fan, H., Liu, X., Fuh, J., Lu, W.F., Li, B.: Embodied intelligence in manufacturing: leveraging large language models for autonomous industrial robotics. *J. Intell. Manuf.* 1–17 (2024). <https://doi.org/10.1007/s10845-023-02294-y>
8. Garg, S., Tsipras, D., Liang, P., Valiant, G.: What can transformers learn in-context? A case study of simple function classes (2023)
9. Hu, Z., et al.: Deploying and evaluating LLMs to program service mobile robots (2023)
10. Ke, Z., Shao, Y., Lin, H., Konishi, T., Kim, G., Liu, B.: Continual pre-training of language models. In: *The Eleventh International Conference on Learning Representations* (2023)
11. McCloskey, M., Cohen, N.J.: Catastrophic interference in connectionist networks: the sequential learning problem. In: *Psychology of Learning and Motivation*, vol. 24, pp. 109–165. Elsevier (1989)
12. Razdaibiedina, A., Mao, Y., Hou, R., Khabsa, M., Lewis, M., Almahairi, A.: Progressive prompts: continual learning for language models (2023)
13. Tan, C., Chen, Y., Shao, W., Chen, W.: Make a choice! Knowledge base question answering with in-context learning (2023)
14. Tan, Q., Ng, H.T., Bing, L.: Towards benchmarking and improving the temporal reasoning capability of large language models (2023)
15. Tan, Y., et al.: Evaluation of ChatGPT as a question answering system for answering complex questions (2023)
16. Wang, J., et al.: Large language models for robotics: opportunities, challenges, and perspectives (2024)

17. Weston, J., Bordes, A., Chopra, S., Mikolov, T.: Towards AI-complete question answering: a set of prerequisite toy tasks. In: Bengio, Y., LeCun, Y. (eds.) 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, 2–4 May 2016, Conference Track Proceedings (2016). <http://arxiv.org/abs/1502.05698>
18. Yuan, J., Tang, R., Jiang, X., Hu, X.: Large language models for healthcare data augmentation: an example on patient-trial matching (2023)



MIDGARD: A Robot Navigation Simulator for Outdoor Unstructured Environments

Giuseppe Vecchio¹, Riccardo E. Sarpietro¹(✉), Francesco Cancelliere¹,
Simone Palazzo¹, Dario C. Guastella¹, Alessandro Strano¹, Ignacio Carlucho²,
Giovanni Muscato¹, Stefano V. Albrecht³, and Concetto Spampinato¹

¹ Department Electric, Electronic and Computer Engineering, University of Catania,
Catania, Italy

riccardo.sarpietro@phd.unict.it

² School of Engineering and Physical Sciences, Heriot-Watt University,
Edinburgh, UK

³ School of Informatics, University of Edinburgh, Edinburgh, UK

Abstract. We present MIDGARD, a simulation platform based on Unreal Engine for training autonomous robots in complex outdoor unstructured environments. It offers photorealistic 3D scenes, procedural scene generation, and integration with ROS and OpenAI Gym. The focus of MIDGARD is on navigation, where an autonomous agent travels from random initial positions to designated target locations avoiding obstacles, enabling researchers to develop and evaluate novel algorithms and navigation methods. We evaluate MIDGARD’s suitability as a research tool by training navigation algorithms based on reinforcement learning; we also assess sim-to-real transfer capabilities in a traversable horizon prediction task, using deep learning models on RGB images only.

MIDGARD builds and docs are available at www.midgarsim.org.

Keywords: simulation · robot navigation · unstructured environments

1 Introduction

Autonomous ground robot navigation in challenging outdoor and unstructured environments presents significant challenges. While learning-based control models have succeeded in structured settings, they struggle when deployed in the wild. One of the main challenges lies in acquiring extensive data across various conditions, from the environment and from the agent, which are vital for training deep learning models. Real-world training is hindered by high costs, vehicle risks, slow training, and limited scene variety, underscoring the necessity for simulation environments. Indeed, simulation has long been employed in the study of autonomous navigation by reducing costs, risks, and training time.

In this work, we present MIDGARD¹, a simulation platform based on Unreal Engine for training autonomous robots in complex outdoor unstructured environments. Harnessing the high photorealism provided by Unreal Engine, MIDGARD overcomes several limitations of state-of-the-art simulators, introducing procedural scene generation for increased environment variability, allowing for configurability and extensibility through HTTP APIs, and supporting integration with ROS and OpenAI Gym.

2 Related Work

Recent years have seen the emergence of high-quality simulation engines: some focus on structured environments (e.g., CARLA [1]), while others simulate the physical behavior of vehicle (e.g., BeamNG [2]).

CARLA (Car Learning to Act) [1] is designed for research in autonomous driving for urban environments. The simulator presents unique challenges due to complex traffic dynamics and supports a flexible configuration of an agent’s sensor suite, including RGB cameras and pseudo-sensors for depth and semantic segmentation.

BeamNG [2] is a physics-based vehicle simulation platform that focuses on simulating the physical behavior of vehicles and their interactions with the environment. The core of BeamNG’s methodology includes soft-body physics and set of sensors, which models how vehicles deform and react to impacts in a way that mirrors real-world physics, terrain friction, collision and damage simulation when vehicles crash.

Although, CARLA and BeamNG are customizable via built-in tools (e.g., custom vehicles or maps), they are not tailored for unstructured outdoor environments and require computer graphics and 3D skills by users. MIDGARD fills this gap by allowing the generation of outdoor unstructured environments within a range of complexity without any effort.

3 MIDGARD

MIDGARD is a simulation platform tailored for the navigation of autonomous robot within complex, outdoor and unstructured photorealistic environments. It features pre-built scenes and extends support for procedural scene generation with customizable levels of complexity. Furthermore, it offers an HTTP endpoint to extend the customization (e.g., with ROS [3] and OpenAI Gym [4]) to support the integration and customization of novel reinforcement learning algorithms, navigation methods and frameworks [5].

An overview of MIDGARD’s architecture is provided in Fig. 1.

MIDGARD ensures environment variability at training time through a procedural scene generation process, which simply requires the specification of scene type and obstacle density. Four scene types are built-in into the simulator,

¹ “Kingdom of mankind”, i.e., Earth, in Norse mythology.

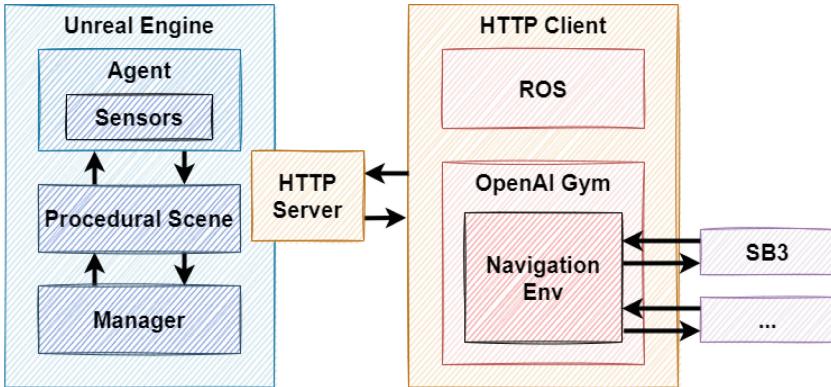


Fig. 1. MIDGARD architecture.

each with its own distinctive characteristics for procedural generation (Fig. 2): (a) *Meadow* offers easy-to-medium complexity with small/medium rocks and bushes while (b) *Forest* scene increases the complexity with trees, branches and puddles; (c) *Volcanic* and (d) *Arctic* have medium-to-high complexity, mostly rocky masses with challenging obstacles' identification due to uniform coloring, making navigation harder.

Scenes can further be customized by specifying rendering parameters, image resolution and image quality (ranging from “Low” to “Cinematic”), camera types, time of the day, and weather conditions.

As its main purpose is to support the design and development of navigation algorithms, MIDGARD provides robot agents with a variety of vision and low level sensors specifically engineered for navigation purposes that can be also used for training reinforcement learning agents (e.g., reward computation, end of episode, target reached, and so on):

- **Vision sensors** (Fig. 3): as (a) on-board cameras that capture raw RGB pixel data, (b) depth information and (c) semantic segmentation data.
- **Low-level sensors:** are vector-based information concerning the agent and environment state measurements, such as spatial information about the robot and the target positions (i.e., x, y, and z coordinates), the Euclidean distance to the target destination, the robot inertial measurements (brake status, linear and angular velocity), LIDAR measurements, and collision detection.

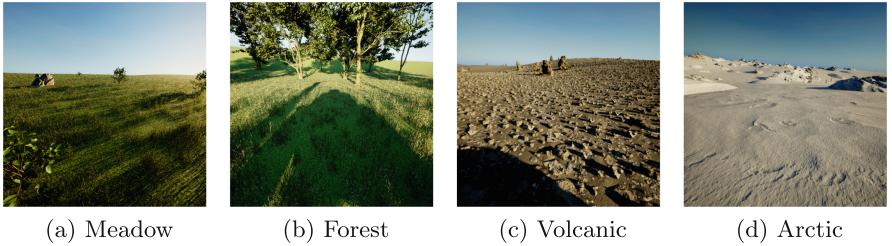


Fig. 2. Pre-built different types of scenes for procedural generation.

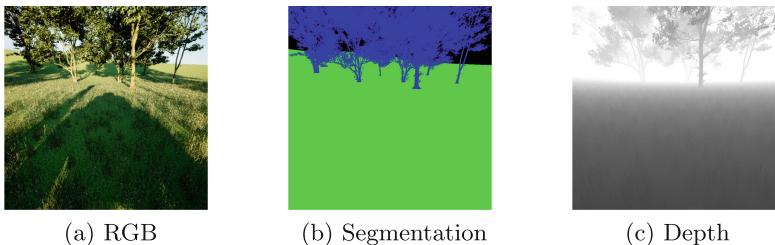


Fig. 3. MIDGARD vision sensors.

4 Experimental Results

We evaluate the suitability of MIDGARD as a framework for developing robot navigation methods by training state-of-the-art reinforcement learning approaches—PPO [6] and SAC [7]²—on *Forest* and *Volcanic* scenes measuring the success rate of the agent in reaching the target. Results from Table 1 show that MIDGARD provides a challenging benchmark to state-of-the-art approaches; the decrease in performance as the distance to target and obstacle density increase (Fig. 4) indicates the effectiveness of procedural scene generation in controlling the task’s difficulty.

² The agent has been trained for 2,500 steps with an update of 80 epochs at each step. The initial learning rate for the actor was 0.0003 and 0.001 for the critic.

Table 1. Success rate at different obstacle density levels and distances from target.

| Scene | Method | Distance | Obstacle density | | |
|----------|--------|----------|------------------|-----|-----|
| | | | 0.1 | 0.5 | 1.0 |
| Forest | PPO | 10 m | 54% | 42% | 23% |
| | | 20 m | 41% | 34% | 15% |
| | SAC | 10 m | 62% | 51% | 37% |
| | | 20 m | 53% | 39% | 27% |
| Volcanic | PPO | 10 m | 48% | 37% | 17% |
| | | 20 m | 38% | 29% | 15% |
| | SAC | 10 m | 60% | 49% | 33% |
| | | 20 m | 48% | 37% | 22% |



(a) Density 0.1

(b) Density 0.5

(c) Density 1.0

Fig. 4. Instances of Forest scene at different obstacles density level.

Furthermore, MIDGARD’s sim-to-real transfer capabilities have been evaluated in real-world in a traversable horizon prediction task [8], by estimating the traversability scores for vertical bands within observed RGB frames. To train the network, authors used a combination of supervised regression loss, with synthetic annotated images from MIDGARD and an unsupervised domain adaptation loss to improve its generalization on real scenes. The traversability predictions are then used to generate velocity commands for the real-world vehicle navigation. The vehicle is directed towards the band with the highest traversability score, and its forward velocity is determined by this score. The angular velocity depends on the relative position of the selected band with respect to the middle band. In case the traversability score falls below a critical value, the vehicle enters a recovery mode, slowly rotating in place while seeking alternative paths. Experimental tests of this approach are performed on a tracked vehicle equipped with a camera and GPU-based computing unit. The experiments demonstrate that the model, trained on simulated MIDGARD data, successfully identifies traversable areas in real-world scenarios, confirming the sim-to-real transfer potential (Fig. 5).



(a) Real capture

(b) Prediction

Fig. 5. Traversability On-field (preediction of non-traversable area in blue).

5 Conclusions

In this work, we presented MIDGARD, a photorealistic training platform, based on Unreal Engine, for supporting research in autonomous navigation in outdoor unstructured environments. It offers a unique procedural scene generation with configurable difficulty, promoting generalization in learning agents feautirng ROS and OpenAI Gym integration and providing versatility.

MIDGARD demonstrated to be suitable to evaluate reinforcement learning algorithms and effectiveness of the traversability in real-world navigation scenario. Future plans include multi-agent support, various vehicle types, improved agent-terrain interaction, and human-in-the-loop control.

Acknowledgement. G. Muscato, C. Spampinato and F. Cancelliere acknowledge financial support from PNRR MUR project PE0000013-FAIR.

D. C. Guastella acknowledges support by the project PON R&I REACT-EU.

References

1. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: Carla: an open urban driving simulator. [arXiv:1711.03938](https://arxiv.org/abs/1711.03938) (2017)
2. BeamNG GmbH. BeamNG.tech. <https://www.beamng.tech/>
3. Thomas, D., Woodall, W., Fernandez, E.: Next-generation ROS: building on DDS. In: ROSCon Chicago 2014. Open Robotics, Mountain View (2014)
4. Brockman, G., et al.: OpenAI gym. [arXiv:1606.01540](https://arxiv.org/abs/1606.01540) (2016)
5. Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., Dormann, N.: Stable-baselines3: reliable reinforcement learning implementations. J. Mach. Learn. Res. **22**(268), 1–8 (2021)
6. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. [arXiv:1707.06347](https://arxiv.org/abs/1707.06347) (2017)
7. Haarnoja, T., Zhou, A., Abbeel, P., Levine, S.: Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor. In: ICML (2018)
8. Palazzo, S., et al.: Domain adaptation for outdoor robot traversability estimation from RGB data with safety-preserving loss. In: IEEE IROS (2020)



Beluga: A Modern Monte Carlo Localization Package for ROS and ROS 2

Gerardo Puga^(✉), Nahuel Espinosa, Michel Hidalgo, Olmer García,
and Ivan Paunovic

Ekumen, Buenos Aires, Argentina

glpuga@ekumenlabs.com

<https://www.ekumenlabs.com/>

Abstract. We present *beluga_amcl*, a novel Robot Operating System (ROS) localization package that implements Adaptive Monte Carlo localization (AMCL), one of the most widely used algorithms for map-based localization in the ROS navigation stack.

The *beluga_amcl* package is fully compatible with the long-known and widely used *nav2_amcl* package, but stands out in that it was designed from the ground-up to be high-quality, production grade open-source software, in line with current expectations of the ROS 2 community.

In this paper we demonstrate that *beluga_amcl* is a robust and performant implementation of the features in *nav2_amcl*, and that it can thus be considered a suitable functional replacement for it. This provides the ROS community with a modern, high-quality, and backwards-compatible MCL package that can be a basis for further extension.

Keywords: ros · monte carlo · particle filter · localization · robotics

1 Introduction

Robot Operating System (ROS) is a framework for robot software development with a long history and a very large community of users. There have been two major revisions: ROS 1 (frequently referred to as just ROS), which targeted robotics research and rapid prototyping, and ROS 2, a ground-up redesign to also address the needs of professional robotics practitioners [3].

The *nav2_amcl* map-based localization package included in ROS is one of the most widely used in mobile robots. It implements the Adaptive Monte Carlo localization algorithm (AMCL) [5] that gives it its name. It has been available in ROS almost since its inception, but it has barely changed over the years: the ROS 2 version is mostly just a port of the ROS 1 package, which was in turn strongly based on a much older software called Player [4].

Being such an early component, the *nav2_amcl* codebase is not in line with the quality standards currently expected by the ROS 2 community for such a key component of the navigation stack. Even though alternative MCL implementations have been developed over the years, they have not been widely adopted by the community (e.g. *QuickMCL* and *gmcl*).

In this paper we present *beluga_amcl*, a novel implementation of map-based AMCL for ROS and ROS 2 that has been designed from the ground-up to be a modern, production-grade open-source localization package. *beluga_amcl* is fully compatible with the existing *nav2_amcl* package, ensuring a straightforward migration path for existing users. Also, the modular nature of the core Beluga library enables the creation of novel localization algorithms and extensions in a much simpler way than previously possible.

In the following sections we describe the design of the *beluga_amcl* localization package; we summarize the code quality strategy and we present performance evaluation results. Finally we discuss future lines of work to further extend the feature-set available to users.

2 Methodology

2.1 Design

The *beluga_amcl* localization package is composed of two layers. The outer layer is responsible for implementing the ROS interfaces, while the inner, core layer is based on the ROS-agnostic Beluga particle filter library that gives the package its name.

This layered architecture allows the creation of both ROS 1 and ROS 2 versions of the package that fully share the same core particle filter with different external interfaces, and even non-ROS projects can benefit from the core particle filter library without having to commit to including any ROS dependencies.

2.2 Features

To provide a migration path for existing *nav2_amcl* users, *beluga_amcl* aims at providing 1-to-1 feature equivalence with it. An up-to-date compatibility table can be found in the project documentation [1].

2.3 Quality Practices

nav2_amcl is a mature and time-tested package; to systematically guarantee the same level of reliability, *beluga_amcl* enforces the Code Quality Practices recommended in the *ROS 2 Developer Guide* [2]. Table 1 summarizes the currently implemented quality practices.

Table 1. Summary of the different quality practices in the *beluga_amcl* package.

| Strategy | Notes |
|-------------------------------|---|
| Linters & static analysis | C++, Python and cmake linters enforcing quality in CI |
| Unit testing | 95% code test coverage, enforced in CI |
| System testing | Regression testing of MCL and AMCL filter instances |
| Performance characterization | Measurement of APE, RSS, CPU load and latency |
| Simulated world evaluation | Accuracy and robustness against unmapped obstacles |
| Benchmarking against datasets | Accuracy against public Magazzino SLAM dataset |

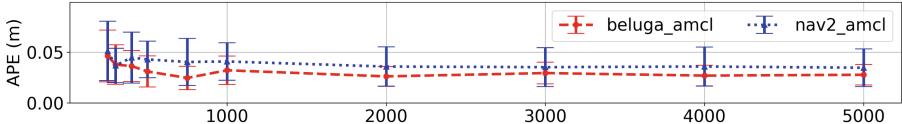


Fig. 1. Mean APE vs particle count with likelihood field model. Error bars at $\pm\sigma$.

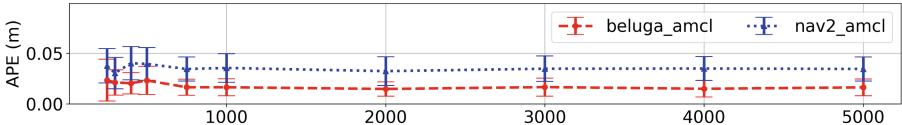


Fig. 2. Mean APE vs particle count with beam sensor model. Error bars at $\pm\sigma$.

3 Results and Discussion

3.1 Memory, CPU Usage and Non-adaptive APE

The Peak Resident Set Size (RSS), average CPU usage, and Absolute Pose Error (APE) were measured¹ as a function of the number of particles while processing a basic synthetic dataset with known ground truth. The particle count was varied between 250 and 5000, and both likelihood field and beam sensor models were measured. All other settings were kept at their default values. A 16 core Intel Core i9-9900 CPU 3.10 GHz host was used to run the performance characterization.

The resulting APE values can be found in Figs. 1 and 2, while the CPU measurements can be found in Table 2. Both packages sustained a peak RSS in the 33MB to 36MB range for all particle counts.

3.2 Simulated Environment with Unmapped Obstacles

A simulated 480 m² office space environment was created using the Gazebo simulator to evaluate the robustness of localization estimates in an environment

Table 2. Average CPU usage (percentage), for both *beluga_amcl* and *nav2_amcl* packages, in both likelihood field and beam sensor model configurations.

| Particles | 250 | 300 | 400 | 500 | 750 | 1000 | 2000 | 3000 | 4000 | 5000 |
|------------------|--------|------|------|------|------|------|------|-------|-------|-------|
| likelihood field | beluga | 3.32 | 3.29 | 3.46 | 3.55 | 3.65 | 3.76 | 3.95 | 4.18 | 4.36 |
| | nav2 | 4.19 | 4.25 | 4.39 | 4.56 | 4.97 | 5.16 | 6.76 | 7.70 | 8.82 |
| beam | beluga | 5.29 | 5.56 | 6.21 | 6.64 | 7.57 | 8.52 | 12.23 | 15.11 | 18.41 |
| | nav2 | 4.85 | 5.47 | 5.52 | 5.81 | 6.67 | 7.41 | 9.84 | 12.31 | 14.05 |
| | | | | | | | | | | 21.52 |
| | | | | | | | | | | 17.05 |

¹ *evo* (<https://github.com/MichaelGrupp/evo>) and *timemory* (<https://github.com/NERSC/timemory>) were used for data capture and evaluation.



Fig. 3. Localization trajectories in simulated environment. The unmapped obstacles present in the environment are shown in a lighter shade of gray.

Table 3. APE metrics for trajectories in Fig. 3.

| | likelihood field model | | | | beam sensor model | | | |
|-------------|------------------------|---------|---------|---------|-------------------|---------|---------|---------|
| | rmse | mean | std | max | rmse | mean | std | max |
| beluga_amcl | 0.127 m | 0.109 m | 0.065 m | 0.308 m | 0.086 m | 0.077 m | 0.038 m | 0.200 m |
| nav2_amcl | 0.134 m | 0.108 m | 0.080 m | 0.431 m | 0.088 m | 0.069 m | 0.054 m | 0.331 m |

with both mapped and unmapped obstacles. A differential drive Kobuki mobile platform equipped with an RPLidar A1 lidar was chosen as the simulated robot. The navigation stack is based on the Humble Hawksbill ROS 2 distribution in a Docker container based on Ubuntu 22.04. Just like before, all of the configuration settings were kept at their default value except for the sensor model.

The recorded localization trajectories for both *beluga_amcl* and *nav2_amcl*, each of them using both likelihood field and beam sensor model, are plotted in Fig. 3 along with the occupancy grid of the environment and the distribution of the unmapped obstacles. The resulting APE metrics can be found in Table 3.

3.3 Benchmarking Using Public Datasets

Table 4 lists APE values measured while benchmarking *beluga_amcl* and *nav2_amcl* against the *hallway_localization* and *hallway_return* Magazino datasets, both of which are part of the Cartographer public datasets.

These datasets were recorded using a Magazino TORU mobile robot platform moving along a long hallway with limited feature availability. They contain 2D laser scans and odometry data. Ground truth and occupancy grid maps were computed offline for these benchmarks using Cartographer’s SLAM algorithm.

Table 4. APE benchmark results measured using the Magazino dataset.

| dataset | amcl | likelihood field model | | | | beam sensor model | | | |
|----------------------|--------|------------------------|---------|---------|---------|-------------------|---------|---------|---------|
| | | rmse | mean | std | max | rmse | mean | std | max |
| hallway localization | beluga | 0.072 m | 0.058 m | 0.042 m | 0.289 m | 0.072 m | 0.059 m | 0.042 m | 0.356 m |
| | nav2 | 0.072 m | 0.058 m | 0.042 m | 0.194 m | 0.068 m | 0.055 m | 0.039 m | 0.192 m |
| hallway return | beluga | 0.074 m | 0.061 m | 0.043 m | 0.334 m | 0.078 m | 0.062 m | 0.047 m | 0.349 m |
| | nav2 | 0.070 m | 0.058 m | 0.039 m | 0.189 m | 0.073 m | 0.058 m | 0.044 m | 0.257 m |

3.4 Discussion

Results suggest both packages exhibit similar performance, with few exceptions. For one, the *beluga_amcl* likelihood field model implementation appears to perform better than that of *nav2_amcl*. In Table 4, on the other hand, peak APE for *beluga_amcl* appears to be worse than that of *nav2_amcl*. This effect is not replicated in the results of Sect. 3.2 for neither sensor model. As such, it remains unexplained and further investigation is required.

4 Conclusion

We have presented *beluga_amcl*, a novel open-source localization package for ROS. We have shown that it constitutes a viable alternative to the widely used *nav2_amcl* package, while providing a modern, high-quality codebase that can be further extended to support newer algorithms and sensor modalities. As an added benefit, the ROS-agnostic Beluga core library is not limited to ROS applications, making it available to the robotics community at large. Future work will consolidate the library and expand its feature-set, broadening its scope of applicability.

References

1. beluga_amcl compatibility notes. https://github.com/Ekumen-OS/beluga/blob/31445270321482c72dc8a7608385a0682880151b/beluga_amcl/docs/PARAMETERS.md. Accessed 10 Oct 2023
2. Ros 2 developer guide. <https://docs.ros.org/en/rolling/The-ROS2-Project/Contributing/Developer-Guide.html#ros-2-developer-guide>. Accessed 19 Oct 2010
3. Macenski, S., Foote, T., Gerkey, B., Lalancette, C., Woodall, W.: Robot operating system 2: design, architecture, and uses in the wild. *Sci. Robot.* **7**(66), eabm6074 (2022). <https://doi.org/10.1126/scirobotics.abm6074>, <https://www.science.org/doi/abs/10.1126/scirobotics.abm6074>
4. Quigley, M., et al.: ROS: an open-source robot operating system (2009)
5. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics. Intelligent Robotics and Autonomous Agents series. MIT Press (2005). <https://books.google.com.ar/books?id=jtSMEAQBAJ>



Distributed Coverage Control for Robotic Systems Employing On-Board Sensors

Federico Pratissoli^(✉), Chiara Buono, and Lorenzo Sabattini

University of Modena and Reggio Emilia, Reggio Emilia 42124, Italy
`{federico.pratissoli,chiara.buono,lorenzo.sabattini}@unimore.it`

Abstract. In this study, we introduce a coverage control strategy for coordinating a fleet of autonomous robots. Many existing approaches in the literature rely on extensive computations and environment partitioning, which necessitate complete knowledge of the environment and the robots' positions for achieving optimal coverage. In contrast, our paper presents a control methodology based on the principles outlined in [1], focusing on a local partitioning of the environment corresponding to each robot's sensing area. This methodology exclusively leverages locally acquired information, ensuring optimal coverage without the need for inter-robot communication. We demonstrate how this approach enables a group of robots to achieve optimal coverage while relying solely on locally sensed data, eliminating the necessity for global information sharing. We validate this methodology through simulations and real-world experiments conducted with a group of mobile robots equipped with LiDAR sensors.

Keywords: Multi-Robot Systems · Coverage Control · Distributed Control

1 Introduction

Various strategies have been introduced for implementing coverage control with networked mobile robots. In particular, the most relevant results to this study are reported in [3,4]. These works provide a simple solution that ensures the convergence of networked robots to a configuration maximizing environmental coverage. It relies on a Voronoi partitioning of the entire environment, which is assumed to be either known [9] or measurable by the robots [8]. Several algorithms exist for constructing an *exact* Voronoi diagram, either by assuming that each robot can obtain the position of all the others [5] or through communication among robots and broadcasting of messages [1,2,6]. However, in most practical scenarios, robots lack information about the location of all other robots and may not have the option to rely on a communication network. Thus, the Voronoi partitioning needs to be studied in a sensor range-constrained scenario.

Therefore, instead of an exact Voronoi cell, we consider a *limited* Voronoi cell, as described in [7], which is a partitioning of the area within the robot's sensing range.

This work was supported by the Socially-acceptable Extended Reality Models and Systems (SERMAS) Project of the European Union's Horizon Europe Research and Innovation Program (GA n. 101070351).

In practical applications, the limited sensing capability of the robot is a constraint, and it is rare for the sensing radius to provide sufficient information to compute the exact Voronoi diagram.

We propose a control strategy that operates within the constraints of the information detected by the sensor, eliminating the need for a communication network, and relies on the Voronoi partitioning of the area within the sensing range (limited Voronoi diagram).

It is important to emphasize that all the aforementioned studies are founded on an exact Voronoi partitioning of the environment. In these approaches, the control algorithm necessitates knowledge of the Voronoi neighbors' locations and the environment's boundary to calculate each robot's Voronoi cell accurately. Additionally, in [7, 8] and the related studies, only simulations and experiments within a Motion Capture System were carried out. These experiments assumed that an individual robot could distinguish between obstacles and neighbors. Furthermore, they assumed the robot could determine its neighbors' center of mass for use in the coverage algorithm.

Conversely, our work addresses a scenario in which the group of robots exclusively relies on their onboard sensors: specifically, we consider mobile robots equipped with LiDAR sensors. We introduce an integrated algorithm that utilizes the information derived from the point cloud in the coverage control process. The control strategy we propose can be effectively applied to a group of robots deployed in an unknown environment, as the only information required by the control law is obtainable through onboard sensors, ensuring local acquisition of data.

Beyond research applications, this approach has practical real-world implications. Consider, for example, a fleet of autonomous drones deployed in a search and rescue mission in a disaster-stricken area. In this scenario, the drones can use our proposed control strategy to autonomously cover the search area, locating survivors or hazardous conditions. The ability to operate without centralized control and rely solely on onboard sensors is a crucial feature for such critical missions.

1.1 Contribution

This paper introduces an innovative control strategy that is integrated with the coverage control approach, enabling a complete implementation on real-world robotic systems. Notably, this implementation relies exclusively on onboard sensors, making it highly practical and applicable in real-world scenarios.

2 The Control Strategy

The fundamental technique used in the coverage-based control is the Voronoi tessellation, whose aim is to achieve a distribution of the robotics agents provided with sensors within an area of interests. The methodology is based on the definition of a performance function to be maximized in order to obtain the optimal coverage of the group of robots over the unknown domain Q . The performance function is chosen to model how reliable is the measurement, at point $q \in Q$, performed by robot i whose position is p_i , as a function of the distance $\|q - p_i\|$. Therefore, the performance function can be defined as non-increasing differentiable function $f(\|q - p_i\|) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$.

As described in [7], we have a Voronoi partitioning $\mathcal{V}(P)$ of the environment Q in, so called, n Voronoi cells $\{V_1, \dots, V_n\}$. The *optimization* function $H : Q \rightarrow \mathbb{R}$ can then be formulated as follows:

$$\mathcal{H}(\mathcal{P}, \mathcal{V}) = \sum_{i=1}^n \int_{V_i} f(\|q - p_i\|) \phi(q) dq, \quad (1)$$

where each robot is in charge of covering its own cell, and a better coverage corresponds to a higher value of the function. The optimization function is influenced by a density function ϕ , which encodes the areas of heightened interest.

Choosing the performance function as: $f(x) = -x^2 1_{[0,r]}(x) - r^2 1_{(r,\infty)}(x)$, as described in [7], we compute the gradient of the optimization function, and we obtain the following control law:

$$u_i = k(C_{V_i} - p_i), \quad (2)$$

where $k \in \mathbb{R}_{>0}$ and C_{V_i} denote respectively a proportional gain and the center of mass of the Voronoi cell V_i of the robot i in position p_i . The configuration of points \mathcal{P} which maximizes the optimization function $\mathcal{H}_V(\mathcal{P})$ coincides with the *centroids* of the respective Voronoi cells. In other words, the solution to the maximization problem is achieved when each agent is located at the centroid of its Voronoi region, such that $p_i = C_{V_i}, \forall i$.

The overall methodology algorithm can be broken down into several key steps, following the algorithm described in Algorithm 1. Each robot is able to compute onboard the Voronoi tessellation of its neighbouring area within its sensor range. The robot compute the Voronoi cell associated with itself. For the Voronoi cell, the centroid is computed. The next step involves guiding the robotic agents to move towards the centroids of their respective Voronoi cells. This movement is performed iteratively: during each iteration, robots recompute the Voronoi tessellation based on their current positions and on the detected neighbours and obstacles, calculate new centroids, and adjust their positions to move closer to these new centroids. The iterative process continues until convergence is achieved. Convergence is typically determined by assessing whether the positions of the robots have stabilized within a predefined tolerance. When convergence is reached, the algorithm terminates.

Algorithm 1: Coverage Control Algorithm with Lloyd's Method

Data: Initialize robot positions and sensing range

Result: Evenly distribute robots to optimize coverage

```

1 while Not converged do
2   for Each robot do
3     Seeds ← Detect neighbours and obstacles(Algorithm 2);
4     Cells ← Compute the Voronoi cells (Seeds);
5     Centroid ← Compute the centroid of the robot assigned cell (Cells);
6     Move the robot to the centroid (Centroid);
7   Check for convergence (e.g., small change in robot positions);

```

In order to compute the Voronoi tessellation, each robot utilizes its on-board sensor to identify points relevant to the tessellation calculation. In our specific scenario, we equipped the robots with a laser scanner on board, providing environmental information through 360 points, each representing one degree. As illustrated in Algorithm 2, the robots process this data by iterating through each point provided by the laser scanner. Each robot identifies all visible objects and their respective dimensions using the sensor data. Subsequently, for the Voronoi tessellation, the nearest point to each object is selected.

Algorithm 2: Detect Nearby Objects

Data: LaserScan message: *_msg*
Result: List of Cartesian positions: *cartesians*

```

1 Function DetectObjs (_msg)
2   Initialize objects: Objects[];;
3   Initialize variables: startObj = 0, endObj = 0;
4   for laser_i to _msg.size() do
5     Update startObj, endObj;
6     if startObj and endObj then
7       Add {startObj, endObj} to Objects;
8   for Each object in Objects do
9     Find the nearest point on the object in polar coordinates;
10  Convert polar coordinates to Cartesian coordinates → cartesians;
11  return cartesians;
```

3 Experiments and Simulations

The coverage methodology finds extensive application in various real-world scenarios, including environmental monitoring, search and rescue missions, precision agriculture, and autonomous robotics. It plays a pivotal role in ensuring that robotic agents efficiently and systematically cover a given area, contributing significantly to the success of diverse applications.

To validate our methodology, we conducted a series of experiments. As depicted in Figs. 1(a) and 1(b), one of these experiments involved a simulation in the Gazebo environment, while another was executed with real mobile robots in an indoor setting. Notably, the latter experiment was conducted without the use of a motion capture system, relying solely on onboard sensors and locally detected information.

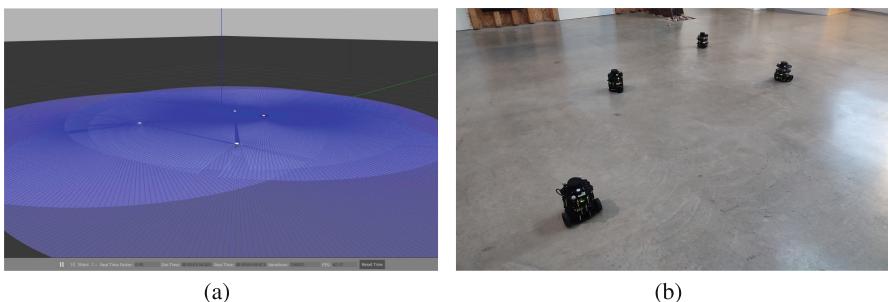


Fig. 1. (a) Simulation in Gazebo environment of 4 mobile robots. (b) Real platform experiment of 4 mobile robots. The mobile robots used are the Turtlebot3 Burger mobile robots.

4 Conclusions

In this study, we introduced a distributed coverage-based control approach to coordinate a fleet of autonomous robots operating in an unfamiliar environment, each equipped with limited sensing capabilities. Our proposed control strategy relies on a limited Voronoi partitioning, utilizing exclusively the information directly perceived by each individual robot. Executed in a distributed manner on-board, the control law makes use of data exclusively from on-board sensors. An extensive set of experiments and simulations was conducted on a fleet of mobile robots to assess the efficacy of the proposed control methodology. Future work will delve into exploring the implementation of this strategy across diverse robotics platforms, such as drones or Vertical Take-Off and Landing (VTOL) systems.

References

1. Alsalih, W., Islam, K., Rodríguez, Y.N., Xiao, H.: Distributed voronoi diagram computation in wireless sensor networks. In: SPAA, p. 364 (2008)
2. Bash, B.A., Desnoyers, P.J.: Exact distributed voronoi cell computation in sensor networks. In: Proceedings of the 6th International Conference on Information Processing in Sensor Networks, pp. 236–243 (2007)
3. Cortes, J., Martinez, S., Bullo, F.: Spatially-distributed coverage optimization and control with limited-range interactions. ESAIM: Control, Optim. Calculus Var. **11**(4), 691–719 (2005)
4. Cortes, J., Martinez, S., Karatas, T., Bullo, F.: Coverage control for mobile sensing networks. IEEE Trans. Robot. Autom. **20**(2), 243–255 (2004)
5. Guruprasad, K., Dasgupta, P.: Distributed voronoi partitioning for multi-robot systems with limited range sensors. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3546–3552. IEEE (2012)
6. Li, Q., Rus, D.: Navigation protocols in sensor networks. ACM Trans. Sens. Netw. (TOSN) **1**(1), 3–35 (2005)
7. Pratissoli, F., Capelli, B., Sabattini, L.: On coverage control for limited range multi-robot systems. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 9957–9963 (2022)

8. Schwager, M., McLurkin, J., Rus, D.: Distributed coverage control with sensory feedback for networked robots. In: *Robotics: Science and Systems*, pp. 49–56 (2006)
9. Teruel, E., Aragues, R., López-Nicolás, G.: A practical method to cover evenly a dynamic region with a swarm. *IEEE Robot. Autom. Lett.* **6**(2), 1359–1366 (2021)



Towards Robust Autonomous Robots Using Statistical Model Checking

Michaela Klauck¹(✉), Ralph Lange¹, Christian Henkel¹, Selma Kchir²,
and Matteo Palmas¹

¹ Robert Bosch GmbH, Bosch Research, Renningen, Germany
`{michaela.klauck,ralph.lange,christian.henkel2,
matteo.palmas}@de.bosch.com`

² List, CEA, Université Paris-Saclay, Palaiseau, France
`selma.kchir@cea.fr`

Abstract. Robots deployed outside controlled environments, often fail to take appropriate actions and require human intervention when facing unexpected situations. Our goal is to advance the capabilities of robots to perform complex tasks *robustly* within unstructured environments by validating their behavior using statistical model checking on the *entire system*. We discuss requirements to a common modeling language from two use cases and present first ideas for making model checking accessible for robotic system validation.

Keywords: robot deliberation · model checking · robustness

1 Problem Statement and Vision

Robust autonomy of robots and intelligent interaction with the environment, even in rare or new situations, is an ultimate goal of robotics research. Methods for planning, decision making, and learning are present in today's robots. In advanced systems, preprogrammed behaviors and actions, planning by logical inference on formalized environment representations, and learning-based approaches are combined for high-level robot deliberation [7]. Still, robots often have trouble coping *robustly* with unknown situations in unstructured environments such as private households, outdoor scenarios, and public places. To some extent, this is due to inadequacies of the aforementioned planning and learning methods.

However, it is also due to a lack of methods and tools to validate complex robotic systems efficiently *as a whole*, despite the longstanding problem of verification and validation (V&V) of autonomous systems [11]. The robotic and AI communities have mostly focused on V&V for *safe* dependable autonomous systems, e.g. in automated driving [14]. Our goal is to instead show the potential of model checking (MC) for the benefit of *robustness* of robot deliberation. In current approaches, often only *individual* components are validated independently disregarding the component's orchestration and the system's interaction

with the environment. Simulation or real hardware tests of the *entire* system are time- and resource-consuming. Between these two extremes lie methods that rely on extensive models of the underlying software components and hardware [3]. The dominant approach is to develop independent components and rely on a communication layer, a middleware like ROS. Some frameworks go beyond this providing additional scaffolding for building coherent systems [1]. Existing approaches, however, do not target the deliberation layer and lack a connection to formal models, which help achieve levels of confidence in the correct behavior unreachable by others [6]. Our experience from product development is that the deliberation component is considered by developers to be the system's most complex software component regardless of the technology. We argue that the lack of appropriate methods and tools to verify deliberation of autonomous systems is a major blocker for the introduction of higher cognitive abilities in industry.

Robustness. The problem of robustness arises in autonomous robotic systems within (i) the system architecture and in (ii) the interaction with the environment. In (i), robustness ensures that all subsystems work together as intended and that the system as a whole is able to fulfill its tasks. The system architecture we consider consists of a deliberation, a skill, and a functional layer, where the latter comprises sensors and actuators to physically interface with the environment. In (ii), robustness ensures that the system can cope with unexpected situations and react appropriately. In that sense, robustness is a performance measure determined by application-specific benchmarks but also (subjectively perceived) intelligence of the robot, and must therefore be specified very elaborately. We want to guarantee robustness of robot behavior by applying MC on the *entire robotic system in its environment*. With this, it can be ensured at design time that the system fulfills its tasks and that it is able to cope with unexpected situations. This guarantee is difficult to achieve with testing alone.

Why Model Checking? MC has so far only been taken into consideration in robotics on *snippets* of the system architecture because it was unclear how to get consistent full models that are verifiable in a reasonable time. E.g., Behavior Trees (BTs) have been decomposed and translated into the input language of a model checker for automated failure mode and effect analysis [4]. Others proposed to extend the BT notation with set theoretic constructs and an approach for generation of minimal cutsets for safety analysis using MC [10]. We propose light-weight techniques where a *full* system model can be extracted partially automatically, e.g. from the skill BT. In addition, we suggest to use *Statistical Model Checking (SMC)* [9], where the model is evaluated by simulating sample executions to get statistical evidence whether a property holds. This is much more scalable than exhaustive MC, corner cases are found faster, and more environment settings can be tested than with unit or real life system tests.

2 Solutions by Bridging Robotics to Model Checking

To achieve our goals, *modeling languages* to formally describe the entire robotic system, and *tools to formally verify* robot behavior have to be found. An overview

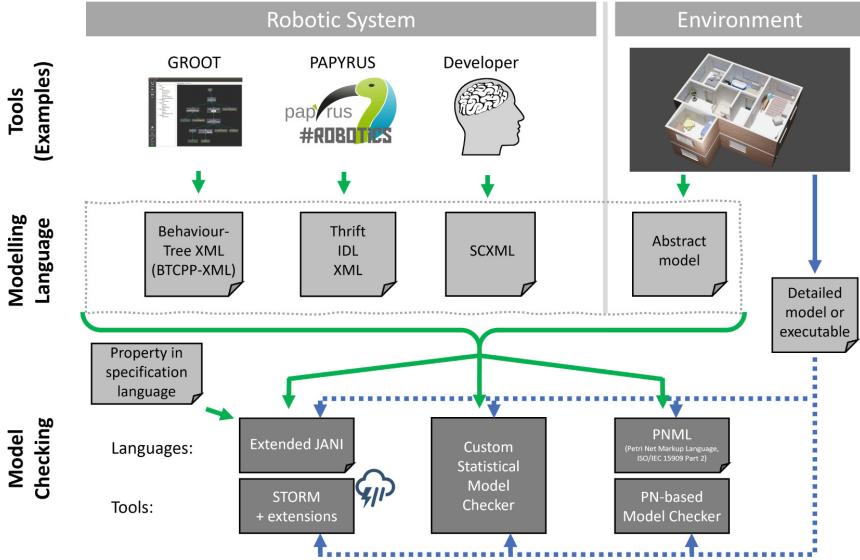


Fig. 1. Modeling languages and MC tooling.

about our proposed solution approaches discussed in the following is given in Fig. 1. Common modeling languages established in the MC community are of interest because then MC tooling is available right away. We aim for a modular language that allows easy modifications. To minimize overhead for developers and modeling discrepancies, we aim at directly using parts of the robot's software, e.g., BTs. The full model consists of information about the (1) tasks, (2) skill layer, (3) functional layer, and (4) the environment. This imposes the following requirements on the modeling language: (1) We assume BTs specifying robotic tasks. The BT interacts with the skills implemented by separate components through a middleware. Thus, the language has to support parallel composition, synchronization, and variables to describe the BT's blackboard. (2) Skills pose the same requirements as BTs since they are algorithms implemented by functional components and logic represented by BTs. (3) Functional components realize complex algorithms in general purpose programming languages. Those need to be abstracted in a formalism suitable for MC. Since many algorithms process complex sensor information subject to noise, we consider non-deterministic and probabilistic processes. (4) Typical 3D environment models used with physics simulators like Gazebo are much too complex as input for MC. Therefore, we consider environments with basic geometric elements expressible with variables. Later, SMC can be combined with a simulator exploring the complex models with a search on the parametrization of the simpler variant. We propose that a fully specified model is split into (i) an environment model, describing the world geometry and objects, (ii) the description of the skill and

functional components deciding what to do when, originally given in program code of the BT, and (iii) the corresponding BT.

A common format to express BTs in robotics is [BTCPP-XML](#) with GROOT, the IDE and editor for BTs. The format is easily extensible and supported by many tools, which makes it a good candidate as a foundation for the modeling language. One path we are pursuing is to start from the XML flavor SCXML and the languages developed in the project SCOPE, part of the EU-funded project [RobMoSys](#). Another line of work allows the direct usage of state-of-the-art model checkers. The idea is to use the common input language for model checkers, [JANI](#) [2], which uses JSON to define networks of quantitative automata with variables. It has been invented to foster tool-interchangeability and interoperability. To make descriptions of environments, functions, and skills easier, the format has to be extended by specific JSON tags fitting to the robotics domain.

After having a formal model of the robotic system in an input format accepted by model checkers, the MC task can be performed directly by state-of-the-art tools. As an alternative to the standard representation of the model for MC as networks of quantitative automata, we propose to translate BTs to Petri Nets (PNs) [13] since most system components are already modeled as BTs and the rest is easily expressible as an extension of the resulting PN. The MC community provides numerous efficient tool support for PNs, e.g., STORM and PRISM [5,8]. Our translation procedure comprises: (1) the translation of the BT to a PN by converting each node, using a template for the respective node type, and the composition of the resulting PNs, followed by several optimization steps [12].

3 Practical Applications

Our goals are closely related to the EU-funded project [CONVINCE](#), in which deliberation capabilities of autonomous robots over extended times are developed along with methods and tooling for verification. Results are integrated in an extensible toolchain, whose core will be open-sourced. We use two of the project’s use cases to demonstrate applicability of our methods.

Vacuum Cleaning Robot. A robot for cleaning apartments autonomously demonstrates the need for context-aware, verifiable, and dynamic deliberation, as cleaning shall be done repetitively and autonomously in unstructured environments. This can only be achieved by dynamically adapting to the environment. End-users expect robots to learn how to handle problematic aspects of their home to avoid getting stuck. The user should not feel the need to tidy up the room and move obstacles out of the way before cleaning. The MC methods presented above can be used during development of robot behaviors to improve the robustness of the software and reduce the need for timely and costly field tests. Properties model checked during development could be that (i) the maximal probability to get stuck is always below a given threshold, (ii) starting from a predefined area around the dock, docking is finished after a given time, or (iii) the relative cleaning coverage along walls with different geometrical complexities is above a certain threshold.

Autonomous Assembly Robot. Assembly appears to be a simple problem that applies a given strategy to a given situation. However, robust assembly is much more complex. Besides selecting and grasping the relevant part to assemble, generating the correct placement poses while taking into account the dependencies between the parts and geometrical constraints of the environment represents a first challenge if the assembly sequence is unknown *a priori*. Robust assembly also deals with environmental uncertainties for detecting defects and assembly anomalies. Thus, developing capabilities for situation understanding through perception, learning, and by anchoring semantic rules to semantic and geometric knowledge representation is a second challenge. Finally, the capability to solve undesired anomalies like assembly failures or to pursue the assembly after human intervention cannot be achieved without dynamic deliberation, which constitutes the third challenge. Properties model checked during development could be that (i) given a task without defective parts, it is finished in a given time, or (ii) if the robot does not manage to fix an assembly anomaly after the application of predefined and learned strategies, entire re-planning of the task is triggered and the part is pulled away whilst waiting for human intervention.

4 Outlook

An important goal is to bring together existing MC and robotic tools and modeling languages by an open-source framework which also commercial providers can link their solutions to. For this, our framework will provide, lightweight adapters and small tools on top of existing software. From CONVINCE, we have launched the [ROS Deliberation Working Group](#) to coordinate with relevant players from the ROS community at an early stage. As soon as a first prototype of the framework exists, we will actively try to integrate relevant tool providers.

Acknowledgements. The work was supported by the European Union’s Horizon Europe Research & Innovation Program under Grant 101070227 (CONVINCE).

References

1. Bruyninckx, H.: Open robot control software: the OROCOS project. In: Proceedings of IEEE International Conference on Robotics and Automation, ICRA, pp. 2523–2528. IEEE (2001)
2. Budde, C.E., Dehnert, C., Hahn, E.M., Hartmanns, A., Junges, S., Turrini, A.: JANi: quantitative model and tool interaction. In: Tools and Algorithms for the Construction and Analysis of Systems, TACAS 2017, ETAPS Proceedings, Part II. LNCS (2017)
3. Foughali, M., Berthomieu, B., Zilio, S.D., Hladik, P.E., Ingrand, F., Mallet, A.: Formal verification of complex robotic systems on resource-constrained platforms. In: Proceedings of the 6th Conference on Formal Methods in Software Engineering (2018)
4. Grunske, L., Lindsay, P.A., Yatapanage, N., Winter, K.: An automated failure mode and effect analysis based on high-level design specification with behavior trees. In: Integrated Formal Methods, IFM 2005. LNCS, vol. 3771. Springer (2005)

5. Hensel, C., Junges, S., Katoen, J., Quatmann, T., Volk, M.: The probabilistic model checker storm. *Int. J. Softw. Tools Technol. Transf.* **24**(4), 589–610 (2022)
6. Ingrand, F.: Recent trends in formal validation and verification of autonomous robots software. In: 3rd IEEE International Conference on Robotic Computing, IRC. IEEE (2019)
7. Ingrand, F., Ghallab, M.: Deliberation for autonomous robots: a survey. *Artif. Intell.* **247**, 10–44 (2017)
8. Kwiatkowska, M.Z., Norman, G., Parker, D.: PRISM 4.0: verification of probabilistic real-time systems. In: Computer Aided Verification, CAV. LNCS. Springer (2011)
9. Legay, A., Lukina, A., Traonouez, L., Yang, J., Smolka, S.A., Grosu, R.: Statistical model checking. In: Computing and Software Science - State of the Art and Perspectives. LNCS, vol. 10000, pp. 478–504. Springer (2019)
10. Lindsay, P.A., Winter, K., Yatapanage, N.: Safety assessment using behavior trees and model checking. In: 8th IEEE International Conference on Software Engineering & Formal Methods, SEFM. IEEE Computer Society (2010)
11. Luckcuck, M., Farrell, M., Dennis, L.A., Dixon, C., Fisher, M.: Formal specification and verification of autonomous robotic systems: a survey. *ACM Comput. Surv.* **52**(5), 100:1–100:41 (2019)
12. Palmas, M., et al.: Translating behavior trees to petri nets for model checking (under submission)
13. Reisig, W.: Understanding Petri Nets - Modeling Techniques, Analysis Methods, Case Studies. Springer (2013)
14. Wishart, J., et al.: Literature review of verification and validation activities of automated driving systems. *SAE Int. J. Connect. Autom. Veh.* **3**, 267–323 (2020)



Simulating Aerial Event-Based Environment: Application to Car Detection

Ismail Amessegher¹, Hajar Fradi¹⁽⁾, Clémence Liard^{1,2}, Jean-Philippe Diguet², Panagiotis Papadakis^{1,2}, and Matthieu Arzel¹

¹ IMT Atlantique, Lab-STICC, UMR CNRS 6285, 29238 Brest, France
hajer.fradi@imt-atlantique.fr

² CNRS, IRL CROSSING, Adelaide, Australia

Abstract. With the primary goal of enhancing the efficiency of drones for research and rescue missions through the exploitation of neuromorphic sensors and event-based vision, our focus in this work lies in setting up a simulated environment that can be used for synthetic data generation. In particular, we employ Unreal Engine to generate scenes suitable for the case of vehicle perception, followed by a dynamic event-based simulation environment in interaction with the AirSim and v2e software tools. The synthetic event data acquired in this simulated environment is shown to provide a valuable resource for training Artificial Intelligence (AI) systems and more particularly for the task of car detection using YOLOv7.

Keywords: Drones · Simulation · Events · Synthetic Data · Car Detection

1 Introduction

The increasing frequency of natural disasters calls for new solutions to improve the effectiveness and efficiency of disaster response services. According to a United Nations review, in 2019 only, around 400 climate-related and geophysical events resulted in tens of thousands of human losses. In this context, first-response by Search-And-Rescue (SAR) robots has shown its potential to operate in unstructured and hostile environments. A SAR team typically composed of aerial and/or ground vehicles operates remotely via a control post [7]. To accomplish such missions, efforts must be directed towards multiple aspects, including:

- Safe movement: enhancing safety with onboard AI assistance that ensures reliable obstacle detection.
- Fast movement: achieving low-latency environment capture for higher drone reactivity.
- Energy preservation: prioritizing artificial vision solutions that are conducive to low energy consumption.

SAR missions would therefore significantly benefit from technologies enabling higher energy efficiency combined with improved navigation autonomy of drones. To fulfill this objective, our focus in this study is to investigate newly emerging bio-inspired sensors known as event or neuromorphic cameras [5]. Thanks to their unique properties including lightweight, energy-efficient, and high-frequency data acquisition, an increasing interest has been shown in exploiting these new sensors for perception as well as the control of drones [3].

It is crucial to note that event-based vision requires additional dataset acquisition and subsequent training, which are more readily accessible for conventional RGB data. Due to various constraints in the extensive deployment and usage of real robots for training and testing, the use of a simulated environment can reduce the risk of damage and compensate for the lack of real data via abundant simulated trajectories following previous works [1]. On this basis, in this work, we integrate complementary tools that jointly serve the goal of generating synthetic event aerial data that are finally used to train a proof-of-concept, custom car detector.

2 Simulated Event-Based Environment

In the literature, object detection or tracking by drones has been predominantly performed using conventional frame-based cameras [2, 4, 8]. Event cameras are newly emerging bio-inspired sensors that asynchronously measure per-pixel brightness changes. Known also as neuromorphic sensors, they are different from conventional frame-based sensors where images are captured at a fixed frame rate. Their resulting output is a stream of events encoding the time, position, and sign of brightness changes. Event sensors have several advantages compared to conventional RGB cameras including high microsecond temporal resolution, very high dynamic range (140 dB vs. 60 dB), low power consumption, and high pixel bandwidth (in the order of kHz) [5]. Given all these advantages, these cameras have witnessed an increasing interest in drone applications.

At present, event-based ground-truth data acquired from airborne drones are scarce and limited in volume and diversity. To compensate for the risk of drone crashes when acquiring data from real images and allow for more accurate annotations in nearly infinite environmental scenarios, we opt for using a simulated pipeline. In particular, inspired by research works that use AirSim simulator [10] and Unreal Engine (UE), we build a simulated event-based environment coupled with the v2e toolbox [6] for synthetic event data creation.

In detail, AirSim (Aerial Informatics and Robotics Simulation) is an open-source simulation platform developed by Microsoft for unmanned robots. It is built using Unreal Engine as its graphics engine [9], which is particularly powerful and versatile. UE boasts impressive graphics capabilities, stunning visual effects, and strong support for virtual and augmented reality. AirSim comes with photo-realistic camera, depth and thermal sensors. For event-based data, the v2e toolbox allows to convert conventional image data into event-based data [6] while adjusting various parameters related to the image conversion, such as

timing, output video dimensions and noise levels to simulate real-world event data collection conditions.

3 Data Generation

Since aerial images are expensive to be collected, and hard to be manually annotated, synthetic data becomes the most feasible solution to provide large amounts of automatically annotated data. For the task of car detection, the training data is created using the Modular Neighborhood pack¹. The scene depicts a suburban neighborhood and contains vehicles, which are suitable for our application of car detection. For RGB images, the expected size is 640×480 and images need to be captured at a frequency of 200 Hz. Since at this high acquisition frequency, data could be significantly blurred, it is worth making efforts in Unreal Engine and AirSim to slow down the simulation. Initially, we could capture images at a frequency of 20 Hz. By making time flow 10 times slower within the simulation, data capture would reach 200 Hz. AirSim allows us also to select the date at which the simulation takes place. This parameter enables data captures with four types of sunlight conditions: day, night, day followed by night, and night followed by day. This helps to test the recognition capabilities under varying lighting conditions.

To construct an image from raw events, various representations could be investigated [13]. In our approach, we opt to aggregate events within a temporal window of $\Delta t = 5$ ms, aligning with the initial frequency of the simulated environment. The resulting collected dataset comprises a total number of 10982 frames, with 7191 for training and 3791 for testing. The training set consists of 3216 samples acquired under daytime conditions and 3975 during the night. Similarly, the test set includes 2301 samples under daytime conditions and 1490 under nighttime conditions. For the annotations, bounding boxes with a size of less than 240 pixels are excluded. Figure 1 shows two sample images (daytime and nighttime), with their corresponding event frames and annotations.

4 Experiments

We choose to evaluate our prototype for the task of car detection, conducting experiments on the YOLOv7 detector [11], motivated by its effectiveness for real-time applications as well as its more accurate results compared to previous YOLO versions [12]. Reportedly, it outperforms other real-time object detection models in both speed and accuracy. In the training phase, we use a mini-batch size equal to 64 and 30 epochs. The model is trained on an NVIDIA RTX A6000 GPU. The performance of car detector in the test set of our created synthetic event data is evaluated in terms of mean Average Precision (mAP) of detections at Intersection Over Union (IOU) equal to 0.5 regarding the ground truth boxes.

¹ <https://www.unrealengine.com/marketplace/en-US/product/modular-neighborhood-pac>.

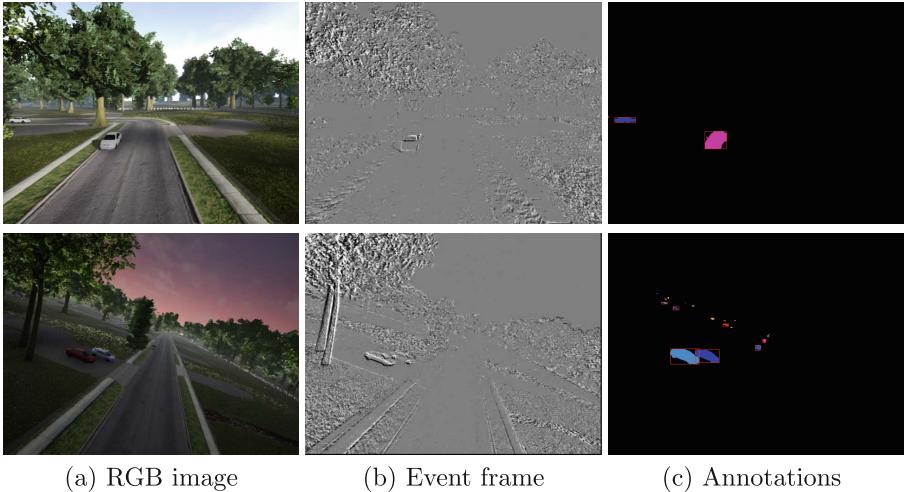


Fig. 1. Two sample images of the simulated dataset with their corresponding generated event frames and annotation boxes in the segmentation map. In the first row, a frame during the day, and in the second row, another sample under nighttime conditions.

The achieved mAP attains 0.63, with precision and recall both reaching 0.93 and 0.63, respectively. These outcomes affirm that we succeed in precisely detecting cars within the simulated data, though there is room for improvement in terms of recall. Increasing true positives by detecting more objects could be achieved by expanding the size of the training data (Fig. 1).

5 Conclusion

In this work, we build a simulation environment based on Unreal Engine, AirSim and v2e toolbox to simulate dynamic aerial events. These tools are assessed in generating event-based data, which is used to train a custom car detection model. By means of experiments using the YOLOv7 detector, the utility of the simulated event data for training a custom model under different lighting conditions. Having used the Modular Neighborhood environment scene driven by its relevance to our use-case, we could subsequently improve the variety of objects and scene conditions via the addition of further scenes.

Acknowledgement. This work has received a French government support granted to the Labex CominLabs excellence laboratory and managed by the National Research Agency in the “Investing for the Future” program under reference ANR-10-LABX-07-01 from September 2022 to December 2024. In this program, the project associated to this work is *LEASARD* (Low-Energy deep neural networks for Autonomous Search-And-Rescue Drones).

References

1. Andrei, M., Panagiotis, P., Serge, G.: An open-source software framework for reinforcement learning-based control of tracked robots in simulated indoor environments. *Adv. Robot.* **36**(11), 519–532 (2022)
2. Cao, Y., et al.: VisDrone-DET2021: the vision meets drone object detection challenge results. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2847–2854 (2021)
3. Dimitrova, R.S., Gehrig, M., Brescianini, D., Scaramuzza, D.: Towards low-latency high-bandwidth control of quadrotors using event cameras. In: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 4294–4300. IEEE (2020)
4. Fradi, H., Bracco, L., Canino, F., Dugelay, J.-L.: Autonomous person detection and tracking framework using unmanned aerial vehicles (UAVs). In: 2018 26th European Signal Processing Conference (EUSIPCO), pp. 1047–1051. IEEE (2018)
5. Gallego, G., et al.: Event-based vision: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(1), 154–180 (2020)
6. Hu, Y., Liu, S.-C., Delbruck, T.: v2e: from video frames to realistic DVS events. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1312–1321 (2021)
7. Kruijff, G.-J.M., et al.: Rescue robots at earthquake-hit Mirandola, Italy: a field report. In: IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR) (2012)
8. Ramachandran, A., Sangaiah, A.K.: A review on object detection in unmanned aerial vehicle surveillance. *Int. J. Cogn. Comput. Eng.* **2**, 215–228 (2021)
9. Sanders, A.: An Introduction to Unreal Engine 4. CRC Press, Boca raton (2016)
10. Shah, S., Dey, D., Lovett, C., Kapoor, A.: AirSim: high-fidelity visual and physical simulation for autonomous vehicles. In: Field and Service Robotics: Results of the 11th International Conference, pp. 621–635. Springer (2018)
11. Wang, C.-Y., Bochkovskiy, A., Liao, H.-Y.M.: YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7464–7475 (2023)
12. Zhao, H., Zhang, H., Zhao, Y.: YOLOv7-sea: object detection of maritime UAV images based on improved YOLOv7. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 233–238 (2023)
13. Zubić, N., Gehrig, D., Gehrig, M., Scaramuzza, D.: From chaos comes order: ordering event representations for object recognition and detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 12846–12856 (2023)



Enhanced Localization of ArUco Markers for Autonomous Robotics: A Comparative Study

Alessandro Minervini^(✉), Jean Carlos Quito Casas, Davide Fassio,
Davide Buoso, Claudio Giuseppe Messina, Francesco Marino,
and Giorgio Guglieri

Politecnico di Torino, Turin, Italy
alessandro.minervini@polito.it

Abstract. Autonomous drone technology increasingly enables their use in diverse applications, offering cost and time benefits in precision agriculture and surveillance. They are especially efficient in search and rescue and exploring hard-to-access areas.

Navigating indoor settings and partially known environments poses significant challenges in autonomous robotics. This paper introduces a novel method that leverages depth image data to substantially improve performance in these contexts. We elucidate the method's design, showcasing its dependability and advantages over conventional approaches. Furthermore, the paper delineates the critical procedures for effective autonomous robot guidance, tackling complex obstacles inherent to the field.

Keywords: ArUco markers · Autonomous UAV · Robotics · Localization · Perception

1 Introduction and State of the Art

Advancements in autonomous drone technology have enabled the integration of these aerial robots into various applications, such as agriculture [1, 2], and rescue operations [3]. ArUco markers (Fig. 1) are central to this advancement, playing a pivotal role in robotic navigation and localization due to their simple yet effective design.

In the latest Leonardo Drone Contest, Leonardo S.p.a motivated 7 Italian Universities to explore UAV and UGV autonomous cooperation in indoor and partially known environments. The 2023 contest focuses on detecting and locating targets marked with ArUco markers, identifiable by UAVs, UGVs, or an Axis camera in the competition area. The DRones Autonomous Flight Team (DRAFT) from Politecnico di Torino participated, providing a novel marker detection method and securing second place in the November 2023 contest. Several methodologies have been developed for ArUco marker detection in autonomous robotics, primarily through analyzing camera RGB images.

After converting these images to grayscale, markers are identified and localized using their known positions in the frame, simplified by OpenCV functions like ‘detectMarkers’ and ‘estimatePoseSingleMarkers’ [4]. Despite these advancements, accurately detecting markers over distances greater than 2 m remains challenging, particularly under competition constraints that discourage proximity for position estimation.

To address this, various methods have been proposed. For instance, one approach achieves accurate position estimation up to approximately 9 m with a 6% error [5], while another uses deep learning to identify markers in challenging lighting conditions [6]. Convolutional neural networks have also been employed for reliable detection even when marker corners are occluded [7]. However, the need for a novel approach with low computational demands and higher precision within 4 m is evident.

This paper proposes a novel method that enhances performance by incorporating depth image information. This method, backed by data collection, demonstrates superior effectiveness and reliability compared to traditional methods. This approach aims to address state-of-the-art challenges in autonomous robot guidance and localization, with several papers introducing methods using fiducial markers for this purpose [8,9].

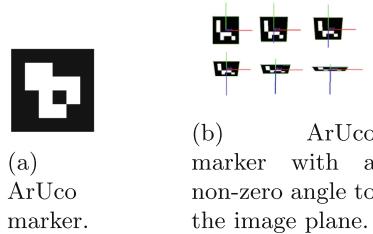


Fig. 1. Overview of ArUco markers.

2 Depth Image Augmented ArUco Pose Estimation

The innovative solution proposed in this paper leverages the additional information from the Intel RealSense D455 depth image [10]. This methodology reduces errors compared to classical approaches and significantly enhances measurement reliability in scenarios where detections occur at extended distances and with non-parallel marker orientation. Following ArUco marker identification, the proposed method seamlessly transitions into the 3D localization process. Once the coordinates of the target corners are obtained, they are used to find the corresponding points in the depth image. To reduce the observation noise, our strategy is to determine the best position through the weighted average of coordinates.

For the coordinates x_i and y_i obtained from the i-th acquisition, with distances d_{xi} and d_{yi} , the weighted average is calculated as follows:

$$\bar{x} = \frac{\sum_{i=1}^n \frac{x_i}{d_{xi}}}{\sum_{i=1}^n \frac{1}{d_{xi}}} \quad \bar{y} = \frac{\sum_{i=1}^n \frac{y_i}{d_{yi}}}{\sum_{i=1}^n \frac{1}{d_{yi}}} \quad (1)$$

In summary, the proposed methodology is illustrated in Fig. 2, and it involves the following steps:

1. Define a bounding rectangle around the inscribed circle.
2. Identify pixels within the inscribed circle for subsequent depth analysis.
3. Retrieve depth values for each pixel, weighted based on multiple depth measurements near each ArUco marker.

Another possible way of understanding the technique is that our solution, by obtaining a robust estimation of the distance of the ArUco, minimizes the reprojection error of the observed markers and can improve the estimate over multiple observations.

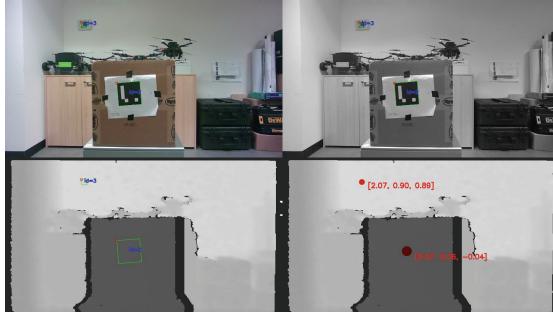


Fig. 2. Depth Image Augmented ArUco detection: RGB and depth image association for position estimate

3 Experimental Results

Experimental results have been collected to compare the classical methodology and the proposed approach exploiting depth image information. The data collection involved placing the marker at various distances from the cameras up to the threshold of the identification range. For each distance, measurements were taken for five different angles of the marker relative to the image plane, ranging from 0 to 50°. A precision laser meter with an accuracy of 0.001 m was employed to measure the marker's distance from the camera. The data collection highlights the low sensitivity of the proposed methodology based on the depth image, both

with increasing distance and marker inclination relative to the image plane. In Fig. 2, the estimated position employing depth image contribution is merely affected by the angle increase, while the RGB methodology shows high sensitivity when distance increases. Figure 3, illustrates the relative measurement error of the distance for each angle. Once again, the methodology employing the depth image outperforms the classical approach, reducing the margin of error. This is particularly evident for distances and angles close to the detection thresholds. Table 1 provides the measurement data for the detection threshold distance.

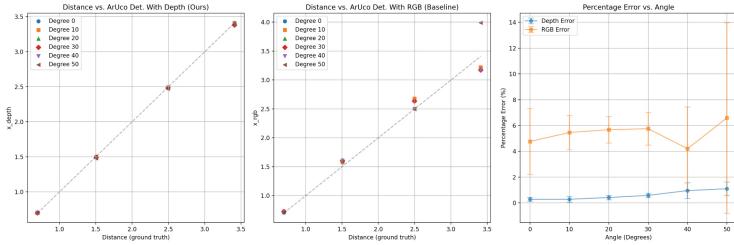


Fig. 3. Classical methodology and depth image association: experimental comparison

Table 1. Comparison of relative error at the maximum distance over the angles.

| Relative Orientation [Deg] | Ground truth (distance) [m] | x_{rgb} [m] | x_{depth} [m] | Relative Error RGB [%] | Relative Error Depth (ours) [%] |
|----------------------------|-----------------------------|---------------|-----------------|------------------------|---------------------------------|
| 0 | 3.408 | 3.193 | 3.411 | 6.309 | 0.088 |
| 10 | 3.408 | 3.220 | 3.410 | 5.516 | 0.059 |
| 20 | 3.408 | 3.190 | 3.415 | 6.397 | 0.205 |
| 30 | 3.408 | 3.175 | 3.381 | 6.837 | 0.792 |
| 40 | 3.408 | 3.174 | 3.391 | 6.866 | 0.499 |
| 50 | 3.408 | 3.992 | 3.390 | 17.136 | 0.528 |

4 Conclusions

This paper presented an innovative approach to enhance the localization of ArUco markers. Focusing on a comparative study between the classical RGB image-based method and a novel depth image-augmented technique. Our research demonstrated that integrating depth information significantly improves

the accuracy and reliability of ArUco marker localization, especially in challenging scenarios involving extended distances and non-parallel marker orientations.

The experimental results clearly showed the superiority of the proposed depth image-augmented methodology over the conventional RGB approach. This was evident in reduced sensitivity to distance and angle variations and a marked decrease in measurement errors under various conditions. The depth image-augmented method proved robust, computationally efficient, and effective in diverse practical applications, extending the potential for more complex and dynamic autonomous robotic missions. In conclusion, the depth image-augmented approach to ArUco marker offers a practical, accurate, and efficient solution for enhancing robotic navigation and positioning.

References

1. Abderahman, R., Alireza, A., Karim, R., Horst, T.: Drones in agriculture: a review and bibliometric analysis. *Comput. Electron. Agric.* **198** (2022). <https://doi.org/10.1016/j.compag.2022.107017>
2. Kulbacki, Marek, Segen, J., et al.: Survey of drones for agriculture automation from planting to harvest. In: 2018 IEEE 22nd International Conference on Intelligent Engineering Systems (INES) (2018). <https://doi.org/10.1109/INES.2018.8523943>
3. Yunus, K., et al.: The potential use of unmanned aircraft systems (drones) in mountain search and rescue operations. *Am. J. Emerg. Med.* **36**, 583–588 (2018). <https://doi.org/10.1016/j.ajem.2017.09.025>
4. OpenCV, Detection of ArUco Markers. https://docs.opencv.org/4.x/d5/dae/tutorial_aruco_detection.html
5. Ferrão, J., Dias, P., Neves, A.J.R.: Detection of Aruco markers using the quadrilateral sum conjuncture. In: Campilho, A., Karray, F., ter Haar Romeny, B. (eds.) ICIAR 2018. LNCS, vol. 10882, pp. 363–369. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93000-8_41
6. Berral-Soler, R., Muñoz-Salinas, R., Medina-Carnicer, R., Marín-Jiménez, M.: DeepArUco: marker detection and classification in challenging lighting conditions. In: Iberian Conference on Pattern Recognition and Image Analysis, pp. 199–210 (2023). https://doi.org/10.1007/978-3-031-36616-1_16m
7. Li, B., Wang, B., Tan, X., Wu, J., Wei, L.: Corner location and recognition of single ArUco marker under occlusion based on YOLO algorithm. *J. Electron. Imaging*, **30** (2021). <https://doi.org/10.1117/1.JEI.30.3.033012>
8. Nahangi, M., Heins, A., Mccabe, B., Schoellig, A.: Automated localization of UAVs in GPS-denied indoor construction environments using fiducial markers. In: 35th International Symposium on Automation and Robotics in Construction (ISARC 2018). <https://doi.org/10.22260/ISARC2018/0012>
9. Mráz, E., Rodina, J., A. Babinec: Using fiducial markers to improve localization of a drone. In,: 23rd International Symposium on Measurement and Control in Robotics (ISMCR). Budapest, Hungary, vol. 2020, pp. 1–5 (2020). <https://doi.org/10.1109/ISMCR51255.2020.9263754>
10. Sonoda, T., Grunnet-Jeps, A.: Depth image compression by colorization for Intel® RealSense™ Depth Cameras. <https://dev.intelrealsense.com/docs/depth-image-compression-by-colorization-for-intel-realsense-depth-cameras>



Adaptive Distributed Coverage Control for Learning Spatial Phenomena in Unknown Environments

Mattia Mantovani^(✉), Federico Pratissoli, and Lorenzo Sabattini

University of Modena and Reggio Emilia, 42124 Reggio Emilia, Italy
`{mattia.mantovani,federico.pratissoli,lorenzo.sabattini}@unimore.it`

Abstract. This study focuses on efficiently deploying a multi-robot system (MRS) to achieve optimal area coverage in the presence of an unknown event that requires monitoring. We introduce a control algorithm based on coverage to enable a team of robots to learn and estimate the spatial process of this region while ensuring good coverage. The robots' observations are influenced by environmental noise. We use Gaussian processes (GP) to model the spatial process, and the multi-robot team optimally covers the estimated process. We evaluated the algorithm through simulations and real platform experiments.

Keywords: Multi-Robot Systems · Coverage Control

1 Introduction

MRS excel in collaborative exploration, focusing on maximizing coverage in important areas to enhance effectiveness [1]. This work addresses spatial coverage in MRS with limited capabilities, using Voronoi partitioning to ensure robots convergence to key areas defined by a density function [1, 7]. Robots often operate in unfamiliar environments with unknown densities. In manufacturing plants, monitoring diverse production areas is crucial to prevent failures. In livestock facilities, it can be used to monitor possible concentrations of pollutants like ammonia, ensuring the health and well-being of the animals. Air quality monitoring is critical in many industrial settings, such as chemical plants or labs, where the safety of workers depends on maintaining specific standards. Consider energy power plants, where this methodology could be used to monitor the integrity of critical components in the production of, for instance, nuclear energy where it is crucial to maintain safety and to prevent potential hazards. MRS can also find use in the control and monitoring of fires in large industrial complexes or areas at high risk of biological contamination, such as laboratories working with dangerous pathogens. Furthermore, it plays a crucial role in search and

This work was supported by the AI-DROW Project through the Italian Ministry for University and Research under the PRIN 2022 program, funded by the European Union – Next Generation EU.

rescue operations, particularly in situations where human intervention would be too hazardous, like exploring collapsed buildings after a disaster.

The GP is a frequently employed tool for reconstructing signals from limited samples and making predictions across their domains [11]. Recent research focuses on optimizing multi-robot coverage for exploration and process estimation [5]. Some methods regulate sampling but overlook unknown areas and exploration-exploitation trade-offs [13]. Others introduce decentralized control strategies for learning spatial fields and coverage [4, 8]. In [3], robots begin in random positions and optimally cover the estimated spatial field without a filtering strategy for acquired samples. Certain approaches involve a two-phase strategy: initial exploration to estimate the density function, followed by exploitation for optimal coverage [10, 12]. These approaches overlook the computational complexity of GP regression, which is $O(N^3)$, and the impact of noise on spatial estimation and robot deployment strategies [2, 6]. Noise greatly affects spatial process estimation accuracy, leading to sub-optimal robot deployment strategies.

Contribution: *This paper presents a novel control strategy for optimal learning and estimation of a spatial process using a team of noisy-sensor-equipped robots in a noisy environment. It uses GPs to model the process, efficiently filtering data to balance exploration and exploitation in learning and coverage maximization.*

2 Coverage and Exploration Problem

This study optimizes spatial field exploration with GPs, leveraging GP-derived data for focused robot team efforts. Bayesian Optimization, detailed in [9], employs an ‘acquisition function’ to balance exploration and exploitation, maximizing utility for efficient learning and information use. The strategy is ‘adaptive’ in the sense that the MRS uses a substitute of the density function that coincides with an acquisition function, in our study: $\phi'_t(x) = \sigma_{t-1}(x) + W_t \mu_{t-1}(x)$. The density function $\phi'_t(x)$ adapts the strategy to guide robots to high utility areas. In particular the utility distribution favors exploration when the density function is unknown ($\phi'_t(x) \sim \sigma_{t-1}(x)$) and favors exploitation when it’s well-explored ($\phi'_t(x) \sim \mu_{t-1}(x)$). The weight W_t is a term to balance the process and is selected according to the preference over the trade off between exploration and exploitation. High values of W_t mean that the exploitation is preferred over the exploration: $W_t = C_1 \arctan(C_2 t)$, where C_1 and C_2 are two parameters, chosen to tune the exploration-exploitation trade off.

3 Experimental Validation and Performance Evaluation

In this study, we tested the algorithm through simulations (Fig. 1) and real hardware (Fig. 3). Parameters were tuned with $C_2 = 1$, leading to $C_1 = 0.111$ and $W_t = 10$ over time, favoring exploration initially and shifting towards convergence. We introduced white Gaussian noise in sensory observations by varying parameters for Σ in the range [0.01–0.3]. Performance evaluation (Figs. 2 and 4)

analyzed Root-Mean-Square Error (RMSE) trends and dataset sizes, affirming the strategy's effectiveness in both simulated and real hardware experiments, with decreasing RMSE and stable sample acquisition by the robots.

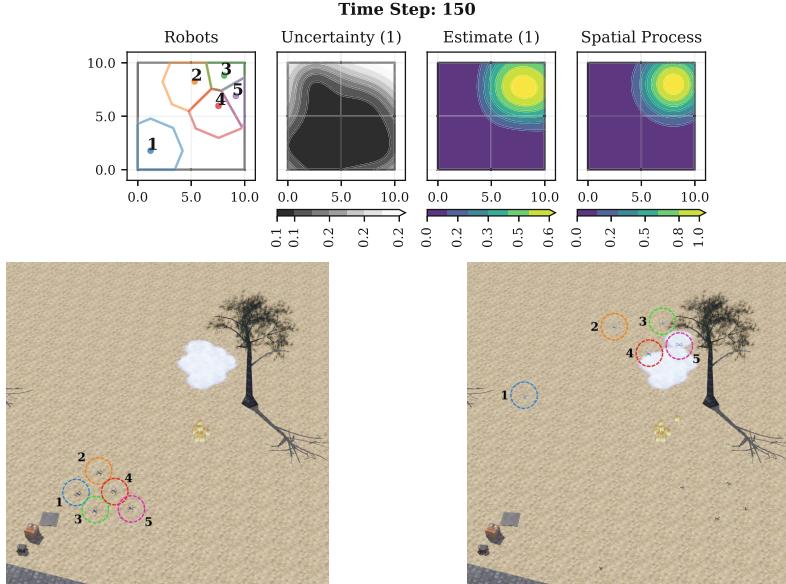


Fig. 1. The image shows a robot team responding to a fire event. Starting from the base station, robots progressively move towards the event (bottom figures). The top figure shows various process stages, from left to right: robots (colored dots) with Voronoi partitions, uncertainty in the first robot spatial field estimation, the spatial field estimated by the first robot, and the actual spatial process. At $ts = 150$, robots optimally cover the event. This example highlights the first robot's GP computation for clarity. (Color figure online)

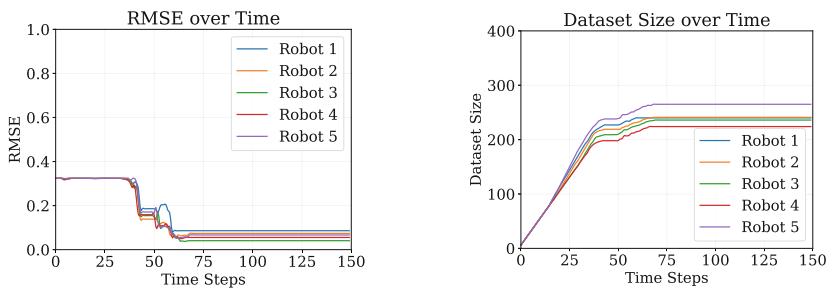


Fig. 2. The first figure shows decreasing RMSE between robot GP-derived spatial processes and ground truth over time, never reaching zero due to operating conditions. The second figure indicates constant robot dataset size, highlighting filter effectiveness in selecting relevant samples for process estimation. Complex processes require more samples for an acceptable estimate.

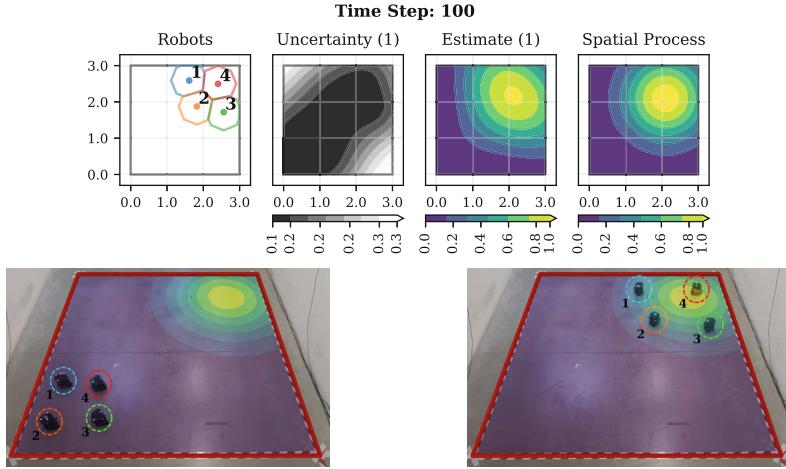


Fig. 3. The figure showcases a team of TurtleBot3 Burger robots exploring an unknown domain and monitoring a simple event. The first sub figure displays the evolving spatial process, while the second and third sub figures depict the robots' initial and final positions. In the final configuration, the robots evenly surround the event density peak.

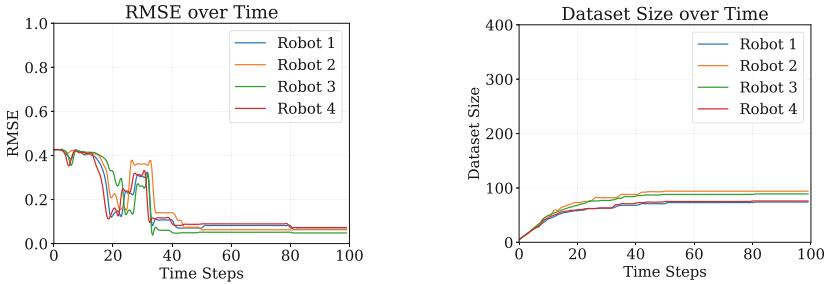


Fig. 4. The first sub figure demonstrates the RMSE between each robot GP estimated spatial process and the ground truth over time. The second sub figure shows that over time the size of the robots datasets tends to remain constant, indicating the filter effectiveness in selecting relevant samples for process estimation.

4 Conclusion and Future Work

In this work, we introduce a novel distributed algorithm designed to improve spatial field estimation and coverage by a group of robots, even in the presence of noisy observations. Our approach utilizes GPs to model the spatial field based on data collected by the robots. We also propose an innovative method to manage dataset size and address computational challenges posed by large and noisy datasets. This control strategy empowers a team of robots to explore unknown environments and reconstruct events in various scenarios, such as monitoring bio-hazardous areas and responding to fires. Our future work will focus on developing

a methodology to handle time-varying spatial processes and explore multi-robot systems with heterogeneous sensing capabilities. Aerial robots excel at macro-level estimations, while ground robots provide detailed information. However, successful implementation will require effective coordination and data exchange between these two distinct robot types.

References

1. Cortes, J., Martinez, S., Karatas, T., Bullo, F.: Coverage control for mobile sensing networks. *IEEE Trans. Robot. Autom.* **20**(2), 243–255 (2004)
2. Jakkala, K.: Deep gaussian processes: a survey. arXiv preprint [arXiv:2106.12135](https://arxiv.org/abs/2106.12135) (2021)
3. Luo, W., Nam, C., Kantor, G., Sycara, K.: Distributed environmental modeling and adaptive sampling for multi-robot sensor coverage. In: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, pp. 1488–1496 (2019)
4. Nakamura, K., Santos, M., Leonard, N.E.: Decentralized learning with limited communications for multi-robot coverage of unknown spatial fields. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 9980–9986. IEEE (2022)
5. Paley, D.A., Wolek, A.: Mobile sensor networks and control: adaptive sampling of spatiotemporal processes. *Ann. Rev. Control, Robot. Auton. Syst.* **3**, 91–114 (2020)
6. Park, C., Huang, J.Z.: Efficient computation of gaussian process regression for large spatial data sets by patching local gaussian processes. *J. Mach. Learn. Res.-JMLR* (2016)
7. Pratissoli, F., Capelli, B., Sabattini, L.: On coverage control for limited range multi-robot systems. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 9957–9963 (2022)
8. Santos, M., Madhushani, U., Benevento, A., Leonard, N.E.: Multi-robot learning and coverage of unknown spatial fields. In: 2021 International Symposium on Multi-Robot and Multi-Agent Systems (MRS), pp. 137–145. IEEE (2021)
9. Schulz, E., Speekenbrink, M., Krause, A.: A tutorial on gaussian process regression: modelling, exploring, and exploiting functions. *J. Math. Psychol.* **85**, 1–16 (2018)
10. Schwager, M., Vitus, M.P., Rus, D., Tomlin, C.J.: Robust adaptive coverage for robotic sensor networks. In: Robotics Research, pp. 437–454. Springer (2017)
11. Viseras, A., Shutin, D., Merino, L.: Robotic active information gathering for spatial field reconstruction with rapidly-exploring random trees and online learning of gaussian processes. *Sensors* **19**(5), 1016 (2019)
12. Wei, L., McDonald, A., Srivastava, V.: Multi-robot gaussian process estimation and coverage: Deterministic sequencing algorithm and regret analysis. In: 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 9080–9085. IEEE (2021)
13. Zuo, L., Shi, Y., Yan, W.: Dynamic coverage control in a time-varying environment using Bayesian prediction. *IEEE Trans. Cybern.* **49**(1), 354–362 (2017)



Insights on Control Barrier Functions Application for Complex Robotic Systems

Filippo Bertoncelli and Lorenzo Sabattini^(✉)

Department of Sciences and Methods for Engineering (DISMI), University of Modena
and Reggio Emilia, Reggio Emilia, Italy
{filippo.bertoncelli,lorenzo.sabattini}@unimore.it

Abstract. Control Barrier Functions (CBFs) are a very useful tool for ensuring the safety and coordination of robots in complex environments. While versatile and offering formal safety guarantees, CBFs face challenges, including computational complexity and potential conservatism. This paper explores the practical differences between two CBF formulations (zeroing and reciprocal) through a case study involving multiple aerial robots coordinating for a desired formation. The results, collected in a simplified simulation scenario, show the different performance achieved with the two formulations.

1 Introduction

Control Barrier Functions (CBFs) are a cornerstone in modern robotics, providing a systematic approach to ensure the safety of robots operating in complex environments [4]. By defining safe operational states and designing control inputs that respect these safety constraints, CBFs help robots avoid obstacles, interact safely with humans, and adhere to their physical limitations. This integration of safety into the control system is crucial for applications ranging from autonomous vehicles to collaborative robots in manufacturing.

The beauty of CBFs lies in their versatility and the formal safety guarantees they offer, which is a significant boon for the adoption of robots in safety-critical applications. However, the implementation of CBFs is not without its drawbacks. The complexity of calculating these functions can be a hurdle, especially for systems with high degrees of freedom. Moreover, the conservative nature of CBFs may lead to overly cautious behaviors that can limit a robots' operational efficiency.

In essence, while CBFs are a powerful tool for enhancing the safety of robotic systems, they must be employed judiciously, balancing safety with performance, and acknowledging the limitations imposed by computational demands and potential conservativeness. In this manuscript we aim at providing an insight on the practical difference between two different types of CBFs: zeroing and reciprocal.

These concepts can be applied to general robotic systems, in particular when safety constraints are relevant (e.g., human-robot interaction, automated driving). In this paper we will focus on a case study involving multiple aerial robots, whose motion is coordinated to achieve a desired formation. This example represents a simplified instance of the general problem of multi-robot or multi-agent coordination, which can be applied to several domains, such as logistics and transportation, area monitoring, and search and rescue. The results we report show the difference in behavior that the two equivalent solutions have when identical parameters are used.

2 Problem Description

As a case study, consider a system composed by a team of 3 drones flying in open space. The aerial robots fly using only information from onboard sensors and do not communicate with each other. Each drone knows its global position at takeoff and estimates the pose of its local reference frame ${}^w\Sigma_i$ through odometry information. In the following, we assume that all the drones fly at the same altitude and move with a bounded planar velocity. Each drone is equipped with a front-facing camera and can detect the neighbors that are in its field of view. Let ${}^i\mathbf{p}_j = [{}^ix_j, {}^iy_j]^\top$ denote the relative position of drone j with respect to drone i expressed in the local frame ${}^w\Sigma_i$. Each drone j that is detected by i 's camera belongs to the set of neighbors of i , named \mathcal{N}_i . Denote the state of each aerial robot i by ${}^w\chi_i = [{}^wx_i \ {}^wy_i \ {}^w\theta_i]^\top$, where ${}^wx_i, {}^wy_i \in \mathbb{R}$ represent the coordinates of the 2D position of Σ_i in Σ_w , while ${}^w\theta_i$ is the yaw angle. The aerial robot is controlled to obtain the following kinematic model

$${}^w\dot{\chi}_i = {}^wR_i {}^i\mathbf{u}_i = {}^wR_i \begin{bmatrix} {}^iv_x \\ {}^iv_y \\ {}^i\omega \end{bmatrix}, \quad (1)$$

where ${}^iv_x, {}^iv_y, \omega \in \mathbb{R}$ represent the linear and angular velocities of the drone in Σ_i .

3 CBF Formulations

Following CBF theory [1], let $\mathcal{C} = \{\mathbf{p} \in \mathbb{R}^m : h(\mathbf{p}) \geq 0\}$ be the set of configurations that satisfy the safety requirements for the system, also known as the safe set \mathcal{C} which is defined as the super level set of a smooth function $h : \mathbb{R}^m \rightarrow \mathbb{R}$ with $\frac{\partial h}{\partial \mathbf{p}}(\mathbf{p}) \neq 0 \forall \mathbf{p} \in \partial \mathcal{C}$. According to Nagumo's theorem [5], with the extension provided in [2] and [3], CBFs are defined by the following relation:

$$\exists \mathbf{u} \text{ s.t. } \dot{h}(\mathbf{p}, \mathbf{u}) \geq -\alpha(h(\mathbf{p})) \iff \mathcal{C} \text{ is invariant,}$$

where α is an extended class \mathcal{K} function. With this condition it is possible to synthesize optimization-based controllers that modify the desired control input \mathbf{u}^* in a minimally invasive fashion, solving the following optimization problem:

$$\begin{aligned} \mathbf{u}(\mathbf{p}) = \arg \min_{\mathbf{u} \in \mathbb{R}^m} & \frac{1}{2} \|\mathbf{u} - \mathbf{u}^*\|^2 \\ \text{s.t. } & \dot{h}(\mathbf{p}, \mathbf{u}) \geq -\alpha(h(\mathbf{p})). \end{aligned} \quad (2)$$

This controller aims therefore at providing the control input \mathbf{u} closest to the desired input \mathbf{u}^* that satisfies the constraints. This formulation of CBF is named *zeroing*, due to the definition of the $h(\cdot)$ function that is zero at the border of the safe set. A different formulation that, instead, employs a function that becomes infinite at the border, is the *reciprocal* control barrier function, where the following quantity is defined: $B(\mathbf{x}) = 1/h(\mathbf{x})$. The corresponding constraint in the optimization problem becomes

$$\dot{B}(\mathbf{x}, \mathbf{u}) \leq \alpha(h(\mathbf{x}))$$

In the considered case study, introduced in Sect. 2, CBFs are employed together with Control Lyapunov Functions (CLFs) to achieve two objectives. The first objective is to maintain each detected neighbor, for each drone, inside the field of view of its camera. The second objective is to achieve a desired formation through CLFs. The formation is achieved by instructing the neighbors to be at target distance from each other. To achieve the first objective we propose the use of the following $h(\cdot)$ functions.

$$h_l(p_j^i) = -\frac{(y_j^i - \tan(\alpha/2) * x_j^i)}{\sqrt{(\tan(\alpha/2))^2 + 1}} \quad (3)$$

$$h_r(p_j^i) = \frac{(y_j^i + \tan(\alpha/2) * x_j^i)}{\sqrt{(\tan(\alpha/2))^2 + 1}} \quad (4)$$

This represents the distance of the neighbor drone j with respect to the left and right border of the camera field of view. The second objective is expressed through the design of a Lyapunov Function $V(\cdot)$.

$$V(p_j^i) = (\|p_j^i\|^2 - d_t)^2 \quad (5)$$

The Lyapunov function is inserted through a constraint in the optimization problem, as follows:

$$\begin{aligned} \arg \min_{^i \mathbf{u}_i \in \mathbb{R}^m} & (^i \mathbf{u}_i - ^i \mathbf{u}_i^*)^T H (^i \mathbf{u}_i - ^i \mathbf{u}_i^*) \\ \text{s.t. } & \dot{h}_r(^i \mathbf{p}_j, ^i \mathbf{u}_i) \geq -\alpha(h_r(^i \mathbf{p}_j)) , \forall j \in \mathcal{N}_i. \\ & \dot{h}_l(^i \mathbf{p}_j, ^i \mathbf{u}_i) \geq -\alpha(h_l(^i \mathbf{p}_j)) , \forall j \in \mathcal{N}_i. \\ & \dot{V}(^i \mathbf{p}_j, ^i \mathbf{u}_i) \leq -\lambda V(^i \mathbf{p}_j) , \forall j \in \mathcal{N}_i. \end{aligned} \quad (6)$$

The slack variable λ is introduced to soften the constraint, thus helping to get feasible solutions.

The reciprocal version of the same problem is obtained as

$$\begin{aligned} \arg \min_{^i\boldsymbol{u}_i \in \mathbb{R}^m, \epsilon} & (^i\boldsymbol{u}_i - ^i\boldsymbol{u}_i^*)^T H (^i\boldsymbol{u}_i - ^i\boldsymbol{u}_i^*) \\ \text{s.t. } & \dot{B}_r (^i\boldsymbol{p}_j, ^i\boldsymbol{u}_i) \leq \alpha(h_r (^i\boldsymbol{p}_j)) , \forall j \in \mathcal{N}_i. \\ & \dot{B}_l (^i\boldsymbol{p}_j, ^i\boldsymbol{u}_i) \leq \alpha(h_l (^i\boldsymbol{p}_j)) , \forall j \in \mathcal{N}_i. \\ & \dot{V} (^i\boldsymbol{p}_j, ^i\boldsymbol{u}_i) \leq \epsilon - \lambda V (^i\boldsymbol{p}_j) , \forall j \in \mathcal{N}_i. \end{aligned} \quad (7)$$

In the optimization problem the desired velocity for the robot $^i\boldsymbol{u}_i^*$ is used to define the desired behavior. In the validation experiments discussed in the next section, we will define it such that drone 1 moves along a circumference.

4 Results and Discussion

Figure 1 depicts the motion of the three drones controlled with CBF/CLF optimization. It is possible to note that the motion obtained using the zeroing version is slightly smoother. Figure 3 displays the values for the $h(\cdot)$ functions for all the drones. To make a comparison, the values of B in the reciprocal version has been transformed to corresponding h values. As can be seen from Fig. 3, the reciprocal version is imposing a more constraining bound on the motion. This explains why the motion is smoother with the zeroing. Figure 2 displays the behavior of V for each pair of neighboring drones. It is evident that, with more freedom of motion, the value of V is subject to more oscillations in the zeroing case.

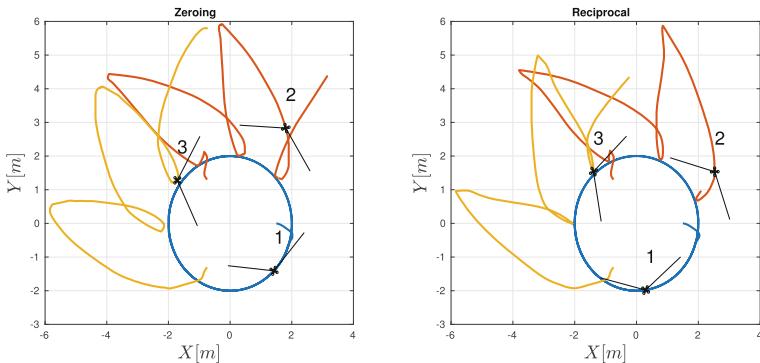
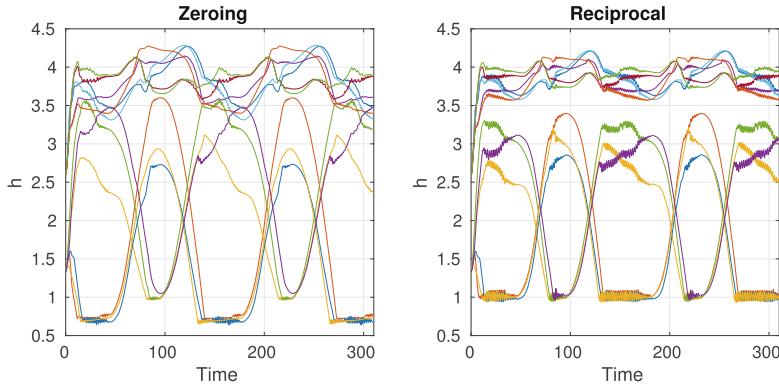
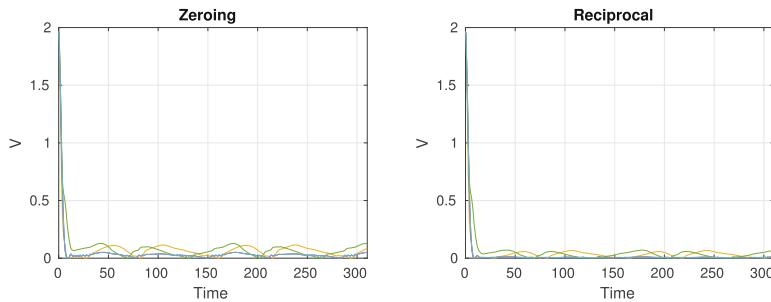


Fig. 1. Motion of the three drones, with drone 1 instructed to follow a circumference

**Fig. 2.** Comparison considering the value of V **Fig. 3.** Comparison considering the value of h

5 Conclusions

In this paper we proposed a comparison between two formulations of CBFs, namely zeroing and reciprocal. Such comparison was performed considering a coordination problem for multi-robot systems. While both formulations lead to correctly achieving the desired objective, we show that some differences in the motion patterns are generated. Results were collected in a simplified simulation scenarios, that however is representative of realistic situation, thus paving the way to application in the real world.

References

1. Ames, A.D., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K., Tabuada, P.: Control barrier functions: theory and applications. In: 18th European Control Conference (ECC), pp. 3420–3431 (2019). <https://doi.org/10.23919/ECC.2019.8796030>
2. Ames, A.D., Grizzle, J.W., Tabuada, P.: Control barrier function based quadratic programs with application to adaptive cruise control. In: 53rd IEEE Conference on Decision and Control, pp. 6271–6278 (2014). <https://doi.org/10.1109/CDC.2014.7040372>

3. Ames, A.D., Xu, X., Grizzle, J.W., Tabuada, P.: Control barrier function based quadratic programs for safety critical systems. *IEEE Trans. Autom. Control* **62**(8), 3861–3876 (2017). <https://doi.org/10.1109/TAC.2016.2638961>
4. Ferraguti, F., et al.: Safety and efficiency in robotics: the control barrier functions approach. *IEEE Robot. Autom. Mag.* **29**(3), 139–151 (2022)
5. NAGUMO, M.: Über die lage der integralkurven gewöhnlicher differentialgleichungen. *Proc. Phys.-Math. Soc. Jpn. 3rd Ser.* **24**, 551–559 (1942). https://doi.org/10.11429/ppmsj1919.24.0_551



Virtual Forward Dynamics Models Applied to Orbital Robotics Scenarios

Mohatashem Reyaz Makhdoomi^{1(✉)}, Vivek Muralidharan¹, Juan Sandoval², Miguel Olivares-Mendez¹, and Carol Martinez¹

¹ Space Robotics Research Group (SpaceR), Interdisciplinary Research Centre for Security, Reliability and Trust (SnT), University of Luxembourg, Esch-sur-Alzette, Luxembourg

mohatashem.makhdoomi@uni.lu

² Laboratoire des Sciences du Numérique de Nantes (LS2N), Ecole Centrale de Nantes, Nantes, France

Abstract. The paradigm of applying the virtual forward dynamics model-based control (VFDM) for robotic manipulation in the field of orbital robotics is presented through two use cases: The first case explores the use of VFDM to emulate on-orbit scenarios using robotic arms in a ground-based facility. The second case explores VFDM for essential on-orbit manipulation tasks performed on a mockup of a control panel (“push-button, release” and “peg-in-hole”). Experiments confirm the feasibility of the approach for its use in space applications. VFDM offers a viable solution for a variety of operational scenarios that lack elaborate models. It excels in exhibiting stability at kinematic singularities during cartesian motion control, all the while allowing the conditioning of the manipulator’s inertia matrix for imposing a desired dynamic behaviour.

Keywords: On-orbit Manipulation · Forward Dynamics · Cartesian Control · Virtual Models · Spacecraft On-Ground Emulation

1 Introduction

Operations such as spacecraft docking, satellite refueling, servicing and assembly, are critical for space exploration and utilization. On-ground facilities are built for validating spacecraft and robotic systems for such missions [1]. Robotic arms form a key component of these facilities. Emulation of on-orbit operations involves direct or indirect interactions with the environment and employing force and torque sensors (F/T) for interaction control.

Robots mainly employed in these facilities are equipped with joint torque sensors that enable low-level F/T control, such as impedance control. However, these robots are generally expensive compared to motion-controlled robots that use admittance control for interaction. Both methods require dynamic models of the system while handling issues such as kinematic singularities, properly conditioned inertia matrix, coupled terms of the Jacobian matrix, etc. As most robots

available in the market are motion-controlled, with the increase in low-cost space ventures, this work investigates methods that improve the interaction capabilities of motion-controlled robots to get similar benefits to torque-controlled robots.

This paper explores the use of Virtual Forward Dynamics Models (VFDM)-based control strategies capable of handling (1) non-trivial issues such as kinematic singularities, properly conditioned inertia matrix, and coupled Jacobian matrix without elaborate dynamics models, and (2) different modalities of interaction. To this effect, the use of VFDM-based control of robotic manipulators is proposed for two space applications. Firstly, combined with an Orbital Dynamics Simulator, the feasibility of VFDM for on-ground validation of on-orbit operations is demonstrated [2]. Secondly, this method also presents a decoupled virtual dynamic model that allows for both a singularity-robust inverse-kinematic algorithm and the possibility of robot compliance control for various applications in interactive robotic manipulation.

2 Virtual Manipulator Model for Forward Dynamics Simulation

Forward dynamics involves modeling the applied forces to predict the motion of a virtual robot, which is then translated to a real robot with the same kinematic consistency to achieve the desired dynamic behavior. Consider the “virtual” robotic manipulator with kinematically identical to its real counterpart modeled as

$$\tau = \mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) \quad (1)$$

where $\mathbf{H}(\mathbf{q})$ is the positive definite inertia matrix; $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ includes the Coriolis and Centrifugal terms; $\mathbf{G}(\mathbf{q})$ denotes the effect of gravity; τ represents the joint torques; $\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}$ are the joint space accelerations, velocities, and positions, respectively. The joint torques τ and end-effector force \mathbf{f} are related to the virtual model’s Jacobian \mathbf{J} as $\tau = \mathbf{J}^T \mathbf{f}$. With a few assumptions: (1) the virtual model does not need to account for gravity, (2) the robot accelerates from zero in each control cycle, and (3) within each cycle the robot motion is very slow. Consequently, the terms $\mathbf{G}(\mathbf{q})$ and $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ are neglected. Rearranging Eq. (1) to solve for acceleration and omitting the notation \mathbf{q} from $\mathbf{H}(\mathbf{q})$ for brevity yields

$$\ddot{\mathbf{q}} = \mathbf{H}^{-1} \mathbf{J}^T \mathbf{f} \quad (2)$$

which is a simplified virtual model, independent of the non-linear joint velocities, described by the “Virtual Model” block in Fig. 1 and relates the Cartesian space forces to instantaneous joint space accelerations. It provides a forward mapping from Cartesian space to the joint space. This simplification paves the way for the implementation of two different interaction strategies.

2.1 Ground Emulation of In-Orbit Scenarios

The upper left and right blocks in Fig. 1 describe the control loop for an emulation scenario. Equation (2) paves the way for the conditioning of the joint-space

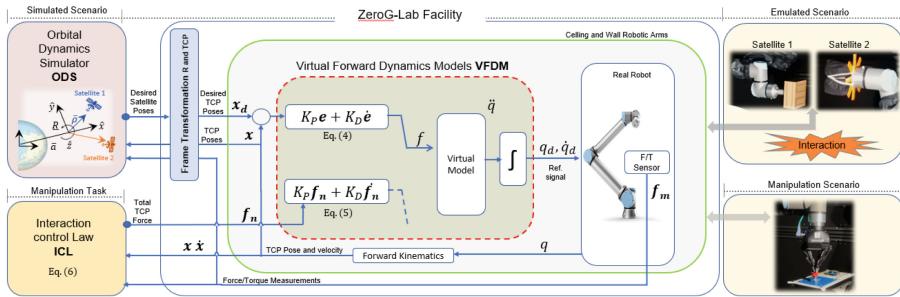


Fig. 1. Block Diagram for Robot Interaction and Emulation

inertia matrix \mathbf{H} where a desired dynamic behavior is obtained by changing the mass distribution of virtual links. This conditioning enables Cartesian Motion Control based on Virtual Forward Dynamics to provide accurate motion tracking and smooth transition between sparsely sampled poses [3]. Additionally, assuming the instantaneous cycle starts at rest, Eq. (2) resembling a Jacobian transpose IK method is rendered stable by dynamic decoupling such as

$$\ddot{\mathbf{x}} = \mathbf{J}\mathbf{q} \rightarrow \ddot{\mathbf{x}} = \overset{0}{\cancel{\mathbf{J}}} \dot{\mathbf{q}} + \mathbf{J}\ddot{\mathbf{q}} = \mathbf{J}\mathbf{H}^{-1}\mathbf{J}^T\mathbf{f} = \mathbf{M}^{-1}\mathbf{f} \quad (3)$$

Through dynamic decoupling $\mathbf{M}^{-1}(= \mathbf{J}\mathbf{H}^{-1}\mathbf{J}^T)$ is made a diagonal, time-invariant matrix across joint configurations to ensure the stability at singular configurations [3].

Formulating a closed-loop control around this virtual model, and defining the error term as $\mathbf{e} = \mathbf{x}_d - \mathbf{x}$, i.e., the difference between the desired target \mathbf{x}_d and current end-effector pose \mathbf{x} is obtained via a forward kinematics (FK) routine. Then, the control input \mathbf{f} to the forward model is given by Eq. 4 and expressed as a proportional derivative PD control law, where \mathbf{e} is a pose vector.

$$\mathbf{f} = \mathbf{K}_P \mathbf{e} + \mathbf{K}_D \dot{\mathbf{e}} \quad (4)$$

Experiment: Controlled Collision During Free-Floating Behavior. Two satellite mockups are mounted on the robots of the ZeroG-Lab facility at the University of Luxembourg [4]. The inertial properties and orbital parameters of the mockups are specified in the orbital dynamics simulator (ODS). An initial velocity drives the mockup toward each other. Upon collision, the forces and moments acting on the mockups are measured by inbuilt F/T sensors of the robot, based on which the on-orbit simulator, through proper force/torque transformations, generates trajectories on the fly. Figure 2 illustrates the setup and the corresponding impact on the robots' trajectory pre and post-collision. A comparable consistency in the simulation (desired) and HIL (actual velocity profile) similar to those available in literature [5] are observed, thereby confirming the feasibility of this method. In these experiments, compliant robot motion

is achieved based on real-time information received by the ODS. Despite the complex motion, VFDM-based control ensures stable robot behavior close to any singular configurations ([see video here](#)).

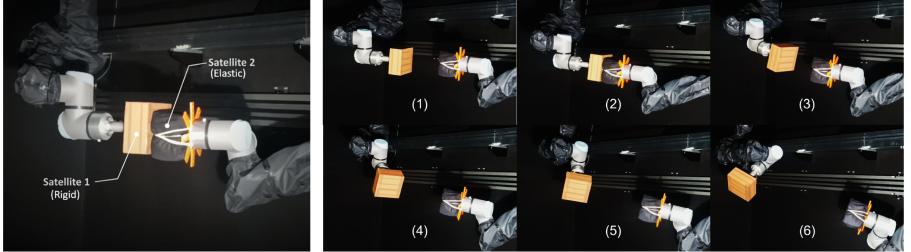


Fig. 2. Impact/Collision Experiment. (Left) Setup. (Right) Sequence of satellite states: (1) Before impact, (2) During impact, (3)-(6) After impact.

2.2 On-Orbit Robotic Manipulation

A closed-loop control is formulated to regulate forces and motion set points describing its interaction state. It involves reference motion tracking in free space in the presence of restoring forces. The bottom left and right blocks in Fig. 1 depict the control scheme. A PD regulator, as in literature, is adopted as [3]:

$$\mathbf{f} = \mathbf{K}_P \mathbf{f}_n + \mathbf{K}_D \dot{\mathbf{f}}_n \quad (5)$$

where \mathbf{K}_P and \mathbf{K}_D are positive semi-definite matrices. In the context of the current implementation of the control framework [6], the direct effect of the damping matrix \mathbf{D} is ignored imposing a spring-like behavior on the system. Consequently, the net force \mathbf{f}_n is described as

$$\mathbf{f}_n = \mathbf{f}_d - \mathbf{f}_m + \mathbf{K} (\mathbf{x}_d - \mathbf{x}) + \mathcal{D}^0 (\dot{\mathbf{x}}_d - \dot{\mathbf{x}}) \quad (6)$$

where \mathbf{f}_d is the desired force, \mathbf{f}_m is the force measured by the force/torque sensor, $\dot{\mathbf{x}}_d$ and \mathbf{x}_d are the desired velocity and the desired pose, while as $\dot{\mathbf{x}}$ and \mathbf{x} are the actual velocity and pose information obtained from forward kinematics. Here \mathbf{K} and \mathbf{D} denote the stiffness and damping matrices, respectively. With Eq. (6) as input to the control law described in Eq.(5), the quantities of acceleration in the virtual model are available and are offered as inputs to the low-level joint controllers of the real robot. These forward dynamics computations are iterative and run independently of the real robot's internal motion control cycle.

Experiment: Compliance Control for Manipulation Tasks. In this experiment, as demonstrated in Fig. 3, two operations were performed under compliance control mode using a robotiq 3-fingers gripper mounted on the UR10 robot of the ZeroG-Lab. Firstly, the press-and-release of an emergency button (left image). Then, a peg-in-hole task (right image). These operations ([See video here](#)) are performed analogously to robotic manipulation activities in space such as instrumentation panel manipulation, component insertion, etc. The robot follows a cartesian motion and upon reaching the goal, interaction is observed. Using the feedback from the force-torque sensor, VFDM helps implement the required compliance, modulating the interaction between the robot and the environment, while guaranteeing stable motion for the successful task execution.

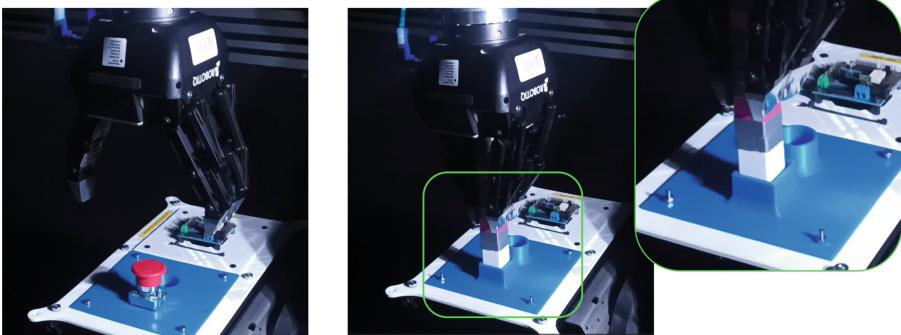


Fig. 3. Manipulation Experiment. (Left) Press-and-release button and (Right) peg insertion (Right).

3 Conclusion

This work showcases the application of a virtual forward dynamics-based compliance control approach in two distinct space-related scenarios. The first experiment demonstrates the method's effectiveness in emulating spacecraft interactions on the ground using a collision-based setup. The second set of experiments highlights its suitability for orbital servicing tasks, including peg-in-hole and instrumentation panel manipulation. The VFDM method excels in addressing kinematic singularities during motion planning and handling scenarios with challenging Jacobian matrices. Nevertheless, it has its limitations, such as limited gravity compensation and the need for manual control parameter tuning, which will be addressed in future work.

References

1. Papadopoulos, E., Aghili, F., Ma, O., Lamariello, R.: Robotic manipulation and capture in space: a survey. *Front. Robot. AI*, 228 (2021)
2. Muralidharan, V., et al.: Rendezvous in cislunar halo orbits: hardware-in-the-loop simulation with coupled orbit and attitude dynamics. *Acta Astronaut.* **211**, 556–573 (2023)
3. Scherzinger, S.: Human-inspired compliant controllers for robotic assembly. PhD thesis, Karlsruher Institut für Technologie (KIT) (2021)
4. Olivares-Mendez, M., et al.: Zero-g lab: a multi-purpose facility for emulating space operations. *J. Space Saf. Eng.* (2023)
5. Aghili, F., Namvar, M.: Scaling inertia properties of a manipulator payload for 0-G emulation of spacecraft. *Int. J. Robot. Res.* **28**(7), 883–894 (2009)
6. Cartesian controllers. https://github.com/fzi-forschungszentrum-informatik/cartesian_controllers. Accessed 30 Aug 2022



Inverse Kinematics for Robotic Manipulators Using Iterative Preconditioned Optimization Algorithms

Tianchen Liu^(✉), Lasitha Weerakoon, and Nikhil Chopra

Department of Mechanical Engineering, University of Maryland,
College Park, MD 20742, USA
{tianchen,lasitha,nchopra}@umd.edu

Abstract. In this paper, we employ previously developed algorithms, the iteratively preconditioned gradient-descent (IPG) algorithm and its variant with momentum terms, to solve the inverse kinematics (IK) problem for robotic manipulators. Our algorithms belong to the class of quasi-Newton methods. By selecting appropriate hyperparameters, the algorithms can potentially achieve faster convergence than commonly used optimization algorithms. A two-link planar robot and a Universal Robots UR3 cobot are used as examples to present the performance of the algorithms.

Keywords: Inverse kinematics · Optimization

1 Introduction

The inverse kinematics (IK) problem is to find a set of joint values that reach a desired pose for the robot's end-effector [2]. While the IK problem can be solved by geometric calculations, closed-form IK solution is challenging for complex robot geometry. Hence, an advanced approach is to formulate IK as a nonlinear least-squares problem to minimize a cost function regarding the joint values and the desired location of the end-effector. Optimization methods are attractive for solving the IK problem due to their ability to handle the complexity and nonlinearity simultaneously and to achieve a high level of numerical precision. Algorithms, such as Broyden-Fletcher-Goldfarb-Shannon (BFGS) [15], damped least-squares method or Levenberg-Marquardt (L-M) algorithm [11,13,14] have been employed for the IK problem. Additional methods include heuristic and nature-inspired optimization algorithms (e.g., [1,12]), especially for finding the global optimal solution.

In this paper, we consider the inverse kinematics for a generic robotic manipulator via the following unconstrained convex optimization problem:

$$\min_q F(q) = \|f(q) - x_d\|_2^2, \quad (1)$$

This work was partially supported by Army Research Office (Award 310251-00001).

where $f(q) : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is the forward kinematics of the manipulator given configuration $q \in \mathbb{R}^n$, and $x_d \in \mathbb{R}^m$ is the desired location of the end-effector.

As a preliminary study, in this paper the objective is to compute the best configuration q^* to reach a particular end-effector pose. Requirement of link orientations and constraints on the joints, including collision avoidance requirements, bounds of joint values, etc., are not considered. These topics will be addressed in our future development of the preconditioned algorithms.

2 Optimization Algorithms

A first-order method to find the optimal solution of (1) is the gradient descent (GD) approach. For iteration k , the difference between desired and actual end-effector position is defined as $\Delta x = x_k - x_d$. We denote the Jacobian as $J = \begin{bmatrix} \frac{\partial f}{\partial q} \end{bmatrix} \in \mathbb{R}^{m \times n}$. By forward kinematics, $x_k = f(q_k)$, then $\Delta q_k = J^{-1} \Delta x_k$ if $n = m$, or $\Delta q_k = J^\dagger \Delta x_k$ if $n \neq m$, where J^\dagger is the pseudo-inverse of J . Hence, an update step can be calculated to drive the actual end-effector pose to converge to desired one: $q_{k+1} = q_k + \alpha \Delta q_k$. $\alpha \in \mathbb{R}^+$ is a selected step size. The process repeats until $\|\Delta q\|$ is less than a preset small positive tolerance value ε .

In this paper, we will investigate to employ two algorithms from our previous work [3, 7] to solve the inverse kinematics problem of robotic manipulators. The algorithms belongs to the class of quasi-Newton methods, and are proposed to find best value q^* that minimizes unconstrained convex optimization problem $F(q)$. The main steps of the algorithms are presented as in Algorithm 1 and 2.

Algorithm 1. IPG algorithm (centralized formulation) [3, 8]

```

Initialize  $q_0 \in \mathbb{R}^n$ ,  $K_0 \in \mathbb{R}^{n \times n}$ . Select scalar constants  $\beta \in [0, 1)$ ,  $\delta \in \mathbb{R}^+$ .
while not converged do
     $q_{k+1} = q_k - \delta K_k g_k$ 
     $K_{k+1} = K_k - \alpha_k [(H_k + \beta I)K_k - I]$ 
end while

```

Algorithm 2. IPG w/ momentum (IPG-Mo) (centralized formulation) [7]

```

Initialize  $q_0 \in \mathbb{R}^n$ ,  $K_0 \in \mathbb{R}^{n \times n}$ . Select scalar constants  $\beta \in [0, 1)$ ,  $\delta \in \mathbb{R}^+$ ,  $\mu_1 \in [0, 1)$ .
while not converged do
     $\hat{g}_{k+1} = g_k + \mu_1 \hat{g}_k$ 
     $q_{k+1} = q_k - \delta K_k \hat{g}_{k+1}$ 
     $K_{k+1} = K_k - \alpha_k [(H_k + \beta I)K_k - I]$ 
end while

```

The notations in the algorithms are as follows: q_0 and K_0 are the initial values for the optimizing variables and the preconditioner matrix. Subscript k indicates the value at iteration k , and g_k and H_k denote the gradient and Hessian

of $F(q)$ with respect to q , evaluated at $q = q_k$. $I \in \mathbb{R}^{n \times n}$ denotes the identity matrix. α_k is the step-size to be selected following convergence conditions. For the variant with momentum terms, μ_1 is a weighting parameter for incorporating previous gradient values. The detailed description and conditions for convergence can be found in literature [3, 7, 8]. Specifically for IK problems, g_k and H_k are evaluated by

$$g_k = J^T(f(q_k) - x_d), \quad H_k = \frac{\partial}{\partial q}(J^T(f(q_k) - x_d)). \quad (2)$$

Then the gradient and Hessian can be computed in our iterative algorithms.

3 Experiments

In this section, the proposed algorithms noted in Sect. 2 are implemented to solve IK problems of two types of industrial robotic manipulators: i) a two-link planar manipulator, and ii) a UR3 cobot. We implemented the GD method for the first example, and utilized the BFGS and L-M algorithms from the IK solver in Matlab Robotics Toolbox [4] to compare the results.

3.1 Two-Link Planar Manipulator

The two-link manipulator is a simple configuration, whose end-effector moves within a planar area. The lengths of links are denoted as $a_1 = 1$ and $a_2 = 1$, and the rotational angles for the joints are $q = (\theta_1, \theta_2)^T$. The end-effector position is denoted as $x \in \mathbb{R}^2$, which can be computed by the following equations:

$$x = \begin{bmatrix} a_1 \cos \theta_1 + a_2 \cos(\theta_1 + \theta_2) \\ a_1 \sin \theta_1 + a_2 \sin(\theta_1 + \theta_2) \end{bmatrix} := f(q) \quad (3)$$

In Fig. 1a, the IK solutions for reaching point $x_d = (0.8, 1.2)^T$ are plotted. GD, BFGS and L-M methods obtain one feasible solution, while our proposed algorithms obtain the other feasible solution. For this desired end-effector point, our algorithms converges in fewer iterations than other methods using $\beta = 0.15$, $\delta = 1$ and $\mu_1 = 0.01$, as shown in the bar chart in Fig. 1a.

The performance for the IPG and IPG-Mo algorithms can be impacted by the selections of the initial guess x_0 and the hyperparameters β, δ, K_0 and μ_1 . We randomly generated 200 positions within the area $\|x\|_\infty \leq 1$, and the algorithm with fewest number of convergence iterations is marked for each point in Fig. 1b. The L-M algorithm converged in fewest iterations for corner positions. However, it is expected that the hyperparameters of IPG and IPG-Mo methods can be tuned to outperform BFGS and L-M algorithms for those desired positions.

3.2 UR3 Cobot

UR3 is a widely used model of the collaborative robots (cobots) manufactured by Universal Robots. The 6-degree-of-freedom (DOF) manipulator is designed

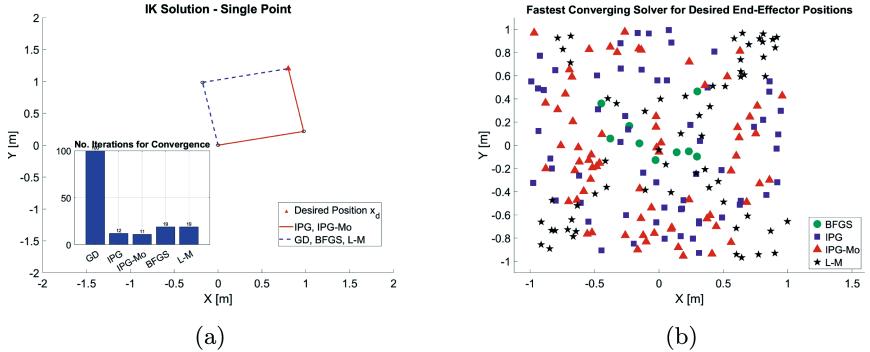


Fig. 1. Two-link manipulator example: (a) IK solutions to $x_d = (0.8, 1.2)^T$, (b) tests for fastest converging algorithm for various desired x_d .

to safely interact and collaborate with human users. The forward and inverse kinematics of the UR robots have been widely studied in previous literature (e.g., [5, 6, 10], etc.). In this section, we use the available kinematic equations to solve the nonlinear optimization problem using our proposed algorithms.

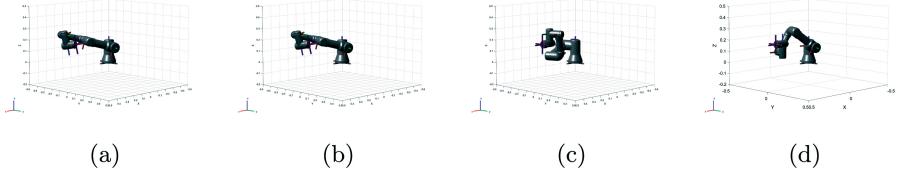


Fig. 2. UR3 manipulator IK solutions to $x_d = (0.4, 0, 0.2)^T$: (a) IPG, (b) IPG-Mo, (c) BFGS, (d) L-M.

Figure 2 shows IK solutions to the desired end-effector position $x_d = (0.4, 0, 0.2)^T$ by IPG, IPG-Mo, BFGS and L-M algorithms. From the visualization, it can be observed that the desired position is achieved by all methods. Since we do not specify the orientation requirements, the poses of the links by the algorithms are different. In this example, our proposed algorithms can potentially outperform other methods regarding number of iterations to converge, however, finding the proper hyperparameter values is more challenging due to more complex configurations.

4 Conclusion

In this paper, we explored the employment of optimization algorithms in our previous work to the inverse kinematics problem. Two robotic manipulator examples are tested to show the performance. However, there are two limitations for this

preliminary research. First, the proposed algorithms have not been compared with other efficient ones (e.g., Halley's method [9]). Second, more generalized IK problems with constraints have not been tested. These issues will be addressed in our future work.

References

1. Ayyildiz, M., Cetinkaya, K.: Comparison of four different heuristic optimization algorithms for the inverse kinematics solution of a real 4-dof serial robot manipulator. *Neural Comput. Appl.* **27**, 825–836 (2016)
2. Buss, S.R.: Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods. *IEEE J. Robot. Autom.* **17**(1–19), 16 (2004)
3. Chakrabarti, K., Gupta, N., Chopra, N.: On accelerating distributed convex optimizations. arXiv preprint: [arXiv:2108.08670](https://arxiv.org/abs/2108.08670) (2021)
4. Corke, P.I.: A robotics toolbox for MATLAB. *IEEE Robot. Autom. Mag.* **3**(1), 24–32 (1996)
5. Husty, M.L., Pfurner, M., Schröcker, H.P.: A new and efficient algorithm for the inverse kinematics of a general serial 6R manipulator. *Mech. Mach. Theory* **42**(1), 66–81 (2007)
6. Kebria, P.M., Al-Wais, S., Abdi, H., Nahavandi, S.: Kinematic and dynamic modelling of UR5 manipulator. In: 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 004229–004234. IEEE (2016)
7. Liu, T., Chakrabarti, K., Chopra, N.: Accelerating the iteratively preconditioned gradient-descent algorithm using momentum. In: the Ninth Indian Control Conference (ICC). IEEE (2023)
8. Liu, T., Chakrabarti, K., Chopra, N.: Iteratively preconditioned gradient-descent approach for moving horizon estimation problems. In: the 62nd IEEE Conference on Decision and Control (CDC). IEEE (2023)
9. Lloyd, S., Irani, R.A., Ahmadi, M.: Fast and robust inverse kinematics of serial robots using Halley's method. *IEEE Trans. Rob.* **38**(5), 2768–2780 (2022)
10. Mathiassen, K., Fjellin, J.E., Glette, K., Hol, P.K., Elle, O.J.: An ultrasound robotic system using the commercial robot UR5. *Front. Robot. AI* **3**, 1 (2016)
11. Nakamura, Y., Hanafusa, H.: Inverse kinematic solutions with singularity robustness for robot manipulator control. *J. Dyn. Syst. Meas. Contr.* **108**(3), 163–171 (1986)
12. Savsani, P., Jhala, R.L., Savsani, V.J.: Comparative study of different metaheuristics for the trajectory planning of a robotic arm. *IEEE Syst. J.* **10**(2), 697–708 (2014)
13. Sugihara, T.: Solvability-unconcerned inverse kinematics by the Levenberg–Marquardt method. *IEEE Trans. Rob.* **27**(5), 984–991 (2011)
14. Wampler, C.W.: Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. *IEEE Trans. Syst. Man Cybern.* **16**(1), 93–101 (1986)
15. Zhao, J., Badler, N.I.: Inverse kinematics positioning using nonlinear programming for highly articulated figures. *ACM Trans. Graph. (TOG)* **13**(4), 313–336 (1994)



Accurate Registration of Transparent Objects in 2D LiDAR SLAM

Gurtajbir Singh Herr^(✉), Lasitha Weerakoon, and Nikhil Chopra

Department of Mechanical Engineering , University of Maryland,
College Park, MD 20742, USA
{gurtaj89,lasitha,nchopra}@umd.edu

Abstract. Transparent objects and surfaces present a unique challenge to the application of LiDAR for indoor mobile robotics. This is due to the fact that the majority of the incident light (except for normal incidence) passes through such objects and results in them becoming “invisible” to the resulting map. In this work, we exploit the sharp intensity spikes that occur when an incident beam strikes a transparent surface perpendicularly, and keep an accurate account of such points. This enables us to make the representation unaffected by the drift accumulated as the robot moves through the environment. Our method utilizes the Pose Graph Optimization (PGO) framework and exhibits superior results when compared to an existing approach that utilizes intensity spikes for detecting transparent surfaces but uses a particle filter based approach for generating the map of the environment. We showcase how, by leveraging the inherent structure of a PGO framework, the transparent objects can be mapped accurately.

Keywords: LiDAR · SLAM · Pose Graph Optimization

1 Introduction

The development of Simultaneous Localization and Mapping (SLAM) algorithms has enabled precise mapping of indoor and outdoor environments [1]. Light detection and ranging sensors (LiDARs) have become increasingly popular for this task due to their attractive features such as precision and long-range operation. However, they perform poorly in the presence of transparent objects and reflective surfaces. As transparent walls and partitions are used more and more for aesthetic appeal and energy-efficient interior lighting in modern building designs, detecting and including such transparent obstacles in the map is required for reliable and safe robot navigation.

A LiDAR relies on the reflection of the incident beam from an object to measure the range to it. Diffuse surfaces disperse the beam equally in all directions, including directly back to the source, thus enabling detection. This dispersion causes a decrease in the received intensity [3]. On the other hand, transparent surfaces transmit most of the incident light while reflecting the rest away. It is

only at a normal angle of incidence that the light is reflected back to the source and this is characterized with a sharp increase in received intensity.

There have been several approaches that have tried to address the challenge of detection and mapping glass points. Foster et al. [3] consider that glass behaves like a diffuse object in a minimal range of angles (near-normal incidence) and thus gets treated as noise by the SLAM framework. They augment standard occupancy grid mapping by tracking the subset of angles from which glass objects are visible and reconstructing the objects from these angles assuming there is some localization error. By examining the intensity profile of the returned laser beam around normal incidence to the glass surface, a simple detection scheme is utilized by Wang and Wang [6]. This is then incorporated into GMapping, an existing particle filter based SLAM approach [5]. Another common practice is to utilize additional sensors and employ sensor fusion to deal with the detection and inclusion of glass into the map [2, 7]. In these approaches, sonars are commonly used as the additional sensing modality.

In our present work, we exploit the sharp spike in reflected beam intensity to detect the presence of glass and other transparent surfaces as introduced in [6]. As the robot moves through the environment these identified points are maintained in a separate data structure. We show how these points can be mapped accurately onto the map of the environment by utilizing the history of poses maintained by the PGO framework. We compare the maps produced by our proposed approach to those generated by the existing method [6].

In the remainder of the paper, we will refer to transparent *glass-like objects* and glass objects, interchangeably.

2 Transparent Object Mapping via a Pose Graph Optimization Framework

2.1 Glass Detection

We inspect the LiDAR scan's intensity profile to identify an intensity peak if present, to detect glass surfaces as introduced in [6]. For false proof detection, we use an intensity threshold value (*thresh*), an intensity gradient (*grad*) and a profile width (*width*) as tunable parameters. Experiments with different transparent materials were conducted and a set of parameters was chosen as shown in Fig. 1.

2.2 Glass Mapping Methodology

The Pose Graph Optimization (PGO) framework [4], as the name suggests, consists of a graph where the nodes of the graph represent the poses of the robot. The edges between the nodes act as the constraints for the optimization problem. Successive nodes are connected by transformations that can be obtained by scan matching or by a reliable odometry source. Over time, drift may accumulate in the system and this results in the pose estimates becoming further from the true

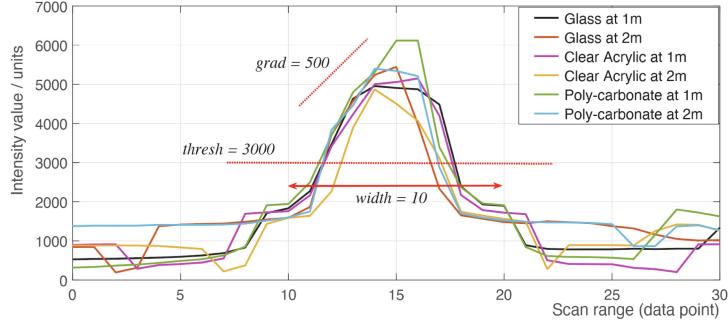


Fig. 1. Intensity profiles for different materials and set of selected parameters

values. This drift can be corrected by incorporation of what are known as *loop closure* constraints. When the robot visits a portion of the environment that it had seen earlier, a *loop closure* constraint is inserted between the current node and a node from its previous visit. When the global optimization (consisting of all the poses and constraints thus far) is performed, the presence of *loop closure* constraints significantly reduces the drift. Our method is able to take advantage of this correction and able to deliver a consistent estimate of the glass points.

We incorporate the glass detection methodology along with a PGO framework. As a new LiDAR scan is received, it is either accepted or rejected by a motion detector based on whether the robot has moved or rotated beyond a certain threshold. This is done to avoid accumulation of redundant information. The accepted scan is then processed through a glass detection algorithm. If the scan contains any glass points, the indices of those points are recorded. This information equips us to find the local co-ordinates of the glass points with respect to the LiDAR via a transformation of the polar co-ordinates (range and angle in the scan) to the local Cartesian co-ordinates in the LIDAR frame. If it is determined that a scan S has glass points at indices given by a set I , the local co-ordinates for each glass point $j \in I$ are obtained as $X_j^{local} = range_j \times \cos(\theta_j)$ and $Y_j^{local} = range_j \times \sin(\theta_j)$ where θ_j can be obtained by using the index of the point j in the scan and the angular resolution of the LiDAR.

A map of the environment, including the identified glass surfaces can be obtained at any time by utilizing the latest trajectory (poses of nodes) and the data structure containing the detected glass points. The global co-ordinates of the detected points can be recovered and plotted onto the map generated by the PGO framework. If a scan S is associated with a node N of the pose graph, and H represents its homogeneous transformation matrix with respect to the origin, the global co-ordinates of the glass point, $\{X_j^{global}, Y_j^{global}\}$ where $j \in I$, are obtained as follows:

$$\begin{bmatrix} X_j^{global} \\ Y_j^{global} \\ 1 \end{bmatrix} = H \begin{bmatrix} X_j^{local} \\ Y_j^{local} \\ 1 \end{bmatrix}$$

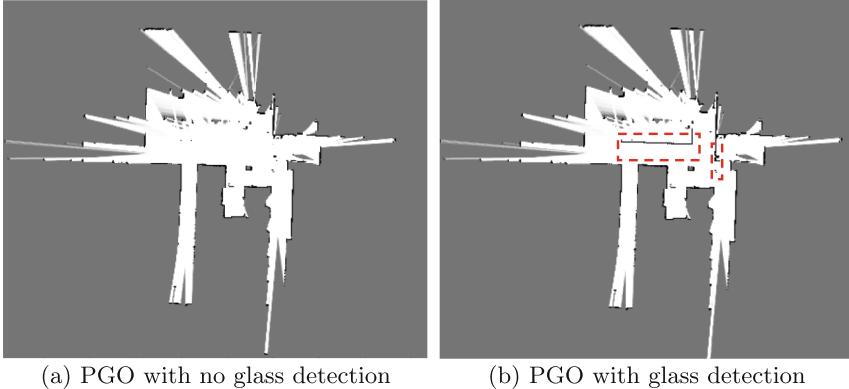


Fig. 2. Comparison of maps generated by a PGO framework without and with glass detection where dashed boxes show glass surface undetected by the former

3 Results and Discussion

The effectiveness of the proposed approach is demonstrated by applying it to two different scenarios. In the first case, we apply an off-the-shelf PGO framework from the MATLAB Navigation toolbox to an open source dataset [6]. The results are compared to the same PGO framework augmented with our proposed solution. The parameters used include an intensity threshold of $thresh = 8000$, an intensity gradient of $grad = 4000$ and a profile width of $width = 5$ as suggested in [6]. The ability of the proposed method to successfully detect and add the transparent surfaces to the final map can be seen by comparing from Fig. 2 (a) and Fig. 2 (b) where the regions marked in dashed red in Fig. 2 (a) are stretches of glass that are absent in Fig. 2 (b).

In the second case, we compare the performance of our augmented PGO framework with the method proposed in [6] on a different segment of the open source dataset. Our method is able to recover a greater portion of the glass surface as shown in Fig. 3 when compared with the *slam_glass* algorithm [6].

It should be noted that in spite of the performance of proposed method, there are some undetected glass sections in maps generated by the proposed method. The proposed glass detection method is contingent on the fact that near normal incidence results in spikes in reflected beam intensity. The glass panels might not have been precisely vertical, or the ground might have had some inclination. Additionally, identifying heterogeneous surfaces and labeling them accordingly can be an important consideration in future work. This is especially important when non transparent shiny surfaces reflect incident beams with high intensity thus mimicking the effect produced by shiny transparent surfaces.

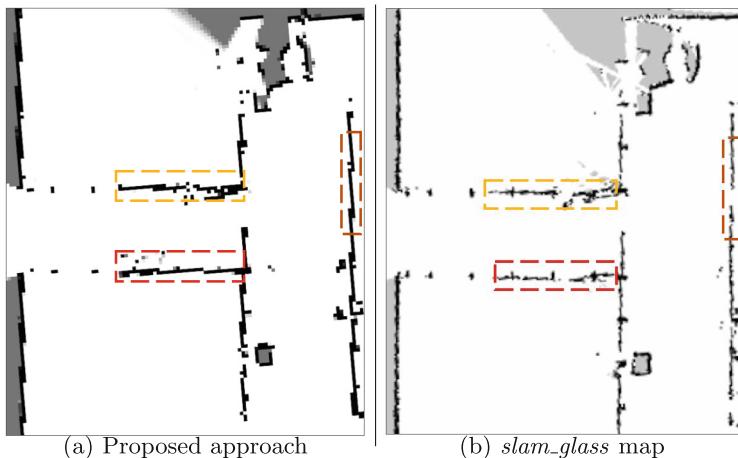


Fig. 3. Generated maps for a part of the dataset from [6]. Dashed boxes in same color correspond to the same region.

References

1. Cadena, C., et al.: Past, present, and future of simultaneous localization and mapping: toward the robust-perception age. *IEEE Trans. Rob.* **32**(6), 1309–1332 (2016)
2. Diosi, A., Kleeman, L.: Advanced sonar and laser range finder fusion for simultaneous localization and mapping. In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, pp. 1854–1859. IEEE (2004)
3. Foster, P., Sun, Z., Park, J.J., Kuipers, B.: Visagge: visible angle grid for glass environments. In: 2013 IEEE International Conference on Robotics and Automation, pp. 2213–2220. IEEE (2013)
4. Grisetti, G., Kümmerle, R., Stachniss, C., Burgard, W.: A tutorial on graph-based slam. *IEEE Intell. Transp. Syst. Mag.* **2**(4), 31–43 (2010)
5. Grisetti, G., Stachniss, C., Burgard, W.: Improved techniques for grid mapping with Rao-Blackwellized particle filters. *IEEE Trans. Rob.* **23**(1), 34–46 (2007)
6. Wang, X., Wang, J.: Detecting glass in simultaneous localisation and mapping. *Robot. Auton. Syst.* **88**, 97–103 (2017)
7. Wei, H., Li, X., Shi, Y., You, B., Xu, Y.: Fusing sonars and LRF data to glass detection for robotics navigation. In: 2018 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 826–831. IEEE (2018)



Generic Nuclear Robotics Architecture (GNRA) A Standard for Nuclear Robotics Electronic Architectures and Interoperability

Ipek Caliskanelli¹(✉), Periklis Charchalakis², Matthew Goodliffe¹, Tomoki Sakue³, Fumiaki Abe³, Elias Stipidis², and Robert Skilton¹

¹ Remote Applications in Challenging Environments, UK Atomic Energy Authority, Abingdon, UK

Ipek.Caliskanelli@ukaea.uk

² Vetrnoics Research Centre, Brighton, UK

³ Tokyo Electric Power Company (TEPCO), Tokyo, Japan

Abstract. The primary objective of this work is to research, design, and draft an open standard for nuclear robotics, aiming to standardise the design, development, integration, interoperability, and overall through life capability management of technologies, devices, and platforms for robotic systems used in nuclear fusion and decommissioning operations. In this paper, we summarise some of the key considerations and features of the proposed standard.

Keywords: nuclear robotics · standards · through life capability management

1 Overview and Background

The commercial robotics industry has achieved great advancements in the past decade, but despite the exponential capability increase it is not easily possible to apply those technologies at large scale directly to sensitive sectors, such as nuclear, due to their strict design, operation, and safety & security requirements. The increasing complexity of the devices, systems, system-of-systems integration, and proprietary/black-box designs make it even more difficult to cost-effectively achieve the desired requirements of the fully integrated system. Reduced capability, risk retention, and exponentially higher costs are often accepted when there is an urgent need for a solution, as evidently seen in defence with the use of urgent operational requirement contracts. In defence, the use of standardisation and encouraging inherent modularity and interoperability in the supply chain has been identified as the most effective solution to overcome these issues. It is considered with high confidence that the same would apply to nuclear robotics. The creation process of such a standard for nuclear robotics could follow similar design steps of standards developed for military/defence applications, such

as the UK's Generic Vehicle Architecture (GVA) and the NATO Generic Vehicle Architecture (NGVA). A nuclear robotics open standard would inherently improve software maintainability, extensibility, modularity, integration efficiency, and operational agility whilst reducing development and integration costs and maximising supply-chain efficiency. For readers benefit, we have described and outlined the key terminology used below as follows:

- **Improved Software Characteristics:** Standardisation process would likely lead to improved software maintainability, extensibility, modularity, and integration efficiency. This could make it easier to develop and update software for nuclear robots.
- **Operational Agility:** Standardisation can lead to increased operational agility, as it allows for easier integration and interoperability between different systems. This can be crucial in dynamic and potentially hazardous environments like those found in nuclear facilities.
- **Cost Reductions:** By providing a common framework, the development and integration costs for nuclear robots could be reduced. Companies and organizations could build on a shared foundation rather than starting from scratch.
- **Supply Chain Efficiency:** A standardised open standard could help streamline the supply chain for nuclear robotics components and software. This could make it easier for different manufacturers and developers to collaborate and provide compatible solutions.
- **Facilitated Innovation:** With a standardised framework in place, developers and engineers can focus on innovation within the defined boundaries, rather than reinventing the wheel for basic functionalities.
- **Interoperability:** The standard could promote interoperability between different systems, potentially allowing robots from different manufacturers to work together seamlessly.

The standard would cover communication (network interfaces, protocols), high-integrity & deterministic operation (networking, operating systems, applications), interoperability (system internal, system external, system-of-systems), Human-Machine Integration, rapid development & integration (middleware, data model), etc. One of the biggest challenges is future-proofing long-lived nuclear solutions which often require integration and interoperability between the existing legacy systems and future system designs. Therefore, future-proofing should lie at the core of this standard.

Currently there is no open standard tailored for Nuclear Robotics available to organisations and industry for use in operations, development, testing, and other activities. However, various types of open standards and technologies do exist in adjacent sectors such as commercial and industrial robotics, robotics-related technologies & devices supply, deterministic/high-integrity/safety critical systems interoperation and control, high-integrity teleoperator communication.

Industrial automation has made great advancement in robotic technologies, autonomous/semi-autonomous operations, centralised and remote managed operations, etc., but it is believed that their design, development, and operational

methodologies have not been advanced, tested or approved for the high-criticality and hazardous operational requirements of the nuclear robotics sector. Security is also becoming an increasingly important requirement for highly connected robotics that operate in hazardous and critical environments, but current industrial and commercial robotics do not inherently provide the security protections that nuclear robotic operations would require to protect against serious threats from foreign State and Non-State Actor threats and terrorism activities.

The advent of Unmanned Ground Vehicles (UGV) integration within the international defence sector provides a very good source of underlying technology innovation and mature expertise in the development of advanced robotic technologies and robotic sub-systems/devices operating in remote high-hazard environments with inherent capabilities for enhanced operational safety and security protection.

Developing an international open standard might be considered as a simple task due to its “Open” nature, but in reality it is a highly complex process that requires strategic organisation, targeted development activities, and long-term agile management. The technical elements of an open standard need to be simultaneously generic enough to support multiple technologies and designs to ensure that there is a high availability of a wide supporting product range and industry investment, while at the same time also be strict enough to ensure simplified and low-overhead interoperability and to protect against overspecialisation where vendors develop solutions that indirectly lock-in the customers to a specific single-sourced product. In addition, the national and/ or international political and commercial aspects need to always be an integral decision-making factor as each nation participant inherently tends to advocate for their own national government and industry interests. Failure to do so can result in an open standard that is well developed, widely available, but rarely used.

2 Proposal

The aim is to investigate and kickstart the creation of an open standard, the “Generic Nuclear Robotics Architecture” (GNRA) and support the development, integration, and interoperability of robotic systems for nuclear applications. The establishment and use of open standards is a tried-and-tested solution for organising and optimising the R&D, supply chain, and overall long-term through-life capability management (TLCM) of complex, diverse, and multi-national multi-vendor platforms and integrated systems.

The project team brought in multi-decade R&D expertise on military land-system electronic architectures standardisation and high-integrity open systems development. Their military land systems experience spans from manned platforms to unmanned platforms (UGVs) and covers expertise on manned-unmanned teaming capabilities, large-scale ultra-remote deployment, high-integrity system operation, high safety requirements, high security requirements, and overall logistics management of international standards development (managing international relations, leading multi-national standardisation committees and technical working groups, technical documentation write-up).

At the beginning of the work, the issues identified by the stakeholder as the current key issues of the nuclear industry, along with future expectations and considerations, would be collected and analysed. This activity would then seed a requirements capture activity, the outputs of which would then be used for the creation of a unified document that will offer guidelines, principles, and recommendation on the development, integration, and overall management of the electronic architectures of robotic systems and devices for the nuclear sector (initially focusing on decommissioning due to urgency of the activity in the UK nuclear sector, but also covering generic nuclear infrastructure operations if possible). Standards and protocols currently available directly and indirectly related to robotic systems would be investigated thoroughly to identify and extract the currently available best hardware, software, and system architecture designs and components, testing principles, integration processes, cost-effectiveness, etc. The captured requirements would then be combined with the standards and technology evaluation outputs to create the first draft of the GNRA Standard documentation.

The developed GNRA Standard documentation would then pass from two additional stages of iterative review, expansion, and maturation, with each draft version presented to the stakeholders for review and then updated and expanded based on the feedback and review processes. The last iteration of the document would be provided at the end of the programme as a release candidate for external dissemination within the nuclear robotics industry.

2.1 Objectives and Scope

The primary objective of this work and therefore this paper was to initiate the creation of a standard with solid technical design and management structures, making it capable of achieving an open standard status. The standard's operating scope was defined to support the international development, integration, interoperability, and life cycle of robotic systems for UK Atomic Energy Authority (UKAEA) applications. The standard's operating environments scope was defined to support:

- Nuclear Decommissioning (structured)
- Nuclear Decommissioning (unstructured)
- UK Atomic Energy Applications (generic)

The standard's capabilities scope was defined to support:

- Operations in high-criticality environments (safety & security)
- Direct compatibility with training and operations planning environment (simulators & digital twins)
- Rapid adaptation to urgent operational requirements (modularity)
- Wide range of OEMs, suppliers, solution providers (interoperability)
- Integrated life cycle management (through life capability management)

Furthermore, taking into account the complexity of creating an open standard, it was deemed necessary that both technical and logistical aspects should

be considered throughout all activities. The technical aspects were expected to cover requirements with respect to technical specifications, technology selection (hardware, software), mechanical designs, and other engineering design considerations. The logistical aspects were expected to cover requirements with respect to the coordination with stakeholders, government organisations, industry, research organisations, etc., for the technical development of the Standard, and the long-term maintenance and overall management of the Standard. It was deemed important that a well-structured draft release of the Standard shall be one of the primary outputs of the programme, as to provide a good baseline for future continuation work.

3 Progress Up-to-Date

The first phase of the programme started with the investigation of the requirements of nuclear decommissioning stakeholders, to familiarise with the inherent needs and expectations, and identify the key technical and logistical issues what should be covered by the standard. The follow-up task conducted a commercial, industrial, and military robotics' technologies review, to identify similar open standards and Electronics architectures used in commercial and industrial robotics. With the preliminary user requirements investigation and technology reviews completed, a more thorough requirements capture was conducted focusing on nuclear decommissioning robotics specific functional, operational, and technical requirements.

As part of the investigation a number of existing middleware solutions was reviewed to familiarise with the low-level software needs for middleware, application development, safety and criticality requirements for nuclear robotic applications. The requirements investigation was heavily influenced by the ongoing UKAEA projects, which is likely to be a subset of requirements that the standard would cover in the future. The requirements were individually analysed and classified into primary and secondary function group, as way to identify commonalities between them and establish distinct technology groups, (e.g. computing hardware, software, networking, training, integrity, autonomy, power, mechanical, etc.), which should be included in and covered by the Standard documentation. These groups were then further clustered together into technical contexts (e.g. platform design, safety, security, power distribution, operator station, etc.) to form distinct documents classified as Annexes. Based on these, a draft structure was agreed with the NDA group stakeholder.

A key activity of the first phase was to thoroughly investigate and design the formal development methodology, guidelines and overall management structure and operating model of the standard itself. This included the investigation of the creation of a management authority, tailored structured working groups, versioning and release management procedures, and general considerations on potentially sensitive and national security matters. Next would be agreeing on the management and custody of this standard with the nuclear stakeholders and forming the working groups.



Robots and Social Sustainability

Bipin Indurkhyā^(✉) and Barbara Sienkiewicz

Cognitive Science Department, Jagiellonian University, Krakow, Poland
bipin.indurkhyā@uj.edu.pl, barbara.sienkiewicz@student.uj.edu.pl

Abstract. Sustainability is no longer a matter of choice but is invariably linked to the survival of the entire ecosystem of our planet Earth. As robotics technology is growing at an exponential rate, it is crucial to examine its implications for sustainability. Our focus is on social sustainability, specifically analyzing the role of robotics technology in this domain by identifying six distinct ways robots influence social sustainability.

Keywords: Companion robots · education · mediator robots · mental health · nudge theory · social inclusion · sustainability · social robots

1 Introduction

Sustainability is no longer an option but a dire necessity. One approach to achieving sustainability is to leverage existing technology [4]. We focus here on how the technology of social robots relates to sustainability. We use the term social robot somewhat broadly to include chatbots and avatars. In recent years, there has been an increased interest in this area: a workshop on sustainability was organized as part of the IROS conference in 2021 [29, 41], and a session on sustainability was held at the European Robotics Forum in 2023 [20]. Nonetheless, sustainability has many facets, and the technology of social robots interacts with it in myriad ways. Here, we identify and analyze some of these interactions.

Sustainability, in general, refers to maintaining the capital, whether it be human capital, social capital, economic capital, or environmental capital [25]. Our focus here is on social sustainability—how the robotic technology can facilitate maintaining the social capital, which includes family relationships, friendships, group dynamics, and so on. We identify six ways in which social robots affect social sustainability: facilitating social inclusion, facilitating physical and mental health, provide companionship, act as mediators, educate about sustainability, and nudge towards sustainable behaviour, which are discussed below in turn.

This research was supported in part by the National Science Centre, Poland, under the OPUS call in the Weave programme under the project number 2021/43/I/ST6/02489.

2 Robots Facilitating Social Inclusion

One major area where social robots are contributing to social sustainability is in facilitating social inclusion. Social robots are deployed for the purposes of communication, emotion expression and perception, social relationship maintenance, interpretation of natural cues, and the development of social competencies [22, 37]. These skills are useful for people with neuro-developmental disorders such as autism [49] or ADHD [10]. It has been shown that children with autism are more willing to talk with social robots than with other children [58], which can be used to facilitate their social interaction skills. Social robot can be a facilitator not only in one-on-one interaction, but also during group activities [69]. A robot Probo was used in the assisted therapy with the children with autism: the robot tells the stories through which it teaches how to react in social situations like ‘hello’, ‘thank you’.

Incorporating people with disabilities into society is a fundamental step towards achieving equality and sustainability. It creates a sense of purpose and belonging for all individuals. Robotic avatars, as demonstrated by Barbareschi [3], play a vital role in this process by not only promoting inclusiveness but also providing opportunities for the disabled to play a role (e.g. a waiter) that would not be possible due to their limitations. This not only enriches their lives but also contributes positively to the society as a whole.

One example of a social robot adding to the quality of life of visually disadvantaged people is provided by [34]: a map-less navigation system *Pathfinder* is designed for blind people to reach destinations in unfamiliar buildings, which is implemented on a robot.

Another group that can benefit from interaction with social robots is children. Using humanoid robot during the rehabilitation of children with cerebral palsy increases their motivation and keeps them focused for longer [30, 42]. Children under medical treatment at the hospital often feel stressed and need additional emotional support. Social robot can provide distraction during the medical procedure, and emotional or well-being support during the hospital stay when parents cannot be there [45]. Social robots can also be used to increase awareness of hidden biases and discrimination. For example, in [55], a robot uses storytelling to teach children about gender equality.

In spite of all these approaches, we must be cautious in evaluating their full impact, as it needs to be considered from the point of view of disabled people themselves [61]. Moreover, issues related to potential psychological harm from robot rejection must also be considered [47, 63].

3 Robots for Physical and Mental Health

Another area where social robots have been effectively deployed to improve quality of life is in the care of the elderly. Cooper et al. [12] point out the needs of our aging society, and how assistive robot can provide physical support and cognitive support. An interesting case in point is the Primo Puel robotic doll, which was

introduced in Japan around 2004. This doll had simple functional features—it talks, giggles, and asks for cuddles—and was designed as a companion for young single girls in the workforce. Perhaps surprisingly, these dolls were a big hit with the elderly population [5]. They were found to be effective in monitoring the health and safety of the elderly, and to provide them companionship. Nonetheless, Sharkey and Sharkey [60] raise a number of ethical concerns related to such technologies, such as deception and infantilisation.

Another example is Paro the seal, which also has a simple design based on a baby harp seal. Covered in antibacterial fur, it can move its head and flippers a little bit. It can purr like a cat and its eyes can open and close. Its design has remained largely unchanged since its inception about twenty years ago. Paro the seal has been used effectively as a therapy aid for different user groups such as the elderly [6,7], people with dementia [31,46,52], and cancer patients [19]. However, even though Paro has been commercialized and used in care settings for more than a decade in multiple countries [27], to adequately assess its impact on quality of life, more studies are needed with larger sample sizes and a rigorous study design [53]. Moreover, as with Primo Puel dolls, some ethical concerns have been raised about the use of this technology, such as self-deception [9].

Researchers have found that elderly users feel certain threats (the need for independence, the need for control, the fear of being replaced, and the need for authenticity) from robots [14], and face a number of psychological vulnerabilities [39].

In recent years, many social robots and chatbots have been introduced as conversational agents that are especially designed for a therapeutic use. For example, the chatbot *Woebot* helps to deal with substance abuse [51], and the chatbot *Replika* is designed to be an accepting, understanding, and non-judgmental companion [74].

However, some negative effects of these conversational agents have also been reported. For example, *Replika* has been blamed for inciting murder and suicide [50]. It has also been argued that chatbots can lead to addiction, thereby harming one's ability to form real-life intimate relationships [74]. Some other ethical concerns of incorporating emotional response in social robots and chatbots are discussed in [28].

4 Robots as Companions

For neurotypical children, social robots can facilitate their cognitive development, boost their creativity [1,33], and teach computational thinking [75].

A more controversial issue is when social robots are treated as romantic partners. One such case is presented in Isabella Willinger's 2019 documentary *Hi, A.I.*, where Chuck, a man from Texas is attempting to start a romantic relationship with a humanoid robot Harmony. Cindy Friedman [23] raises some crucial ethical concerns about such use of technology from an African philosophical perspective. One obvious problem with such sexbots is that they promote addiction [8]. A number of other ethical issues related to this technology are being hotly debated [24,43,65].

Similarly, [54] discusses the potential use of social robots for coping with the social anxiety disorder, which is promising given the effectiveness of using social robots in interaction with children with ASD. However, there is a risk of harmful side effects: for example, the companion social robot might become the only friend for the person with a high social anxiety, thereby causing further isolation.

Another way in which social robots are likely to make a significant impact on the quality of life in the near future is in doing household chores [35]. However, as the study by Vaussard *et al.* [70] on the impact of robotic vacuum cleaners on home ecosystem has shown, one needs to look at the bigger picture by studying how such technology integrates with the user's space and perception to assess its overall impact.

5 Robots as Mediators

Social robots have also been found to be effective in mediating negotiations. For example, Druckman *et al.* [17] compared the effectiveness of three different mediator platforms (teleoperated robotic, human, and computer screen-based) on negotiation outcomes, and found the robotic platform to be the most effective. It also generated more agreements outside of the pre-negotiation scenarios, and led to a more positive perception of the negotiation experience for the participants.

In another recent study, Weisswange *et al.* [72] have explored how social robots can be used for mediating complex social relationships in the real world. They have identified six different roles for robot mediators, and discussed mediation behaviours and measures to evaluate the quality of mediation interventions.

6 Robots Educating About Sustainability

Examining the intersection of robotic technologies with education and sustainable development is not new [15]. A more recent view is provided in [48]. We briefly present a few examples here.

Li *et al.* [38] address culturally-responsive computing (CRC) by introducing the concept of co-creation with a robot agent. In this approach, the robot agent itself is co-created through participatory design with adolescent girls and a social robot agent. It identifies themes about who has the power to make decisions, what decisions are made, and how to maintain social relationships.

A recent case study considers the problem of solving the waste problem at multi-day music festivals [71]. By using co-design workshops, the authors explored sociotechnical strategies to solve this problem together with the festival participants. As a result, they identified the challenges and opportunities for sustainable human-computer interaction on carving out a design response to a difficult problem by situating relations, meaning making, telling invisible stories, and finding leverage points.

In another example, Ferreira and Miškovic [21] use robot as a tool to share magnificent underwater life with the goal that this will create a desire to preserve this environment. Also, social robots can help build good habits like hand hygiene [13, 57]. Robots contribute to sustainable development not only through explicit teaching but also by effectively addressing and alleviating education challenges, such as the use of social robots in school anti-bullying programs [56].

7 Robots Nudging Towards Sustainable Behaviour

Nudging [26, 66], also called Shikakeology [44], means changing the behavior of people by design. A typical use case is to decide what you want people to choose, and to set that as the default option. This is because most people go with the default option. For example, if the default option is to be an organ donor (after your death) then there are many organ donors. Other examples include painting religious icons on a public wall to discourage men from urinating against it.

To promote socially responsible, sustainable behavior, nudge theory has been successfully applied in many areas in many countries around the world [64]. However, other studies have found that nudges do not always work, and sometimes end up nudging people in a direction opposite to the one that was intended [16]. Engelen [18] found that some health-promoting nudges are not ethical. It has also been argued that nudging violates personal autonomy [59].

There are several ways in which nudging can be used to promote sustainable behaviour [32, 36, 40]. One study specifically explored the effect of different types of agents in promoting normative behaviour [68] and found that a robot by its presence itself can induce normative (sustainable) behavior.

However, there are two major problems. One is that nudging, no matter how benevolent and how much based on good intentions, is essentially a kind of manipulation, and manipulation erodes trust [11]. It has been argued that nudging is incompatible with informed consent [62]. Another study [2] focused on how people felt about opt-in format (by default your data is not shared, but you have to check a box to share) vs opt-out format (by default your data is shared, but you have to check a box to not share) in privacy statements. They found that participants who read the opt-in format felt significantly more control and trust than participants who read the opt-out format. Generally, people resent being manipulated, and if they sense it, they tend to go against the nudge, even at the risk of harming themselves.

Another issue is that any nudge creates a backdoor for commercial or political exploitation, as it depends on the intentions and transparency of the creators. Examples of unethical use of nudges include misleading wording, and order and presentation bias, to persuade the user for hasty purchases [73].

8 Conclusions

We have identified six aspects of social sustainability where robots are having a major impact. In many ways, robots are facilitating social interaction, especially in spanning across space and time limitations, and also overcoming cognitive limitations such as from Autism spectrum disorder (ASD). However, the technology for social robots is also creating alienation leading towards an emotionally dysfunctional society such as imagined in [67].

References

1. Ali, S., Moroso, T., Breazeal, C.: Can children learn creativity from a social robot? In: Proceedings of the 2019 on Creativity and Cognition, pp. 359–368 (2019)
2. Arcand, M., Nantel, J., Arles-Dufour, M., Vincent, A.: The impact of reading a web site's privacy statement on perceived control over privacy and perceived trust. *Online Inf. Rev.* **31**(5), 661–681 (2007)
3. Barbareschi, G., et al.: I am both here and there parallel control of multiple robotic avatars by disabled workers in a café. Association for Computing Machinery (2023)
4. Bashir, H.: Leveraging technology to communicate sustainability-related product information: evidence from the field. *J. Clean. Prod.* **362**, 132508 (2022)
5. British Broadcasting Corporation (BBC): My Special Partner (n.d.). http://news.bbc.co.uk/2/hi/programmes/this_world/golden_years/4373857.stm. Accessed 16 Dec 2024
6. Bemelmans, R., Gelderblom, G.J., Jonker, P., de Witte, L.: Effectiveness of robot Paro in intramural psychogeriatric care: a multicenter quasi-experimental study. *J. Am. Med. Directors Assoc.* **16**(11), 946–950 (2015)
7. Birks, M., Bodak, M., Barlas, J., Harwood, J., Pether, M.: Robotic seals as therapeutic tools in an aged care facility: a qualitative study. *J. Aging Res.* **2016** (2016)
8. Boot, G.: Lifelike and cheap sex robots made with 3d printers will fuel surge in addiction. The Sun (2018). <https://www.thesun.co.uk/tech/7998023/lifelike-and-cheap-sex-robots-made-with-3d-printers-will-fuel-surge-in-addiction/>. Accessed 29 Mar 2023
9. Calo, C.J., Hunt-Bull, N., Lewis, L., Metzler, T.: Ethical implications of using the Paro robot, with a focus on dementia patient care. In: Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence (2011)
10. Cervantes, J.A., López, S., Cervantes, S., Hernández, A., Duarte, H.: Social robots and brain-computer interface video games for dealing with attention deficit hyperactivity disorder: a systematic review. *Brain Sci.* **13**(8), 1172 (2023)
11. Choo, L., Kaplan, T.R., Zultan, R.: Manipulation and (Mis)trust in prediction markets. *Manag. Sci.* **68**(9), 6716–6732 (2022)
12. Cooper, S., Ros, R., Lemaignan, S.: Challenges of deploying assistive robots in real-life scenarios: an industrial perspective. In: 32nd IEEE International Conference on Robot and Human Interactive Communication (2023)
13. Deshmukh, A., Darda, K.M., Mhatre, M.M., Pandey, R., Jadhav, A.R., Cross, E.: Enhancing hand hygiene practices through a social robot-assisted intervention in a rural school in India. In: Proceedings of the ICSR 2023, pp. 244–253. Springer Nature Singapore, Singapore (2023)

14. Deutsch, I., Erel, H., Paz, M., Hoffman, G., Zuckerman, O.: Home robotic devices for older adults: opportunities and concerns. *Comput. Hum. Behav.* **98**, 122–133 (2019)
15. Dias, M., Mills-Tettey, G., Nanayakkara, T.: Robotics, education, and sustainable development. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation, pp. 4248–4253 (2005)
16. Dimant, E., van Kleef, G.A., Shalvi, S.: Requiem for a nudge: framing effects in nudging honesty. *J. Econ. Behav. Organ.* **172**, 247–266 (2020)
17. Druckman, D., et al.: Who is best at mediating a social conflict? Comparing robots, screens and humans. *Group Decis. Negot.* **30**, 395–426 (2021)
18. Engelen, B.: Ethical criteria for health-promoting nudges: a case-by-case analysis. *Am. J. Bioethics* **19**(5), 48–59 (2019), pMID: 31068115
19. Eskander, R., Tewari, K., Osann, K., Shibata, T.: Pilot study of the paro therapeutic robot demonstrates decreased pain, fatigue, and anxiety among patients with recurrent ovarian carcinoma. *Gynecol. Oncol.* **130**(1), e144–e145 (2013)
20. European Robotics Forum: Social robots: Applications and sustainability priorities (2023). <https://erf2023.sdu.dk/timetable/event/social-robots-applications-and-sustainability-priorities/>. Accessed 12 Sep 2023
21. Ferreira, F., Miškovic, N.: Addressing the sustainability of the adriatic sea using marine robotics. In: Proceedings of the IEEE IROS 2021 Workshop on Role Robot. Achieving UN's Sustainable Development Goals, p. 2 (2021)
22. Fong, T., Nourbakhsh, I., Dautenhahn, K.: A survey of socially interactive robots. *Robot. Auton. Syst.* **42**(3–4), 143–166 (2003)
23. Friedman, C.: Ethical concerns with replacing human relations with humanoid robots: an ubuntu perspective. *AI Ethics* **3**, 527–538 (2023)
24. González-González, C.S., Gil-Iranzo, R.M., Paderewski-Rodríguez, P.: Human-robot interaction and sexbots: a systematic literature review. *Sensors* **21**(1) (2021)
25. Goodland, R.: Encycl. Glob. Environ. Change 5, chap. Sustainability: Human, Social, Economic and Environmental, pp. 481–491. John Wiley & Sons, Inc., Hoboken, NJ (USA) (2002)
26. Halpern, D.: Inside the Nudge Unit: How Small Changes can Make a Big Difference. WH Allen, London, UK (2015)
27. Hung, L., et al.: The benefits of and barriers to using a social robot paro in care settings: a scoping review. *BMC Geriatr.* **19**, 1–10 (2019)
28. Indurkhy, B.: Ethical aspects of faking emotions in chatbots and social robots. In: 32nd IEEE International Conference on Robot and Human Interactive Communication (2023)
29. IROS Workshop: Sustainable robotics (2021). <https://www.sustainablerobotics.org/iros-2021-workshop-summary/>. Accessed 12 Sep 2023
30. Kachmar, O., Kozyavkin, V., Ablikova, I.: Humanoid social robots in the rehabilitation of children with cerebral palsy. In: REHAB 2014 (2014)
31. Kang, H.S., Makimoto, K., Konno, R., Koh, I.S.: Review of outcome measures in paro robot intervention studies for dementia care. *Geri. Nurs.* **41**(3), 207–214 (2020)
32. Kasperbauer, T.J.: The permissibility of nudging for sustainable energy consumption. *Energy Policy* **111**, 52–57 (2017)
33. Kory-Westlund, J.M., Breazeal, C.: Assessing children's perceptions and acceptance of a social robot. In: Proceedings of the 18th ACM International Conference on Interaction Design and Children, pp. 38–50 (2019)

34. Kuribayashi, M., et al.: Pathfinder: designing a map-less navigation system for blind people in unfamiliar buildings. In: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, pp. 1–16 (2023)
35. Lehdonvirta, V., Shi, L.P., Hertog, E., Nagase, N., Ohta, Y.: The future (s) of unpaid work: How susceptible do experts from different backgrounds think the domestic sphere is to automation? *PLoS ONE* **18**(2), e0281282 (2023)
36. Lehner, M., Mont, O., Heiskanen, E.: Nudging-a promising tool for sustainable consumption behaviour? *J. Clean. Prod.* **134**, 166–177 (2016)
37. Li, H., John-John, C., Tan, Y.K.: Towards an effective design of social robots. *Int. J. Soc. Robot.* **3**(4), 333–335 (2011)
38. Li, Y., et al.: I want to be unique from other robots: positioning girls as co-creators of social robots in culturally-responsive computing education. In: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, pp. 1–14 (2023)
39. Liu, S.X., Shen, Q., Hancock, J.: Can a social robot be too warm or too competent? Older Chinese adults' perceptions of social robots and vulnerabilities. *Comput. Hum. Behav.* **125**, 106942 (2021)
40. GGossen, M., Jager, S., Hoffmann, M. L., Bießmann, F., Korenke, R., Santarius, T.: Nudging sustainable consumption: A large-scale data analysis of sustainability labels for fashion in German online retail. *Front. Sustain.* **3**, 922984 (2022)
41. Mai, V., et al.: The role of robotics in achieving the united nations sustainable development goals—the experts' meeting at the 2021 IEEE/RSJ IROS workshop [industry activities]. *IEEE Robot. Autom. Mag.* **29**(1), 92–107 (2022)
42. Martí Carrillo, F., et al.: Adapting a general-purpose social robot for paediatric rehabilitation through in situ design. *ACM Trans. Hum.-Robot Interact. (THRI)* **7**(1), 1–30 (2018)
43. Masterson, A.M.: Designing a loving robot: a social construction analysis of a sex robot creator's vision. *Hum.-Mach. Commun.* **5**, 99–114 (2022)
44. Matsumura, N., Fruchter, R., Leifer, L.: Shikakeology: designing triggers for behavior change. *AI Soc.* **30**, 419–429 (2015)
45. Moerman, C.J., Van Der Heide, L., Heerink, M.: Social robots to support children's well-being under medical treatment: a systematic state-of-the-art review. *J. Child Health Care* **23**(4), 596–612 (2019)
46. Moyle, W., Bramble, M., Jones, C., Murfield, J.: Care staff perceptions of a social robot called paro and a look-alike plush toy: a descriptive qualitative approach. *Aging Mental health* **22**(3), 330–335 (2018)
47. Nash, K., Lea, J.M., Davies, T., Yogeeswaran, K.: The bionic blues: robot rejection lowers self-esteem. *Comput. Hum. Behav.* **78**, 59–63 (2018)
48. Park, I.W., Han, J.: Teachers' views on the use of robots and cloud services in education for sustainable development. *Clust. Comput.* **19**, 987–999 (2016)
49. Pennisi, P., et al.: Autism and social robotics: a systematic review. *Autism Res.* **9**(2), 165–183 (2016)
50. Possati, L.M.: Psychoanalyzing artificial intelligence: the case of replika. *AI Soc.* 1–14 (2022)
51. Prochaska, J.J., et al.: A therapeutic relational agent for reducing problematic substance use (woebot): Development and usability study. *J. Med. Internet Res.* **23**(3) (2021)
52. Pu, L., Moyle, W., Jones, C.: How people with dementia perceive a therapeutic robot called paro in relation to their pain and mood: a qualitative study. *J. Clin. Nurs.* **29**(3–4), 437–446 (2020)

53. Pu, L., Moyle, W., Jones, C., Todorovic, M.: The effectiveness of social robots for older adults: a systematic review and meta-analysis of randomized controlled studies. *Gerontologist* **59**(1), e37–e51 (2018)
54. Rasouli, S., Gupta, G., Nilsen, E., Dautenhahn, K.: Potential applications of social robots in robot-assisted interventions for social anxiety. *Int. J. Soc. Robot.* **14**(5), 1–32 (2022)
55. Romain Maure, B.B.: Participatory design of a social robot and robot-mediated storytelling activity to raise awareness of gender inequality among children. In: 32nd IEEE International Conference on Robot and Human Interactive Communication (2023)
56. Sanoubari, E., Young, J., Houston, A., Dautenhahn, K.: Can robots be bullied? A crowdsourced feasibility study for using social robots in anti-bullying interventions. In: 2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN), pp. 931–938. IEEE (2021)
57. Sasidharan, S., Pasupuleti, D., Das, A.M., Kapoor, C., Manikutty, G., Rao, R.: Haksh-E: an autonomous social robot for promoting good hand hygiene among children. In: IROS 2021 Workshop on The Roles of Robotics in Achieving the UN's Sustainable Development Goals (2021)
58. Scassellati, B., Admoni, H., Mataric, M.: Robots for use in autism research. *Annu. Rev. Biomed. Eng.* **14**, 275–294 (2012)
59. Schmidt, A.T., Engelen, B.: The ethics of nudging: an overview. *Philos Compass* **15**(4), e12658 (2020)
60. Sharkey, A., Sharkey, N.: Granny and the robots: ethical issues in robot care for the elderly. *Ethics Inf. Technol.* **14**(1), 27–40 (2012)
61. Shew, A.: Against Technoableism: Rethinking Who Needs Improvement. W.W. Norton, New York, NY, USA (2023)
62. Simkulet, W.: Informed consent and nudging. *Bioethics* **33**(1), 169–184 (2019)
63. Spisak, S.E., Indurkhya, B.: A study on social exclusion in human-robot interaction. *Electronics* **12**(7) (2023)
64. Sunstein, C.R., Reisch, L.A.: Trusting Nudges: Toward A Bill of Rights for Nudging. Routledge, Abingdon, UK (2019)
65. Sætra, H.S.: Loving robots changing love: towards a practical deficiency-love. *J. Futur. Robot Life* 109–127 (2022)
66. Thaler, R.H., Sunstein, C.R.: Nudge: Improving Decisions about Health, Wealth, and Happiness. Yale University Press, New Haven, Connecticut, USA (2008)
67. Torras, C.: The Vestigial Heart. MIT Press, Cambridge (MA), USA (2018)
68. Tussyadiah, I., Miller, G.: Nudged by a robot: responses to agency and feedback. *Ann. Tour. Res.* **78**, 102752 (2019)
69. Vanderborght, B., et al.: Using the social robot probo as a social story telling agent for children with ASD. *Interact. Stud.* **13**(3), 348–372 (2012)
70. Vaussard, F., et al.: Lessons learned from robotic vacuum cleaners entering the home ecosystem. *Robot. Auton. Syst.* **62**(3), 376–391 (2014)
71. Vella, K.: Pick me up before you go-go: sociotechnical strategies for waste in music festival campsites. In: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, pp. 1–13 (2023)
72. Weisswange, T.H., et al.: What could a social mediator robot do? Lessons from real-world mediation scenarios (2023). Accessed 14 Sep 2023
73. Wendel, S.: Designing for Behavior Change: Applying Psychology and Behavioral Economics. O'Reilly Media, Sebastopol (2020)
74. Xie, T., Pentina, I.: Attachment theory as a framework to understand relationships with social chatbots: a case study of replika (2022)

75. Yang, W., Luo, H., Su, J.: Towards inclusiveness and sustainability of robot programming in early childhood: child engagement, learning outcomes and teacher perception. *Br. J. Educ. Technol.* **53**(6), 1486–1510 (2022)



Planning and Inverse Kinematics of Hyper-Redundant Manipulators with VO-FABRIK

Cristian Morasso¹, Daniele Meli¹⁽⁾, Yann Divet², Salvatore Sessa²,
and Alessandro Farinelli¹

¹ University of Verona, Verona, Italy

daniele.meli@univr.it

² Hibot Corp., Tokyo, Japan

Abstract. Hyper-redundant Robotic Manipulators (HRMs) offer great dexterity and flexibility of operation, but solving Inverse Kinematics (IK) is challenging. In this work, we introduce VO-FABRIK, an algorithm combining Forward and Backward Reaching Inverse Kinematics (FABRIK) for repeatable deterministic IK computation, and an approach inspired from velocity obstacles to perform path planning under collision and joint limits constraints. We show preliminary results on an industrial HRM with 19 actuated joints. Our algorithm achieves good performance where a state-of-the-art IK solver fails.

Keywords: Inverse Kinematics · Robot Planning · Obstacle Avoidance · Hyper-Redundant Manipulators · Velocity Obstacles

1 Introduction

Hyper-redundant Robotic Manipulators (HRMs) guarantee high dexterity and are particularly suitable for operation in highly constrained environments, e.g., aerospace, search-and-rescue, and maintenance [10]. However, planning and control of HRMs in Cartesian space becomes extremely challenging as the number of Degrees of Freedom (DoFs) increases. In fact, Inverse Kinematics (IK) cannot be solved in closed form, but task-specific optimization of joint configuration within constraints as joint limits and collision avoidance is required [14].

Standard optimization algorithms either do not guarantee fast convergence to the solution [15], or do not ensure repeatability [5], which is essential for reliability in safety-critical scenarios as smart industry. Similarly, purely geometric approaches require assumptions on the specific robot and task at hand [9], hence are hardly generalizable. When considering complex kinematic structures

C. Morasso and D. Meli—Equal Contribution

This work has been supported by Hibot Corp.

The authors thank Prof. Joshua Vaughan (Univ. Louisiana at Lafayette) for his technical support, and Prof. Paolo Fiorini (Univ. Verona) for his mentoring.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2024
C. Secchi and L. Marconi (Eds.): ERF 2024, SPAR 32, pp. 195–199, 2024.

https://doi.org/10.1007/978-3-031-76424-0_35

as HRMs, gradient-free methods, e.g., genetic algorithms [12], or deep neural networks [8] are prominent solutions. Such algorithms are however expensive in terms of training data and computational time. Furthermore, they do not provide any repeatability or safety guarantee [4]. The drawbacks of optimization-based approaches become ever more severe when solving the combination of IK and path planning problem [17].

In this paper, we introduce VO-FABRIK, an algorithm combining Velocity Obstacles (VO) [2,6] and Forward and Backward Reaching Inverse Kinematics (FABRIK) [1]. FABRIK guarantees repeatable iterative fast IK computation. VO is integrated into FABRIK iterations and used for path planning at the end effector. Differently from alternatives as artificial potential fields [7], VO prunes unsafe (colliding) velocities, thus guaranteeing the fast computation of collision-free configurations. Existing FABRIK applications in robotics consider few DoFs [13] or do not guarantee joint limits and collision avoidance [16]. Instead, we test VO-FABRIK on a (simulated) HRM with 19 rotational DoFs from Hibot Corporation (Float Arm), intended to navigate in cluttered hazardous environments and acquire detailed data for inspection and maintenance. We show the increased performance of VO-FABRIK against BioIK [11], a state-of-the-art IK solver based on evolutionary algorithms.

2 Background

2.1 Velocity Obstacles (VO)

The VO algorithm computes the set of admissible velocities \mathcal{V}_a for a robot R , given a target G and N obstacles defined by their radii (assuming spherical shapes for simplicity), current position and relative velocity to R . At a given time step, the robot would reach its target with a velocity \mathbf{v}_{pref} towards G . To avoid collisions, for each i -th obstacle a *collision cone* CC_i is computed as the set of robot velocities which will cause collision with i -th obstacle. \mathbf{v}_{pref} may then be modified, in order to fit in $\mathcal{V}_a = \mathcal{V} \setminus \bigcup_{i=1}^N CC_i$, where \mathcal{V} is the full velocity space for the robot.

2.2 FABRIK

FABRIK algorithm solves the IK for a N -link kinematic chain iteratively, performing backward and forward phases and neglecting collision avoidance. It starts from the goal end-effector position \mathbf{p}_G , and the positions $\{\mathbf{p}_i\}_{i=0}^N$ of link extrema (\mathbf{p}_0 being the base of the robot). In the *backward phase* (Fig. 1a), \mathbf{p}_N is set to \mathbf{p}_G . Then, a segment from \mathbf{p}_N to \mathbf{p}_{N-1} is drawn, with the same length as N -th link. Its extremal point represents the new position \mathbf{p}'_{N-1} . The procedure is repeated for all links up to the base. Then, in the forward step, the reverse algorithm is applied, setting the newly computed \mathbf{p}'_0 to the original one (the base must remain fixed). Forward and backward steps are executed until $\|\mathbf{p}_N - \mathbf{p}_G\|_2 < \epsilon \in \mathbb{R}^+$. In this way, FABRIK guarantees minimal displacement between consecutive joint configurations.

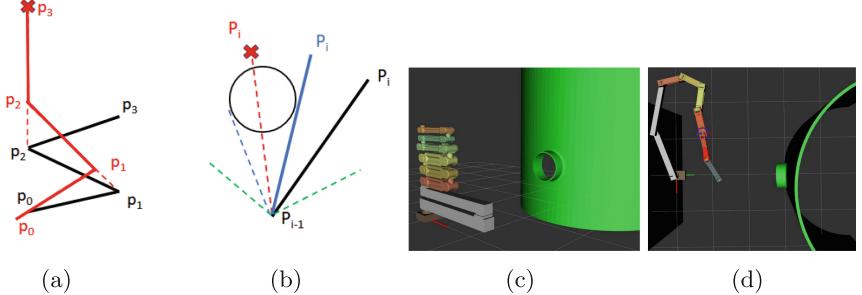


Fig. 1. a) FABRIK backward iteration from black to red configuration; b) VO-FABRIK avoiding collision for link i , setting blue configuration instead of red to avoid **obstacle** within **joint limits**; the Float Arm c) wrapped and d) extended.

Algorithm 1. VO-FABRIK - IK Phase

Require: positions of link extrema $\{\mathbf{p}_i\}_{i=0}^N$; length of links $\{l_i\}_{i=1}^N$; goal and base positions $\mathbf{p}'_N, \mathbf{p}_B$; set of obstacles \mathcal{O} ; joint limits $\{jl_i\}_{i=0}^{N-1}$; joint angles $\{\alpha_i\}_{i=0}^{N-1}$

- 1: **while** not converged **do**
- 2: % Backward phase
- 3: $\mathbf{p}_N = \mathbf{p}'_N$
- 4: **for** $i = N - 1; i \geq 0; i --$ **do**
- 5: $\mathbf{p}_i \leftarrow \text{BACKWARD}(\mathbf{p}_{i+1}, \mathbf{p}_i, l_{i+1})$
- 6: $\mathcal{L} \leftarrow \text{COMPUTE_POS}(\{\mathbf{p}_j\}_{j=0}^i)$
- 7: **for** $\text{odo} \in \mathcal{O} \cup \mathcal{L}$
- 8: Compute CC
- 9: $jl_i \leftarrow jl_i \cap \text{CONSTRAINTS}(CC, \mathbf{p}_{i+1}, \mathbf{p}_i)$
- 10: $\alpha_i \leftarrow \text{COMPUTE_SAFE}(jl_i)$
- 11: $\mathbf{p}_i \leftarrow \text{FK}(\alpha_i)$
- 12: % Forward phase
- 13: $\mathbf{p}_0 = \mathbf{p}_B$
- 14: **for** $i = 1; i \leq N; i ++$ **do**
- 15: $\mathbf{p}_i \leftarrow \text{FORWARD}(\mathbf{p}_{i+1}, \mathbf{p}_i, l_{i+1})$
- 16: ... % Analogous to backward phase, with inverted index order

return $\{\mathbf{p}_i\}_{i=0}^N, \{\alpha_i\}_{i=0}^{N-1}$

3 VO-FABRIK

VO-FABRIK consists of two phases: path planning and IK computation. Given a target G to be reached by the robot, in the *motion planning phase* the velocity of the end effector is set to \mathbf{v}_{pref} in the direction of G , with arbitrary module, resulting in a displacement from \mathbf{p}_N to \mathbf{p}'_N within a predefined time step t_s (according to the truncated VO paradigm [3]). Then, in the *IK phase* (Algorithm 1), FABRIK starts with the backward step for each i -th link at Line 5. Then, the closest point to i -th link is computed (Line 6) on each following link in the kinematic chain (hence, to the base). A virtual obstacle is added at these

positions, and both virtual and real external obstacles are used to compute the set of collision cones CC (Lines 7–10), with radius equal to the link’s or obstacle’s thickness. Since i -th link can only pivot around \mathbf{p}_i , collision cones for each obstacle here represent ranges of angular velocities (applied for t_s) which are safe for i -th link. An example is shown in Fig. 1b in 2D, with blue lines representing CC for a circular obstacle. CC can then be converted to a range of feasible solid angles¹, to be intersected with i -th joint limits (Line 11). Actual joint angles are then selected from the safe range (Line 12) and applied to update \mathbf{p}_i with forward kinematics (Line 13). Similarly, the forward phase is performed and the process is iterated until convergence to \mathbf{p}'_N is achieved. The full VO-FABRIK algorithm terminates when \mathbf{p}'_G is finally reached.

Table 1. Quantitative results.

| | Wrapped (Fig. 1c) | Extended (Fig. 1d) | |
|-------------------|-------------------|--------------------|-------------------|
| | VO-FABRIK | BioIK | VO-FABRIK |
| Joint disp. [rad] | 0.01 ± 0.13 | N/A | 0.007 ± 0.052 |
| Time per step [s] | 0.015 ± 0.007 | N/A | 0.015 ± 0.007 |
| | | | 1.93 ± 0.40 |

4 Experiments

We implemented VO-FABRIK in Python, and tested it in a simulated environment with the Float Arm, shown in Fig. 1c. The Float Arm enters through a narrow opening and explores a cavity up and down. The trajectory is calculated using a time step of $t_s = 0.2$ s. Starting from home configuration shown in figure, BioIK² is not able to complete the task, due to self-collisions. We then extend the home configuration as in Fig. 1d. Table 1 shows that VO-FABRIK achieves comparable average joint displacement per step, while guaranteeing much faster computation per step.

5 Conclusions and Future Works

We presented VO-FABRIK, an algorithm for fast repeatable motion planning and IK guaranteeing safe collision avoidance within joint limits, thanks to the combination of FABRIK and VO. Our methodology is particularly suited for HRMs but can generalize to any kinematic configuration. In the future, we will further assess the performance of our algorithm with multiple scenarios and robots. Moreover, we will extend the capabilities of our algorithm to also consider for link and end-effector orientation, currently not supported from FABRIK.

¹ Solid angles and joint limits can be converted to pitch and yaw by applying simple 2D projections, depending on the kinematic model of the robot.

² Collision checking is managed via MoveIt (<https://moveit.ros.org/>), motion planning is based on VO. BioIK minimizes joint displacement.

References

1. Aristidou, A., Lasenby, J.: Fabrik: a fast, iterative solver for the inverse kinematics problem. *Graph. Models* **73**(5), 243–260 (2011)
2. Bonanni, L., et al.: Monte Carlo planning for mobile robots in large action spaces with velocity obstacles. In: Proceeding of the 10th Italian Workshop on Artificial Intelligence and Robotics (AIRO) (2023)
3. Claes, D., et al.: Collision avoidance under bounded localization uncertainty. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1192–1198. IEEE (2012)
4. Corsi, D., et al.: Formal verification of neural networks for safety-critical tasks in deep reinforcement learning. In: Uncertainty in Artificial Intelligence, pp. 333–343 PMLR (2021)
5. Deo, A.S., Walker, I.D.: Minimum effort inverse kinematics for redundant manipulators. *IEEE Trans. Robot. Autom.* **13**(5), 767–775 (1997)
6. Fiorini, P., Shiller, Z.: Motion planning in dynamic environments using velocity obstacles. *Int. J. Robot. Res.* (1998)
7. Ginesi, M., et al.: Dynamic movement primitives: volumetric obstacle avoidance using dynamic potential functions. *J. Intell. Robot. Syst.* **101**, 1–20 (2021)
8. Kouabon, A.J., et al.: A learning framework to inverse kinematics of high DOF redundant manipulators. *Mech. Mach. Theory* **153**, 103978 (2020)
9. Mu, Z., et al.: A segmented geometry method for kinematics and configuration planning of spatial hyper-redundant manipulators. *IEEE Trans. Syst. Man, Cybern. Syst.* **50**(5), 1746–1756 (2018)
10. Mu, Z., et al.: Hyper-redundant manipulators for operations in confined space: typical applications, key technologies, and grand challenges. *IEEE Trans. Aerosp. Electron. Syst.* (2022)
11. Ruppel, P., et al.: Cost functions to specify full-body motion and multi-goal manipulation tasks. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 3152–3159 (2018)
12. Ruppel, P., Hendrich, N., Starke, S., Zhang, J.: Cost functions to specify full-body motion and multi-goal manipulation tasks. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 3152–3159. IEEE (2018)
13. Santos, P.C., et al.: M-fabrik: a new inverse kinematics approach to mobile manipulator robots based on fabrik. *IEEE Access* (2020)
14. Siciliano, B., Khatib, O., Kröger, T.: Springer Handbook of Robotics, vol. 200. Springer (2008)
15. Wang, J., Li, Y., Zhao, X.: Inverse kinematics and control of a 7-DOF redundant manipulator based on the closed-loop algorithm. *Int. J. Adv. Rob. Syst.* **7**(4), 37 (2010)
16. Zhao, L., et al.: Collision-free kinematics for hyper-redundant manipulators in dynamic scenes using optimal velocity obstacles. *Int. J. Adv. Robot. Syst.* (2021)
17. Zhong, J., Wang, T., Cheng, L.: Collision-free path planning for welding manipulator via hybrid algorithm of deep reinforcement learning and inverse kinematics. *Complex Intelli. Syst.* 1–14 (2021)



A Scalable Multi-robot System for Cooperative Exploration

Vincenzo Scognamiglio^(✉), Riccardo Caccavale, Alberto Finzi,
and Vincenzo Lippiello

PRISMA Lab, Department of Electrical Engineering and Information Technology,
Università degli Studi di Napoli Federico II, Via Claudio 21, 80125 Naples, Italy

{vincenzo.scognamiglio2,riccardo.caccavale,alberto.finzi,
vincenzo.lippiello}@unina.it

Abstract. In the last few years, mobile robots have become widespread in a lot of applications that require a certain level of autonomy. To overcome the performances of single-robot systems, multi-robot systems have been adopted in several domains achieving good results in exploration, search and rescue, and surveillance operations. This work presents an architectural system that focuses on scalability, making the multi-agent system able to include or exclude an agent without impacting the whole system. The framework has been designed to overcome communication issues that are typical of this kind of system. The proposed solution adopts a concise shared representation of reality which takes care only of the necessary information to allow cooperative exploration of multiple robots.

Keywords: multi-agent systems · autonomous navigation · exploration

1 Introduction

The research on autonomous systems is moving towards the exploitation of multi-agent frameworks capable of performing complex operations minimizing time and costs. This trend is recognizable when looking at the most relevant robotics competitions that, in the last years, have focused on the employment of multi-robot systems. The DARPA SubT Challenge [1] is one of the most famous competitions that require the development of a heterogeneous multi-robot system composed in general of robots with different locomotion systems, such as aerial, legged, and wheeled robots. The participant teams had to deal with communication problems that are caused by the large quantity of data that the robots have to communicate to collaborate during exploration and they had to take care of the possibility of losing one or more agents and continuing the exploration without faults of the multi-robot system execution. In this work, a framework for autonomous exploration which employs a multi-robot system is presented. The focus is on the scalability feature of the proposed system, which allows addition

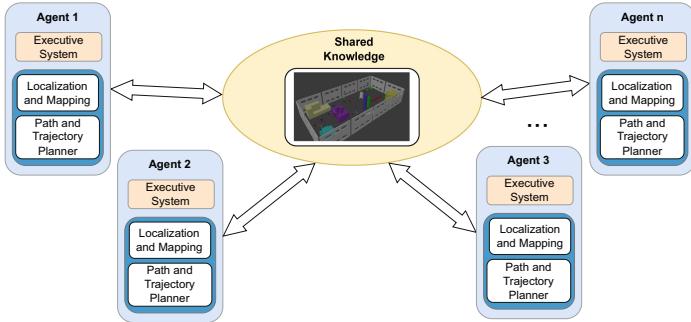


Fig. 1. Software architecture

and removal of an agent without modification in the behavior of the system. This is achieved thanks to a distributed *executive system* on each agent that exploits a synthetical representation of common activities and lightweight communication. The rest of this paper is divided into three sections, in Sect. 2 an overview of the complete systems is provided, then Sect. 3 illustrates a case study where the system has been tested: the Leonardo Drone Contest. In Sect. 4 observations about the behavior of the multi-robot system and a discussion about further possible improvements are shown.

2 System Architecture

The framework aims to represent a system for multi-robot cooperative autonomous exploration, hence, the core of the system is the agents. Every single agent has specific navigation algorithms that allow localization and collision-free movement in the environment. Once basic abilities of navigation are achieved by any robot, the distributed executive system enters into play. The executive system is in charge of understanding the overall status relying on the shared knowledge of the system and then commanding tasks to robots surveilling their correct execution. Every agent can access the shared knowledge of the environment that contains fixed target poses. This shared information has been chosen to lighten up the communication effort of every agent. The complete architecture can be described by Fig. 1, the functioning of every module is explained below.

2.1 Robot-Specific Navigation

Usually, mobile robots deployed in autonomous exploration tasks are characterized by a standard navigation stack, which depends on the specific locomotion system and relies on localization and mapping procedures. These operations can be achieved through the use of Simultaneous Localization and Mapping (SLAM). Many state-of-the-art algorithms provide a software implementation of this mechanism and generally differ depending on the type of sensors they

accept as input and the type of map they generate. The other module fundamental for autonomous navigation is the path and motion planner which is strictly correlated to the locomotion, kinematic, and dynamic constraints of the mobile platform. For a wheeled robot, in general, it is possible to plan movements in a 2-D plane while for an aerial robot, the implementation of a 3-D planner is necessary. Furthermore, the planner has to take care of the dynamic constraints of each agent. Once the robot has a localization system and a planner, it needs only to define the way to explore the environment. The decision strategy depends in general on many aspects of the task such as the knowledge of the environment that has to be explored. In the presented framework this task is assigned to the executive system which, having access to the shared knowledge representation, can make decisions to give a desired behavior to the multi-robot system.

2.2 Executive System

Execution and supervision of complex activities are performed by endowing each member of the team with an agent-specific executive system similar to the one proposed in [2,3]. The system is composed of 3 modules: a Long Term Memory (LTM) containing task descriptions for the specific agent, a Working Memory (WM) that maintains a hierarchical representation of the structured tasks during their execution, and a Behavior Based System (BBS) providing a symbolic representation of the agent's primitives, i.e., of controllers and perceptual modules to be actually executed (Fig. 2, left). In our cooperative exploration scenario, we design in each LTM an instance of the *exploration task* that is specifically tailored to the peculiarities and the capabilities of the agent's class (rover, drone, fixed camera, etc.). During the execution, agent-specific tasks are retrieved from LTMs and allocated into the robots' WMs in order to be monitored and further decomposed into primitive nodes/behaviors. This executive system provides agents with distributed (local) intelligence, supporting the autonomous and independent execution of the exploration task. Each agent's LTM contains a schema representation of an exploration task that is tailored to the agent-specific capabilities and can be online instantiated and executed. During the execution, if the agent is idle, a new target is sampled from the list of possible goals in the shared knowledge. The selected target is then used to instantiate an exploration task into the systems WM which is monitored and orchestrated by the executive system until finalized. When the task is over, the outcome of the execution (failure or success) is indirectly communicated to the team by adjusting the sampling probabilities of targets in the shared knowledge.

2.3 Shared Knowledge Representation

In the proposed framework, all the agents of the system can access shared knowledge from the different robots. In particular, agents have to constantly publish their pose through the transformation tree of the multi-robot system, in this way, the executive system on every agent can keep track of the pose of the whole team and it can decide the next task following the logic explained in Sect. 2.2.

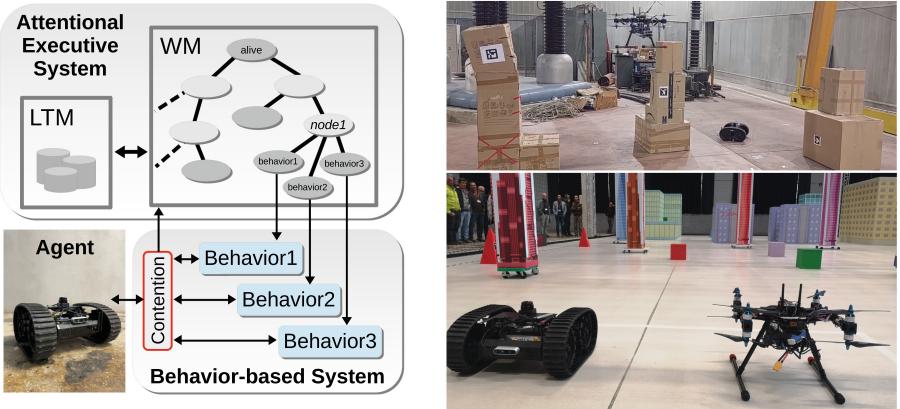


Fig. 2. Representation of the executive system for a single agent (left) and multi-robot system deployed in our flying arena (right-up) and in the Leonardo Drone Contest arena (right-down)

Furthermore, it is possible to define a list of pre-selected fixed targets that will be included in the transformation tree, these target poses will be used by the executive system as map goals to explore the environment. These targets can be defined in the setup phase of the system by leveraging the previous knowledge of the environment if there is any. This synthetic representation of the environment has proven to be efficient during extensive tests because it reduces to the bone the necessary amount of data to be exchanged by the agents (only poses and task accomplishment/failure are communicated).

3 Case Study

To test the framework presented above, a heterogeneous multi-robot system composed of a wheeled robot and a drone has been implemented. The ground platform has four in-wheel independent motors and they are coupled by tracks that make it kinematically equivalent to a differential drive robot. The aerial platform is a coaxial octacopter. The ground vehicle is equipped with a 2D TOF 360 Lidar and a frontal RGB-D camera, while the aerial vehicle has a tracking camera that retrieves an odometry measurement, and a RGB-D camera. The choice of this sensor configuration comes from the regulation of the challenge where the system was employed in: the Leonardo Drone Contest [8]. To achieve robust and stable autonomous exploration, both robots need navigation algorithms, as specified in Sect. 2.1. To this end, the ground robot exploits the ROS navigation stack [6], using Gmapping [4] for SLAM and TEB [7] for motion planning. As for the aerial agent, we deployed RTAB-Map [5] for 3D mapping, using depth images from the RGB-D camera and the odometry from the tracking camera to refine global positioning, and a custom RRT-based 3D motion planner endowed with a fifth-order polynomial trajectory generator for obstacle free

navigation. Before the competition (Fig. 2, right-down), the whole system was tested extensively in the PRISMA Lab’s flight arena (Fig. 2, right-up), recreating a contest-like scenario.

4 Conclusions

The proposed approach to develop multi-robot autonomous exploration focuses on the possibility of adding or removing (even losing) agents. This ability is achieved by implementing a distributed executive system on top of lightweight communication and synthetic shared knowledge of the environment. The system has been tested by extensive missions and during the Leonardo Drone Contest. The multi-agent framework achieved a fully autonomous mission of more than 15 min until the batteries were completely discharged. As further improvements, the map constructed by the several agents could be shared and fused to get a more accurate determination of the explored environment. Also, the executive system can be developed to specify more complex behaviors.

References

1. Agency, D.A.R.P.: Darpa subterranean (subt) challenge (2021). <https://www.darpa.mil/program/darpa-subterranean-challenge>
2. Caccavale, R., Finzi, A.: Plan execution and attentional regulations for flexible human-robot interaction. In: 2015 IEEE International Conference on Systems, Man, and Cybernetics, pp. 2453–2458 (2015). <https://doi.org/10.1109/SMC.2015.429>
3. Caccavale, R., Finzi, A.: A robotic cognitive control framework for collaborative task execution and learning. *Top. Cogn. Sci.* **14**(2), 327–343 (2022). <https://doi.org/10.1111/tops.12587>. <https://onlinelibrary.wiley.com/doi/abs/10.1111/tops.12587>
4. Grisetti, G., Stachniss, C., Burgard, W.: Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Trans. Rob.* **23**(1), 34–46 (2007). <https://doi.org/10.1109/TRO.2006.889486>
5. Labb  , M., Michaud, F.: Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *J. Field Robot.* **36**(2), 416–446 (2019). <https://doi.org/10.1002/rob.21831>. <https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21831>
6. Marder-Eppstein, E., Berger, E., Foote, T., Gerkey, B., Konolige, K.: The office marathon: robust navigation in an indoor office environment. In: 2010 IEEE International Conference on Robotics and Automation, pp. 300–307 (2010). <https://doi.org/10.1109/ROBOT.2010.5509725>
7. R  smann, C., Hoffmann, F., Bertram, T.: Integrated online trajectory planning and optimization in distinctive topologies. *Robot. Auton. Syst.* **88**, 142–153 (2017). <https://doi.org/10.1016/j.robot.2016.11.007>. <https://www.sciencedirect.com/science/article/pii/S0921889016300495>
8. S.p.A., L.C.: Leonardo drone contest (2023). <https://www.leonardo.com/it/innovation-technology/open-innovation/drone-contest>

AI and Robotics



Learning-Based Ground Vehicle Navigation in Outdoor Unstructured Environments

Simone Palazzo, Dario C. Guastella, Giuseppe Vecchio,
Riccardo E. Sarpietro^(✉), Giuseppe Sutera, Francesco Cancelliere,
Giovanni Muscato, and Concetto Spampinato

Department of Electrical, Electronic and Computer Engineering Catania, University
of Catania, Catania, Italy
riccardo.sarpietro@phd.unict.it

Abstract. Autonomous navigation in outdoor unstructured environments is still an open challenge in field robotics, due in part to the difficulty to recognize and evaluate distances from obstacles and to identify type and slope of terrain. We present our current research on autonomous ground robot navigation in outdoor environments. Lying at the intersection of robotics and artificial intelligence, we investigate vision-based methods, integrating unsupervised learning and domain adaptation techniques, for improved sim-to-real capabilities. We validate the proposed methods with on-field experiments on real unmanned ground vehicles, thus assessing the feasibility of the developed navigation methods.

Keywords: ground robot navigation · unstructured environments · deep learning · domain adaptation

1 Introduction

Identifying navigable paths and executing appropriate control actions are critical for enabling a robot to navigate safely and autonomously in unstructured outdoor environments. This potential ability is essential in various domains, e.g., search and rescue, precision agriculture, and planetary exploration. Terrain traversability assessment has become a commonly used strategy, largely thanks to the progress made in learning-based methodologies [1]. In some research, terrain traversability is addressed as a traversal cost regression task, where inverse reinforcement learning [2] is utilized to generate a cost map for path planning. Other methods aim to categorize different terrain types [3] or distinguish between areas that are “traversable” versus “non-traversable” [4].

Nonetheless, there is still no established technique for converting the results of traversability analysis into concrete driving commands. More recent off-road navigation approaches rely on end-to-end models that directly convert exteroceptive data (a mix of geometric and visual information) into control commands

[5]. While these approaches are efficient, they obscure the link between the perceived environment and the resulting driving decisions, as they lack intermediate layers that can be easily understood by humans.

This paper presents our recent advancements in autonomous outdoor navigation. We introduce an innovative approach to traversability estimation that relies solely on RGB data [6]. Additionally, we explore a learning-based framework for traversability assessment that integrates supervised learning on simulated datasets with self-supervision [7] and domain adaptation [8] using real-world, unlabeled samples. Finally, we propose a navigation strategy that leverages these traversability predictions to guide the vehicle through uncharted environments, validated through both field experiments and simulations in challenging, obstacle-rich terrains.

2 Methodology

Traversability Formulation. We frame traversability estimation as a vector regression task, where each element of the target vector represents the traversability score for a specific region in the input image [6]. Specifically, the RGB images are partitioned into vertical bands, and a set of traversability scores corresponding to the traversable horizon within each band is regressed (Fig. 1).

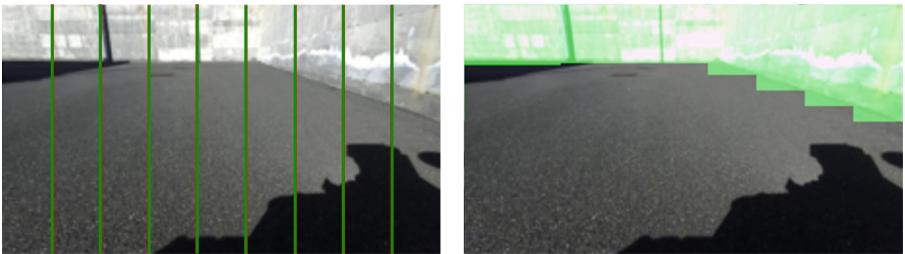


Fig. 1. Illustration of an input RGB image divided into 9 vertical bands for traversability estimation [6].

For the prediction network, we use a DeepLabV2 [9] architecture as the backbone, followed by a regression module that estimates the traversability scores for all image bands.

Self-supervised Initialization. Our model is trained on data from two distinct domains: the *source domain* (synthetic images with computer-generated traversability labels) and the *target domain* (real-world images without manual annotations during training).

We employ a domain adaptation strategy, as described in [6], to enable robust traversability predictions across both synthetic and real environments. To minimize computational overhead and avoid overfitting, we do not fine-tune the

DeepLabV2 backbone during training. Instead, we address domain shift issues through a self-supervised initialization phase. This phase pre-trains the model on both real and synthetic datasets without traversability labels to learn generalizable features before applying supervised learning. The self-supervision method used in this work is SwAV [7], a clustering-based technique that learns data representations by predicting the cluster assignment of one image view based on the representation of another. This method ensures consistency between different views of the same image.

We modify the original SwAV framework to work with features extracted from the model’s backbone rather than directly using raw pixel data. This adjustment enforces local consistency between features derived from the final convolutional layer of the traversability estimation network.

Unsupervised Domain Adaptation. Although self-supervision on both synthetic and real datasets helps the model learn initial features for both labeled and unlabeled data, it does not fully address the distribution gap between the domains. As a result, a model trained on synthetic data may struggle to generalize effectively to real-world inputs due to persistent differences in feature distributions. To mitigate this, we incorporate *gradient reversal layers* (GRL) [8]. In practice, the traversability estimation branch, trained in a supervised manner on the synthetic domain, is supplemented by an additional domain classification branch. This branch aims to identify whether an input image belongs to the source or target domain. The primary goal is to force the model to learn domain-invariant features that make it challenging for the classification branch to differentiate between domains. In essence, if features from both domains are indistinguishable, the traversability estimation branch should perform consistently across both.

Navigation Control. Using the traversability predictions from the process described above, our navigation algorithm directs the robot toward a vertical segment in the camera view that is deemed traversable, ensuring that neighboring segments are also safe to traverse. This strategy allows for adaptive path corrections. The segment’s position within the image determines the vehicle’s steering direction, while the traversability score regulates forward speed: lower scores result in slower speeds, while higher scores lead to faster movement.

3 Experimental Results

We constructed a real-world dataset by manually controlling an unmanned, rubber-tracked ground robot designed for navigating outdoor terrains. The robot, equipped with a camera, captures 1280×720 RGB images at 15 fps. The dataset contains 419 curated images. The synthetic dataset was generated using a simulator built with Unreal Engine, which provides photorealistic scenes and supports automatic data labeling [10]. The synthetic data, collected within a meadow environment, spans a large $3,600 \text{ m}^2$ area that closely mimics the real-world conditions. The dataset includes more than 2,200 RGB frames, each annotated with traversability information generated by raycasting techniques within



Fig. 2. Traversability outputs with different components of the proposed method.

the simulator, detecting collisions with scene objects. Outputs of different model variants are shown in Fig. 2, illustrating the improved traversability predictions of our final model. When trained solely on synthetic data and tested directly on real-world inputs (shown in the “Base” column), the model exhibits significant errors. While domain adaptation improves performance on simpler terrain, adding self-supervision further enhances results but sometimes results in overly optimistic predictions in specific regions. Combining self-supervision with domain adaptation yields the best results, aligning predictions more closely with ground-truth data and ensuring safer, more conservative outputs.

In field tests, we evaluated the model’s performance in a new, challenging real-world setting, which included obstacles like rocks of varying sizes, branches, and dense vegetation—conditions more complex than those found in the training dataset. The experiments confirmed the model’s ability to accurately identify traversable areas in environments that differ significantly from the training scenarios, highlighting its robustness and generalization capacity. Figure 3 shows

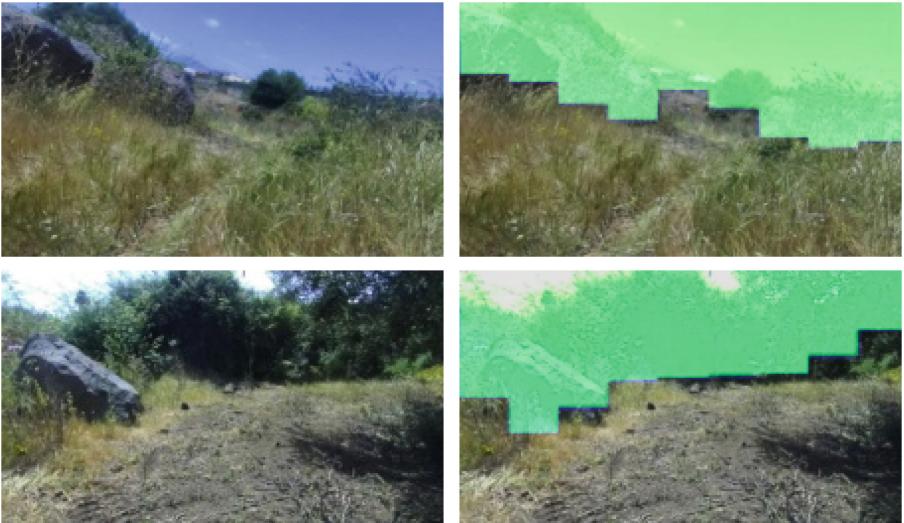


Fig. 3. Examples of successful traversability inference from the field tests: real images (left) and corresponding non-traversable area predictions marked in green. (Color figure online)

examples of successful traversability predictions in real-world field tests, where green marks indicate correctly identified non-traversable regions.

4 Conclusions

In this paper, we explored a traversability estimation approach that combines self-supervision and domain adaptation techniques. Field experiments conducted in novel and unstructured outdoor settings demonstrate that the proposed method effectively predicts traversability, enabling autonomous navigation in challenging environments. In the future, we plan to enhance the model's handling of occlusions caused by vegetation and obstacles like rocks. One possible improvement involves integrating traversability estimates with confidence scores from a segmentation model such as DeepLabV2, allowing the model to mark regions as non-traversable if confidence levels fall below a certain threshold.

Acknowledgement. G. Muscato, C. Spampinato, G. Sutera and F. Cancelliere acknowledge financial support from PNRR MUR project PE0000013-FAIR. D. C. Guastella acknowledges support by the project PON R&I REACT-EU.

References

1. Guastella, D.C., Muscato, G.: Learning-based methods of perception and navigation for ground vehicles in unstructured environments: a review. *Sensors*, **21**(1) (2021)
2. Pflueger, M., Agha, A., Sukhatme, G.S.: Rover-IRL: Inverse reinforcement learning with soft value iteration networks for planetary rover path planning. *IEEE Robot. Autom. Lett.* **4**(2), 1387–1394 (2019)
3. Rothrock, B., Kennedy, R., Cunningham, C., Papon, J., Heverly, M., Ono, M.: “SPOC: deep learning-based terrain classification for mars rover missions. In: *AIAA SPACE 2016* (2016)
4. Chavez-Garcia, R.O., Guzzi, J., Gambardella, L.M., Giusti, A.: Learning ground traversability from simulations. *IEEE Robot. Autom. Lett.* **3**(3), 1695–1702 (2018)
5. Nguyen, A., Nguyen, N., Tran, K., Tjiputra, E., Tran, Q.D.: Autonomous navigation in complex environments with deep multimodal fusion network. *IROS 2020*, 5824–5830 (2020)
6. Palazzo, S., et al.: Domain adaptation for outdoor robot traversability estimation from RGB data with safety-preserving loss. In: *IROS 2020*, pp. 10014–10021 (2020)
7. Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised learning of visual features by contrasting cluster assignments. In: *Advances in Neural Information Processing Systems* (2020)
8. Ganin, Y., Lempitsky, V.S.: Unsupervised domain adaptation by backpropagation. In: *International Conference on Machine Learning*, pp. 1180–1189 (2015)
9. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *arXiv preprint arXiv:1606.00915* (2016)
10. Vecchio, G., et al.: MIDGARD: a simulation platform for autonomous navigation in unstructured environments. *arXiv:2205.08389* (2022)



Enhancing Robotic Demonstration-Based Learning Method with Preliminary Visual Target Localization

Pasquale Foggia, Francesco Rosa^(✉), and Mario Vento

University of Salerno, 84084 Fisciano, SA, Italy
{pfoggia,frosa,mvento}@unisa.it

Abstract. A significant challenge in robotics is teaching robots to replicate tasks from a single visual demonstration. Imitation Learning is a valuable approach that allows training end-to-end control architectures that can replicate the intent of the demonstrator. However, a common issue is that these systems frequently manipulate the incorrect object. Our study introduces a novel approach that leverages the ability to explicitly solve relevant problems for task resolution, such as target object localization. Our validation shows that the proposal overtakes the leading method thanks to its ability to locate the target object.

Keywords: robot learning · imitation learning · multi-task learning

1 Introduction

An interesting problem in cognitive robotics is teaching a robot the ability to replicate a task after being shown a demonstration of what the user wants to be accomplished. This ability would be extremely useful in modern industrial settings, including dynamic environments where humans can ask a robot to bring a tool or perform assembly operations, which can vary based on the manipulated object and the assembly procedure itself. A versatile and intuitive programming approach is essential to realize these applications, eliminating the need for tedious task-specific coding. Drawing inspiration from how humans learn through visual demonstrations, we can create a system capable of replicating tasks based on observed actions. This approach enables human operators to program the robot effortlessly by merely recording the task execution when a new assembly operation is required, enhancing flexibility and adaptability. The problem can be formalized as the learning of a parametric *policy function* $\pi(c, s_t)$, that takes as input a representation c of the desired task (e.g. a suitable encoding of the demonstration video) and of the *state* of the system s_t (representing the inputs from the available sensors at time t); π returns the action a_t that the robot must perform at time t to proceed towards the completion of the task (for instance, a_t could be the position/orientation of the end effector, the forces or torques to apply at the joints etc.).

In Imitation Learning, π is learned using a set of examples of executions of the task where the action, at each instant t of the execution, is provided by a human expert (for instance, using a tele-operation interface [6]). The learning problem is framed as a supervised learning, in which the system tries to minimize the difference between its behavior and that of the human expert. Authors in [1] proposed a preliminary work in this context, training a policy represented by a Transformer based architecture, that is able to replicate different variations of the pick-place task starting from a demonstration of the desired variation. Authors in [4] have further developed this idea by training a policy that responds to video demonstrations for multiple tasks. The policy is designed not only to handle various versions of the same task but also to adapt to entirely different tasks. The problem related to Imitation Learning is that the quantity being optimized in the learning phase is not directly related to the success of the task. Thus, it may happen that while the learned policy π on the average chooses an action very similar to that of the human expert, it may fail the task because of an error in a critical point of the trajectory; for instance, the “last-centimeters errors” reported in [3], or the manipulation of the wrong object among the similar ones present in the scene. All the proposed methods share a common approach based on end-to-end architectures, that directly translate inputs into the corresponding action. This approach requires the model to learn an implicit representation that encodes both the intent of the task and the current scene state, like the location of the object of interest, to inform the control module. In this paper, we present a modification to this common approach, where the system is forced, during the training phase, to produce an explicit representation of the position of the object that has to be manipulated. Our idea is that in this way the system will be more focused on this aspect of the task, and will be less likely to pick the wrong object, improving the success rate of the task. We have validated this hypothesis by performing an experimental comparison between the state-of-the-art MOSAIC system [4], and a version of it modified according to our proposal, using a large dataset generated according to the original MOSAIC paper.

2 Proposed Method

Our starting point was the state-of-art method, MOSAIC [4], depicted in Fig. 1a. This framework comprises two primary components: the MOSAIC backbone and the Control Module. The former processes the current state s_t , that is the image currently seen by the camera, and the command c^i , represented using 4 frames from the demonstration video. Its objective is to form an embedding z_t that implicitly encodes both the environment state and the command intent. This is used by the Control Module that has to generate a valid action. However, the learning procedure, which focus on matching the target agent trajectories distribution, might not effectively incorporate in the embedding some crucial details for solving tasks, such as the target object location. Based on these considerations, we have chosen to make explicit some relevant information, such as the target object location. To achieve this, we have developed a new component, named *Conditioned-Target Object Detector* (CTOD). This module takes

the current state and the command to determine the target object position using a category-independent bounding box, i.e., it identifies the location without recognizing the category. The CTOD utilizes a two-stage Mask-RCNN [2] for bounding-box regression. It takes in input a feature-wise modulated agent scene embedding, obtained through the FiLM conditioning layer [5], used to obtain a representation that effectively integrates the command information about the target object into the scene embedding. After the CTOD is fully trained, it is integrated into the larger MOSAIC framework, as depicted in Fig. 1b. While learning the policy the CTOD parameters are kept fixed. Particular emphasis is placed on enhancing the Control Module so that it can effectively utilize the positional information of the target, which is encoded as a bounding box. By incorporating this step into the training routine, we intentionally introduce a beneficial bias that simplifies the learning of action generation, making the overall problem easier.

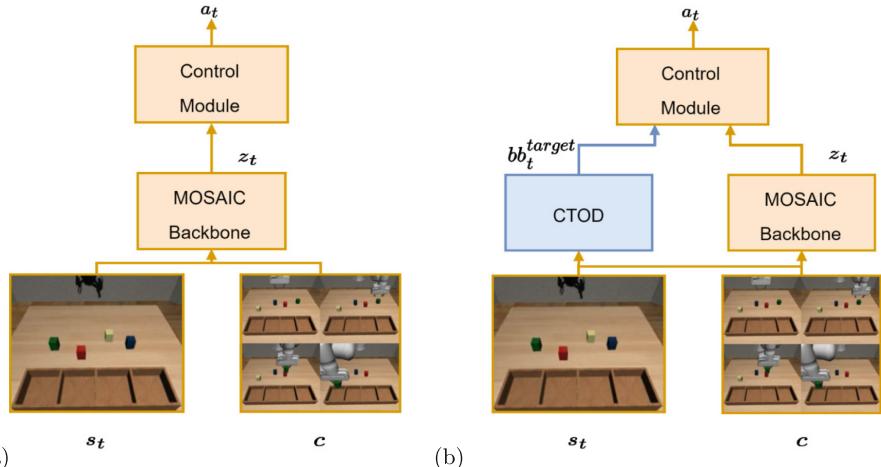


Fig. 1. Comparison between the classic end-to-end architecture (a) and our staged architecture (b), where the CTOD module explicitly compute the position of the target object and feeds it to the Control Module.

3 Experimental Results

To evaluate the effectiveness of our approach, we performed a comparative analysis between the state-of-art MOSAIC method [4] and our proposed system. This comparison is centered around two specific tasks: Pick-and-Place, and Nut-Assembly, as illustrated in Fig. 2. Both tasks are composed of multiple variations, 16 for pick-and-place, 9 for nut-assembly. For replicability, we employ the Robo-suite simulation environment [7]. Notably, with respect to the original dataset

used in MOSAIC we introduced two significant changes: we replaced the target agent with the UR5e robot and the demonstrator with the Panda robot; moreover we used a slightly different environment. These alterations ensure that the generated dataset aligns with our real-world experimental settings, as one of our goals is to assess our proposed system in real-world scenarios. About dataset generation, we adopted the same procedure described in [4]. We generated **100** trajectories for each task variation and for both the target agent and the demonstrator. Moreover, during training each agent trajectory was paired with all the demonstrator videos for a given variation. Regarding testing, we run **10** separate trials for each variation. Each trial was represented by a novel instance, generated by varying object arrangements. Each set of trials was repeated 3 times, without altering the generated scenes. All models were tested with the same scenarios for fair performance comparison.

Alongside the MOSAIC baseline, and our proposed method MOSAIC-CTOD. We also tested an ideal system, indicated as MOSAIC-GT-BB, that receives in input the ground-truth bounding box; this method can be seen as an upper bound for our system performance.

The methods were evaluated using three key metrics: – *Reaching Rate*, indicating the frequency of the robot reaching the target; – *Picking Rate*, reflecting the accuracy of the robot in grasping the target object; – *Success Rate*, measuring the overall task completion rate. Table 1 reports the averaged performance of the best checkpoint tested after training. From the results, it is evident that the inclusion of bounding-box prediction empowers the system to consistently identify, reach, and handle the target object, demonstrating robustness to variations in object configurations. Notably, there was a marked improvement in the overall success rate for both tasks. Detailed analysis of rollout videos revealed that the controller steers the robot towards the predicted bounding boxes, emphasizing the importance of this information in guiding action generation. However, most errors were attributed to the misidentification of the final position.

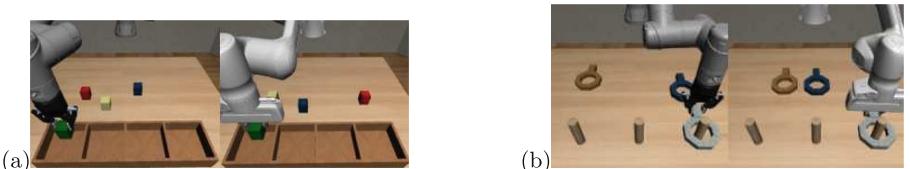


Fig. 2. Example of pick-place (a) and nut-assembly (b) scenarios. The UR5-e, positioned on the left, serves as the agent, whereas the Panda, positioned on the right, acts as the demonstrator.

Table 1. Performance comparison on pick-place and nut-assembly tasks. MOSAIC is the baseline, MOSAIC-GT-BB is an ideal method (not realizable), MOSAIC-CTOD is the proposed method.

| Task | Method | Reaching Rate [%] | Picking Rate [%] | Success Rate [%] |
|--------------|---------------------|-------------------|------------------|------------------|
| Pick-Place | MOSAIC | 63.96 ± 1.44 | 62.50 ± 1.65 | 59.17 ± 0.72 |
| | <i>MOSAIC-GT-BB</i> | 100.00 ± 0.00 | 97.29 ± 0.95 | 74.79 ± 2.00 |
| | MOSAIC-CTOD | 99.33 ± 0.36 | 91.46 ± 2.19 | 77.92 ± 1.80 |
| Nut-Assembly | MOSAIC | 37.41 ± 2.79 | 35.93 ± 1.69 | 33.70 ± 3.28 |
| | <i>MOSAIC-GT-BB</i> | 99.63 ± 0.64 | 97.41 ± 0.64 | 65.93 ± 0.64 |
| | MOSAIC-CTOD | 98.89 ± 1.11 | 97.41 ± 2.03 | 64.07 ± 0.64 |

4 Conclusion

In our study, we addressed the challenge of instructing a robot to perform tasks by mimicking actions shown in demonstration videos. A key issue we tackled is the target misidentification. To resolve this, we developed a new component called the *Conditioned-Target Object Detector* (CTOD), designed to accurately identify the target object. Our experimental validation revealed that integrating the CTOD into an end-to-end learning system significantly enhances the robot's task execution success rate. This improvement is due to the system's ability to use relevant information when deducing actions, which in turn simplifies the learning process for the control function.

References

1. Dasari, S., Gupta, A.: Transformers for one-shot visual imitation. In: J. Kober, F. Ramos, C. Tomlin (eds.) Proceedings of the 2020 Conference on Robot Learning, Proceedings of Machine Learning Research, vol. 155, pp. 2071–2084. PMLR (2021). <https://proceedings.mlr.press/v155/dasari21a.html>
2. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-CNN. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2980–2988 (2017). <https://doi.org/10.1109/ICCV.2017.322>
3. Jang, E., et al.: Bc-z: zero-shot task generalization with robotic imitation learning. In: A. Faust, D. Hsu, G. Neumann (eds.) Proceedings of the 5th Conference on Robot Learning, Proceedings of Machine Learning Research, vol. 164, pp. 991–1002. PMLR (2022). <https://proceedings.mlr.press/v164/jang22a.html>
4. Mandi, Z., Liu, F., Lee, K., Abbeel, P.: Towards more generalizable one-shot visual imitation learning. In: 2022 International Conference on Robotics and Automation (ICRA), pp. 2434–2444 (2022). <https://doi.org/10.1109/ICRA46639.2022.9812450>
5. Perez, E., Strub, F., De Vries, H., Dumoulin, V., Courville, A.: Film: visual reasoning with a general conditioning layer. In: Proceedings of the AAAI conference on artificial intelligence, vol. 32 (2018)

6. Zhang, T., et al.: Deep imitation learning for complex manipulation tasks from virtual reality teleoperation. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 5628–5635 (2018). <https://doi.org/10.1109/ICRA.2018.8461249>
7. Zhu, Y., et al.: robosuite: a modular simulation framework and benchmark for robot learning. arXiv preprint [arXiv:2009.12293](https://arxiv.org/abs/2009.12293) (2020)



Towards Robotic 3D Surface Processing with Global Local Neural Region Descriptor Fields

Anish Pratheeckumar^{1,2(✉)}, Markus Ikeda¹, Andreas Pichler¹,
and Markus Vincze²

¹ Profactor Gmbh, Im Stadtgut D1, 4407 Steyr-Gleink, Austria
`{anish.pratheeckumar,markus.ikeda,andreas.pichler}@profactor.at`
² Vision for Robotics Laboratory, ACIN, TU Wien, 1040 Vienna, Austria
`vincze@acin.tuwien.ac.at`

Abstract. Recent developments show one-shot region of interest knowledge transfer across category level objects utilizing implicit neural networks. This work extends the current state of the art with a hybrid global-local feature extraction and documents the first practical application of this technology targeting robotic surface processing.

1 Introduction

3D object surface processing is an integral part of manufacturing including operations such as polishing, grinding, painting etc., [1]. Many of the surface processing operations are still performed manually [1]. Shortage of skilled workers, high labor costs etc., motivate the utilization of industrial robots which enables flexible, and low cost process execution [1]. Most research works with regard to robotic surface processing consider a specific object undergoing mass production or local regions with a generic process execution [1, 2]. Considering the current trend shift in manufacturing from mass production to mass personalization [3] we have to account for a high mix low volume environment. This calls for moving a significant step ahead in the direction of process knowledge adaptability to diverse objects within a specific object category. In this paper, we attempt to address such an application of the process knowledge transfer for robotic 3D surface processing. Specifically, given the process associated with a particular surface region of interest (ROI) on a reference object, the robot treats similar surface region on an unseen category level target object, as illustrated in Fig. 1.

We realize this goal by following our recently proposed geometric surface region correspondence [7] approach, utilizing implicit networks [4]. The implicit networks in general learns on the object boundary in 3D space, typically by ground truth data such as binary inside/outside occupancy [6], signed distance functions etc. Such a model is able to implicitly represent diverse objects within

This work was accomplished within the Lighthouse project supported by the Austrian Institute of Technology (AIT).

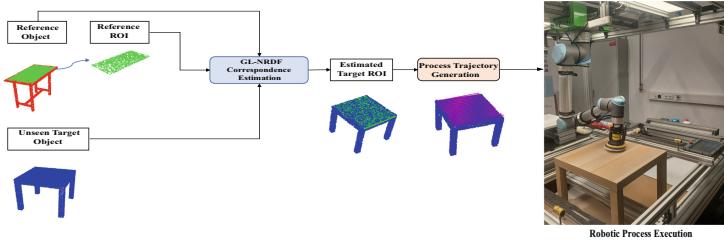


Fig. 1. Robotic 3D surface processing via process knowledge transfer with GL-NRDF.

an object category. The network architecture is typically encoder-decoder in nature, where the encoder extract the 3D geometry feature embedding, and the decoder is trained to predict for any given 3D query point the corresponding occupancy state on the object boundary. A recent work introduced Neural Descriptor Field (NDF) [4] representations obtained by the concatenated layer activations of learned implicit network decoder layers, which helps in achieving point level correspondences. Another work by Liu et al. showcase improvements in the shape dense correspondence by application of object constituent part embedding based self, and cross object reconstructions [8]. Inspired by these developments we recently proposed a surface region correspondence estimation approach by incorporating a novel descriptor level self, and cross object reconstruction training, optimizing the descriptors explicitly across diverse objects within a category [7]. We refer to these optimized descriptors for an ROI on the object surface as Neural Region Descriptor Field (NRDF). The NRDF provides a desirable representation as the optimization promotes the descriptors of corresponding surface locations on diverse category level objects to be as similar as possible. This facilitates direct utilization of the NRDF of a reference object surface ROI to estimate the corresponding target ROI. In this work, we extend NRDF and introduce Global Local Neural Region Descriptor Field (GL-NRDF) which further optimizes the NRDF representation by incorporating a hybrid global-local feature extraction approach.

2 Global Local Neural Region Descriptor Fields

Given a 3D reference object o_r point cloud $p_r \in \mathbb{R}^3$ and its surface boundary S_r , we initially indicate a surface ROI $s_r \subsetneq S_r$ to which a predetermined robotic process is assigned. Now the objective of the category level process transfer based on region correspondence is to identify a corresponding surface ROI s_t on an unseen target object o_t point cloud p_t such that the same robotic process could be executed. To this end, we propose to model the object geometry with an implicit neural network as illustrated in Fig. 2. Initially an object geometry is encoded using a global-local encoder. In contrast to our recent work [7] which considers only global PointNet [5] encoder GE , here we consider a dual encoding approach in which the global and local features are extracted. For the local feature z_l , we

introduce local encoder LE by adapting the approach in [6], which include local pooling with a shallow version of PointNet. Such local feature extraction leads to encoded feature volume z_l , and to extract point level features from this we apply a trilinear interpolation [6]. The interpolated local features and the global features z_g are concatenated along with the arbitrary query points of interest mapped to higher dimension with a linear layer. The concatenated features are then given as input to a global-local decoder B . This encoder-decoder network is then trained on ground truth data of boundary occupancy such that for any arbitrary 3D query point the model predicts the corresponding occupancy state.

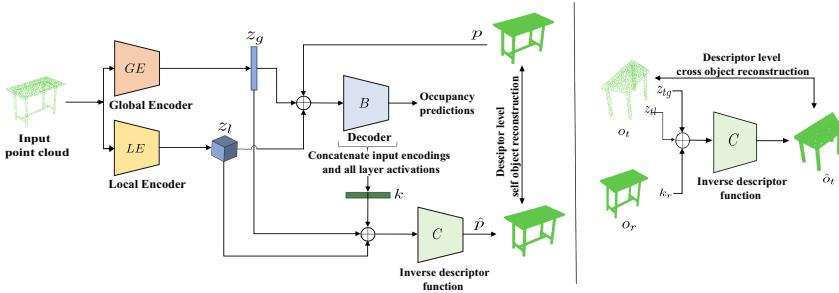


Fig. 2. GL-NRDF architecture: (left) end-to-end network architecture and the descriptor level self object reconstruction, (right) descriptor level cross object reconstruction.

The current model is able to represent the object in an implicit manner. Here the NDF descriptors k are extracted by concatenating the activations of all layers of the global-local decoder along with the input feature encoding. We then follow our recent proposed approach of performing an optimization of these descriptors to learn explicit 3 dimensional variations of the object geometry by applying a descriptor level self, and cross object reconstructions [7]. An illustration of the reconstruction approaches are shown in Fig. 2. The reconstructions are realized utilizing an inverse descriptor function C which takes as input the global feature z_g , and local feature volume z_l along with the descriptors. The feature volume is down sampled and flattened to facilitate concatenation with the global features z_g , and the descriptors k . The objective here is that with descriptor level self object reconstruction, the feature descriptors are explicitly optimized to associate with the geometry of the 3D object as the descriptors are made capable of retrieving the corresponding geometry with the inverse descriptor function. Furthermore we enforce the extracted descriptors to be consistent with the diverse objects within a category by performing cross object reconstructions such that given any arbitrary reference object descriptor k_r , and a target object global z_{tg} and local feature z_{tl} encoding the target shape is retrieved.

Optimization by descriptor level reconstruction result in the global-local descriptors of corresponding locations on the objects in a category to be as close as possible. We refer to such optimized descriptors for an ROI on the

object surface as GL-NRDF. We propose to utilize these GL-NRDF descriptors for estimating the corresponding region across diverse objects of a category. Specifically, when a reference object and a reference ROI is provided, we extract the optimized GL-NRDF descriptors and use this along with the target object global-local feature encoding to extract a corresponding surface ROI on the target object.

3 Experiments and Results

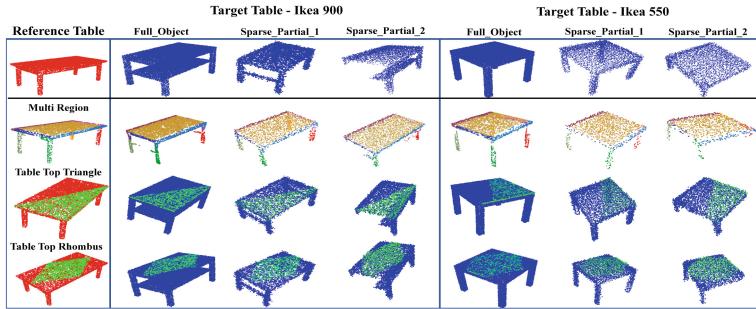


Fig. 3. Qualitative results. First row - the input reference object (red) and the target objects (blue). Second row - multiple reference surface ROI (Multi Region) on the reference object, and the GL-NRDF estimated corresponding regions on target objects. Third and Fourth row - Recovery of partial table top portions.

Table 1. Evaluation for surface ROI correspondence estimation with CD-L1 metric

| Regions | Ikea 900 | | | Ikea 550 | | |
|--------------------|--------------|-------|-------|--------------|-------|-------|
| | FO | SP1 | SP2 | FO | SP1 | SP2 |
| Multi Region | 0.033 | - | - | 0.018 | - | - |
| Table Top | 0.016 | 0.019 | 0.020 | 0.018 | 0.030 | 0.033 |
| Table Top Triangle | 0.015 | 0.019 | 0.018 | 0.021 | 0.026 | 0.027 |
| Table Top Rhombus | 0.015 | 0.020 | 0.022 | 0.026 | 0.031 | 0.043 |

Towards a real world demonstration of the proposed approach, we consider a use case of polishing an unseen table top surface, given a reference table object and a reference ROI on it. The reference ROI is associated with a particular robotic process. Now for the unseen target table object the goal is to estimate the corresponding ROI and facilitate execution of the same process, but adapted to

the estimated target object region. We select the Ikea 900 and 550 objects as the unseen target tables. The qualitative results of estimating corresponding regions with the proposed GL-NRDF approach is shown in Fig. 3. Here we showcase the robustness of the proposed approach by considering the partial object point clouds with added random noise of mean 0 and a standard deviation of 7.5 mm. Sparse_Partial_1 (SP1) and Sparse_Partial_2 (SP2) indicates approximately 25% and 40% missing information respectively. The qualitative results in Fig. 3 shows that GL-NRDF is capable of estimating correspondences for the noise added and partial point cloud instances with reasonable quality in comparison to the Full_Object (FO). We further perform a quantitative evaluation of the correspondence estimation accuracy, measured with the L1 Chamfer Distance (CD-L1) metric [9], as shown in Table 1. Here it is observed that the deviation for SP1 and SP2 in contrast to FO case is on average only 0.006 and 0.009 respectively.

Real World Execution Setup: Our robot environment includes a workspace with UR10e Universal robot, and a low cost *Orbbec Astra Plus* 3D camera for real time point cloud data acquisition. The complete workspace set up with the unseen Ikea 550 table and the overall pipeline for the process execution is shown in Fig. 1. The known reference table (red) and its surface ROI (green), along with the unseen target Ikea 550 table point cloud data are taken as input by the pre-trained GL-NRDF model which estimate the corresponding target surface ROI. Followed by this a process trajectory generator module will generate process trajectory specific to the estimated ROI with the same process type associated with the reference surface ROI. The generated trajectory is then executed by the robot on the target table, and a video demonstration of the same is available online (<https://youtu.be/uiAISYvcChc>).

4 Conclusion and Discussion

In this work we presented GL-NRDF, towards solving robotic 3D surface processing in a low volume high mix industry environment. The successful real world demonstration on table objects showcase an initial proof of concept of the proposed application. In addition to the presented region level process transfer, the proposed method could be utilized in industry for enabling a complete object level surface process automation. This could be realized with multi region process knowledge transfer. Future work will explore the scalability, and the discriminative ability in estimating correspondences for complex surface patches.

References

1. Zeng, X., Zhu, G., Gao, Z., Ji, R., Ansari, J., Lu, C.: Surface polishing by industrial robots: a review. *Int. J. Adv. Manuf. Technol.* **125**(9–10), 3981–4012 (2023)
2. Wen, Y., Hu, J. , Pagilla, P.R.: A novel robotic system for finishing of freeform surfaces. In: International Conference on Robotics and Automation (ICRA), pp. 5571–5577. IEEE (2019)

3. Barata, J., Cardoso, J.C., Cunha, P.R.: Mass customization and mass personalization meet at the crossroads of Industry 4.0: a case of augmented digital engineering. *Syst. Eng.* (2023)
4. Simeonov, A., et al.: “Neural descriptor fields: $Se(3)$ -equivariant object representations for manipulation. In: International Conference on Robotics and Automation (ICRA), pp. 6394–6400. IEEE (2022)
5. Qi, C.R., et al.: Pointnet: deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEE CVPR, pp. 652–660 (2017)
6. Peng, S., Niemeyer, M., Mescheder, L., Pollefeys, M., Geiger, A.: Convolutional occupancy networks. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12348, pp. 523–540. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58580-8_31
7. Pratheeepkumar, A., Ikeda, M., Hofmann, M., Pichler, A., Vincze, M.: Surface region of interest retrieval for 3D objects with implicit neural region descriptor fields. (Submitted to International Conference on Robotics and Automation (ICRA)) (2024). unpublished
8. Liu, F., Liu, X.: Learning implicit functions for topology-varying dense 3d shape correspondence. *NeurIPS* **33**, 4823–4834 (2020)
9. Fan, H., Su, H., Guibas, L.J.: A point set generation network for 3d object reconstruction from a single image. In: Proceedings of the IEEE CVPR, pp. 605–613 (2017)



Enabling Cognitive Robotics Through Autonomous Motion Planning and Model-Driven Intuitive Programming

Antonio Venezia¹, Ileana Romata¹, Giorgio Nicola², Roberto Fausti², Massimiliano Nitti², Adriano Liso², Vito Renò², Nicola Pedrocchi², Manual Beschi^{2,3}, Nicola Longo¹, Giovanni Di Stefano¹, and Simone Panicucci^{1(✉)}

¹ Comau S.p.A., Via Rivalta 30 Grugliasco, Italy
simone.panicucci@comau.com

² National Research Council of Italy, Rome, Italy
nicola.pedrocchi@stiima.cnr.it

³ Università degli Studi di Brescia, Brescia, Italy
manuel.beschi@unibs.it

Abstract. This paper explores innovation in cognitive robotics to automate complex mission-critical processes. By autonomously generating its models of the surrounding environment, the system localizes its target, detects any obstacle, calculates cycle time optimization and collision-free path planning and makes decisions that intuitively refine its movements and actions. All this technical complexity is hidden behind a simple and effective user interface based on low-code programming. Comau and CNR have embedded AI-powered methodologies in an innovative model-driven intuitive programming framework, applied to a self-adaptive system that autonomously inspects helicopter blades measuring up to 7 m. The focus will be given to the system's versatility while performing tapping tests and multispectral surface inspection along the helicopters' blades.

Keywords: Cognitive Robotics · Industrial Automation · Multigoal Sampled based motion planning · I4.0 Challenge

1 Introduction

1.1 Cognitive Robotics and Industrial Applications

A step-change technology in industrial robotics is a realistic digital twin (DT) embedded with methodologies that assign jobs to human users and autonomous agents according to process models and online simulations [3]. However, autonomous tools that can solve the complexity of industrial tasks are still far from being available in the market. At the same time, a fully autonomous system introduces likely over-complexity and is often unnecessary, given human operators' expertise. Therefore, the actual industrial needs consist of having

tools supporting humans to interact easily with a powerful DT and the possibility that DT allows the plug-and-play of intelligent agents to solve specific tasks as needed.

Today, having high-quality DT is quite simple, as a digital representation of a dynamic industrial environment can be achieved using ROS [4]. However, its integration with industrial platforms overwhelms a robot integrator. Furthermore, the ROS user interfaces are far from the industrial standard and cannot cope with a cognitive approach to programming.

The challenge, therefore, is having platforms that allow a cognitive interaction between human operators and DT tools easily and intuitively. Historically, enabling ease and cognitive robot programming leads to passing from Code-Driven to Model-Driven Engineering [6], also known as low-code programming. Model-driven engineering aims to master complexity by splitting a complex problem into smaller parts, solving each, and finally putting the solutions together to form a comprehensive solution for the original problem [7]. Although it is not a recent framework, the first industrial applications are taking advantage of it (e.g., [2]) only recently. Nevertheless, despite the quality of market products, users still need more tools to focus on abstraction and process modeling while almost ignoring the “geometric” aspects of the application. An industrial-like cognitive and autonomous system should be able to turn symbolic/logical tasks and goals into geometric targets, constraints, and scene management.

This short communication describes the programming software (SW) tool that Comau has designed to achieve a cognitive approach to human-system interaction and developed to integrate ROS straightforwardly. This design feature has allowed the integration of an innovative multigoal motion planner and the most advanced techniques for vision systems.

1.2 Motivation

Section 3 shows the quality inspection of helicopter blades addressed using this framework. This is a challenging task characterized by high requirements in terms of accuracy and reliability. These inspections can take many hours, and many unanticipated situations could arise. A Process Engineer decides the task flow, identifies areas that must be analyzed, and identifies a series of measurements that must be performed. The position of the objects to be measured is highly inaccurate, and using fixtures is unfeasible. Helicopter blades are also bent in a way that is difficult to simulate. As such, a robot integrator would struggle to generate the part program with the thousands of points that collision-free trajectories must reach. The complexity is also given by the fact that the robot must offer at least redundant axes or a mobile platform, and the task itself is often under-constrained (e.g., the rotation around the tool axis is free). Given the massive dimension of the Inverse Kinematics (IK) null space, such complexity is also challenging for autonomous systems. This application is paradigmatic for a broad family of industrial challenges. It demonstrates the need for programming architectures, where the user defines the process flow and constraints, and the architecture turns it autonomously into a complete working program.

2 The Architecture

The architecture proposed here is a layered framework based on a micro-service architecture that relies on Docker [1] to encapsulate each methodology. The main layers are: (i) A cognitive model-driven user interface; (ii) A Perception Layer for the Scene Registration; (iii) A Multigoal Collision-Free Motion Multigoal (Fig. 1).

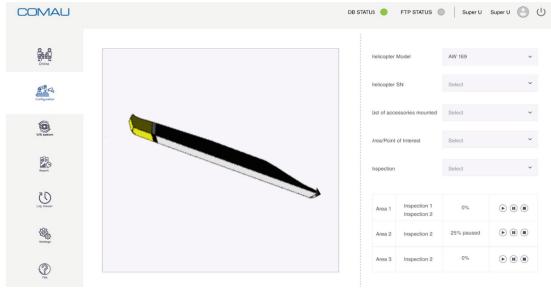


Fig. 1. Comau’s cognitive model-driven user interface for easy deployment

2.1 Cognitive Model-Driven User Interface

The platform developed and deployed by Comau allows two main options for the user: first, it can be fully customized; second, it consists of a block diagram programming interface (model-driven engineering paradigm). This block diagram programming allows the system to store a repository of completed algorithms, which can be put in a cascade to create a customized process pipeline. Those available algorithms represent atomic software functionalities (from simple tasks, such as changing I/O values, to complex tasks, like collision-free and optimized path planning) that inherit and extend abstract classes and methods to be compatible with the underlying software platform. This, in turn, allows the process expert to be the system programmer since it can completely ignore activities such as user interface, camera calibration, *etcetera*. The Model-Driven, therefore, opens a different cognitive interaction between human operators and systems, and it will allow in the future also for the integration of AI modules for the autonomous generation of process flows (exploiting the co-pilot functionalities offered by AI that are now in the hype for the code programmers).

2.2 Context Self-awareness and Objects Representation in the Scene

Among the most classic problems in robotics is the registration of a point cloud resulting from multiple camera acquisitions. The benefit of Comau’s SW architecture is that the most advanced algorithms can be used and packaged in a

container that communicates directly with the robot using REST API. The integrated methodology is a multi-step algorithm. The first step is the background cancellation. Given a set of points in a defined space, the DBSCAN algorithm groups together closely packed points. The tuning of such a methodology is lightweight and semi-autonomous, according to the mean point-to-point distance in the Point Cloud. This value depends on the 3D sensor used to acquire the point cloud. Once the object is detected and the different point clouds are stitched, the second step is to register it on its CAD model, that is, find a transformation that aligns the point cloud with the CAD model. We used a combination of the RANSAC [5] based on FPFH features for the preliminary registration and the ICP algorithm for accurate alignment, implementing a coarse-to-fine approach.

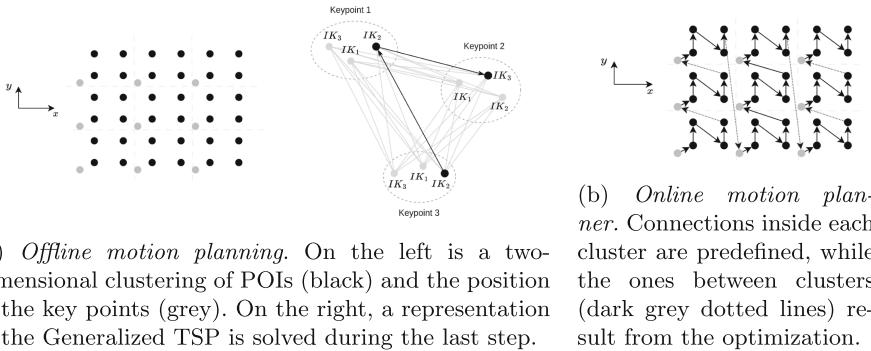


Fig. 2. Multigoal Motion Planner

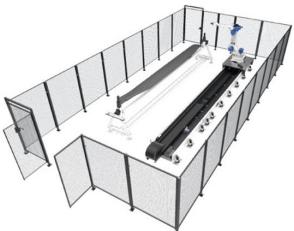
2.3 Multigoal Collision-Free Motion Planning

The Comau SW tool integrates an innovative multi-goal collision-free motion planner, dealing with thousands of underconstrained Cartesian waypoints and multiple-redundant kinematics. The intelligent approach consists of a two-stage algorithm with an offline routine to determine a global solution and an online stage that registers the solution over the actual scene and finds collision-free paths. This architecture tackles both the high dimension of the solution space and the computation constraints of the industrial application.

Offline Motion Planning consists of: (i) Clustering; (ii) Reachability analysis; (iii) Key-points sorting. The first step is to cluster the inspection points (POI). Each cluster is assigned a key point (KP) positioned above the cluster. The second step computes many inverse kinematics (IK) for each KP, resulting in a list of reachable and non-colliding joint configurations, considering the null space given by the task redundancies. Finally, we sort the KPs, defining in which order the clusters will be traversed during execution. Under the condition that

a tour must pass through exactly one configuration for each KP, the problem is a Clustered Traveling Salesman Problem over an undirected graph, and we used a nearest-neighbor approach. The offline procedure outputs the clusters of POIs and an ordered set of KPs and joint configurations.

Online Motion Planning considers the registered objects in the environment. First, offline results are registered on the actual scene. Then, for each KP, many IK are computed based on the ones determined offline, and the path between KPs is restored. Each cluster's path passes through all its POIs, starting from the KP, following the order of the points in the ordered point cloud frame Fig. 2(b). The last point of each cluster is connected to the KP of the next cluster, following the optimized result of the offline procedure. The policy used to plan the point-to-point path is double. A direct connection is planned if there are no obstacles between two nodes, or a sampling-based path planning algorithm to avoid a collision if any obstacle blocks the connection (Fig. 3).



(a) Robot on a linear Axis for the quality inspection of a blade



(b) Robot with the Tapping Tool in processing the blade

Fig. 3. Leonardo Helicopters demonstrator for the SW architecture

3 Demonstration

Given the aim of this short communication, we report some results for a quality inspection task. The first set of experiments was on the tapping of a helicopter blade. This task performs impact-controlled collision at thousands of points. The blade is manually positioned over holders. The blade is many meters long, and even a tiny error in the pose estimation results in extremities up to tens of centimeters. The CAD may also differ from the actual blade, and some areas could be covered by features not present in the CAD model. The results were remarkable. The user was able to program the tapping task easily. The error in the blade registration is constantly lower than 2 cm on average, considering the blade's total dimensions and blending nature. The total number of under-constrained Cartesian goals is tens of thousands. For each of these, tens of IK solutions were evaluated in the null space. The offline solution took nearly 45 min, while the online solution for each area took just a few minutes.

Acknowledgements. We thank our colleagues at Leonardo Helicopters, whose insight and access to their facilities have greatly assisted the research.

References

1. Docker: Docker (2013). <https://www.docker.com/>. Accessed November 2023
2. DragAndBot: DragAndBot GmbH (2017). <https://www.dragandbot.com/>. Accessed November 2023
3. Javaid, M., Haleem, A., Suman, R.: Digital twin applications toward industry 4.0: a review. *Cogn. Robot.* **3**, 71–92 (2023)
4. Open Software Robotics Foundation: ROS (2013). <https://www.ros.org/>. Accessed November 2023
5. Raguram, R., Frahm, J.-M., Pollefeys, M.: A comparative analysis of RANSAC techniques leading to adaptive real-time random sample consensus. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *ECCV 2008*. LNCS, vol. 5303, pp. 500–513. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88688-4_37
6. Schlegel, C., Steck, A., Brugali, D., Knoll, A.: Design abstraction and processes in robotics: from code-driven to model-driven engineering. In: Ando, N., Balakirsky, S., Hemker, T., Reggiani, M., von Stryk, O. (eds.) *SIMPAR 2010*. LNCS (LNAI), vol. 6472, pp. 324–335. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-17319-6_31
7. Schlegel, C., Lotz, A., Lutz, M., Stampfer, D.: Composition, separation of roles and model-driven approaches as enabler of a robotics software ecosystem. In: *Software Engineering for Robotics*, pp. 53–108. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-66494-7_3



Acroba Gym: A Unity-Based ROS Compliant Simulator for Robotics

Marco Ojer^{1,2(✉)}, Xiao Lin¹, Iñigo Mendizabal-Arrieta^{1,3},
and Antonio Tammaro¹

¹ Vicomtech Foundation, Basque Research and Technology Alliance (BRTA),
Mikeletegi Pasealekua 57, 20009 San Sebastian, Basque Country, Spain
moyer@vicomtech.org

² Computer Science and Artificial Intelligence Department, Faculty of Informatics,
University of the Basque Country (UPV/EHU), Manuel Lardizabal pasealekua 1,
20009 Donostia, Basque Country, Spain

³ Group of Automation, Robotics and Computer Vision (AUROVA), University of
Alicante, San Vicente del Raspeig S/N, Alicante, Spain

Abstract. This paper presents a cross-platform, ROS compliant and versatile virtual gym, which provides a visually appealing simulated environment with high-fidelity to validate robotic systems without the need for physical hardware and support training/testing artificial intelligence models with robots in a controlled digital space. The presented virtual gym is built based on Unity with seamlessly integrated with the ROS, where the main contributions are centered around following key aspects: simulated scene definition, features on robot/sensor/tools simulation and machine learning integration.

Keywords: robotics · simulation · robot operating system · machine learning

1 Introduction

Robotics simulators are widely used by academic and industry practitioners as they serve to validate several developments in a safe, reliable and accessible manner.

In recent years, there has been a significant academic focus on the utilization of robotics simulators as training environments for Artificial Intelligence (AI) models. These simulators provide controlled and virtual spaces in which AI agents can learn through interactions with the simulated environments and then transfer these learned behaviours to the real world. There are several popular robotics simulators such as Gazebo [1], CoppeliaSim [2] and Mujoco [3] which share features such as robot modelling, robot control, environment modelling, sensor modelling, physics modelling. Several studies such as [4–8] made comparisons among some them. The consensus drawn from these evaluations suggests that there is no universally superior simulator, as the suitability of a particular

simulator largely depends on the specific task and environment it is employed for. Industrial practitioners benefits from robotics simulation as a validation tool that servers to reduce integration time, support flexible automation and enable production optimization [9]. Industrial scenarios tend to have a wider range of custom elements that interact among them within scene.

In this work, we propose a robotics simulator which is designed to serve both academic and industrial practitioners, the proposed simulator is based on Unity [10], a widely used development engine mainly focused on video games. In comparison with aforementioned robotics simulators, Unity benefits are its high rendering capabilities, fair physics simulation, virtual and augmented reality features, cross-platform compatibility, user-friendly environment definition, a vast amount of plugins for scene and process customization and robust support from a large and active community.

One of the main contributions is that this simulator is Robot-Operating System (ROS) compatible, since ROS is a remarkably popular robotics framework widespread in the academy and that, in the recent years, is entering the industry domain. This implies that the virtual simulated robot exhibits the same control interfaces as real ROS compliant robotic systems.

2 Scene Definition

The definition of a scene is a crucial element in the simulation process. Various simulators often employ distinct formats for scene description. Within the ROS ecosystem, the Universal Robot Description Format (URDF) is widely utilized for this purpose. Importantly, URDF is increasingly gaining compatibility with other platforms beyond ROS, making it a versatile and widely applicable standard. The process of constructing a scene via URDF is a tedious work that involves by direct XML writing and visualization in an manual iterative process. To alleviate this process, Acroba-gym includes a URDF exporter, so users can generate scenes interactively through the Unity editor and then export it to a URDF file, which can later be used by ROS. The URDF exporter also takes into account the visual collision element separation during the exportation stage.

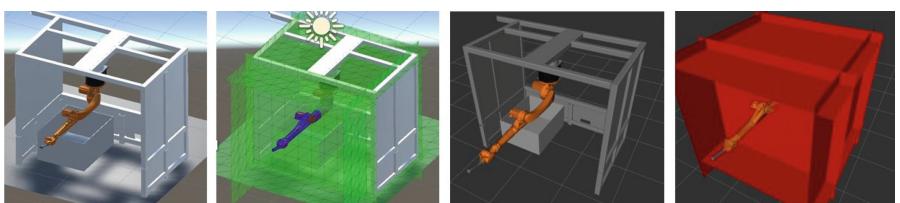


Fig. 1. Visual and collision scenes in Unity (left) and exported to urdf (right).

3 Features

Acroba-gym offers a range of essential features common to robotics systems. These include real-time access to robot joint and cartesian configuration data, as well as control interfaces via trajectory, joint or cartesian commands following the ROS standards. Moreover, it provides interfaces for runtime scene management such as object pose monitoring and modification, object addition/deletion, and visual and physical parameter customization.

Additionally, Acroba-gym includes some customized features, that are particular to certain industrial scenarios such as:

- **Vibratory Tables:** They are employed to simplify the grasping task, specially on small object by spreading and reorienting pieces across the surface, vibration can be controlled into specific directions and magnitudes.
- **Human Modelling:** Humans can be added and moved along predefined paths, see Fig. 2. This feature is very practical for safety validations.
- **RGB-D Camera Sensors:** Depth information can also be added to visual sensors output. Moreover, sensors' optical parameters are exposed and can be defined, in order to mimic real hardware and facilitate pointcloud reconstruction, see Fig. 4
- **Force-Torque Sensor:** Leveraging the Unity physics engine, we implemented a sensor capable of computing the interaction forces between other objects.

3.1 Qualitative Simulation

In the manufacturing domain, there are processes whose simulation involves highly complex physics calculations, and therefore they can not be done within universal physics engines. In such cases, qualitative simulations are employed. Qualitative simulation involves an analysis without precise numerical data, where the behaviour of the environment follows some heuristics. Acroba-gym provides some qualitative behaviours such as:

- **Magnetic and Sucking behaviour:** This is qualitatively simulated by defining an attachment region fixed to the tool. If the tool is activated, it will attach all objects in contact with this region.
- **Material Removal:** It is implemented by applying a transparent texture to a visual object. For instance, when drilling a surface, the drill encounters the surface, and the in-contact textured pixels become transparent. The system also tracks the drilled path so that when a closed curve is described, the inner surface is also removed, simulating the hole-drilling process, see Fig. 2.

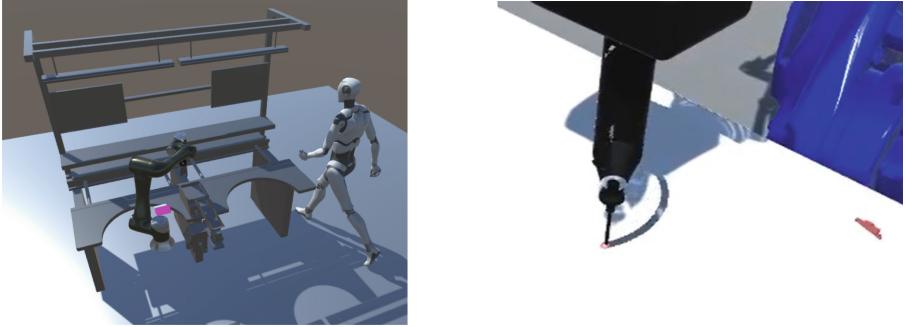


Fig. 2. Simulated human in the scene (Left), drilling simulation (Right).

4 Machine Learning Applications

4.1 Reinforcement Learning

Acroba-gym is also intended to work as a platform to train and test reinforcement learning policies. It provides runtime simulation management tools such as simulation reset and time speed modification. The aforementioned scene management capabilities allow domain randomization of the initial visual and physical conditions, which helps to leverage learning, see Fig. 3. In addition, it is designed to handle multiple simulation instances, allowing for parallel execution of simulations.

Furthermore, for seamless integration with actual hardware, the simulation allows for the flexible adjustment of sensor and control signal frequencies before each simulation run. Given that the interfaces follow the ROS standard, models trained using the acroba-gym can be effortlessly transferred to real-world scenarios that adhere to this standardized framework.

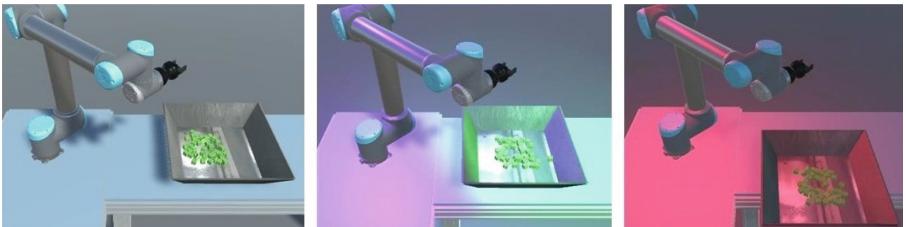


Fig. 3. Randomization of lighting setup and object position.

4.2 Synthetic Data Generation

Acroba-gym visual sensors can also provide a pixel-wise mask of the objects' instances in the image, see Fig. 4. This information is highly valuable in generat-

ing ground-truth data for computer vision applications. Therefore, acroba-gym provides tools for synthetic data generation and benefits from its high-fidelity rendering capabilities compared with other simulators.

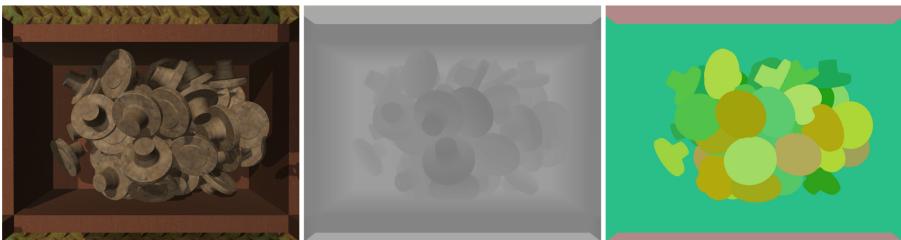


Fig. 4. Available visual information: RGB sensor image (Left), depth sensor image (Middle), object mask images (Right).

5 Conclusions

The presented work can result beneficial for both academia and industry. On one hand, industry practitioners can benefit from an easy yet complete environment customization where qualitative simulation can leverage simulation of complex manufacturing processes. On the other hand, academics can benefit from a training platform for AI models, particularly in the domains of control and vision. The choice of Unity as a base engine adds superior flexibility and rendering capabilities in comparison with other simulators. Our goal is to empower people to simulate complex procedures that are not possible or difficult to model in the current robotics simulations.

Acknowledgement. This research was carried out within the project ACROBA which has received funding from the European Union' Horizon 2020 research and innovation programme, under grant agreement No 101017284.

References

1. Koenig, N., Howard, A.: Design and use paradigms for ch1gazebo, an open-source multi-robot simulator. In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No. 04CH37566), vol. 3, pp. 2149–2154 (2004)
2. Rohmer, E., Singh, S.P.N., Freese, M.: CoppeliaSim (formerly V-REP): a versatile and scalable robot simulation framework. In: Proceedings of the International Conference on Intelligent Robots and Systems (IROS) (2013)
3. Todorov, E., Erez, T., Tassa, Y.: MuJoCo: a physics engine for model-based control. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 5026–5033 (2012)

4. Collins, J., Chand, S., Vanderkop, A., Howard, D.: A review of physics simulators for robotic applications. *IEEE Access* **9** (2021)
5. Körber, M., Lange, J., Rediske, S., Steinmann, S., Glück, R.: Comparing popular simulation environments in the scope of robotics and reinforcement learning (2021)
6. Erez, T., Tassa, Y., Todorov, E.: Simulation tools for model-based robotics: Comparison of bullet, Havok, MuJoCo, ODE and PhysX. In: *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, pp. 4397–4404 (2015)
7. Nogueira, L.: Comparative analysis between gazebo and V-REP robotic simulators (2014)
8. Ivaldi, S., Padois, V., Nori, F.: Tools for dynamics simulation of robots: a survey based on user feedback (2014)
9. Sanneman, L., Fourie, C., Shah, J.A., et al.: The state of industrial robotics: emerging technologies, challenges, and key research directions. *Found. Trends® Robot.* **8**(3), 225–306 (2021)
10. Haas, J.K.: A history of the unity game engine (2014)



Exploitation of Similarities in Point Clouds for Simplified Robot Programming by Demonstration

Philipp Möhl, Markus Ikeda, Michael Hofmann^(✉), and Andreas Pichler

Profactor GmbH, Im Stadtgut D1, 4407 Steyr-Gleink, Austria
pmoehl@profactor.at

Abstract. Programming by demonstration empowers robot operators to program robots without coding. This approach finds particular relevance in scenarios involving repetitive assembly processes composed of similar components arranged in patterns. In industrial environments with limited CAD data, like SMEs, point cloud data from scans serves as a practical starting point for further investigations. Recognizing component patterns within point clouds and analyzing demonstrated actions in specific areas are vital for guiding robots in automation tasks.

Our work introduces innovative methods designed to identify patterns within point clouds, enabling the efficient analysis of demonstrated actions and their subsequent execution by robotic systems. A primary focus of our approach is the computation of similarities within point clouds, a crucial step in pinpointing regions suitable for repetitive actions. We are particularly intrigued by primitive-based patterns, as these representations align closely with industrial object surfaces. Given the prevalence of geometric representations in industry, recognizing recurring geometries through primitive-based descriptions represents a pertinent research objective. We present the problem within a practical application context and explore the feasibility of comprehensively identifying recurrent geometries using primitive-based features.

Keywords: Geometric reasoning · programming by demonstration · point cloud · deep learning · human machine interaction

1 Introduction

Robotic assistance in industry is crucial for task precision and reducing repetitive work, yet achieving this precision in varied scenarios is challenging. This study focuses on using machine learning to enable robots to adapt actions to new objects with minimal external input, promoting self-sufficiency. Our main goal is to demonstrate intelligent methods' effectiveness in achieving autonomy in industrial settings. We employ robots for tasks like polishing, oiling, painting, and welding on unfamiliar objects, proposing a demonstration-based approach using 3D cameras and tracking tools. This paper presents our approach to improving robotic performance in dynamic industrial environments.

1.1 Problem Statement

The primary challenge of our study is to replicate actions from a reference point cloud P_1 to a new point cloud P_2 , both derived from scanning operations. The key task is identifying corresponding regions in P_2 that align with those in P_1 . We employ a latent space encoder that considers the entire point cloud's context for mapping points to their encodings. This problem formulation raises intriguing questions, including the application of point cloud registration, which aligns point clouds from different sources, considering various attributes.

In our industrial context, geometric attributes play a significant role in determining correspondences (cf. Fig. 1). We focus on replicating actions, which are typically performed on solid man-made objects and can often be broken down into basic geometric shapes (primitives like planes, spheres, and cylinders). Our goal is to develop a context-based encoder tailored to these industrial applications, emphasizing the significance of primitive-based surface characteristics.

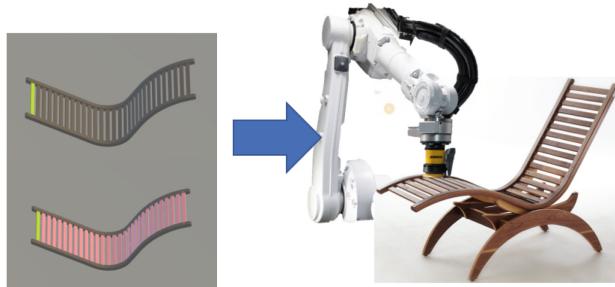


Fig. 1. Application Setup

1.2 Application

In our present application phase, we narrow our focus on a specific sub-task within the outlined problem, wherein we constrain correspondences to the replication of specific regions. This approach accommodates inherent noise differences that emerge in scan data. Thus, we operate on a single point cloud, selecting a distinct region and aiming to pinpoint all corresponding replicas within that point cloud (cf. Fig. 1). The identified correspondences are then provided to the robot as transformation matrices.

2 Methodology

2.1 Related Works

We want to highlight approaches addressing 3D object correspondences in a broader context, typically involving keypoints, semantic parts, or regions.

Keypoint-based methods [2] target feature point correspondences between objects. In contrast, part-based approaches [4] seek correspondences among similar component parts. Region correspondence approaches [5] identify corresponding regions of interest without specific part constraints. While our strategy aligns with the keypoint approach, our objective is to attain outcomes similar to region-based approaches.

2.2 Description

Our method comprises two key components, a point-wise latent space encoder and a distance-based region-growing approach for region reconstruction.

For encoding, we employ neural networks, specifically adapting the first part of HPNet [6], which has demonstrated promising outcomes in primitive-based segmentation, particularly on the mechanical parts dataset ABCParts [3]. Instead of mean-shift clustering used in HPNet for point cloud segmentation, we repurpose this part as a primitive-aware feature embedding module. By integrating the fine-tuned PointNet [1] as our backend, we encode explicit and implicit geometric information for primitive identification, facilitating comparison based on relevant geometric attributes.

After feature encoding on both point clouds, we calculate distance vectors between key-points in the selected region and the other point cloud. These vectors guide our region-growing strategy, which starts with the point of the lowest distance vector and expands, considering region size as a validation metric.

In our application, we intend to employ this method for point cloud registration. Hence, we utilize the predicted region to find an initial transformation and refine it with ICP. In this registration process, the second point cloud consistently corresponds to the original scan, with a previously matched sub-point cloud removed. We iterate this process until all replications are identified, where feature encoding is performed only once at the beginning to enhance runtime efficiency.

3 Results

3.1 Experiments

The experiment focused on a point cloud scan of our object of interest, comprising approximately 21,000 points after preprocessing and artifact removal.

The objective was to reidentify all rails in the lounger from an initial pick of 3,561 points. Given the significant information loss associated with down-sampling due to complexity, we divided the large point cloud into 30 smaller patches, each with up to 7,000 points to meet neural network and computational requirements. Patching along the z-axis (cf. Fig. 2) proved as the most effective strategy.

After patching, we encoded over all patches to maintain context. We experimented with overlapping patches and individual registration, but this method

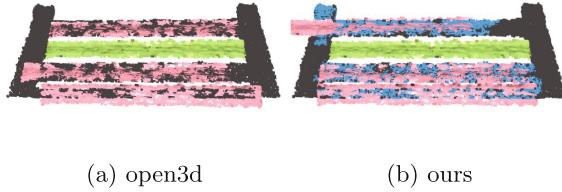


Fig. 2. Results on three patches. Source in green, matches in red, regions in blue. (Color figure online)

yielded less favorable results. We compared our method against open3d’s registration pipeline [7], focusing on metrics processing time, correspondence overlap, and inlier RMSE, across 10 runs (results in Table 1).

We observed that while both our method and open3d showed similar performance initially, manual post-selection of matches enhanced both methods’ effectiveness. Notably, our region-growing approach demonstrated superior performance compared to the combination with ransac-based global registration. The analysis also revealed a consistent relative arrangement of points within primitives of the same type (see Fig. 2).

Table 1. Selected scores and average wall time for full scanned object.

| Method | Time [s] | Overlap | RMSE |
|---------------|----------|-------------|-------------------|
| open3d | 107.13 | 0.79+/-0.09 | 0.00154+/-0.00042 |
| open3d+manual | 168.21 | 0.94+/-0.02 | 0.00101+/-0.00023 |
| ours | 130.06 | 0.71+/-0.12 | 0.00115+/-0.00031 |
| ours+manual | 137.31 | 0.82+/-0.14 | 0.00105+/-0.00022 |
| ours+ransac | 142.15 | 0.63+/-0.10 | 0.00116+/-0.00033 |

3.2 Discussions

The results indicate that our method performs less effectively when directly compared to specialized approaches in the narrow sub-problem of point cloud registration. However, it’s important to remember that a general-purpose method is unlikely to outperform highly specialized ones in specific scenarios. Our method’s primary goal is not solely focused on identifying replications but rather to achieve a comprehensive geometric representation of the region, aligning with our broader aim to apply it to various sub-problems. We advocate for specialized algorithms in these areas. Our intention is merely to emphasize that new methods have the potential to bridge the gap between generalized algorithms and highly specialized ones.

Additionally, we acknowledge limitations related to learning methods. Adapting methods to new data domains, known as data drift, can limit their generalization across domains and lead to recurrent retraining costs. We aim to strike a balance between data dependency and robustness across diverse situations, similar to the bias-variance trade-off. However, once effectively trained, learning methods require less parameter tuning compared to other approaches, which can be sensitive to parameter choices.

Many methods for similar tasks fix the number of points in training objects, mainly due to improved network performance on specific tasks and computational constraints. However, this limitation can be problematic, particularly for detail-rich objects, where downsampling may result in information loss. To address this challenge, one approach involves breaking the object into patches of the required size and applying the method to each patch, though this introduces its own set of challenges, particularly in determining the optimal patching strategy, which can vary by object.

We argue that learning-based methods can bridge the gap in general correspondence finding, but should not be used yet in situations where a specific method is available. Our future work will broaden testing across a more diverse object set. We will particularly explore new applications, like replicating a wider range of complex actions on to fully unseen objects, to demonstrate the extensive potential of our approach.

Acknowledgments. This work was accomplished within the AIT-Lighthouse project (Pillar 2) and the EFRE-FTI-DemoDatenPro project as well as financed by the Austrian Institute of Technology (AIT) and by research subsidies granted by the government of Upper Austria.

References

1. Qi, C.R., et al.: Pointnet: deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 652–660 (2017)
2. Shi, C., et al.: Keypoint matching for point cloud registration using multiplex dynamic graph attention networks. IEEE Rob. Autom. Lett. **2**, 8 (2021)
3. Sharma, G., Liu, D., Maji, S., Kalogerakis, E., Chaudhuri, S., Méch, R.: PARSENET: a parametric surface fitting network for 3D point clouds. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12352, pp. 261–276. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58571-6_16
4. Zhao, H., et al.: Point transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 16259–16268 (2021)
5. Denitto, M., et al.: Region-based correspondence between 3d shapes via spatially smooth biclustering. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 4270–4279 (2017)
6. Yan, S., et al.: HpNet: deep primitive segmentation using hybrid representations. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 2733–2742 (2021)
7. Choi, S., et al.: Robust reconstruction of indoor scenes. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5556–5565 (2015)



Intuitive Cobot Programming for Small-Medium Enterprises

Niccolo Lucci¹⁽⁾, Elias Montini^{1,2}, Isacco Zappa¹, Andrea Maria Zanchettin¹,
and Paolo Rocco¹

¹ Politecnico di Milano, Piazza Leonardo da Vinci 32, Milan, Italy
`{niccolo.lucci,isacco.zappa,andreamaria.zanchettin,paolo.rocco}@polimi.it,`
`elias.montini@supsi.ch`

² University of Applied Sciences and Arts of Southern Switzerland, Lugano,
Switzerland

Abstract. AI-powered and intuitive robotic programming represents a game changer in manufacturing industries that demand flexibility and promptness to production changes. This work proposes an intuitive and code-free method that exploits Programming by Demonstration. The method relies on executable semantic behaviour trees that model demonstrated *Skills* and encapsulate their semantic representation, the robot's primitives, and all the possible modes for completing them. Such models can be intuitively created by unskilled operators, ensuring cost-effective adaptation to changing production requirements. The method has been validated through experiments in an industrially relevant use case.

1 Introduction

Collaborative robots aim to bring disruptive innovation to European Small-Medium Enterprises (SMEs) through programming intuitiveness and production flexibility. However, for the moment, SMEs failed to exploit their true potential. Indeed, the claim that collaborative robots are user-friendly and quickly deployable for new *Tasks* is only partially valid.

Therefore, intuitive methods, like the robot's free drive, should be exploited to let the operator demonstrate the *Task* to the robot [1]. From the *Task*'s demonstration, the robot should understand its semantic [4]. Understanding is provided by semantic knowledge, modelled through the evaluation of *Predicates* that abstract raw information into high-level knowledge (like PDDL). On the other hand, robotic primitive execution and reactivity are granted by behaviour trees [2]. Finally, all these features should be encompassed inside a system accessible by unskilled operators through a Graphical User Interface (GUI) [3].

This work leverages kinesthetic demonstrations to deduce the action semantics, along with the several modes to execute the action, how each of these modes influences and alters the robotic environment throughout their accomplishment, and ultimately how the robot executes them. The model is abstracted from the

specific scenario and reused with different parameters specified by a semantic planner. Such a model is based on the concept of behaviour trees that grant promptness and reactivity to unexpected changes. The system is integrated into an intuitive GUI, deployed on a portable device, that lets unskilled operators program the robot easily.

2 Methodology

2.1 Definitions

A set of symbols $P(C)$ and their groundings needs to be provided to evaluate the state of the environment in each time instant k during the operator demonstration. In particular, these symbols, also known as *Predicates*, can be evaluated as True or False once their parameters C are specified. Consequently, these *Predicates* discretise the continuous state of a demonstration, in particular $S_k = \{P(\bar{C}) \mid P(\bar{C}) = \text{True}\}$.

Therefore, a *Skill* demonstration starts at S_k and ends at S_{k+1} . Within this interval, Predicates $P(\bar{C})$ can be added $\text{add}(\bar{C})$, deleted $\text{del}(\bar{C})$ or remain constant $\text{equal}(\bar{C})$. Such elements are useful to understand under which conditions the *Skill* is applicable $\text{Pre}(C)$ and what are the consequences $\text{Eff}(C)$ [4]. However, the *Skill* needs to be executable from the robot that needs to be aware of all the semantic changes throughout the *Skill* execution. Such a particular element is called $\text{Modes}(C)$. Therefore, a *Skill*, or operator op , is defined as a tuple $op = \langle C, \text{Pre}(C), \text{Eff}(C), \text{Modes}(C) \rangle$.

A *Skill* can be executed in different ways. Therefore, the $\text{Modes}(C)$ are a set of $\text{Mode}_j(C)$. Each of them contains a sequence of changes γ_{ji} that contain the robotics primitives $\phi_{ji}(C)$ (movements or gripper actuation) and how the workspace changes by adding $\text{Cond}_{ji}^+(C)$ or removing $\text{Cond}_{ji}^-(C)$ *Predicates*:

$$\text{Mode}_j(C) = \{\gamma_{ji} \mid \gamma_{ji} = \langle \text{Cond}_{ji}^-(C), \text{Cond}_{ji}^+(C), \phi_{ji}(C) \rangle\}$$

2.2 Learning Modes Semantic and Primitives

To detect the semantic changes during \bar{j} -th *Mode*, the interval ranging from S_k to S_{k+1} is divided into $S_{k,i}$. The minimal detectable symbolic variations populates $\text{Cond}_{\bar{j}i}^+(\bar{C})$ and $\text{Cond}_{\bar{j}i}^-(\bar{C})$ as follows:

$$\begin{aligned} \text{Cond}_{\bar{j}i}^+(\bar{C}) &= S_{k,i} \setminus S_{k,i-1} = \{P(\bar{C}) \mid P(\bar{C}) \in S_{k,i} \wedge P(\bar{C}) \notin S_{k,i-1}\} \\ \text{Cond}_{\bar{j}i}^-(\bar{C}) &= S_{k,i-1} \setminus S_{k,i} = \{P(\bar{C}) \mid P(\bar{C}) \in S_{k,i-1} \wedge P(\bar{C}) \notin S_{k,i}\} \end{aligned}$$

where i starts from 1 and $S_{k,0} = S_k \forall j$. Taking as an example Fig. 1 with the *Predicates* similar to the ones defined in [4], three changes are detected. In Fig. 1c, 1f and 1g $\text{Cond}_{\bar{j}i}^+$ and $\text{Cond}_{\bar{j}i}^-$ are shown for each γ_{ji} of the \bar{j} -th *Mode* with $i = 1, 2, 3$. During a demonstration, the operator can actuate the gripper

or save the robot's waypoints with respect to the robot base. However, saving the TCP pose with respect to the robot base cannot be generalised to different scenarios. Therefore, every time a semantic change γ_{ji} is detected, all the points saved between $S_{k,i}$ and $S_{k,i-1}$ are translated according to a specific reference frame [4]. Each gripper actuation and waypoint populate the ϕ_{ji} of $Mode_j(\bar{C})$.

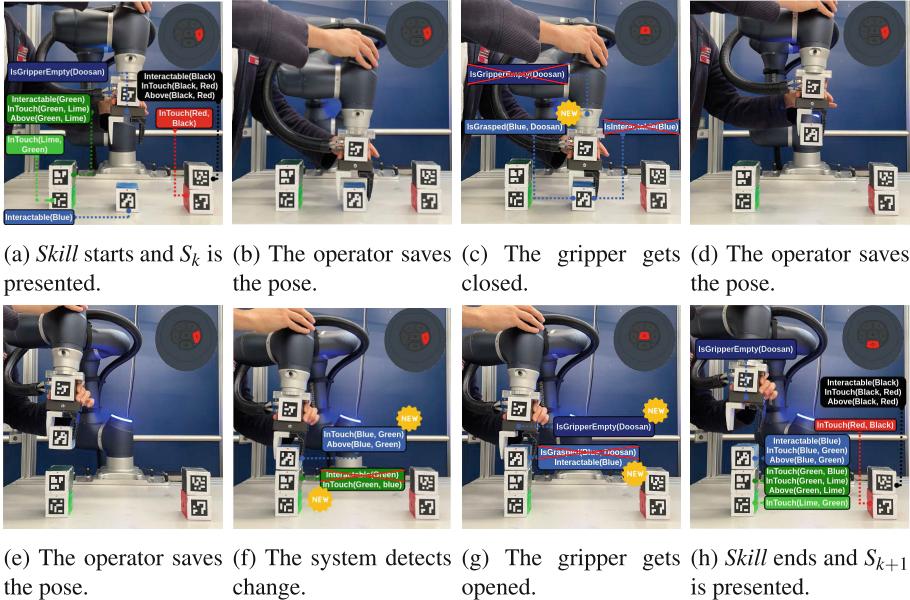


Fig. 1. The Figure shows the operator executing a demonstration: saving waypoints and actuating the gripper. Meanwhile, the system monitors the scene's semantics.

2.3 Semantic Behaviour Tree

Each *Skill* has been demonstrated with specific workspace components. To make the *Skill* generic and reusable, it is required to abstract the parameters of $Pre(\bar{C})$, $Eff(\bar{C})$ and $Modes(\bar{C})$. Each specific parameter of \bar{C} has a type that eases the abstraction. Consequently, $Pre(\bar{C})$ and $Eff(\bar{C})$ of each *Skill* can be written in PDDL.

On the other hand, the generalised $Modes(\bar{C})$ are translated in semantic behaviour trees. The shape of this tree needs to account for the sequentiality of each γ_{ji} composing the $Mode_j(\bar{C})$. Each γ_{ji} is structured as a tree (i.e. Figure 2). First, it checks if its $Cond_{ji}^+(C)$ is not already satisfied. In case the $Cond_{ji}^+(C)$ are not satisfied, $Cond_{ji}^-(C)$ are checked using a *ReactiveSequence* (R_\rightarrow). The *ReactiveSequence* guarantees continuous monitoring of the γ_{ji} 's correct execution. If such a control flow node returns *Failure*, a semantic replanning of the

high-level *Skills* is executed to account for the changed environment. In particular, the *Failure* state issues a replanning to the PDDL planner which provides the robot with a new plan that accounts for the failure. Afterwards, there are the *Move* and *Open/Close Gripper* commands whose parameters are substituted by exploiting the same mapper used for the $Pre(C)$ and $Eff(C)$. Once all the robotics primitives have been executed, the $Cond_{ji}^+(C)$ are rechecked. If they are all satisfied, passing to $\gamma_{j,i+1}$ is possible. For brevity, only γ_{j1} referred to Fig. 1c, is represented in Fig. 2. Likewise, the *Skill* is transformed into a semantic behaviour tree. In particular, the tree is composed of all its *Modes* modelled as semantic behaviour trees plus the *Skill*'s effects $Eff(C)$ at the end. This last component guarantees the *Skill*'s correct completion. The representation of the non-generalised *Skill*'s semantic tree of Fig. 1 with two *Modes*(\bar{C}) can be seen in Fig. 3.

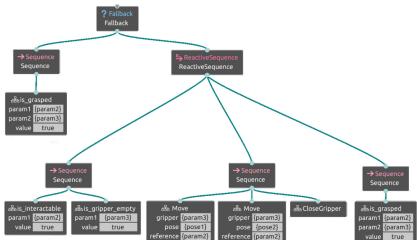


Fig. 2. The Figure shows the semantic tree for γ_{j1} from Fig. 1c.

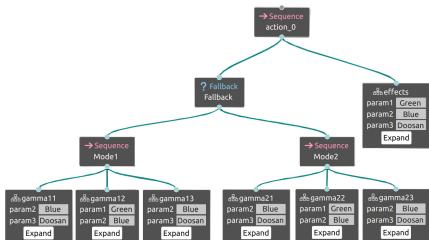


Fig. 3. The Figure shows the behaviour tree of the performed *Skill* in Fig. 1.

3 Use Case

3.1 Environment Description

The system has been validated in an industrial use case targeting kitting applications. Kitting involves positioning distinct but related items into a single unit or kit. Figure 4 provides an overview of the setup and the GUIs specifically developed to support the operator. The use case has been implemented using a Kuka LBR iiy 3 equipped with an electric gripper from Schunk and Roboception as an external camera that tracks all the components in the workspace relying on Aruco markers. The operator exploits the GUI to communicate with the system during the Teaching and Operation Phases. As shown in Fig. 4, the GUI is composed of two pages: one dedicated to the Teaching and one to Operation (i.e., *Task* execution). During the Teaching, the operator can start and stop the recording of a *Skill* and then save it. During the recording, the operator moves the robot using hand guiding, saves waypoints and actuates the gripper. During the Operation Phase, the user specifies the intended *Task* that the robot has to achieve by pressing “Save”. Afterwards, the GUI prompts feedback about the kit composition, as depicted in Fig. 4 and the operator can start the robot.

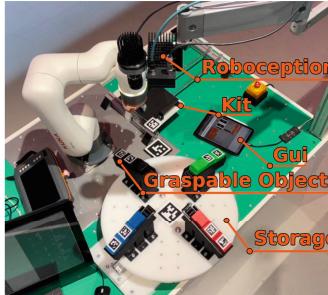


Fig. 4. The Figure shows the Kitting application and the GUI.

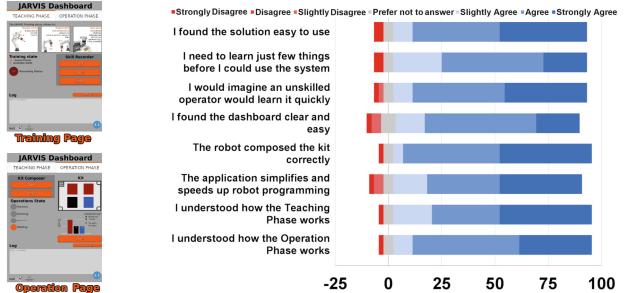


Fig. 5. The Figure shows the survey's result.

3.2 Experiment Description and Results

The system was showcased during the Automatica 2023 fair for the Kuka Innovation Award 2023 competition¹, and the validation was conducted involving 44 users. The experiment consists of:

1. The system and the Teaching Phase are introduced to the user.
2. The user can try the Teaching Phase, but they cannot save further *Skills* since this experiment targets the validation of the Operation Phase.
3. The Operation Phase is introduced.
4. The user manually composes the kit he/she wants the robot to assemble, shows it to the camera and saves the goal.
5. The robot composes the kit. The user can disturb the execution by rotating the spinning table or removing components from the sliders, evaluating the replanning functions of the system.

The results are presented in Fig. 5. In particular, the visitors submitted, on average, positive feedback (91%) about the interface usability to reprogram the robot for a different kit, clarity regarding what the robot learnt during the Teaching Phase, and how to interact with it during the Operation Phase. On the other hand, 4.2% did not evaluate the interface positively, while 4.8% did not express a preference.

4 Closing Comments

This work introduces a no-code learning framework that allows the operator, through a single demonstration, to transfer *Task* knowledge to the robot. The framework embeds a methodology that defines a semantic behaviour tree for each demonstrated *Skill*, encapsulating their semantics and how they are executed.

¹ Video of the application: <https://www.youtube.com/watch?v=zm-7KqjwoT8>.

The semantic behaviour tree allows the system to continuously monitor the environment of the robotic agent during the *Skill*'s execution, issuing a replanning of the high-level plan when the current *Skill* fails. The framework is integrated within a GUI that hides its complexity and lets unskilled operators program the robot *Task* without writing any code.

References

1. Billard, A., Calinon, S., Dillmann, R., Schaal, S.: Robot Programming by Demonstration, pp. 1371–1394. Springer, Heidelberg (2008)
2. Colledanchise, M., Almeida, D., Ögren, P.: Towards blended reactive planning and acting using behavior trees. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 8839–8845. IEEE (2019)
3. Liang, Y.S., Pellier, D., Fiorino, H., Pesty, S.: iRoPro: an interactive robot programming framework. Int. J. Soc. Rob. **14**(1), 177–191 (2022)
4. Zanchettin, A.M.: Symbolic representation of what robots are taught in one demonstration. Robot. Auton. Syst. **166**, 104452 (2023)



Real-Time 3D Reconstruction Adapted for Robotic Applications in Construction Sites

Dimitrios Katsatos¹, Dimitrios Alexiou¹✉, Theodora Kontodina¹, Ioannis Kostavelis^{1,2}, Dimitrios Giakoumis¹, and Dimitrios Tzovaras¹

¹ Information Technologies Institute, Centre for Research and Technology Hellas, Thessaloniki, Greece

{dkatsatos,dalexiou}@iti.gr

² International Hellenic University, Thessaloniki, Greece

Abstract. The integration of robot vision techniques, specifically focused on 3D reconstruction, assumes paramount significance in the construction sector, serving as a key enabler for fulfilling the imperative digitalization prerequisites inherent to the principles of Industry 4.0. This study proposes a real-time 3D reconstruction pipeline, based on common algorithms, that utilizes both RGB and depth information. Specifically, it delves into a comprehensive evaluation of InfiniTAM [5] and introduces a novel pipeline, that involves the integration of InfiniTAM with either ORB-SLAM3 [1] or RTAB-Map [3], aiming to enhance the accuracy of 3D surface reconstruction, especially in the context of robotic operations. The insights derived from this study facilitate the implementation of a robust 3D reconstruction methodology applicable to the construction industry.

Keywords: 3D reconstruction · civil engineering · robotics

1 Background

The construction industry, known for its labor-intensive nature and intricate operational contexts, is progressively incorporating 3D reconstruction methodologies, such as Building Information Modeling (BIM) systems. Within this context, robotic vision systems play a pivotal role, aiming to optimize operational efficiency, enhance safety protocols, and enable comprehensive digital representations of construction projects for improved planning, execution, and management of the relevant processes [9]. Recently, 3D reconstruction approaches have been emphasized on deep learning techniques, specifically on Neural Radiance Fields (NeRF) [4]. In the domain of robotic-based 3D reconstruction techniques for the construction sector, Inspection-NeRF [2] utilizes RGB-D images capturing surface defects, obtained via a wall-climbing robot, to generate a 3D model and its bounding box, aligning it with the NeRF implicit boundary. Additionally, in [7], semantic segmentation, employing pre-trained deep learning algorithms, is

applied to a 3D model generated by Instant-NGP to facilitate the construction of BIMs from a sequence of images captured at construction sites. Despite advancements in NeRFs, these methodologies necessitate precise recovery of absolute scale and exhibit extended computational time requirements, especially when dealing with expansive scenes, such as construction sites. In particular, 3D reconstruction pipelines that integrate camera pose estimation methods and dense meshing frameworks have demonstrated their superior feasibility for 3D reconstruction in time-sensitive construction projects, as examined in [6,8]. Kinect-Fusion acted as a precursor to real-time TSDF volumetric approaches, resulting to InfiniTAM, which encapsulates a camera pose estimation module with a keyframe-based relocizer and offers globally-consistent TSDF or surfel-based reconstruction based on RGB-D input. In this study, InfiniTAM is investigated as a real-time and modular method. Its integration with well-established visual SLAM algorithms, namely ORB-SLAM3 and RTAB-MAP, is explored to achieve more accurate results.

2 Methodology

In this study, we utilized the InfiniTAM algorithm for 3D reconstruction and investigate the substitution of its camera pose estimation component to enhance estimation accuracy in challenging construction areas. InfiniTAM, relies on RGB-D data input to generate a colored mesh. It employs camera pose estimation and infers TSDF-based volumetric reconstructions by utilizing hash tables. The TSDF reconstruction method involves segmenting the scene into rigid subscenes and optimizing their relative poses to construct a coherent overall map. Specifically, during camera pose estimation, it aligns surface measurements with the model prediction. During our experimentation, we observed poor relocalization performance in conditions where partial and noisy surface measurements were captured with reduced overlap, as illustrated in Fig. 2. Consequently, we strategically exploited its modular design to enhance 3D reconstruction accuracy by seamlessly integrating external camera pose estimations into the existing workflow. To accomplish this, we adopted well-established approaches, such as ORB-SLAM3 and RTAB-Map, leveraging their capabilities to bolster the reliability of the visual odometry estimation process amidst challenging construction site conditions. The aforementioned external camera pose estimation approaches enhance estimated odometry accuracy using robust loop closure detection and graph optimization techniques. Both ORB-SLAM3 and RTAB-Map demonstrate adaptability in handling dynamic environments. Our implementation strategy entails the deployment of the ZED 2i stereo camera for extracting depth information in semi-indoor environments characterized by challenging illumination conditions, where conventional RGB-D cameras may encounter limitations.

3 Experimental Evaluation

To access the performance of these approaches, data were captured at a semi-indoor construction site, focusing on vertical surfaces before and after concrete's

appliance, as depicted in Fig. 1. The datasets consisted of RGB and depth images with a resolution of 640×360 pixels, captured by a ZED 2i camera, a prevalent sensor in robotic applications, mirroring robotic platform motion. Additionally, a high-resolution supplementary 3D point cloud was captured by a FARO Focus S-150 terrestrial laser scanner to establish ground truth.



Fig. 1. Samples of (2) ground truth point clouds, before and after shotcrete application, respectively.

For performance metrics, fitness-score (f-score), inliers root mean square error (IRMSE), and mean re-projection error (MRE) were employed. The f-score indicates the overlap between the target and the reference point cloud, whereas IRMSE measures the average of the mean square point-to-point distance of the correspondences (inliers). MRE evaluates the distance between the projected 2D locations of 3D points in the synthesized model and their actual 2D locations in the captured images.



Fig. 2. Series of (3) real-time reconstructions, starting from left to right, with InfiniTAM (INF), ORB-SLAM3-InfiniTAM (OS-INF) and RTAB-Map-InfiniTAM (RM-INF), respectively.

The application of 3D reconstruction in the construction field requires near-real-time performance alongside the generation of accurate 3D models for large scenes. The experimentation showed that vanilla InfiniTAM is highly sensitive to the sensor's abrupt movements, resulting in poor relocalization performance. Consequently, it presents a misaligned and duplicated wooden panel, and ultimately an uncompleted 3D model, as illustrated in the first dataset of Fig 3. In



Fig. 3. Qualitative comparison of methods based on the first and second dataset was presented to the upper sequences from left to right with INF, OS-INF, and RM-INF results, respectively, while a cropped surface of each dataset’s results, isolated for evaluation purposes, was depicted to the lower sequences.

Table 1. Comparative Evaluation of Different Methods

| | Dataset-01 | | | Dataset-02 | | |
|--------|------------|-------------|------------|------------|-------------|------------|
| | MRE (px) | Fitness (%) | IRMSE (cm) | MRE (px) | Fitness (%) | IRMSE (cm) |
| INF | 2.234 | 90.746 | 1.895 | 2.894 | 90.546 | 1.925 |
| OS-INF | 1.852 | 96.492 | 1.622 | 2.187 | 93.934 | 1.770 |
| RM-INF | 1.960 | 96.041 | 1.677 | 2.006 | 95.667 | 1.773 |

contrast, the adaptation of a slower capturing method in the second dataset, led to successful odometry estimation, regarding vanilla InfiniTAM. Additionally, the proposed OS-INF and RM-INF methods outperformed in both datasets, owing to their advanced relocalization and optimization techniques. For evaluation purposes, a surface of the scene was isolated for examination. In this regard, vanilla InfiniTAM managed to accurately reconstruct the region of interest in the first dataset, despite its later failure in the rest of the scene. In the second dataset, the qualitative assessment revealed InfiniTAM’s deformation on the examined surface, as depicted in the bottom sequence in Fig. 3. Conversely, robust OS-INF and RM-INF excelled with accurate and high-texture surfaces, resulting in higher f-scores, along with lower IRMSE and MRE, as presented

in Table 1. Specifically, OS-INF demonstrated slightly better performance compared to RM-INF in terms of MRE and exhibited similar fitness scores regarding the first dataset, whereas RM-INF outstanded in the second dataset, delivering a precise 3D representation, including the intricate geometric details of the surface after shotcreting application. Overall, RM-INF emerged as the most effective approach.

4 Conclusion

This study proposed a real-time robotic-based method for 3D reconstruction within construction areas. The experimental evaluation showcased that vanilla InfiniTAM failed to accurately estimate odometry in challenging scenarios, whereas the external odometry estimation of RTAB-Map or ORB-SLAM3 proved to be a feasible integrative solution. In conclusion, the collaborative combination of InfiniTAM and RTAB-Map yielded the highest precision.

Acknowledgment. This work has been supported by the EU Horizon Europe funded project “RobetArme” under the GA No: 101058731. We extend our gratitude to Christiansen & Essenbaek A/S (CEAS) for providing access to their mock-up construction site premises for data acquisition.

References

1. Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M.M., Tardós, J.D.: ORB-SLAM3: an accurate open-source library for visual, visual-inertial and multi-map SLAM. CoRR abs/2007.11898 (2020)
2. Hong, K., Wang, H., Yuan, B.: Inspection-NeRF: rendering multi-type local images for dam surface inspection task using climbing robot and neural radiance field. Buildings **13**(1) (2023)
3. Labb , M., Michaud, F.: RTAB-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation: Labb  and michaud. J. Field Robot. **36** (2018)
4. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: representing scenes as neural radiance fields for view synthesis (2020)
5. Prisacariu, V.A., et al.: InfiniTAM v3: a framework for large-scale 3D reconstruction with loop closure. CoRR abs/1708.00783 (2017)
6. Shang, Z., Shen, Z.: Real-time 3D reconstruction on construction site using visual SLAM and UAV (2018)
7. Hachisuka, S., Tono, A., fisher, M.: Harbingers of NeRF-to-BIM: a case study of semantic segmentation on building structure with neural radiance fields (2023)
8. Yeh, C.H., Lin, M.H.: Robust 3D reconstruction using HDR-based slam. IEEE Access **9**, 16568–16581 (2021)
9. Zhao, S., et al.: Application and development of autonomous robots in concrete construction: challenges and opportunities application of unmanned aerial vehicle. Drones **6**(12), 424 (2022)



An Intelligent Robotic Platform for Fruit Selective Harvesting

Clemente Lauretti¹(✉), Christian Tamantini¹, Alessandro Zompanti²,
Sara Cimini³, Laura De Gara³, Marco Santonico², and Loredana Zollo¹

¹ Unit of Advanced Robotics and Human-Centred Technologies, Campus Bio-medico University of Rome, Rome, Italy

{c.lauretti,c.tamantini,l.zollo}@unicampus.it

² Unit of Electronics for Sensor Systems, Campus Bio-medico University of Rome, Rome, Italy

{a.zompanti,m.santonico}@unicampus.it

³ Unit of Food Science and Nutrition, Campus Bio-medico University of Rome, Rome, Italy

{s.cimini,l.degara}@unicampus.it

Abstract. This paper introduces a robotic platform designed specifically for selective fruit harvesting. The platform combines (i) an intelligent module for detecting fruit and classifying its ripeness using multispectral sensing, and (ii) a DMP-based motion planning system that enables a robotic manipulator to pick fruits and place them into a basket. We tested this platform using the Tiago robot, which was equipped with a 7-DoF robotic arm, an RGB-D camera, and a VIS-NIR multispectral device. The robot was programmed to identify six stages of tomato ripeness and to perform harvesting tasks at nine different positions. The results confirmed the platform effectiveness in assessing various ripeness levels, with an average classification accuracy of 93.72%, and its ability to adapt its movements to different targets, achieving a maximum position and orientation error of 8 mm and 0.09 rad, respectively. Additionally, the platform successfully performed all tasks, achieving a 100% success rate.

Keywords: Agricultural Robotics · Selective Harvesting · Robot Motion planning · Multi-spectral Sensor

1 Introduction

Robots could be highly effective in agricultural settings, particularly for labour-intensive and repetitive tasks such as fruit picking [1], which can expose farmers to unsafe conditions and chronic occupational health issues. To achieve effective and precise selective harvesting, it is crucial to overcome several challenges, including the capabilities to: i) identify fruits and determine their exact location on the plants; ii) assess their stage of ripeness; and iii) strategically plan robot

movements while accommodating the diverse conditions of the agricultural environment [2]. This paper introduces a robotic system featuring an intelligent fruit selection tool that uses multispectral sensing and a motion planner inspired by human movements using Dynamic Motion Primitives (DMPs) [3, 4]. The multispectral sensing component, comprising an RGB camera and VIS-NIR multispectral sensors [5], captures extensive spectral data on crops. This data allows the robot to precisely identify and categorize the ripeness of fruits using machine learning techniques such as ReliefF and Support Vector Machine (SVM). The implementation of DMPs enables the robot to tailor its movements to various types of crops and environmental conditions, ensuring safe operations [3]. The effectiveness of this platform was validated experimentally using the Tiago robot, which successfully reached nine target positions and accurately classified six stages of tomato ripeness: green mature, breaker, turning, pink, light red, and red-ripe.

2 The Proposed Platform for Fruit Harvesting

Figure 1 shows a block diagram of the proposed robotic platform, which is composed of two main components: i) an intelligent fruit selection module grounded on multispectral sensing that accurately detects and classifies fruits, and ii) a motion planner based on DMPs designed to adapt robot movement to varying types of crops and environmental conditions. Detailed descriptions of each module are reported in the following.

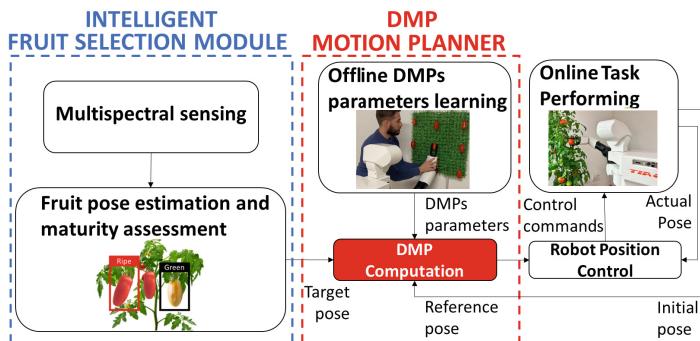


Fig. 1. The proposed platform for selective harvesting (block scheme)

2.1 The Fruit Selection Module

The intelligent fruit selection module is composed of two main components: i) a fruit detection and pose estimation module that employs an RGB-D camera, and

ii) a fruit ripeness classification module that utilizes a VIS-NIR multi-spectral device. Fruit detection and pose estimation are carried out using a modified Yolov3 neural network. This network processes images from an RGB-D camera (specifically the Astra RGB-D camera produced by Orbbec), delivering fruit pose information as output. On the other hand, the classification of different fruit ripeness stages is performed through sophisticated algorithms based on automatic feature selection and predictive classification models. Automatic feature selection is executed using the ReliefF method, which examines the spectral data gathered by the multi-spectral device, automatically identifying critical features linked to fruit ripeness, regardless of the fruit type. These features are subsequently used to train predictive models via supervised learning techniques, such as SVM, to discern various ripeness stages. The hardware components of the multi-spectral device include: i) a broadband LED (specifically the SMB1N-BB450 from Roithner Lasertechnik GmbH, which emits light across a spectrum of 340–1050 nm), and ii) two multi-spectral sensors (the C12880MA and the C14384MA, both produced by Hamamatsu Photonics, which operate within the visible to near-infrared spectra).

2.2 The Motion Planner Based on DMPs

The learning-by-demonstration approach based on DMPs involves two main stages. The first stage, referred to as the offline task learning phase, involves capturing the trajectories executed by a human demonstrator. During this phase, the demonstrator manually guides the robotic arm through the desired motions using a hands-on technique. The embedded position sensors within the arm record the joint movements. These recorded trajectories are then analyzed to extract DMP parameters using a Locally Weighted Regression (LWR) algorithm.

In the second stage, known as the online task performing phase, the DMP parameters relevant to the specific sub-task and target position are retrieved from the dataset using a lookup table method. The DMPs for each Cartesian Degree of Freedom (DoF) are computed as a sum of Gaussian kernels, which are weighted by the extracted DMP parameters. These computations are then used as inputs for controlling the robot position.

The theoretical framework for DMPs is described by the following equation:

$$\tau \ddot{y} = \alpha_y (\beta_y (g - y) - \dot{y}) + f_y \quad (1)$$

In this equation, τ represents a time constant, α_y and β_y are positive constants, y_0 and g denote the initial and final points of the trajectory, respectively, and f_y is a forcing term that creates an attractor landscape. This term is formulated as:

$$f_y(x) = \frac{\sum_{i=1}^N \Psi_i(x) \psi_i}{\sum_{i=1}^N \Psi_i(x)} x(g - y_0) \quad (2)$$

Here, $\Psi_i(x)$ is a Gaussian function, expressed as:

$$\Psi_i(x) = \exp\left(-\frac{1}{2\sigma^2}(x - c_i)^2\right) \quad (3)$$

In these equations, σ_i , c_i , and N indicate the width, centers, and the number of Gaussian functions, respectively. The weight parameters ψ_i (the DMP parameters) are tuned to fit the recorded trajectory, and x is a state variable used to eliminate time dependency from the system.

The extraction of DMP parameters ψ_i as detailed in Eq. (2) is achieved through a LWR algorithm. For a detailed description, see [4].

3 Experimental Validation

3.1 Validation and Results for the Fruit Selection Module

The fruit selection module was experimentally tested on tomatoes, one of the most widely consumed fruits globally. We gathered 450 VIS-NIR spectral measurements from tomatoes at various ripeness stages: Green Mature (GM), Breaker (B), Turning (T), Pink (P), Light Red (LR), and Red-Ripe (RR). A database of multispectral signals was collected, and supervised learning algorithms (SVM), were trained and validated using this database. The classification accuracy for each tomato ripeness stage achieved by our method is: 94.29% for GM, 90.00% for B, 88.24% for T, 85.00% for P, 86.67% for LR, and 97.17% for RR.

3.2 Validation and Results for the Motion Planner

The experimental validation of the motion planner was conducted in two phases. In the first phase, a human demonstrator taught the robot the harvesting tasks for four distinct target positions (the corners as depicted in Fig. 2-a) using a hands-on method [6]. The DMP parameters were computed from these demonstrations and stored in a database [7]. In the second phase, the robot autonomously executed the task at nine predefined target positions, as shown in Fig. 2-a, and at five specific positions on a real plant where red tomatoes were located (shown in Fig. 2-b). Performance metrics, specifically Target Position Error (TPE) and Target Orientation Error (TOE), were computed for the DMPs [8] to assess the system generalization capabilities. The results are illustrated in Fig. 3).



Fig. 2. Position of the fruits to be reached by the robot

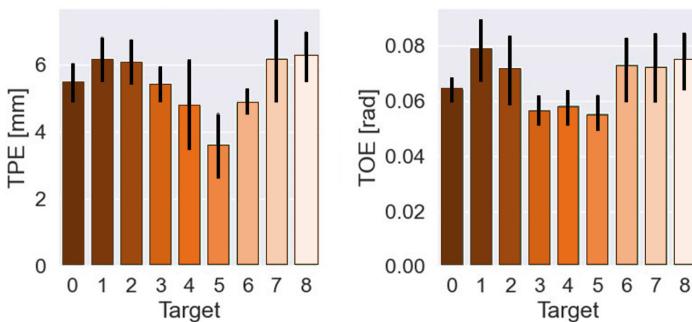


Fig. 3. Results obtained for the motion planner based on DMPs

4 Conclusions

The results underscore the robust performance of the proposed platform, which successfully assessed various ripeness stages with an average classification accuracy of 93.72%. Additionally, the platform demonstrated good generalization capabilities in its motion planning, achieving maximum position and orientation errors of 8 mm and 0.09 rad, respectively.

Acknowledgment. This work is supported by the Italian Ministry of Education, Universities and Research (Miur) with the project FUTURE AI RESEARCH (FAIR) CUP: C53C22000800006. Clemente Lauretti is funded by the PON “Ricerca e Innovazione” 2014–2020.

References

1. Bac, C.W., Van Henten, E.J., Hemming, J., Edan, Y.: Harvesting robots for high-value crops: state-of-the-art review and challenges ahead. *J. Field Robot.* **31**(6), 888–911 (2014)

2. Mahmud, M.S.A., Abidin, M.S.Z., Emmanuel, A.A., Hasan, H.S.: Robotics and automation in agriculture: present and future applications. *Appl. Modell. Simul.* **4**, 130–140 (2020)
3. Lauretti, C., Tamantini, C., Zollo, L.: A new DMP scaling method for robot learning by demonstration and application to the agricultural domain. *IEEE Access* (2024)
4. Lauretti, C., Tamantini, C., Tomè, H., Zollo, L.: Robot learning by demonstration with dynamic parameterization of the orientation: an application to agricultural activities. *Robotics* **12**(6), 166 (2023)
5. Lauretti, C., et al.: A low-cost multispectral device for in-field fruit ripening assessment. *IEEE Sens. J.* (2023)
6. Lauretti, C., et al.: Comparative performance analysis of M-IMU/EMG and voice user interfaces for assistive robots. In: 2017 International Conference on Rehabilitation Robotics (ICORR), pp. 1001–1006. IEEE (2017)
7. Tamantini, C., Cordella, F., Lauretti, C., Zollo, L.: The WGD-a dataset of assembly line working gestures for ergonomic analysis and work-related injuries prevention. *Sensors* **21**(22), 7600 (2021)
8. Lauretti, C., Grasso, T., de Marchi, E., Grazioso, S., di Gironimo, G.: A geometric approach to inverse kinematics of hyper-redundant manipulators for tokamaks maintenance. *Mech. Mach. Theory* **176**, 104967 (2022)



AI-Enabled Disaster Response Planning for Multi-robot and Autonomous Systems via Task Scheduling and Path-Finding

Matteo Lanzarini^(✉) and Lorenzo Marconi

Alma Mater Studiorum, Università di Bologna, 40136 Bologna, Italy
`{matteo.lanzarini3,lorenzo.marconi}@unibo.it`

Abstract. With the increasing interest in autonomous vehicles and robots, new systems that can handle heterogeneous Multi-Robot and Autonomous Systems (MRAS) are needed. In this paper, we want to propose a system to coordinate and manage a generic unmanned team of land and aerial heterogeneous robots in a highly dynamic environment, to address emergencies and hazardous environments, such as in Disaster Response (DR) scenarios via a rapid scheduling and allocation algorithm. To do this we propose a greedy heuristic algorithm to solve this dynamic problem while also considering all the major constraints a fleet of robots could incur, by decomposing the whole problem and optimising over each of its sub-parts.

Keywords: Scheduling · Allocation · Disaster Response · Multi-Robot and Autonomous Systems · Hierarchical Path Finding

1 Multi-robot Systems in Disaster Scenario

Only in 2019, over 90 million people around the world were affected by natural disasters [1], and only between 2000 and 2014 over one and a half million people lost their lives due to natural disasters [2,3]. The last Turkey-Syrian earthquake of February 2023 alone killed more than 40,000 people. Newer Disaster Response (DR) technology is needed to help human operators find and take care of victims of these kinds of events. During a disaster or in the aftermath, robots can work as proxies for human operators to explore hazardous environments, do faster reconnaissance with more sensing capabilities, and deal with heavy loads such as carrying equipment or injured people. Since 2001 at least 40 cases of robot-assisted intervention for DR are documented [4–9], but only 4 of them include a Multi-Robot Systems (MRS) and only 2 involved Multi-Robot and Autonomous Systems (MRAS). As noted in [10], in the last decade multi-robot systems-related work has progressed consistently, but there is still a capability divide between current proficiency and those needed for MRAS for Disaster Response that are: reliability, scalability, autonomy, and heterogeneity. This paper aims to contribute to providing MRAS with a planning tool that could address all of these characteristics.

2 Our Contributions

Numerous research and applications exist that combine task allocation and motion planning for MRS [11–15], but in a disaster response scenario like the ones presented are not suitable, because these are reactive approaches that preclude the operator with the capability to evaluate the execution of the mission in advance. Instead, if the operation is planned a priori, the human operator could supervise it and make adjustments, while the human rescuers could coordinate with the robots. Proposing a schedule and task assignment for an MRAS to the operator could lead to insight into the future of the mission, and could be used for long and mid-term prediction and consequential modification by the operator, as suggested in [16].

As it is presented, this issue falls into a very complex type of dynamic programming combined heterogeneous Vehicle Routing Problem (VRP) with synchronization and precedence constraints, implying that we can not compute a fixed solution a priori because the scenario will evolve during execution. In [17] a complete formulation of a Combined vehicle routing problem is presented, where multiple vehicles have to serve multiple locations, with precedence and synchronization constraints and time windows. For our case, we lost the actual usage of time windows, but we added capacities and resource constraints, together with relative execution time. Due to the heterogeneity of agents, we need to consider differentiated travel time and risk to reach each task position (an aerial agent will be faster and avoid ground obstacles more easily than a ground one). The map could also change during mission execution and thus also travel times and hazardous areas. We address this problem by integrating a fast path-finding algorithm to evaluate the distance and risk of reaching each task from each agent position. In addition to [16], we want to approach the problem by considering more agents and tasks for each mission, while also adding cooperative tasks and synchronous tasks, keeping the evaluation time as small as possible. These things are hard to be done with an Auction-Based approach, so we need to find a different one, that should be as quick as that one, and that eventually could lead to more optimal results. To solve all of the above problems, we propose a meta-heuristic greedy approach to provide a fast and satisfactory solution, in which a compromise between performance and optimality is required, that should be able to be computed on demand to quickly respond to unexpected events or operator's demand.

3 Methodology

We will consider a generic mission in a Disaster Response scenario. We have a set of agents, which could be UGV and UAV of various kinds, each one with a different set of capabilities. This set represents capabilities as sensing equipment and every type of ability that a robot in the group has that has no limited resources associated, e.g. an Infrared Sensor or an actuated robotic appendage to remove debris. Then we have the set of resources that could represent a set of first aid

kits carried by a ground agent, that has to deliver them to people in need; in general, could be every kind of equipment that has limited usage in quantities. Then we have the set of tasks in a mission. Each task could require a set of capabilities and some resources in any possible combination. A task could be done only by those agents that have enough resources and the desired capabilities set. Each task is associated with a position, an estimated duration, and a priority that will influence the order of execution of the mission. There are also precedence constraints, e.g., in a wildfire scenario, the agent cannot remove debris until the fire in the area is extinguished. We also added constraints to implement various kinds of synchronization and cooperation. We want to define an algorithm that aims to increase task throughput while prioritizing high-priority tasks, minimizing risk, and achieving mission completion in the shortest possible time, that is also fast and reliable. Our Meta-Heuristic Greedy approach consists of a multi-step local optimization, using alternately the Simplex algorithm to solve allocation problems and A* for the path-finding part until the complete scheduling is achieved. At each step, before the actual assignment, a pre-optimization is needed. We want to maximize the number of tasks assigned at each step while also considering the most valuable in priority and importance, selecting a subset of them that could be served by the free agent via a maximization mixed-integer linear problem (MILP) solved with the simplex algorithm.

The objective function of this problem assigns a fixed cost to each task: the higher the priority, the higher the cost. Also, each agent will have a weight that describes the desired involvement, e.g., to prefer the usage of more resilient vehicles over more fragile ones. In our heuristic approach, to avoid blocking multiple agents in an endless waiting condition, we use a specific constraint such that no task is assigned unless the task that it has to wait for is assigned itself at the same time. Once this first problem is solved, it is possible to proceed to the actual evaluation of travel costs via the path-finding algorithm between free agents and the chosen tasks. For computation efficiency purposes, paths are computed only for agent-task-compatible pairs.

We implemented a hierarchical path-finding algorithm combined with influence maps. We divide it into two phases: an offline one, slower but that can be done before the start of the mission, and an online phase, which is faster and will be called multiple times during the scheduling operation. In the offline phase, we apply the Delaunay Triangulation Algorithm [18], to create an undirected graph that covers the entirety of the map. With this representation, we decrease the number of information needed to fully describe the environment, while also keeping an accuracy level comparable to cell and grid approaches. Once the graph is computed, it can be used unchanged for each call of the online phase Fig. 1(a). In the online phase, we overlap an influence map over the graph of the offline phase Fig. 1(b). The influence map is a grid in which each square represents the cost of travelling over that region. It could be considered as the crossing's difficulty (like impervious terrain for UGV) or the zone's danger (e.g. the flame and heat of a wildfire). Then we run the A* algorithm [19] over the graph, where the cost of an edge depends on the sum of values of each square of the influence map crossed by

it. The lower number of nodes needed for a satisfactory solution with respect to a grid approach allows A^* to be fast enough to rely on it online. At this point the actual assignment is done, using the Simplex algorithm again to solve another Mixed Integer Linear Problem, to find the best agent-task pairs. The assignment cost for each pair depends on the result of the path-finding step, representing the length of the path, the risk traversed or a combination of both. Then we find the first-to-end task between the ones just assigned. The simulations move to that time step and update the state of each agent (resources, position and if is on duty or not) and, eventually, the influence map(i.e. extinguishing a fire lowers the risk of an area). This whole procedure will be repeated until there are no more tasks to assign or an infeasible condition is reached, such as incompatibility between agents' capabilities and tasks' requirements (e.g., no agent has the required resource for a task). Whenever there is a change in the tasks list or other optimization parameters, or the environment presents unexpected alteration like a new hazard or climate change, the scheduling could be invoked again, using as starting conditions the new information and the actual state of the team at the time of the new call.

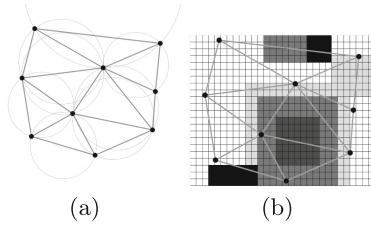


Fig. 1. (a) Example of Delaunay Triangulation for a random set of points (b) A simple example of the graph computed by the Delaunay triangulation algorithm overlapped with the influence map. Black regions are physical objects or no-crossing zones, and grey-scale squares represent a differentiated degree of risk or travelling hardness. Edges that cross black areas are not traversable.

4 Testing and Evaluation

To evaluate our solution, we implement in ROS a simulation to represent different scenarios to test the responsiveness of the algorithm and check if it satisfies all imposed constraints. We tried multiple setups, with varying numbers of agents, simpler or more dense environments, and an increasing number of tasks to assign and schedule. With up to 30 diversified tasks and 5 to 6 heterogeneous agents, the average response time recorded is typically under 2 s and is under 1 s for less complex scenarios, based on various office laptops used for testing. The imposed constraints are all respected, proving the solution to be valid, but the trade-off between the optimal ideal solution and the computed one is yet to be evaluated, due to the elevated complexity of the complete problem considered.

5 Consideration and Result

The overall aim was to obtain a tool that could be used during difficult and dangerous operations such as Search and Rescue and Disaster Response, and that can manage resiliently to unexpected events and quickly changing environments. The proposed approach could be used to coordinate a Multi-Robot and Autonomous System in very different situations and configurations, working well with heterogeneity and also with varying number of elements. Due to the need for more autonomy in the MRAS ecosystem, our solution could free more human operators from scheduling and coordination and leave dangerous action to robots.

References

1. IFRC. World Disaster Report 2020 (2021)
2. IFRC. World Disaster Report 2010 (2010)
3. IFRC. World Disasters Report 2015 (2015)
4. Murphy, R.R., Tadokoro, S., Kleiner, A.: *Disaster Robotics*, pp. 1577–1604. Springer (2016). https://doi.org/10.1007/978-3-319-32552-1_60
5. Kruijff-Korbayová, I., et al.: 2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 278–279 (2016)
6. Fernandes, O., Murphy, R., Adams, J., Merrick, D.: 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 1–6 (2018)
7. Adams, J., Hart, L., McBride, J., Merrick, D., Murphy, R.: 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 1–2 (2018)
8. Whitman, J., Zevallos, N., Travers, M., Choset, H.: 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 1–6 (2018)
9. Murphy, R.R., et al.: *J. Field Robot.* **33**(4), 476 (2016)
10. Drew, D.S.: Multi-agent systems for search and rescue applications. *Curr. Robot. Rep.* **2**(2), 189–200 (2021). <https://doi.org/10.1007/s43154-021-00048-3>
11. Moon, S., Oh, E., Shim, D.H.: *J. Intell. Robot. Syst.* **70**(1–4), 303 (2013)
12. Tan, K.C., Jung, M., Shyu, I., Wan, C., Dai, R.: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 5278–5283. IEEE (2020)
13. Li, Z., Li, X., et al.: *Open J. Appl. Sci.* **7**(10), 511 (2017)
14. Nanjanath, M., Gini, M.: *Robot. Auton. Syst.* **58**(7), 900 (2010)
15. Nanjanath, M., Gini, M.: Proceedings 2006 IEEE International Conference on Robotics and Automation, ICRA 2006, pp. 2781–2786. IEEE (2006)
16. Galati, G., Primatesta, S., Rizzo, A.: *J. Intell. Robot. Syst.* **109**(2), 24 (2023)
17. Bredström, D., Rönnqvist, M.: *Eur. J. Oper. Res.* **191**(1), 19 (2008)
18. Sloan, S.: *Adv. Eng. Softw.* **9**(1), 34 (1978)
19. Foead, D., Ghifari, A., Kusuma, M.B., Hanafiah, N., Gunawan, E.: *Procedia Comput. Sci.* **179**, 507 (2021). 5th International Conference on Computer Science and Computational Intelligence 2020



Road Pavement Inspection with UAVs Beyond Visual Line of Sight in Long-Range Operations

Javier Curado-Soriano^(✉), Francisco J. Pérez-Grau, and Antidio Viguria

CATEC, Wilbur y Orville Wright 19, 41309 La Rinconada, Seville, Spain
jcurado@catec.aero
<https://www.catec.aero/en>

Abstract. This article describes the aerial platform developed in the framework of the OMICRON project for the improvement and optimisation of road maintenance operations. The aerial platform performs road pavement quality inspection in long-range drone flight scenarios. A customized multi-purpose drone adapted to the payloads used, a high-resolution camera to obtain the digital model of the road with photogrammetry and small, lightweight cameras to Detect And Avoid (DAA) possible threats in the environment while flying in Beyond Visual Line Of Sight (BVLOS) operations have been developed. For photogrammetry, a Sony full-frame camera controlled (triggering and metadata management) with a Raspberry Pi is used. DAA processing is carried out onboard on an Nvidia Jetson Orin NX, using Artificial Intelligence (AI) and transmitting real-time video to the Ground Control Station (GCS).

Keywords: UAV · Inspection and Maintenance · Photogrammetry · BVLOS

1 Custom Aerial Platform Development

During the requirements analysis phase of the project, the conditions for the final demonstration of the entire system were defined. Among the most important requirements was the fact that at least 2 Km of road had to be surveyed. This requirement impacts directly on the selection of the aerial platform. Different commercial platforms were analyzed, but several devices had to be mounted, and in some cases the autonomy was not sufficient or in other it was not possible to modify the original layout to fit all the devices.

At this point it was decided to develop a custom aerial platform using the previous knowledge and skills acquired in previous R&D (Research and Development) projects such as Piloting or Aerial Core. Taking advantage of the fact that it is a new development and taking into account that the same platform could be used for other sectors with minor modifications (construction, mining,

topography, ...) a design with high flight autonomy and load capacity has been developed.

Figure 1 shows the first sketch, a 3D design of the aerial platform with the photogrammetry payload attached (where the processing unit, cabling, power electronics and all necessary accessories were arranged) and finally the actual UAV.



Fig. 1. Design development of the C750

The final platform specifications are shown in Table 1.

Table 1. C750 specifications

| | |
|---------------------------------|---|
| Main Engine | T-Motor MN605-KV170 |
| Main rotor propeller | T-Motor 22x6.6 Prop |
| Speed control of the main motor | (ESC) ALPHA 60A 12S |
| Motor battery | Tattu Plus 1.0 22000mAh 22.2V 25C 6S1P LiPo |
| External pilot situation | TBS Tracer 2.4 GHz R/C System - Mission Planner |
| Autopilot | Cube Orange + Ardupilot |
| Datalink | Ubiquiti R5AC-Lite |
| MTOW | 25 Kg |
| OEW | 15 Kg |
| Upper limit Payload weight | 10 Kg |
| Autonomy OEW | 25 min |
| Autonomy MTOW | 15 min |
| Range | 3.5 Km |
| Ceiling | 1524 m |
| Maximum speed | 18 m/s |

2 Digital Model for Quality Inspection

From a maintenance operations management point of view, it is crucial to identify the most damaged areas in order to prioritise work orders there.

The proposed system allows the inspection of large areas of pavement without the need to have employees physically inspecting the road, reducing the risk to employees and shortening the time taken to obtain results.

The digital model of the pavement and vertical elements surrounding the road, such as traffic signs or information panels, is obtained by applying photogrammetric techniques. Aerial photogrammetry is used to create two dimensional (2D) or three dimensional (3D) models from aerial photographs, usually it requires photographs of two or more angles of the same area in order to map the image, and it may or may not involve computer software.

The basic principle behind all photogrammetric measurements is the geometrical-mathematical reconstruction of the paths of rays from the object to the sensor at the moment of exposure. The most fundamental element therefore is the knowledge of the geometric characteristics of a single photograph [1].

Similar to the images we perceive with our eyes, a photograph is the result of a central projection, also known as single point perspective. The distances of the central point of convergence (the optical centre of the camera lens, or the exposure station) to the sensor on one side and the object on the other side determine the most basic property of an image, namely its scale.

Figure 2 shows the triangles established by a ground distance D , and the flying height above ground H_g on the terrain side and by the corresponding photo distance d and the focal length f on the camera side are geometrically similar for any given D and d : the scale S or $1/s$ of the photograph is the same at any point. The relationship between the different distances is described in Eq. 1.

$$S = 1/s = f/H_g = d/D \quad (1)$$

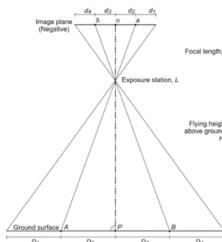


Fig. 2. Vertical photograph take over completely flat terrain [1]

This simplified model becomes more complex when the terrain is not completely flat or the photographs are oblique to the terrain but the basic idea is the same. The geometrical principles and equations mentioned above are sufficient for simple measurements. However, for the precise calculation of the 3D coordinates, it is necessary to mathematically reconstruct the ray paths both inside

and outside the camera with high precision. The parameters needed to describe the geometry of the optical paths are given by the inner and outer orientations of the camera.

The inner orientation of an aerial camera comprises the focal length (measured at infinity), the parameters of the radial distortion of the lens and the position of the so-called principal point in the image coordinate system [1,2]. The external orientation includes the X , Y , Z position of the camera in the ground coordinate system and the three rotations of the camera ω , ϕ , κ relative to this system. The elements of exterior orientation can be determined theoretically with modern high-tech GPS/INSS system simultaneous to image acquisition.

Using differential rectification and orthorectification procedures it is possible to obtain the correct place for each pixel in its planimetric position [3]. In this manner, the digital model obtained in a laboratory testing is showed in Fig. 3.



Fig. 3. Generated digital model, blue rectangles represent each acquired image

3 Detection and Avoidance System

In order to inspect long distances (about 2 Km) as proposed in the project, it is necessary to have a system that monitors the environment and is able to send alerts to the pilot in case it detects any kind of threat, as these operations are Beyond Visual Line Of Sight (BVLOS).

BVLOS operations usually require the presence of external observers to communicate any incidents to the UAV pilot, but in sometimes the external observer's point of view is not reliable (Fig. 4).

There are detection and avoidance systems based on different technologies, such as RADAR, acoustic and optical. In this case, optical systems seem to be the most appropriate. A camera array can be mounted on top of the UAV, which would not add excessive weight, and image processing can be done on board.

The selected cameras are industrial devices with gigabit Ethernet connections, although USB cameras with the same form factor exist, they have been rejected due to the insufficient number of USB ports on the processing device and the known electromagnetic interference on USB cables when running UAV motors (Fig. 5).



Fig. 4. External observer point of view. UAV (400 ft) and aircraft (600 ft) [4]



Fig. 5. Detail of the detection and avoidance system cameras

The system developed, besides to obtaining the images from the cameras and processing them with the trained Artificial Intelligence (AI) model, it has the capability to send the image streams in real time to the Ground Control Station (GCS). On the one hand there is the acquisition of the camera images, where raw frames are modified as needed (rotation, resize, reencoding, ...) and sent as h264 video streams to the MediaMTX RTSP (Real Time Streaming Protocol) server via GStreamer (open-source multimedia framework) pipelines. From there, these streams serve as input to the threat detection module, as well as to GCS visualization tools without requiring additional processing.

Threat detection module receives and decodes the incoming h264 streams into raw frames and performs object detection on them. These outputs are advertised via ROS (Robot Operating System) communications. A threat status integer indicator is published via a ROS topic, and so are ROI (region of interest) crops of the detected items when a threat is found. After inference, ROI information is painted on the images, which are reencoded into h264 and sent back to the RTSP server, advertised in a different URI (uniform resource identifier).

In the GCS, the pilot can easily identify the threat (Fig. 6), estimate the actual risk and validate or ignore the alert. If the alert is ignored, the exploration mission is continued and if accepted, the autonomous flight mission is modified to avoid the threat and will be resumed when conditions permit.

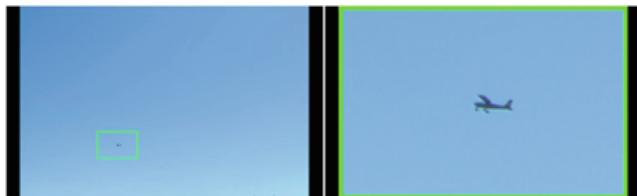


Fig. 6. Threat representation in GCS

References

1. Aber, J., Marzolff, I., Ries, J.: Photogrammetry. Small-Format Aerial Photography, pp. 23–39 (2010)
2. Fraser, C.: Digital camera self-calibration. ISPRS J. Photogrammetry Remote Sens. **52**, 149–159 (1997)
3. Fryer, J.: Elements of Photogrammetry, with applications in GIS, book review. Photogrammetric Rec. **16**, 1038–1039 (2000)
4. Vance, S., Wallace, R., Loffi, J., Jacob, J., Dunlap, J., Mitchell, T.: Detecting and assessing collision potential of aircraft and small unmanned aircraft systems (sUAS) by visual observers. Int. J. Aviation Aeronaut. Aerosp. **4**, 4 (2017). <https://commons.erau.edu/ijaaa/vol4/iss4/4>



SandRo, a Robot for Beach Waste Cleanup

Francesco Cancelliere^(✉), Luca Reitano, Dario Calogero Guastella, Giuseppe Sutera, Concetto Spampinato, and Giovanni Muscato

Department of Electrical, Electronic and Computer Engineering, University of Catania, Catania, Italy
francesco.cancelliere@phd.unict.it

Abstract. Small and large waste on beaches represents a challenge to protect the natural environment. Autonomous robots can provide valuable support in addressing such a compelling environmental issue. This work presents SandRo, an off-road mobile manipulator developed as part of a multi-robot system to automate detecting, collecting, and sorting waste on beaches. The implementation and the on-field results obtained are presented and discussed for the real-world case studies investigated in the BIOBLU project framework, demonstrating the feasibility and effectiveness of our solution.

Keywords: Unmanned Ground Vehicles · beach waste removal · vision-based manipulation · sustainable cleaning

1 Introduction

Our environment is under attack from the increasing presence of marine waste caused by human activities along coastal areas. This pressing concern requires immediate attention and action to protect our oceans and the diverse marine life that inhabits them. Thus, it is necessary to intervene and implement sustainable solutions to address this urgent issue.

To this end, robotics and automation can definitely help to mitigate and efficiently address this issue. The novel solution presented in this paper is an example of how a robotic platform can be successfully deployed on beaches for the autonomous garbage cleanup.

2 Related Work

Current autonomous robotic solutions for waste collection on sandy beaches are equipped with a sieving system consisting of meshes connected to a vibrating mechanism [1,2]. The cleaning process is typically performed by following a coverage path of the considered area. However, this approach is inefficient, both in terms of time and power consumption, as the robot has to traverse the whole area. Furthermore, it can be employed only in sandy beaches and is invasive for the environment, since also natural elements are removed (e.g., small rocks and plants, insects, etc.). Hence, it is imperative to explore alternative techniques that are more efficient, eco-friendly, and widely deployable.

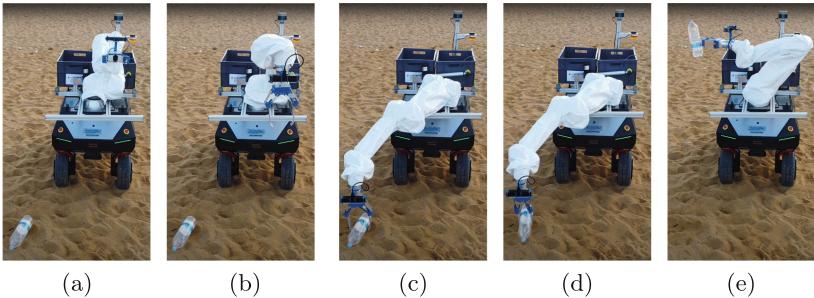


Fig. 1. A sequence of the picking up and sorting task for a plastic bottle.

3 SandRo

To tackle all the limitations described above, we developed SandRo, an off-road robotic platform, with four steerable drive wheels, and a robotic collaborative arm equipped with a suitable gripper. It has been developed as part of the multi-robot system for automatic waste beach cleanup in the framework of the BIOBLU Interreg Italia-Malta Project [3], as shown in the video below¹.

The proposed multi-robot system is composed of a ground station and an aerial platform cooperating to complete the entire cleanup mission successfully. In particular, the UAV flies over the area of interest, recognizes the waste items on the sand, and sends the list of the items' GPS coordinates. The UGV receives this list to navigate towards the waste items, thus resulting in a much more efficient waste collection mission. If an item is locally identified through the UGV's onboard camera, the robotic arm picks it up and places it in one of the designated bins. At the end of the mission, the UGV empties the bins into a waste compactor.

The sensors suite on board of the robot includes a 3D LIDAR, an RTK-GPS, and two stereoscopic cameras, one of which is mounted in a camera-in-hand configuration. ROS (Robot Operating System) [4] is the software used for managing and supervising all the different on-board components.

3.1 Navigation

The navigation of SandRo is managed by the global and the local path planners. The global path planner has to solve a *traveling salesman problem* since it has to calculate the best path for sequentially visiting all and once the points in a map. This issue is an NP-hard problem since the computation time increases exponentially with the number of elements. Each position can be seen as a node in a graph. However, if the number of nodes is high, it is common to divide all the nodes into smaller graphs (Fig. 2a), clustering near nodes with a k-means algorithm and focusing on single clusters (Fig. 2b).

¹ <https://youtu.be/s3dd51d9fxk>.

The local path planner is a DWA (Dynamic Window Approach) and allows the robot to avoid close obstacles. The data provided by the stereoscopic cameras and the LIDAR allows the obstacle detection.

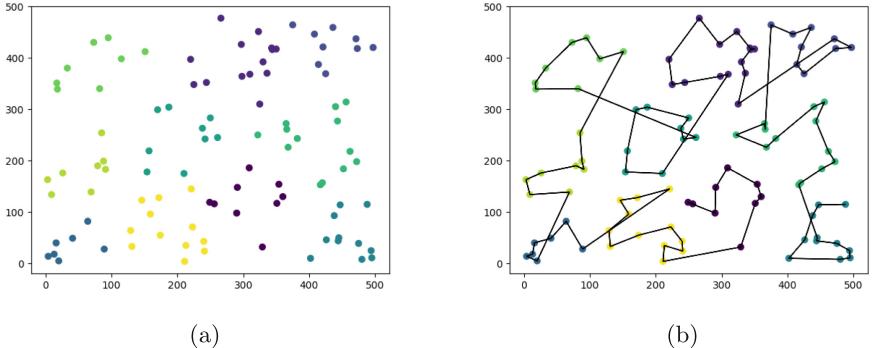


Fig. 2. Path Planning Algorithm: (a) Clusters (b) Computed Path

3.2 Waste Pose Estimation

In order to reliably and accurately pick up the waste, the robot needs to detect and estimate the object's pose. The arm-mounted stereoscopic camera is used with YOLOv8 [5], which is a state-of-art deep learning model that performs object detection with bounding boxes and binary segmentation. After approaching an object, the arm is moved on top of the area to analyze and run the object detection network (Fig. 1b). It is necessary to convert the bounding box prediction from YOLOv8 and the relative angle of the object from PCA (Principal Component Analysis) into a complete pose by converting the output from pixel units within the image frame. This means obtaining the object's position and orientation in 3D space, which is crucial for properly controlling the arm and gripper to collect the item. The binary mask (Fig. 3c) is then processed with PCA to estimate the object orientation (Fig. 3d).



Fig. 3. Bounding Box (b), Segmentation (c), and Orientation (d) estimation of a bottle (a)

The Z , X , and Y coordinates are obtained by applying the central projection camera model [6] as reported in Eq. (1):

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = K^{-1} \begin{bmatrix} xZ \\ yZ \\ Z \end{bmatrix} \quad (1)$$

where

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The Z coordinate is estimated as the average value of the depth points corresponding to the prediction's bounding box pixels because the objects can either float or be sunken in the sand. For this reason, a custom gripper tool (Fig. 4) has been developed, which allows the sand grains or pebbles (up to 2cm) to pass through.

Namely matrix K of Eq. (2) is the matrix projecting 3D points in the camera coordinate frame (X, Y, Z) to 2D pixel coordinates (x, y) using the focal lengths (f_x, f_y) and principal point (c_x, c_y), which are estimated by camera calibration.

At this point, (X, Y, Z) coordinates of the item and its orientation with respect to the gripper are obtained.

Thus, the collection task is performed through classical inverse kinematics, using the robotic arm control implementation available in the MoveIt framework [7]. The gripper first vertically aligns and rotates above the object (Fig. 1c), and after that the gripper goes down to collect it (Fig. 1d). The final step is to place the object in one of the selected bins, as shown in Fig. 1e.

The object pose algorithm was evaluated in the lab by comparing its computed angle against the ground truth angle measured by adding an ArUco marker above the waste. The mean error measured is 0.96° C, with a variance of 0.61. This error is low enough that even if a plastic bottle or a can, which are our main target waste objects, has an estimated angle error on the upper bound, the gripper is wide enough to still pick it. The gripper also is able to detect the presence of an object, using torque sensors, and tries to pick again an object if it fails to do so.



Fig. 4. The custom gripper tool designed and realized for waste picking.

4 On-Field Tests

The platform has been tested in multiple on-field experiments in Milazzo (Italy) and Gozo (Malta). Both sites are characterized by different scenarios and kinds of terrains: pebbles and fine sand. The robot can work in tandem with the whole system and perform multiple missions without human intervention. During testing, waypoint navigation and beach waste collection were successful in both real-world scenarios. The platform was able to properly discern false positive objects reported by the UAV. The video below² clearly shows the essential steps of the grasping process.

5 Conclusion

In this work we presented SandRo, a robot for automatic beach waste cleanup. The presented results prove the effectiveness of the proposed approach, corroborated by the real case studies considered for the BIOBLU project. The main novelty of the proposed approach compared to those available in the literature lies in its sustainability, ensured by mission efficiency and direct pick-and-place. In this manner, the cleanup process has minimal impact on the beach environment, with no manual-human intervention.

As a future development further methods will be investigated for the item pose estimation, relying on point cloud detection and processing, to improve the assessment of the object graspability.

Acknowledgement. G. Muscato, C. Spampinato, G. Sutera and F. Cancelliere acknowledge financial support from PNRR MUR project PE0000013-FAIR. D. C. Guastella acknowledges support by the project PON R&I REACT-EU.

References

1. Ichimura, T., Nakajima, S.: Development of an autonomous beach cleaning robot “hirottaro”. In: 2016 IEEE International Conference on Mechatronics and Automation, pp. 868–872. IEEE (2016)
2. Priya, J.S., Balaji, K., Thangappan, S., Sudhakaran, G.Y.: Beach cleaning bot based on region monitoring. In: 2019 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC), pp. 1–4. IEEE (2019)
3. Cicceri, G., et al.: An intelligent hierarchical cyber-physical system for beach waste management: the bioblu case study. IEEE Access 1 (2023)
4. Quigley, M., et al.: ROS: an open-source robot operating system. In: ICRA Workshop on Open Source Software, Kobe, Japan, vol. 3, no. 3.2, p. 5 (2009)
5. Jocher, G., Chaurasia, A., Qiu, J.: YOLO by Ultralytics (2023). <https://github.com/ultralytics/ultralytics>
6. Corke, P.: Image Formation, pp. 319–357. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-54413-7_11
7. Chitta, S.: Moveit!: an introduction. In: Robot Operating System (ROS) The Complete Reference, vol. 1, pp. 3–27 (2016)

² https://youtu.be/dNyUeop_Ihc.



Domain-Specific Fine-Tuning of Large Language Models for Interactive Robot Programming

Benjamin Alt^{1(✉)}, Urs Keßner^{1,2}, Aleksandar Taranovic², Darko Katic¹, Andreas Hermann¹, Rainer Jäkel¹, and Gerhard Neumann²

¹ ArtiMinds Robotics, 76131 Karlsruhe, Germany
benjamin.alt@artiminds.com

² Karlsruhe Institute of Technology, Autonomous Learning Robots Lab,
76131 Karlsruhe, Germany

Abstract. Industrial robots are applied in a widening range of industries, but robot programming mostly remains a task limited to programming experts. We propose a natural language-based assistant for programming of advanced, industrial robotic applications and investigate strategies for domain-specific fine-tuning of foundation models with limited data and compute.

Keywords: large language models · robot programming · fine-tuning

1 Introduction

Industrial robot programming is cumbersome and costly, especially for advanced applications requiring force sensing and vision. Skill-based approaches simplify programming complex robot tasks by sequencing learned or pre-programmed sub-skills such as grasping or peg-in-hole fitting. However, skill-based programming this still requires expert knowledge, e.g., physical dependencies between acceleration and contact forces, but also technical aspects, such as the applicability of skills in collision-prone or low-tolerance environments. With the introduction of large-scale pretrained foundation models such as LLaMA [6], there is an emerging body of work on fine-tuning large language models (LLMs) for use in various application domains [1, 9]. We propose a language-based programming assistant, which offers interactive dialogues about skills, example use-cases and expected robot behavior. We present three alternative model families, trained by three different domain-specific fine-tuning approaches. The variants are evaluated by comparing their BERTScore [8] performance as well as a user survey with industry experts.

2 Methods

As a testbed, we use the ArtiMinds Robot Programming Suite (RPS), an integrated development environment (IDE) for industrial robots. With it, robots

are programmed using parameterizable robot skills (called “templates”) such as “Grasp”, “Insert”, etc. The resulting robot program is a tree of parameterized templates, which is compiled into executable code. The assistant answers questions of three types:

Providing High-Level Explanations of Templates. Given a question like “What does a ‘Move to State’ template do?”, it should answer something like “It moves the end-effector on a collision-free trajectory to a goal specified in configuration space.” The complexity of answers range from simple, as illustrated here, to highly complex, particularly for force-controlled skills whose behavior depends in part on interactions with the environment.

Providing Examples for the Usage of Templates. To a question like “When should I use a ‘Move to State’ template?”, it should answer “A ‘Move to State’ template can be used to efficiently move the robot arm through environments in which collisions can occur.” Examples can involve concrete application domains, such as painting, gluing or welding for a “Path Loader” template.

Providing Step-by-Step Explanations of Expected Robot Behavior. To a question like “What motions will a robot make when executing a ‘Grasp’ template?”, the assistant should answer “It will execute a collision-free approach motion, open the gripper, move closer to the object, close the gripper, and depart with a collision-free motion.” Step-by-step explanations provide useful information for very high-level templates such as “Peg in Hole”.

We investigate how LLMs can be fine-tuned under the compute and data constraints typical for industrial small- or medium-sized enterprises (SMEs). This implies limitations to a single server-grade graphics processing unit (GPU), e.g., Nvidia A100, with 80 GB VRAM. Traditional fine-tuning requires up to 780 GB given 65B parameters [2]. To reduce model size, we leverage QLoRA adaptation with 4-bit NormalFloat quantization and double quantization [2]. We investigate three data sparse training regimes: Fine-tuning an instruction-following model on domain-specific data, which has been pretrained in different domains; fine-tuning a streaming model for instruction-following on domain-specific data; and fine-tuning a streaming model on a domain-specific streaming data, succeeded by domain-specific instruction-following training. Comparing these training regimes determines whether the use of general-purpose pre-trained models provides useful priors to simplify fine-tuning. Moreover, we investigate the impact of prefix fine-tuning [3] on model quality.

2.1 Datasets

Streaming Dataset: One of the investigated training regimes involves the fine-tuning of a general-purpose streaming model such as LLaMA on domain-specific data. We create a dataset \mathcal{D}_{stream} by parsing the ArtiMinds RPS user manual

into a plain-text representation. Non-informational elements like the cover page, table of contents, index pages, and copyright paragraphs were removed to focus on the technical content.

Instruction-Following Datasets: We constructed a domain-specific dataset to evaluate whether a model could be trained to answer questions using only natural language instructions. The dataset contains questions about the usage of ArtiMinds RPS templates, asking for template descriptions, use cases, and step-by-step descriptions extracted from the RPS documentation. Zhang and Soh [7] found that models may be overly sensitive to minor differences in input prompts, resulting in unintended variability in the generated responses. To address this, we generated 10 question variants for each information topic using ChatGPT 3.5 [4]. The resulting dataset \mathcal{D}_{instr} contains a total of 250 instruction-label pairings.

Prefix Fine-Tuning is a technique in which a specific prefix is added to each prompt to guide the model into the correct domain [3]. To determine how this affects domain-specific fine-tuning, we created an additional version $\mathcal{D}_{instr}^{prefix}$ of the instruction-following dataset, with a custom, domain-specific prefixes, while \mathcal{D}_{instr} applies the default Alpaca prefix [5].

2.2 Model Variants

Alpaca Finetuned on Instruction Task: We finetune three pretrained Alpaca models on our domain-specific instruction-following dataset \mathcal{D}_{instr} . The 7B and 13B models were fully fine-tuned (see Taori et al.), while the 30B model was finetuned using QLoRA, and the resulting LoRA adapters merged back into the original foundation model. Double 4-bit NormalFloat quantization [2] is applied. We perform QLoRA with rank $r = 8$, $\alpha = 16$ and dropout = 0.05 for all 4 self-attention matrices.

LLaMA Finetuned on Instruction Task: Instead of fine-tuning a pretrained instruction-following model, domain adaptation could also be achieved by fine-tuning a streaming model on an instruction-following dataset in the new domain. We use LLaMA models (7B, 13B and 30B) and quantize them to double-quantized 4-bit NormalFloat. We train a LoRA adapter on the instruction-following task \mathcal{D}_{instr} , using the same hyperparameters as in Sect. 2.2.

Merged LLaMA (Streaming and Instruction Task): A third variant of domain adaptation is to perform domain-specific fine-tuning in two stages: First on a streaming task, and then on an instruction-following task, both in the new domain. Using the same pretrained and quantized LLaMA models as in Sect. 2.2, we first train a LoRA adapter on the streaming dataset \mathcal{D}_{stream} . Then, we train an additional LoRA adapter on \mathcal{D}_{instr} .

Prefix Fine-Tuning. To assess to what extent prefix fine-tuning facilitates domain adaptation, we train additional variants of all models mentioned above on $\mathcal{D}_{stream}^{prefix}$ and $\mathcal{D}_{instr}^{prefix}$ respectively.

3 Experiments

3.1 BERTScore Evaluation

To assess the semantic correctness of the trained models’ responses, we compute BERTScores [8] with an evaluation dataset. Nine instructions and reference responses to questions about RPS templates were manually written by domain experts. BERTScore recall, precision and F1 scores were calculated between each response and its reference.

The results are reported in Table 1. Alpaca models without prefix performed best with an F_{BERT} of 0.8. We observe that directly fine-tuning a pretrained instruction-following model yields the best results, and that model size seems uncorrelated to BERTScore performance.

Table 1. BERTScores and survey results for the 10 best-performing models, sorted by F_{BERT} . Factual correctness (C) and domain adherence (D) are binary features, perceived helpfulness (H) is ranked on a five-point Likert scale (1 - low, 5 - high).

| | R_{BERT} | P_{BERT} | F_{BERT} | C | D | H | N^\dagger |
|-----------------------|------------|------------|------------|--------|--------|--------|-------------|
| Alpaca 30B | 0.8156 | 0.7945 | 0.8042 | 0.3830 | 1.0000 | 3.0851 | 5 |
| Alpaca 7B | 0.8044 | 0.7979 | 0.8007 | 0.2041 | 0.8776 | 1.9796 | 5 |
| Alpaca 13B w. p.* | 0.7868 | 0.7929 | 0.7894 | 0.4000 | 0.8600 | 2.5000 | 5 |
| Alpaca 7B w. p. | 0.7873 | 0.7780 | 0.7818 | 0.2881 | 0.5085 | 2.4407 | 6 |
| LLaMA 13B w. p. | 0.8238 | 0.7421 | 0.7798 | 0.1800 | 0.7800 | 2.4600 | 5 |
| Alpaca 30B w. p. | 0.7722 | 0.7189 | 0.7438 | 0.0506 | 0.2785 | 1.3038 | 8 |
| Merged LLaMA 7B w. p. | 0.8017 | 0.6891 | 0.7386 | 0.2317 | 0.9146 | 2.2317 | 9 |
| Merged LLaMA 30B | 0.7952 | 0.6780 | 0.7311 | 0.1224 | 0.8776 | 2.0000 | 5 |
| LLaMA 13B | 0.7910 | 0.6747 | 0.7266 | 0.2143 | 0.6429 | 2.1714 | 7 |
| LLaMA 30B | 0.7631 | 0.6784 | 0.7174 | 0.1277 | 0.7872 | 1.8298 | 5 |

* with prefix † number of survey responses

3.2 User Survey

To obtain feedback about real-life performance, we conduct a survey with domain experts. The evaluation set contains 40 instructions, some of which outside the question types seen in training (e.g., “What is the difference between a ‘Move to Point’ template and a ‘Move to State’ template?”). A total of 33 engineers

participated in the survey, each completing at least one questionnaire containing 10 randomly assigned prompts and model responses. Each prompt-response pair was evaluated on factual correctness (“The answer is precise and factually correct. (yes/no)”), domain adherence (“The answer remains in the domain of ArtiMinds RPS. (yes/no)”) and perceived helpfulness (“Rate the helpfulness of the response on a scale from 1 to 5.”). The results are shown in Table 1. Echoing the BERTScore results, Alpaca models performed best with respect to correctness, domain adherence and helpfulness. It must be noted, however, that all trained models have noticeable shortcomings with respect to correctness and overall helpfulness. The LLaMA-based models in particular struggled with sentence structure, often repeating words and predicting unknown tokens. In contrast, Alpaca-based models generated responses with satisfactory format. Prefix fine-tuning did not have a discernible impact on model performance.

4 Conclusions

We propose a natural-language-based programming assistant which offers human-like, interactive dialog for assistance with complex industrial robot programming. The assistant offers explanations of skills, example use-cases and expected robot behavior. We train three families of LLMs realizing three different approaches for domain-specific fine-tuning and evaluate them with respect to a quantitative semantic similarity metric as well as a user survey. The results indicate that domain-specific fine-tuning of instruction-following models achieves robust domain transfer. However, the results are not (yet) sufficient for practical use. The models often exhibit strong hallucinations (including out of domain responses) and repeat words or phrases or even switch from English to German. This suggests that fine-tuning alone is insufficient to imprint the new knowledge into the model with limited data. Prompting strategies, which eschew fine-tuning in favor of carefully designed prompts, are promising alternatives, and have become technically feasible with the most recent generation of foundation models. Likewise, larger-scale user surveys are required to draw more robust and nuanced conclusions about real-world model performance.

References

- Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., Androutsopoulos, I.: LEGAL-BERT: the muppets straight out of law school. arXiv preprint [arXiv:2010.02559](https://arxiv.org/abs/2010.02559) (2020)
- Dettmers, T., Pagnoni, A., Holtzman, A., Zettlemoyer, L.: QLoRA: efficient fine-tuning of quantized LLMs. In: Advances in Neural Information Processing Systems (2023)
- Li, X.L., Liang, P.: Prefix-tuning: optimizing continuous prompts for generation. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, vol. 1 Long Paper, pp. 4582–4597. Association for Computational Linguistics (2021)

4. OpenAI: ChatGPT (2023). <https://chat.openai.com>
5. Taori, R., et al.: Alpaca: a strong, replicable instruction-following model (2023)
6. Touvron, H., et al.: LLaMA: open and efficient foundation language models. arXiv preprint [arXiv:2302.13971](https://arxiv.org/abs/2302.13971) (2023)
7. Zhang, B., Soh, H.: Large language models as zero-shot human models for human-robot interaction. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2023)
8. Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., Artzi, Y.: BERTScore: evaluating text generation with BERT. arXiv preprint [arXiv:1904.09675](https://arxiv.org/abs/1904.09675) (2020)
9. Zheng, O., Abdel-Aty, M., Wang, D., Wang, C., Ding, S.: TrafficSafetyGPT: tuning a pre-trained large language model to a domain-specific expert in transportation safety. arXiv preprint [arXiv:2307.15311](https://arxiv.org/abs/2307.15311) (2023)



Exploiting Roadside Sensor Data for Vehicle Manoeuvring Assistance

Federico Princiotto^{1(✉)}, Edoardo Bonetto¹, Guido Gavilanes¹,
Matteo Minotti¹, Daniele Brevi¹, Claudio Pastrone¹, Fabrizio Gatti²,
and Ezio Chiocchetti²

¹ LINKS Foundation, Turin, Italy

federico.princiotto@linksfoundation.com

² TIM S.p.A., Service Innovation Division, Turin, Italy

Abstract. This paper introduces a novel approach for tracking and predicting the movements of vehicles at an intersection using only roadside sensor data. The prediction of the vehicles' movements is used to estimate their manoeuvres and to assist other vehicles, that are approaching the intersection, by suggesting suitable manoeuvres to follow. The solution introduced in this paper was tested in a real road intersection with everyday traffic and results showed a limited error that is acceptable for the goal of this system which is to predict the vehicles' manoeuvres.

Keywords: Intelligent Transportation · V2x · sensor · manoeuvre

1 Introduction

Ensuring road safety in urban transportation is a critical aspect. A particular concern is given to the intersections where conflicts between different road users can result in accidents and injuries. Recently, advancements in sensing technologies have led to the development of Intersection Movement Assist (IMA) systems that can support drivers in safely and efficiently navigating intersections. The spread of Vehicle-to-Vehicle (V2V) communication allows connected vehicles to receive this information from the infrastructure. However, the presence of mixed traffic (i.e., not-connected and connected vehicles sharing the roads) is the expected scenario in the future and solutions should take this aspect into account.

In this paper, we propose an implementation of an IMA service that leverages a combination of roadside sensors (i.e., camera and LiDAR) to track and predict the movements of vehicles at intersections. This solution uses only information retrieved from the roadside sensors to predict the vehicle's dynamics, without requiring the vehicles at the intersection to send any information to the system.

This work was supported by the PoDIUM project (101069547) funded by the European Commission.

The next sections are organized as follows: Sect. 2 describes the state-of-the-art related to the IMA service, Sect. 3 introduces the IMA service, the software architecture and its implementation, Sect. 4 provides results of this system in a real scenario and Sect. 5 summarizes the work performed.

2 Related Works

Several research works in the literature already tackled the IMA service and they use different technological approaches for increasing road safety. However, most of the solutions rely especially on information provided by the vehicles. In contrast, it is rare to implement an IMA that is only based on the exploitation of roadside sensor information. For the sake of space, we introduce only the most relevant ones in the following of this section.

Advanced algorithms for path prediction are proposed in [1]. This solution, however, requires that each vehicle is equipped with an On Board Unit (OBU) connected to the CAN-bus of the vehicle for retrieving information about the position, and the dynamics of each vehicle. In a similar way, [2] proposes a solution in which Vehicle-to-Infrastructure communication is used to allow vehicles to send their position to a system that predicts their movement and generates collision alerts.

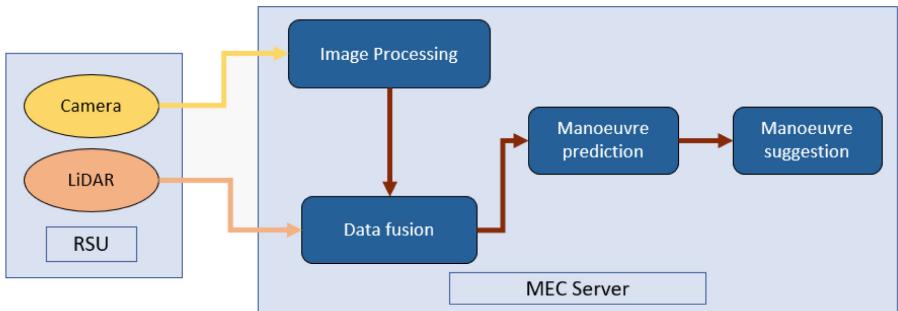


Fig. 1. High-level schema of the software architecture

3 Intersection Movement Assist Service

The IMA service provides assistance to a connected vehicle that approaches an intersection. The connected vehicle receives information from the IMA service about which manoeuvres are safe to perform and which may cause collisions with other vehicles. For example, at a specific moment, it could be safe to turn right, but not to turn left or to go straight. These suggestions depend on the trajectories of other vehicles. In this work, we assume that the other vehicles could be

not connected, and their trajectories are predicted using only the information gathered from the roadside sensors.

The IMA service is deployed using the two following components: the Road-Side Unit (RSU) and a server located at the edge of the mobile network, a.k.a Multi-access Edge Computing (MEC) server [3]. The RSU is installed at the intersection and it forwards using 5G connectivity the data from physical sensors (i.e., camera, and LiDAR) towards the MEC server. The processing of the data is done at the MEC server which guarantees low communication latencies thanks to the edge computing paradigm.

3.1 Software Architecture

The implementation of the IMA service consists of a pipeline of different software modules. These modules are responsible for forwarding to the MEC server the sensor data, which are collected at the RSU, for processing the data to retrieve vehicles' dynamics information, for predicting the future vehicles' movements and for providing the manoeuvre's feedback to the connected vehicles. Figure 1 illustrates a high-level view of the software pipeline. The modules of the pipeline are the following:

- **Image Processing:** it analyses the video stream from the camera to detect and track all the road users;
- **Data fusion:** it merges the information of the Image Processing module with the information provided by the LiDAR's point cloud to estimate the vehicles' positions;
- **Manoeuvre prediction:** it predicts the manoeuvres of vehicles using only the position and dynamics retrieved from the roadside sensor data;
- **Manoeuvre suggestion:** it provides manoeuvre suggestions to the connected vehicles based on the outcomes of the Manoeuvre prediction module.

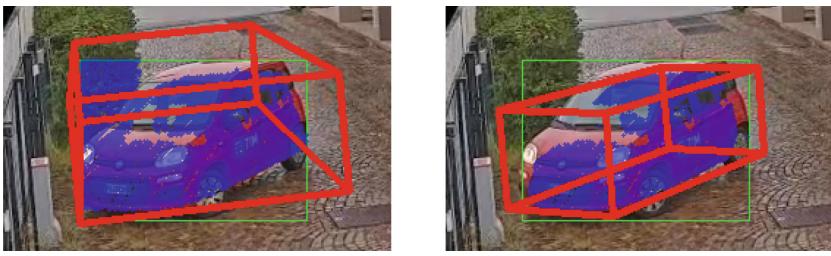


Fig. 2. 3D bounding box examples

The fusion of data from the camera and the LiDAR allows associating a subset of points from the LiDAR's point cloud to each detected road user. This

subset of points allows defining a 3D bounding box around the road user and computing the road user's centroid. This step is fundamental to estimate the position of the road user accurately, and then to predict its trajectory.

There are some factors that can degrade the position and volume estimation of the road user. The most common reason is that background points are wrongly associated with the detected object. This typically happens when the detection box is bigger than the real object. A mitigation strategy consists of applying to the interested portion of the point cloud a *Statistical Outlier Removal* algorithm. This algorithm removes points from the point cloud based on their density [4]. Figure 2a shows the 3D Bounding Box without the mitigation, while Fig. 2b shows the result when the Statistical Outlier Removal is applied.

The knowledge of the vehicle's position over time enables the prediction of its future trajectory. In this work, a *Unscented Kalman filter* implementing the *Constant Turn Rate and Acceleration (CTRA)* model [5] is used to estimate the real state of the vehicle and to predict its movement. The Unscented Kalman filter uses the positions over time of the vehicle to estimate velocity, heading, acceleration and turn rate. This information is then used to predict the future trajectory using the model's state transition function, with the assumption of constant acceleration and turn rate. This assumption limits the effectiveness of the prediction to short-term prediction (i.e., a few seconds) since acceleration and turn rate are not typically constant. This model has indeed already been proven to be a useful tool for short-term prediction [6]. To map the future position to a manoeuvre, the predicted position is checked against a heatmap trained on historical trajectory data. The heatmap associated with each RGB channel gives the probability of a manoeuvre to be performed by a vehicle in that position.

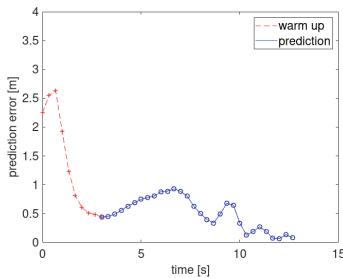


Fig. 3. Prediction error



Fig. 4. Manoeuvre prediction

4 Results

The results provided in this paper have been obtained by analysing a real "T" intersection where an RSU was available. The prediction error of the future position of the not-connected vehicles has been selected to measure the effectiveness

of the IMA service. Although this step does not represent the final stage of the software pipeline, it is crucial to have precise predictions to generate realistic manoeuvre predictions. Moreover, it serves as a reasonable metric for automating the retrieval of numerical values regarding software precision. This error is computed as the difference between the position predicted one second before and the real position that the vehicle has when one second elapsed.

Figure 3 shows the prediction errors over time for a sample vehicle. The first red section corresponds to a warm-up phase that is characterized by a high error because the Unscented Kalman filter needs a few iterations to correctly align itself with the real situation. After this phase, it is possible to see that the prediction error is always within one meter.

The analysis of the selected intersection provides a prediction average error of 0.4 m after the warm-up phase. As we can see in Fig. 4, this result is acceptable for the IMA service: the objective is not to predict with absolute accuracy the future position of the vehicle, but it is to use the future position trajectory of the vehicle and the heatmap data to predict the driver's chosen manoeuvre at the intersection between the available ones.

5 Conclusions

This paper presents a complete IMA service that predicts vehicle movements and provides manoeuvre suggestions to enhance road safety. The IMA service monitors the road segment of interest relying on information from roadside sensor data to detect vehicles and predict their movements. No information needs to be sent from vehicles or from other road users.

Deep learning algorithms are used to detect and track road users from the camera video stream, then a data fusion algorithm is applied to merge this information with the distance information provided from the LiDAR. An Unscented Kalman filter is used to estimate the vehicle's dynamics which are then used to predict the future vehicle's positions. Exploiting these predictions, it is possible to identify the vehicle's predicted trajectory and to use this information to provide suggestions about manoeuvres to connected vehicles. Results were retrieved from analysing a real intersection where an RSU was available. After a short warm-up phase, the Unscented Kalman filter can provide trajectory predictions with a limited error that is suitable to the scope of the IMA service.

References

1. Multi-access edge computing (mec). <https://www.etsi.org/technologies/multi-access-edge-computing>
2. Statistical outlier removal. https://pcl.readthedocs.io/projects/tutorials/en/latest/statistical_outlier.html
3. Fu, Y., Li, C., Luan, T.H., Zhang, Y., Mao, G.: Infrastructure-cooperative algorithm for effective intersection collision avoidance. *Emerg. Technol. Transport. Res. Part C* (2018)

4. Schubert, R., et al.: Comparison and evaluation of advanced motion models for vehicle tracking. In: 2008 11th International Conference on Information Fusion (2008)
5. Tang, B., et al.: Turn prediction at generalized intersections. In: 2015 IEEE Intelligent Vehicles Symposium (IV) (2015)
6. Tsogas, M., et al.: Unscented kalman filter design for curvilinear motion models suitable for automotive safety applications. In: 2005 7th International Conference on Information Fusion (2005)



Combining Local and Global Perception for Autonomous Navigation on Nano-UAVs

Lorenzo Lamberti^{1,✉}, Georg Rutishauser², Francesco Conti¹,
and Luca Benini^{1,2}

¹ DEI, University of Bologna, Bologna, Italy
lorenzo.lamberti@unibo.it

² Integrated Systems Laboratory, ETH Zürich, Zürich, Switzerland

Abstract. A critical challenge in deploying unmanned aerial vehicles (UAVs) for autonomous tasks is their ability to navigate in an unknown environment. This paper introduces a novel vision-depth fusion approach for autonomous navigation on nano-UAVs. We combine the visual-based PULP-Dronet [1] convolutional neural network for semantic information extraction, i.e., serving as the *global perception*, with 8×8 px depth maps for close-proximity maneuvers, i.e., the *local perception*. When tested in-field, our integration strategy highlights the complementary strengths of both visual and depth sensory information. We achieve a 100% success rate over 15 flights in a complex navigation scenario, encompassing straight pathways, static obstacle avoidance, and 90° turns.

1 Introduction

With their sub-10 cm diameter and tens of grams in weight, autonomous nano-sized unmanned aerial vehicles (UAVs) hold the potential to be employed in a wide range of applications, from inspection of hazardous environments [2] to warehouses [3]. Their small size enables access to confined spaces [1,4], and safe operation near humans [3,5], making them suitable for indoor environments.

Autonomous navigation is a pivotal requirement for any UAV exploring an environment while avoiding obstacles. State-of-the-Art (SoA) navigation systems typically combine global and local planning techniques that are computationally heavy, relegating their execution to high-end devices such as GPUs [6]. On the one hand, global planning is in charge of setting high-level destination goals, for example, ensuring the UAV is passing through pre-defined checkpoints. On the other hand, local planning handles close proximity maneuvers, like obstacle avoidance, ensuring progress toward the goals set by the global planner.

The miniaturized form factor of nano-drones imposes significant constraints on their onboard computing devices, ultimately relying on low-power Micro-Controller Units (MCUs) with a power envelope of ~ 100 mW [1]. Consequently, SoA navigation algorithms on autonomous nano-UAV rely on computationally affordable solutions rather than complex planning strategies. The exploration

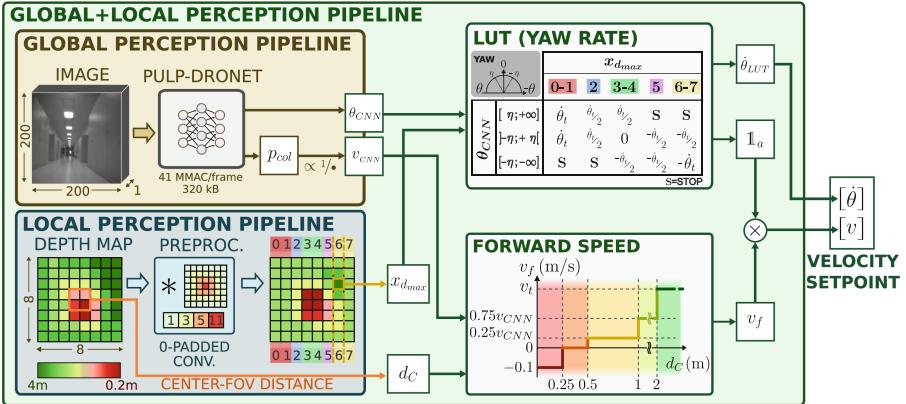


Fig. 1. Our global + local perception pipeline.

policies deployed by [7,8] have to rely on basic state machines and single-beam VL53L1x Time-of-Flight (ToF) sensors for collision avoidance. However, the limited field of view (FoV) (15°) of this ToF sensor challenges the detection of narrow obstacles [9]. Müller *et al.* [4] developed a depth-based navigation system for nano-UAVs, utilizing a novel multizone ToF sensor and implementing a lightweight decision tree based on 8×8 px depth maps. While ToF sensors offer geometrically precise measurements of three-dimensional objects in the environment, they lack semantic information. Visual-based navigation approaches, such as the SoA PULP-Dronet [1] convolutional neural network (CNN), are better equipped to process semantically rich non-volumetric visual cues, such as lanes on a floor, that are invisible to ToF sensors. On the other hand, these methods do not extract any precise geometric information about objects in the FoV, posing challenges for obstacle avoidance. While some research has explored the synergy between vision and depth sensors on nano-drones [5,9], their focus has not been on autonomous navigation applications.

In this work, we take one step towards bringing the global and local planning paradigm aboard nano-UAVs. We present a pipeline for autonomous navigation on nano-drones that integrates both *global* and *local* perception. The *global perception* pipeline utilizes the PULP-Dronet CNN for semantic information extraction, while the *local perception* outputs close-proximity (~ 4 m) obstacle-free navigation area. We fuse the visual-depth information with a lightweight lookup table approach, which maps all possible combinations of outputs from the local and global perception pipelines to drone control commands.

Our fused perception pipeline achieved a 100% success rate on a set of fifteen flights, successfully navigating through straight pathways, executing 90° turns, and avoiding static obstacles. In the same scenario, using only the global perception pipeline led to failures in static obstacle avoidance. Conversely, when relying solely on the local perception pipeline, the system failed to handle 90° turns due to the absence of semantic features in the input data. Our results

highlight the benefit of combining depth and vision sensory inputs to enhance nano-UAV navigation while also revealing the limitations of depending solely on one sensor type in complex navigation scenarios.

2 System Design

Robotic Platform. We use the Bitcraze Crazyflie 2.1, a 27-gram, 10-cm diameter nano-drone. The onboard STM32 MCU processor manages sensor interfacing and low-level flight control tasks. We extend the platform with three printed circuit boards (PCBs). The *AI-deck* serves as the visual navigation engine, featuring a GWT GAP8 9-core RISC-V-based System-on-Chip (SoC) and a grayscale QVGA Himax HM01B0 camera with a 115° FoV. In the GAP8 architecture, a single core handles system operations, including UART communication with the STM32 MCU, while the general-purpose 8-core cluster is used for CNN processing. We use the open-source *Multizone Ranger Deck* [4], featuring a front-looking VL53LC5CX1 matrix ToF sensor. This sensor, connected to the STM32 over the I2C bus, acquires 8×8 px depth maps with a 65° FoV at a frequency of 15Hz and with a range of 0.2–4 m. Last, the *Flow deck v2*, a downward-facing PCB that provides the STM32 MCU with height measurements from the ground and horizontal motion estimation based on optical flow.

Global + Local Perception Pipelines. Our autonomous navigation pipeline for nano-UAVs, depicted in Fig. 1, comprises several components, all executed onboard. The *global perception pipeline* is constituted by the PULP-Dronet [1], a CNN for visual-based navigation on nano-UAVs. We use the pre-trained weights of [1]. The CNN takes 200 × 200px images as input and produces two outputs: a steering angle θ_{CNN} and a collision probability p_{col} . This CNN runs on the 8-core cluster of the GAP8 SoC at 19FPS. As in [1], we post-process the PULP-Dronet outputs, computing the forward speed v_{CNN} inversely proportional to p_{col} , while the steering angle θ_{CNN} is rescaled into a yaw-rate.

The *local perception pipeline* executes entirely on the STM32 MCU. We acquire the 8×8 px depth map from the front-looking matrix ToF sensor. Then, we convolve the depth map with a 2-D Gaussian kernel, requiring only 1600MAC operations, to identify the most free area within the input. The input is zero-padded to maintain consistent input-output dimensions and to favor navigation toward the center of the ToF FoV when no obstacles are present. Finally, we determine the x-coordinate ($x_{d_{max}}$) of the pixel with the longest distance in the 8 × 8 depth map. This coordinate ranges from 0 to 7, denoting a left turn in the 0–2 range, a right turn in the 5–7 range, and no turn otherwise.

To build our final navigation pipeline, we fuse the outputs coming from the global + local perception pipelines. PULP-Dronet’s output $\theta_{CNN} \in [-1, 1]$ is discretized with a threshold $\eta = 0.1$. We use a lookup table (LUT) to generate output pairs $(\mathbf{1}_a, \dot{\theta}_{LUT})$ for every possible combination of $(\theta_{CNN}, x_{d_{max}})$ inputs. Here, $\dot{\theta}_{LUT}$ denotes the output yaw rate, while $\mathbf{1}_a$ is a binary agreement indicator that sets the drone’s forward speed: it assumes a value of 0 when θ_{CNN} and $x_{d_{max}}$ indicate opposite steering directions, as represented by \mathbf{S} in the LUT, or

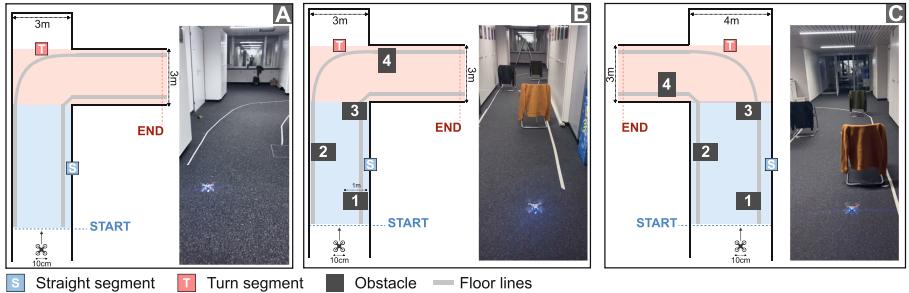


Fig. 2. The testing setups for scenario 1 (A), scenario 2 (B), and scenario 3 (C).

it is 1 otherwise. The output yaw-rate can be either zero, indicate left and right turns at the maximum target yaw rate ($\dot{\theta}_t$ for left, $-\dot{\theta}_t$ for right), or indicate left and right turns at half of the maximum target yaw rate ($\dot{\theta}_t/2$ and $-\dot{\theta}_t/2$, respectively). The forward speed of the drone v_f , which is gated by $\mathbb{1}_a$, is adjusted in step increments as depicted in Fig. 1, and depends on the average distance (d_C) measured by the four central pixels of the ToF sensor. To test the global perception pipeline in isolation, we use the CNN outputs as in [1], and for isolated testing of the local perception pipeline, we force the CNN outputs to $p_{col} = 0$ and $\theta_{CNN} = 0$.

3 Results

We assess the navigation capabilities in an office corridor with a width of 3–4m (Fig. 2). We set up three testing scenarios, each comprising a *straight* pathway followed by a 90-degree *turn*, highlighted in Fig. 2 with blue and red colors, respectively. Scenarios 1 and 2 entail a right turn (A and B), with the former obstacle-free and the latter containing four 1m-wide obstacles. Scenario 3 (C) mirrors scenario 2, featuring a left turn and four 1m-wide obstacles. We draw white lines on the floor, resembling the car lanes of the PULP-Dronet training dataset [1]. We assess the performance of the three perception pipelines discussed in Sect. 2 by conducting five flight tests per scenario, totaling 45 flights. We set the drone’s target height to 0.5 m, the target forward speed (v_t) to 1.5 m/s, and $\dot{\theta}_t$ to 60°/s. A test is considered successful when the drone successfully flies through the corridor from **START** to **END** without collisions.

Table 1 reports the success rate of the drone for each section. Examples of the flight tests are available in the supplementary video¹. In obstacle-free scenario 1, the global perception pipeline succeeds 100% of the time, successfully recognizing the visual clues on the floor. However, this perception pipeline never succeeds in scenarios 2 and 3, colliding with the obstacles in every trial. These results confirm that PULP-Dronet v2 struggles to tackle static obstacle avoidance, as mentioned

¹ https://youtu.be/J703fo_zIKQ.

in [1]. Conversely, the local perception pipeline reliably avoids static obstacles with a 100% success rate in straight segments. Nonetheless, it consistently fails to execute turns in all scenarios, ultimately getting stuck at the end of the corridor, as the ToF can not recognize semantic elements of the environment, e.g., the floor's line markings.

Finally, we test our fused global + local perception pipeline, which scores a 100% success rate across both the straight and turn segments of the corridor, both when navigating in an obstacle-free corridor (scenario 1) and when tackling obstacle avoidance (scenario 2 and 3). In conclusion, our fused global + local perception pipeline captures the benefits of both the depth-based and vision-based sensory inputs: the ToF ranging measurements allow for reliable short-range static obstacle avoidance, while the PULP-Dronet understands semantic cues from its visual input, allowing the drone to follow the shape of the corridor.

Table 1. The success rate of the three perception pipelines on three scenarios.

| Perception Section | Global | | Local | | Global+Local | |
|-----------------------|----------|------|----------|------|--------------|------|
| | Straight | Turn | Straight | Turn | Straight | Turn |
| Scenario 1 | 100% | 100% | 100% | 0% | 100% | 100% |
| Scenario 2 | 0% | N/A | 100% | 0% | 100% | 100% |
| Scenario 3 | 0% | N/A | 100% | 0% | 100% | 100% |

4 Conclusion

We presented a pipeline for autonomous navigation on nano-UAVs that combines global and local perception. The global perception exploits the visual-based PULP-Dronet CNN to extract visual cues from the input images, while the local perception relies on an 8×8 px Time-of-Flight (ToF) sensor capturing reliable close-proximity occupancy maps. Our fused perception pipeline achieved a 100% success rate in fifteen flights, navigating through straight pathways, avoiding static obstacles, and executing a 90° turn.

Acknowledgments. We thank D. Palossi and D. Christodoulou for their contribution to this work.

References

1. Niculescu, V., Lamberti, L., Conti, F., Benini, L., Palossi, D.: Improving autonomous nano-drones performance via automated end-to-end optimization and deployment of dnns. *IEEE J. Emerg. Selected Topics Circ. Syst.* **11**(4), 548–562 (2021)
2. Anderson, M.J., et al.: A bio-hybrid odor-guided autonomous palm-sized air vehicle. *Bioinsp. Biomimet.* **16**(2), 026002 (2020)

3. Awasthi, S., et al.: UAVs for industries and supply chain management. ArXiv [arxiv:2212.03346](https://arxiv.org/abs/2212.03346) (2022)
4. Müller, H., et al.: Robust and efficient depth-based obstacle avoidance for autonomous miniaturized UAVs. *IEEE Trans. Rob.* **39**(6), 4935–4951 (2023)
5. Crupi, L., et al.: Sim-to-real vision-depth fusion CNNs for robust pose estimation aboard autonomous nano-quadcopters. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7711–7717 (2023)
6. Palossi, D., et al.: An energy-efficient parallel algorithm for real-time near-optimal uav path planning. In: Proceedings of the ACM International Conference on Computing Frontiers, ser. CF 2016, pp. 392–397. Association for Computing Machinery, New York (2016)
7. McGuire, K., et al.: A comparative study of bug algorithms for robot navigation. *Robot. Auton. Syst.* **121**, 103261 (2019)
8. Lamberti, L.: Bio-inspired autonomous exploration policies with CNN-based object detection on nano-drones. In: Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1–6 (2023)
9. Pourjabar, M., et al.: Multi-sensory anti-collision design for autonomous nano-swarm exploration. In: 2023 30th IEEE International Conference on Electronics, Circuits and Systems (ICECS), pp. 1–5 (2023)



GAP9Shield: A 150GOPS AI-Capable Ultra-low Power Module for Vision and Ranging Applications on Nano-drones

Hanna Müller¹, Victor Kartsch^{1(✉)}, and Luca Benini^{1,2}

¹ Integrated Systems Laboratory, ETH Zürich, Zürich, Switzerland

{hanmuell,victor.kartsch,lbenini}@iis.ee.ethz.ch

² DEI, University of Bologna, Bologna, Italy

Abstract. The evolution of AI and digital signal processing technologies, combined with affordable energy-efficient processors, has propelled the development of hardware and software for drone applications. Nano-drones, which fit into the palm of the hand, are suitable for indoor environments and safe for human interaction; however, they often fail to deliver the required performance for complex tasks due to the lack of hardware providing sufficient sensing and computing performance. Addressing this gap, we present the GAP9Shield, a nano-drone-compatible module powered by the GAP9, a 150GOPS-capable SoC. The system also includes a 5 MP camera for high-definition imaging, a Wi-Fi-BLE module, and a 5-directional laser-based ranging subsystem, enabling obstacle avoidance capabilities. Compared with similar state-of-the-art systems, GAP9Shield provides a 20% higher sample rate (RGB images) while offering 15% weight reduction. In this paper, we also highlight the energy efficiency and processing power capabilities of GAP9 for object detection using deep learning (YOLO), localization using a particle filter, and mapping, which can run within a power envelope of below 100 mW and at low latency (as 17 ms for object detection), highlighting the transformative potential of GAP9 for the new generation of nano-drone applications.

Keywords: nano-drone · nano-UAV · RISC-V · TinyML

1 Introduction

The evolution of drone technology is marked by swift progress and expansion across diverse sectors, including military, industrial, and commercial, providing solutions for navigation, secure communication, reconnaissance, package delivery, smart healthcare, and precision farming. Enhanced computational capabilities are enabling the development of advanced functionalities in nano-drones, now capable of autonomously exploring and interacting with the environment while

being safe humans nearby and capable of operating indoors, even in confined spaces [1–3].

Among the many available nano-drones systems, the 27 g, 10 cm Crazyflie(CF)¹ quadcopter has gained widespread adoption in the research community due to its open-source, modular architecture, which supports the quick development and testing of drone applications. Regarding the expansion modules, the Multi-ranger and AI decks are the most critical for deploying cutting-edge Obstacle Avoidance (OA) and image recognition missions. The Multi-Ranger shield is a 5-directional (5D) VL53L1-based lightweight low-power omnidirectional range device, enabling applications to perform computationally inexpensive bio-inspired obstacle avoidance tasks efficiently [4]. The AI-deck is an AI vision-based engine featuring a low-power QVGA grayscale camera (Himax HM01B0) and GAP8, an 8-core microcontroller unit (MCU) for efficient onboard processing. While the AI deck exhibits sufficient processing power and sensing for missions in controlled environments, its performance is not sufficient in more complex scenarios and tasks due to its monochrome, low-resolution camera, and the limitations of the GAP8 in floating-point processing capabilities, memory size, and operational frequency. Integrating efficient processors and cameras is essential to enhance and fully exploit the capabilities of tiny drones. Similarly, considering the fundamental requirement for OA during image acquisition and processing, combining range and high-resolution vision acquisition and processing into a single system is desirable, which could also optimize weight and size.

GAP9, the latest Parallel-Ultra-Low-Power (PULP) processor from Green-Waves Technologies², represents a leap forward in addressing the processing power breach for embedded systems. Furthermore, in addition to an extensive array of standard peripherals, it features a MIPI CSI2 interface, allowing compatibility with high-definition cameras. Such features set the stage for developing a new generation of powerful, robust and reliable AI-driven nano-drones.

In response to the evolving demands and potential unlocked by AI, hardware, and imaging sensing innovations, this paper introduces the GAP9Shield. The GAP9Shield is a GAP9-based CF pluggable shield incorporating a high-definition OV5647 camera and the Wi-Fi-BLE-capable NINA W102 module, enabling efficient onboard vision-based processing. The device can also transmit 3-channel RGB QVGA images at a frame rate 20% higher (7 FPS) than the conventional single-channel, monochrome AI-deck while also allowing streaming VGA images at 4 FPS. The shield also integrates a 5D VL53L1-based ranging subsystem for omnidirectional distance measurements, achieving a 20% weight reduction compared to the combined AI-deck+Ranger Deck setup. This work also highlights the energy efficiency and computational capabilities of GAP9 in object detection with deep learning (YOLO), localization, and mapping that can be run on GAP9 with a power envelope below 100 mW and a latency ranging from 18–250 ms. As a final contribution, we plan to provide the complete plat-

¹ <https://www.bitcraze.io/products/crazyflie-2-1/>.

² <https://greenwaves-technologies.com/gap9-processor/>.

form schematics and PCB diagrams as open-source resources, aiming to foster the development of next-generation nano-drone applications.

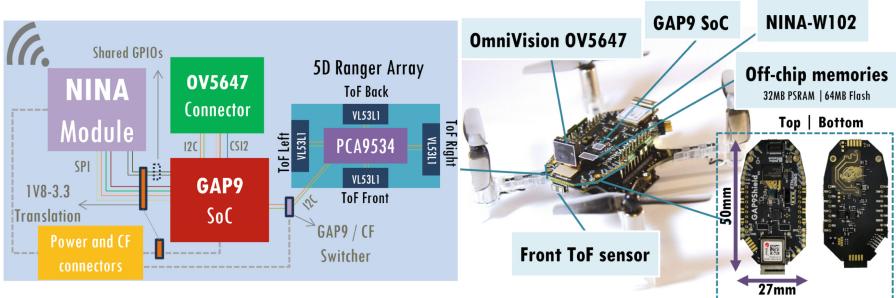


Fig. 1. System overview (and block diagram) of the GAP9Shield on a CF drone.

2 System Description

Figure 1 presents a block diagram of the complete system and a summary of the features of each element within the shield. All components are integrated into a 6-layer 50×27 PCB. The following provides additional details of each component.

2.1 GAP9 SoC

The GAP9 System-on-Chip (SoC), by GreenWaves Technologies, features a 9-core RISC-V compute cluster with AI acceleration (NE16), transprecision floating-point support, which allow for $330 \mu\text{W}/\text{GOP}$ of energy efficiency and 15.6 GOPs for DSP and 32.2 GMACs for ML task of performance. Additionally, it features a single-core RISC-V controller to offload tasks to the cluster and manage peripherals. GAP9 supports various interfaces, including Serial Audio Interfaces and a CSI-2 camera interface, making it versatile for connecting to new devices such as current HD cameras. GAP9 features dynamic frequency (up to 370 MHz), voltage scaling, automatic clock gating, and a sleep power as low as $45 \mu\text{W}$, which allows application-specific fine-tuning of resources for optimized energy consumption. The SoC features 1.6 MB of L2 RAM and 2 MB of non-volatile in-package memories. On GAP9Shield, GAP9 is coupled with an APS256XXN-OBR PSRAM memory from (AP Memory) for an additional 256 Mbit of volatile memory and the MX25UM51245GXDI00 (Macronix) for a supplementary 512 Mbit of nonvolatile memory.

2.2 Camera

GAP9Shield is coupled with the OV5647, a high-performance, low-voltage 5-megapixel CMOS image sensor, allowing for automatic image control while providing several image formats (QSXGA to QVGA) that can be sampled at various frame rates (15 fps to 120 fps for QSXGA and QVGA, respectively). The camera supports 8/10bit raw RGB output formats that can be transferred to the host device through interfaces such as SCCB, DVP, or MIPI/CSI (single or dual lines). The OV564 is coupled to GAP9 through a single CSI2 line on the GAP9Shield.

2.3 Wi-Fi Module

The NINA-W102 is a compact ($10\text{ mm} \times 14\text{ mm}$), stand-alone multi-radio module integrating an ESP32 MCU with 2.4 GHz Wi-Fi, Bluetooth wireless capabilities, and a 3dBi-gain PIFA antenna. The ESP32 MCU is a dual-core SoC, and in our implementation, Wi-Fi and SPI tasks (to communicate with GAP9) are split among the cores to maximize data throughput.

2.4 5D Ranging Array

GAP9Shield also integrates an array of 5 range sensors, pointing to the right, left, front, back, and up directions. Ranging is achieved with the VL53L1 Time-of-Flight (ToF) laser-ranging sensor (ST microelectronics), allowing fast and accurate distance ranging over 400 cm with full field-of-view (FoV) at a maximum sample rate of up to 60 Hz. On the GAP9Shield, the array of VL53L1s can be read directly through GAP9 or the STM32 MCU on the CF.

3 Results

3.1 System Characterization

Physical Properties: We measured our device’s weight and volume at approximately 6 g and 4050 mm^3 ($3 \times 50 \times 27\text{ mm}^3$), respectively. Compared to the existing CF setup with similar functions (AI and multi-ranger decks), our device is 15% lighter and 30% smaller in volume³, which helps improving flight performance and battery life of the CF. **5d Ranging Array:** We characterized the VL53L1 sensors on the shield, employing a setup where the sensors are multiplexed over the I2C connection. The highest range frequency achieved was approximately 40SPS, with the maximum power consumption recorded at 150 mW. **OV5647:** We characterize QVGA and VGA resolutions for their suitability in applications with limited resources and greater image detail in more complex applications, respectively. Our tests show frame rates of 15 fps for QVGA and 45 fps for VGA. The lower-than-expected frame rate for QVGA stems from a driver issue, whereas VGA’s performance matches the expected

³ The AI and ranger decks together weigh about 7 g and occupy 6480 mm^3 .

outcomes based on the manufacturer’s specs for a single-channel CSI2. Power usage was recorded at 80 mW for QVGA at 15 fps and 120 mW for VGA at 45 fps. **Wi-Fi Throughput and Image Streaming:** We evaluated the NINA module’s performance by sending TCP and UDP packets to a remote PC acting as a server on an existing network. The module achieved speeds of 8 Mbit/s for TCP and 12 Mbit/s for UDP, with power consumption around 250 mW for TCP and 260 mW for UDP. Image streaming via TCP yielded a 3-channel RGB JPEG QVGA frame rate approximately 20% higher than the AI-deck (around 7 FPS) and 4 FPS for the VGA resolution.

3.2 GAP9 Performance for Nano-drone Applications

The GAP9Shield (in prototype version) has already successfully been used for many applications, such as object detection with YOLO variants, localization, Mapping (SLAM). **Object detection:** In [5], a slightly modified YOLOv5 architecture was deployed, reaching an execution time of 38 ms while consuming 2.5 mJ. In [6], different YOLO variants were compared between three edge vision hardware platforms, one of them GAP9. The inference of their smallest network with an acceptable accuracy takes 17 ms and consumes 1.59 mJ. **Monte Carlo Localization (MCL):** In [5], MCL4 is used alongside YOLO-based object detection on GAP9 for localization with semantic cues onboard nano-drones. MCL can run at sensor rate (15 Hz for the ToF and odometry update and 5 Hz for the camera update) while using an average of 23 mW during MCL execution. **SLAM:** In [7], the authors deploy NanoSLAM on the GAP9Shield within a power envelope of only 87.9 mW in less than 250 ms.

4 Conclusions

This paper presented the GAP9Shield, a compact, lightweight, sensor-rich, versatile edge-AI platform for nano-drones. It can be used for dataset collection with its Wi-Fi streamer at up to 7fps at below 260 mW as well as for onboard deployment of State of the Art (SoA) algorithms, as YOLO, MCL and SLAM at below 100 mW. GAP9Shield, hence, unlocks new opportunities in sectors like greenhouse farming, inspection, and accurate image recognition and navigation in cluttered conditions.

References

1. Palossi, D., Gomez, A., Draskovic, S., Keller, K., Benini, L., Thiele, L.: Self-sustainability in nano unmanned aerial vehicles: a blimp case study. In: Proceedings of the Computing Frontiers Conference (2017)
2. Birk, A., Wiggerich, B., Bülow, H., Pfingsthorn, M., Schwertfeger, S.: Safety, security, and rescue missions with an unmanned aerial vehicle (UAV). J. Intell. Robot. Syst. **64**(1), 57–76 (2011)

3. Palossi, D., et al.: Fully onboard AI-powered human-drone pose estimation on ultra-low power autonomous flying nano-UAVs. *IEEE Internet Things J.* 1 (2021)
4. Lamberti, L., Bompani, L., Kartsch, V.J., Rusci, M., Palossi, D., Benini, L.: Bio-inspired autonomous exploration policies with CNN-based object detection on nano-drones. In: Design, Automation & Test in Europe Conference & Exhibition (DATE), 1–6. IEEE (2023)
5. Zimmerman, N., Müller, H., Magno, M., Benini, L.: Fully onboard low-power localization with semantic sensor fusion on a nano-UAV using floor plans. In: 2024 IEEE International Conference on Robotics and Automation (ICRA), arXiv preprint [arXiv:2310.12536](https://arxiv.org/abs/2310.12536) (2024)
6. Moosmann, J., et al.: Ultra-efficient on-device object detection on AI-integrated smart glasses with tinyissimoyolo. arXiv preprint [arXiv:2311.01057](https://arxiv.org/abs/2311.01057) (2023)
7. Niculescu, V., Polonelli, T., Magno, M., Benini, L.: NanoSLAM: enabling fully onboard slam for tiny robots. *IEEE Internet Things J.* (2023)



Fusing Multi-sensor Input with State Information on TinyML Brains for Autonomous Nano-drones

Luca Crupi^{1(✉)}, Elia Cereda¹, and Daniele Palossi^{1,2}

¹ Dalle Molle Institute for Artificial Intelligence, USI-SUPSI, Lugano, Switzerland
{luca.crupi,elia.cereda,daniele.palossi}@idsia.ch

² Integrated Systems Laboratory, ETH Zürich, Zürich, Switzerland

Abstract. Autonomous nano-drones (~ 10 cm in diameter), thanks to their ultra-low power TinyML-based brains, are capable of coping with real-world environments. However, due to their simplified sensors and compute units, they are still far from the sense-and-act capabilities shown in their bigger counterparts. This system paper presents a novel deep learning-based pipeline that fuses multi-sensorial input (i.e., low-resolution images and 8×8 depth map) with the robot’s state information to tackle a human pose estimation task. Thanks to our design, the proposed system – trained in simulation and tested on a real-world dataset – improves a state-unaware State-of-the-Art baseline by increasing the R^2 regression metric up to 0.10 on the distance’s prediction.

Keywords: Sensor fusion · TinyML · Autonomous nano-drones

1 Introduction

Miniaturized autonomous nano-drones, from sub-30 g blimps [6] to palm-sized quadrotors [2], are becoming increasingly popular in academia and industry thanks to their broad applicability [8]. They can assist human operators in rescue missions, inspect narrow places, and monitor indoor facilities. Additionally, thanks to their small footprint and weight, they can safely operate close to people, being harmless. However, the vast versatility of autonomous nano-drones comes at the price of simple and cheap electronics, leading to ultra-constrained computational resources and simplistic sensors, e.g., low-resolution cameras.

Recent works [2, 4, 9] have shown how, despite their limitations, nano-drones can achieve a high level of intelligence using tiny deep learning (DL) models fed with multiple sensors and relying only on sub-100 mW System-on-Chips (SoCs). The vast majority of these works address DL-based perception tasks focusing on understanding the surrounding environment, such as estimating the relative pose of an object or a human [2, 4] or avoiding obstacles [5, 9]. We define these as *allocentric* tasks, in which the prediction refers to an external object/subject, in contrast to *egocentric* tasks, in which the subject is the robot itself.

In DL models for egocentric tasks, such as state estimation and visual odometry [1, 3, 7], exploiting the robot’s state as an auxiliary input is common. However, whether it can also benefit allocentric tasks is still an open research question, which we aim to explore by introducing state information into a convolutional neural network (CNN) for the allocentric task of human pose estimation. We start from a State-of-the-Art (SoA) baseline CNN, running on a Greenwaves Technologies (GWT) GAP8 SoC and fed with 160×96 pixel images and an 8×8 depth map, both acquired aboard a nano-drone and used to estimate the distance (x) and the lateral displacement (y) between the drone and a human in front of it. Then, we extend this model by feeding part of the nano-drone’s state as an additional input of the CNN, and we present an ablation study exploring *i*) how different state information (i.e., only pitch or roll, and pitch+roll) and *ii*) how different fusion techniques affect the final CNN’s regression performance.

Our exploration leads to four alternative fusion approaches, resulting in just as many CNNs we deploy and characterize on the GAP8 SoC aboard our target nano-drone. Despite the CNNs training being done only in simulation, our results from a real-world test set (more than 3.5k image-depth pairs) highlight how the introduction of either of the state angles improves the regression performance of our neural network, measured with an R^2 increase of +0.10 on x , and +0.02 on y , at an increase of the computational cost, measured with the number of multiply-and-accumulate (MAC) operations, of just 0.11% if compared with the state-unaware SoA baseline model [2].

2 System Description

Our work starts from the state-unaware vision+depth CNN presented in [2] for the human pose estimation task. We share the same target nano-drone, i.e., the Bitcraze Crazyflie 2.1, and the same computational and sensorial resources, such as the STM32 microcontroller, the GWT GAP8 octa-core SoC, the Himax HM01B0 monocular grayscale QVGA camera, and the ST VL53LC5CX Time-of-Flight (ToF) multi-zone ranging sensor. Like in [2], we process grayscale 160×96 images and 8×8 px depth maps, while we predict the (x, y) position coordinates of the human subject, as shown in Fig. 1. The depth information is fused with a *mid fusion* approach (i.e., the best-performing in [2]) that introduces the depth map as one of the feature maps in input to the CNN’s 6th convolutional layer.

Our state-aware CNNs use the *roll* (φ) and *pitch* (θ) components of the drone attitude, with the two alternative representations shown in Fig. 1-A: either a two-dimensional *state map* in which the elements repeat in a Bayer-like pattern, or a two-element (φ, θ) *state vector*. The former representation is more suitable for fusion approaches in which the state should be spread across multiple neural connections, such as in Fig. 1-B1 called *input fusion*, and in Fig. 1-B2 called *mid fusion*. The sizes of the two-state maps vary to match each convolutional layer’s input dimensions, i.e., 160×96 and 10×6 . Employing a two-element state vector, instead, is a cheaper solution – i.e., less memory and MAC operations – suitable for introducing the state directly into the last CNN’s fully connected

layer, depicted in Fig. 1-B3 and called *late fusion (direct)*, or after processing it with a 2-layer multilayer perceptron (MLP), Fig. 1-B4. As in [2], for all variants, input dropout is employed at training time to use all three inputs equally, i.e., camera, depth, and state.

Finally, we train and validate the four models with the same 700k-sample dataset (from the Webots simulator) and procedures as in [2]. We apply domain randomization, varying the environment (walls, floors, and decoy objects) and the human subject (chosen among 27 3D models). Our procedure randomly selects the drone’s attitude for each sample, such that our state variables (roll and pitch) are uniformly distributed over the $[-20^\circ, 20^\circ]$ range. All CNNs are trained with stochastic gradient descent at a learning rate of 0.001 for 100 epochs, selecting the model with the lowest validation loss. We evaluate the models’ performance on a challenging 3500-sample real-world test set, collected in a room equipped with a motion capture system, where the drone flights stress roll and pitch in the same $[-20^\circ, 20^\circ]$ range.

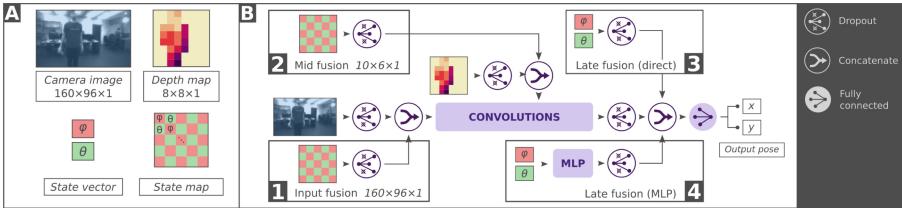


Fig. 1. A) Our CNN inputs. B) CNN architecture exploration, based on the SoA vision+depth backbone from [2]. Our proposed state-aware models either with 1) input fusion, 2) mid fusion, 3) late fusion (direct), and 4) late fusion with MLP.

3 Experimental Results

In this section, we present our ablation study on the proposed fusion techniques, i.e., input, mid, and two late fusions (MLP and direct), combining them with different states (i.e., pitch, roll, pitch+roll) and the effect of a dropout layer applied on the state during training. Each combination results in a different model (24 in total) compared against the state-unaware SoA baseline proposed in [2]. Each model is trained 5 times, resulting in 125 distinctive instances, presented in Fig. 2 and evaluated with the R^2 score on the x and y output variables. The R^2 score measures regression performance in a normalized range $[-\infty, 1]$. A perfect predictor scores 1, while a dummy predictor that always predicts the average of the test set would score 0.

The results in Fig. 2 highlight that introducing any state is consistently beneficial (with the only exception of input fusion without dropout and pitch+roll states), regardless of dropout and fusion technique, with the sum of the R^2

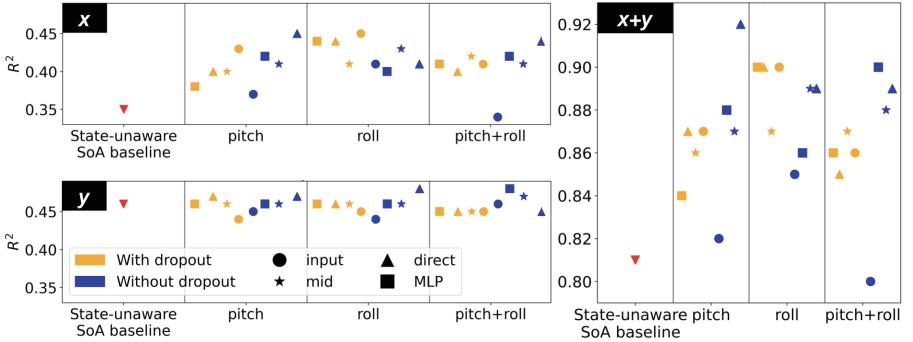


Fig. 2. R^2 comparison of our fusion techniques (i.e., input, mid, direct, and MLP) with and without dropout and with different states (i.e., pitch, roll, and pitch+roll) vs. the state-unaware SoA baseline [2]. Each marker is the average of 5 different training.

improvements on x and y up to +0.11 w.r.t. the state-unaware SoA baseline model. Applying the dropout on the state input does not affect the R^2 standard deviation, contrary to the results from [2]. In this regard, input dropout in [2] was used on the vision and depth sensorial inputs to force the network to consider both information streams. Instead, in this case, the model does not overfit on the state even without dropout because it would be insufficient to estimate the person’s position without a camera or ToF sensor. Out of all CNNs, the best-performing model uses only the pitch state, without dropout and with late fusion (direct). Its R^2 increases by +0.10 on the x and by +0.01 on the y output w.r.t. the state-unaware SoA baseline, with a negligible increase in the computational cost, i.e., 4 MAC. The configuration input fusion on pitch+roll without dropout is the only one where the performance slightly decreases on x (-0.01 R^2).

Table 1. R^2 peak performance, memory occupancy, and computational cost for the state-unaware SoA baseline and the four different state fusion techniques. We report the maximum $R2$ for each variable, x and y , and for each fusion method.

| | SoA baseline | Input | Mid | Direct | MLP |
|-------------------------|--------------|-----------|-----------|-----------|-----------|
| $R^2(\text{peak } x/y)$ | 0.35/0.46 | 0.45/0.45 | 0.43/0.47 | 0.45/0.48 | 0.44/0.48 |
| Memory [kB] | 304.4 | 305.2 | 305.5 | 304.4 | 304.8 |
| MAC/frame [M] | 14.7 | 17.8 | 14.8 | 14.7 | 14.7 |

Table 1 further reports the R^2 peak performance for each output (x and y from different models), the memory occupancy, and the computational cost for each state fusion technique and for the state-unaware SoA baseline. Among the four architecture variants, input fusion and direct late fusion achieve the

highest regression performance improvements (+0.10 and +0.11, respectively) w.r.t the SoA baseline. However, input fusion is the most expensive in computational and second most expensive in memory cost, respectively +3.1IMMAC (+21%) and +0.8kB (+2.6%) w.r.t. the baseline. The other three variants (i.e., mid, MLP, and direct fusion) increase the MAC workload by less than 0.11%. However, the two late-fusion variants require less memory compared to mid-fusion (+1.3% increase vs. +3.6%, respectively). Thus, considering the trade-off between regression performance and memory and computational costs suggests that the direct late-fusion model is the most convenient. These results highlight the benefit provided by the fusion of the state in the pose estimation task with minimal or no computational cost addition (between 0 and 0.11% more MAC for the mid, direct, and MLP approaches) and with a limited increase of the memory usage, i.e., between 0 and 0.36% w.r.t. the SoA baseline.

4 Conclusion

This work introduces a novel study on how to fuse state information into a lightweight multi-sensorial (camera and depth map) CNN for the allocentric task of human pose estimation aboard an autonomous nano-drone. Our analysis highlights the impact on the regression performance of different states (roll, pitch, roll+pitch) and fusion techniques (input, mid-fusion, and late fusions). Our key findings consistently show the benefit of the state fusion: mean R^2 improvement of 0.06 on the x variable w.r.t. the SoA baseline model. Our best model, the late fusion approach, increases the R^2 up to 0.10 and 0.01 on x and y with a negligible overhead in memory or computation.

References

- Clark, R., et al.: VINet: visual-inertial odometry as a sequence-to-sequence learning problem. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, no. 1 (2017)
- Crupi, L., et al.: Sim-to-real vision-depth fusion CNNs for robust pose estimation aboard autonomous nano-quadcopters. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7711–7717 (2023)
- Han, L., et al.: DeepVIO: self-supervised deep learning of monocular visual inertial odometry using 3D geometric constraints. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 6906–6913 (2019)
- Kaufmann, E., et al.: Beauty and the beast: optimal methods meet learning for drone racing. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 690–696. IEEE (2019)
- Liu, C., et al.: Nano quadcopter obstacle avoidance with a lightweight monocular depth network. IFAC-PapersOnLine **56**(2), 9312–9317 (2023). 22nd IFAC World Congress
- Palossi, D., et al.: Self-sustainability in nano unmanned aerial vehicles: a blimp case study. In: Proceedings of the Computing Frontiers Conference, pp. 79–88 (2017)

7. Palossi, D., et al.: Ultra low-power visual odometry for nano-scale unmanned aerial vehicles. In: Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1647–1650 (2017)
8. Shakhatreh, H., et al.: Unmanned Aerial Vehicles (UAVs): a survey on civil applications and key research challenges. IEEE Access: Pract. Innov. Open Solutions **7**, 48572–48634 (2019)
9. Zhang, N., et al.: End-to-end nano-drone obstacle avoidance for indoor exploration. Drones **8**(2), 33 (2024)



Optimized Deployment of Deep Neural Networks for Visual Pose Estimation on Nano-drones

Matteo Riss¹(✉), Francesco Daghero², Beatrice Alessandra Motetti¹,
Daniele Jahier Pagliari¹, Enrico Macii², Massimo Poncino¹,
and Alessio Burrello²

¹ Department of Control and Computer Engineering, Politecnico di Torino,
10129 Turin, Italy

{matteo.risso, beatrice.motetti, daniele.pagliari,
massimo.poncino}@polito.it

² Interuniversity Department of Regional and Urban Studies and Planning,
Politecnico di Torino, 10129 Turin, Italy

{francesco.daghero, enrico.macii, alessio.burrello}@polito.it

Abstract. Miniaturized autonomous unmanned aerial vehicles (UAVs) are gaining popularity due to their small size, enabling new tasks such as indoor navigation or people monitoring. Nonetheless, their size and simple electronics pose severe challenges in implementing advanced onboard intelligence. This work proposes a new automatic optimization pipeline for visual pose estimation tasks using Deep Neural Networks (DNNs). The pipeline leverages two different Neural Architecture Search (NAS) algorithms to pursue a vast complexity-driven exploration in the DNNs' architectural space. The obtained networks are then deployed on an off-the-shelf nano-drone equipped with a parallel ultra-low power System-on-Chip leveraging a set of novel software kernels for the efficient fused execution of critical DNN layer sequences. Our results improve the state-of-the-art reducing inference latency by up to $3.22 \times$ at iso-error.

Keywords: Nano-drones · TinyML · NAS · CNN · Fused Layers

1 Introduction and Related Works

Nano-sized unmanned aerial vehicles (UAVs), commonly named “nano-drones”, are increasingly used for navigation in GPS-denied environments and to operate near humans, thanks to their small size (sub-10 cm) and low weight (sub-40 g). However, their limited on-board computational and memory capacity pose substantial challenges to achieving full autonomy. In particular, they make it impossible to utilize large deep learning models for perception [10, 11]. A particularly relevant task for nano-UAVs is *human pose estimation* [11], which enables applications such as “people monitoring” and “follow-me”. Recent research on this task has concentrated on optimizing tiny Convolutional Neural Networks (CNNs) to operate within tight nano-UAVs hardware constraints, marking a

significant stride towards obtaining reasonable perceptual performance on such platforms [3, 9, 11]. The work of [3], in particular, underscored the critical role of automated optimization methods such as *Neural Architecture Search* (NAS) in facilitating the design of efficient architectures that balance computational efficiency with task performance. However, [3] only scratched the surface of a potentially vast research direction, applying a NAS algorithm aimed at shrinking an input CNN architecture (called “seed”) through the elimination of unimportant feature maps [12], closely resembling *structured pruning*. Other NAS methods allow exploring broader (yet less fine-grained) search spaces, e.g., by selecting between different alternatives for each layer of the CNN [2, 7, 8, 13]. In this work, we show that the sequential application of these two families of NASs (layer selection and model shrinking) can yield superior results in terms of the pose estimation accuracy versus latency and memory trade-offs.

In addition to an optimized network architecture, another key component to enable real-time CNN-based perception on nano-UAVs is the availability of efficient low-level software kernels, fully exploiting the hardware available onboard. To this end, this work proposes the usage of optimized kernels for the execution of *fused DepthWise (DW) and PointWise (PW) convolution* on the Parallel Ultra Low-Power (PULP) multi-core clusters available in recent nano-UAV platforms [10]. Sequences of DW and PW layers are common in tiny CNNs (inspired by MobileNets [6]), and fusing them significantly reduces the amount of intermediate memory transfers, thus improving end-to-end latency compared to single-layer kernels for the same hardware, such as the ones in [5].

Through the combined optimization of a CNN architecture (with two chained NAS steps) and of the corresponding inference software stack, we outperform the state-of-the-art (SoTA) on human pose estimation for nano-UAV-class devices [3, 9], obtaining up to 13.78% lower Mean Absolute Error (MAE), or reducing latency by up to 3.22 \times at iso-error. Our work highlights the key importance of multi-level automated deployment flows for tinyML on tiny drones.

2 Materials and Methods

Target Platform: We focus on the Bitcraze Crazyflie 2.1 nanodrone equipped with a greyscale camera and the GAP8 SoC [4]. GAP8 comprises a single-core fabric controller (FC) and an 8-core PULP cluster (CL). The FC orchestrates memory transfers and delegates demanding computations to the CL. CL cores share a 64 kB L1 memory and the SoC includes a 512 kB L2 memory. A Direct Memory Access (DMA) unit handles transfers between the two memories.

Complexity-Driven Architecture Search: Figure 1 (left) shows the proposed architecture optimization flow, which combines two SotA NASs, *Supernet* [8] and *PIT* [12], using the implementations of the PLiNIO open-source library [7]. Both are so-called Differentiable NASs (DNASs), i.e., they jointly train (with gradient descent) the standard weights of the network W and some additional parameters θ , which control architectural choices such as the type of layer (for

Supernet) or the number of output features of each layer (for *PIT*). The red box of Fig. 1 shows the loss function minimized by both NASs, where \mathcal{L} is the task-specific loss (e.g., Mean Squared Error) and \mathcal{C} is an added complexity term such as the number of parameters or operations of the network, as a function of θ . The balance between the two is controlled by the scalar λ . In particular, we used the size complexity defined in [12].

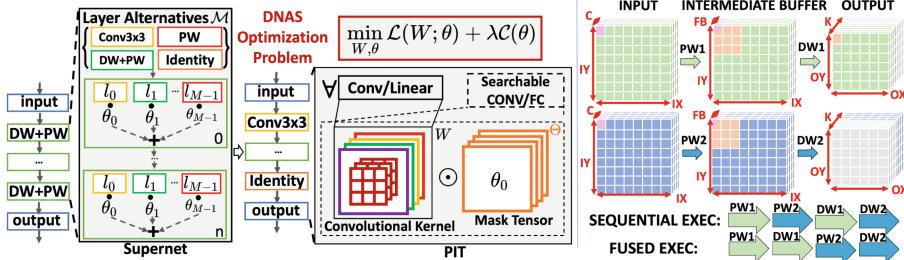


Fig. 1. NAS-based optimization flow (left); Optimized PW+DW kernel (right).

We use a MobileNetV1 architecture as blueprint for the SuperNet step, since it demonstrated SotA performance on human-to-nanodrone pose estimation [3, 9]. For each DW+PW block of the original network, the NAS selects between: i) The original block; ii) A single PW layer; iii) A standard 2DConv with 3×3 filter; iv) A “no-operation” to optionally skip the block. As depicted in Fig. 1, each of the $M = 4$ alternatives is coupled with a θ_i parameter. At the end of training, the alternative associated with the largest θ_i is selected. The architectures obtained with the Supernet are further optimized with PIT, which applies a fine-grained structured pruning, eliminating unimportant output channels from each layer. Namely, the weight tensor of each Conv or linear layer is paired with a set of binary trainable masks θ , which are trained to control whether a specific feature is removed ($\theta_i = 0$) or kept ($\theta_i = 1$).

Fused Kernel for Efficient Inference: DW layers are notoriously difficult to accelerate due to their more limited data reuse options compared to standard 2D Conv. PULP-NN [5], a SoTA open-source kernel library for GAP8, handles this by changing the input data layout from Height-Width-Channels (HWC) to Channels-Height-Width (CHW). However, this causes an increase in data transfers, as the re-ordering operation has to be repeated multiple times when tiling the layers (i.e., loading parts of the data in L1 memory before performing computations) [1]. To reduce this overhead, we implement a new fused kernel, computing a PW+DW sequence entirely in L1, as shown in Fig. 1 (right). We accelerate PW+DW (and not DW+PW) sequences, as this enables exploiting the independence of DW computations across input/output channels (C/K). Specifically, we compute a subset (FB) of the PW output channels storing them in an additional L1 buffer of size $IX * IY * FB$ (IX / IY = input rows/columns); then,

we execute the DW operation on such buffer. The whole process is repeated until all K output channels have been produced. Since the DW is parallelized over channels, we set $FB = 8$, i.e., equal to the cores of GAP8, to maximize utilization with the minimum possible L1 memory overhead.

3 Experimental Results

We train and test our CNNs on the dataset introduced in [11], and with the same data splits of [9] Fig. 2 shows the results obtained with the cascaded application of the Supernet and PIT NASs compared with the SotA networks (red circles) of [9] in the N. of Parameters vs. MAE plane. All the results are uniformly quantized to INT8 data format using [7]. We use a slightly modified version of the test set, as in [9], where the labels have been adjusted to compensate for a calibration inaccuracy of the measurement tools detected through manual inspection.

We apply the SuperNet NAS using two different MobileNetV1 variants as blueprints: a standard one (denoted as *Large*) and one with a width-multiplier of $0.25 \times$ (*Small*). The orange and blue triangles of Fig. 2 denote two promising output architectures obtained with this first NAS step. In particular, we selected the blue triangle (SuperNet Small) as the smallest network achieving a $MAE < 1$. Conversely, we selected the orange one (SuperNet Large) as the network achieving the lowest MAE. The found architectures details are as follows: SuperNet Large substitutes the first three DW+PW layers of the vanilla MobileNetV1 with standard 2DConv layers. Conversely, SuperNet Small substitutes only the first DW+PW block with a 2DConv, while skipping entirely the last DW+PW block.

We then apply PIT to both these models, obtaining the rich collection of Pareto-optimal architectures, depicted in Fig. 2 with blue and orange stars. Noteworthy, our smallest model achieves a $9 \times$ size compression at iso-MAE w.r.t. the most accurate SotA model [9]. Conversely, in the large N. of Parameters regime, our solutions can improve the SotA by up to 13.78%. Note that these results showcase the benefits of the combined usage of both NASs, given that SotA networks have been obtained using PIT only [3].

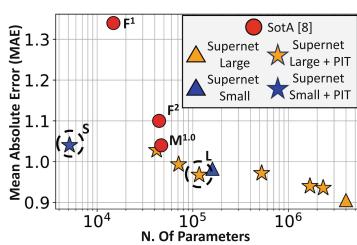


Fig. 2. Optimized architectures vs SotA from [9].

Table 1. Deployment results.

| Model | Ker | MAE | Lat[ms] | Mem[kB] |
|-------------|-----|------|---------|---------|
| SotA | | | | |
| F^1 | U | 1.34 | 7.06 | 14.8 |
| F^2 | U | 1.1 | 8.82 | 44.5 |
| M^1 | U | 1.04 | 21.76 | 46.8 |
| Ours | | | | |
| S | U | 1.04 | 7.4 | 5.14 |
| | F | | 6.76 | |
| L | U | 0.97 | 32.5 | 116.63 |
| | F | | 31.43 | |

We deploy models on GAP8 mirroring the setup of [9]. Table 1 shows the deployment results of selected CNNs from Fig. 2 (those labeled with a letter) from our work and the SoTA. We deploy the smallest (S) and most accurate (L) CNNs found by our NAS chain, excluding models that do not fit the L2 memory of GAP8 (512 kB). For each model, we report the test set MAE, the weight memory footprint (Mem), and the latency (Lat). We also report the kernel used for all PW+DW sequences (Ker), which is either our proposed fused implementation (F) or an unfused one (U) from vanilla PULP-NN.

When considering the S model, we achieve the same MAE at far lower latency (-65%) w.r.t. the most accurate SoTA model (M1). The speedup increases to 68.1% (a further 8.6% reduction) when utilizing our fused PW+DW kernel. If we compare the same model to the least accurate SoTA one (F1), we achieve significantly better MAE (-0.3) with more than $30\times$ fewer parameters. When using fused kernels, we also still reduce latency by 4.2%. The L model, on the other hand, outperforms the most accurate SoTA CNN in terms of MAE (6.98% reduction). Fused kernels are less beneficial for this model, as most of the latency is due to the initial standard convolutional layers. Nonetheless, they still grant a latency reduction of 3.27% compared to a standard deployment. Noteworthy, as shown in [3], reducing perception latency (i.e., improving the frames per second that the model can process) has been shown very beneficial to improve the performance of nano-drones control loops.

4 Conclusions

We have presented a multi-stage, fully-automated optimization pipeline for visual human-to-nanodrone pose estimation, including two chained NAS methods, respectively for layer selection and model pruning, and an optimized implementation of fused PW and DW convolutions to improve CNN inference latency, and consequently the drone’s controller reaction time, by cutting intermediate memory transfers. Overall, our work serves to demonstrate the fundamental importance of deployment optimization pipelines for TinyML on tiny drones.

References

1. Burrello, A., et al.: DORY: automatic end-to-end deployment of real-world DNNs on low-cost IoT MCUs. *IEEE Trans. Comput.* 1253–1268 (2021)
2. Burrello, A., et al.: Enhancing neural architecture search with multiple hardware constraints for deep learning model deployment on tiny IoT devices. *IEEE Trans. Emerg. Top. Comput.* 1–15 (2023)
3. Cereda, E., et al.: Deep neural network architecture search for accurate visual pose estimation aboard nano-UAVs. In: *IEEE ICRA* (2023)
4. Flamand, E., et al.: GAP-8: A RISC-V SoC for AI at the edge of the IoT. In: *IEEE 29th ASAP*, pp. 1–4 (2018)
5. Garofalo, A., et al.: Pulp-NN: a computing library for quantized neural network inference at the edge on RISC-V based parallel ultra low power clusters. In: *26th IEEE ICECS*, pp. 33–36. IEEE (2019)

6. Howard, A.G., et al.: Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint [arXiv:1704.04861](https://arxiv.org/abs/1704.04861) (2017)
7. Jahier Pagliari, D., et al.: Plinio: a user-friendly library of gradient-based methods for complexity-aware DNN optimization. In: FDL, pp. 1–8 (2023)
8. Liu, H., Simonyan, K., Yang, Y.: Darts: differentiable architecture search. arXiv preprint [arXiv:1806.09055](https://arxiv.org/abs/1806.09055) (2018)
9. Motetti, B.A., et al.: Adaptive deep learning for efficient visual pose estimation aboard ultra-low-power nano-drones. [arXiv:2401.15236](https://arxiv.org/abs/2401.15236) (2024)
10. Palossi, D., et al.: An open source and open hardware deep learning-powered visual navigation engine for autonomous nano-UAVs. In: 15th DCOSS, pp. 604–611 (2019)
11. Palossi, D., et al.: Fully onboard AI-powered human-drone pose estimation on ultralow-power autonomous flying nano-UAVs. IEEE IoTJ **9**(3), 1913–1929 (2022)
12. Rocco, M., et al.: Lightweight neural architecture search for temporal convolutional networks at the edge. IEEE Trans. Comput. (2022)
13. Tan, M., et al.: Mnasnet: platform-aware neural architecture search for mobile. In: IEEE/CVF CVPR, pp. 2820–2828 (2019)



Real-Time Fully Convolutional Networks for Peer-to-Peer Nano-drone Visual Localization and LED State Estimation

Nicholas Carlotti^(✉), Mirko Nava, and Alessandro Giusti

Dalle Molle Institute for Artificial Intelligence (IDSIA), USI-SUPSI, 6962 Lugano,
Ticino, Switzerland

nicholas.carlotti@idsia.ch

Abstract. We design and validate a neural network for relative pose estimation and led-based communication using only visual data coming from a nano-drone’s onboard grayscale, low-resolution camera. The model is small enough to be deployed and run in real-time on an onboard hardware accelerator, achieving an inference rate of 39 FPS.

Keywords: Nano-drones · FCN · Perception · TinyML

1 Introduction

Estimating the precise 3D position of a peer robot is a problem faced in many multi-agent robotic systems: in collaborative applications, robots have to navigate the environment without colliding with peers [2], while in non-collaborative applications, malicious robots must be detected and intercepted [10]. To localize peer robots, approaches rely on sensor data, with radio [2, 4, 9] and cameras [3, 5, 8, 10] being the most popular options. Radio technologies such as UWB sensors allow distance estimation of a transmitting source. Güler et al. [4] triangulate the position of peer quadcopters emitting a specific radio signal with three UWB sensors; while others use a single radio sensor and combine sensors’ readings with the robot’s odometry to triangulate the position of peer drones [2, 9]. Purely visual approaches rely on the camera feed to estimate the position of a robot. A widespread class of approaches relies on a robot-mounted visual fiducial marker to reconstruct the pose of the robot relative to the camera [3, 5, 8]. Other visual approaches recover depth information from stereo cameras and use it to triangulate the position of a robot in space [10].

In the context of tiny robots, approaches requiring additional hardware to be mounted on the platform are undesirable due to the increase in payload and energy consumption, e.g., drastically reducing the flight time of a nano-drone. Moreover, visual approaches often employ large neural network models [11] that cannot run on embedded hardware in real-time. We tackle the issues by designing a deep learning model that efficiently runs on an embedded hardware platform, the Greenwaves Technologies GAP8 Parallel Ultra-Low Power (PULP)

board [7] deployed on a Crazyflie 2.1 nano-drone. The model predicts the relative 3D position of a flying peer from a single grayscale monocular image. Additionally, we train the network to classify the state of LEDs mounted on the drone, enabling visual communication between the robots. While state-of-the-art approaches directly regress the robot’s position [1], we propose to use a Fully Convolutional Network (FCN) [6] predicting image-space location, distance, and led state maps; further, we introduce methods to recover the robot’s position from the maps. Results indicate that our approach outperforms the state-of-the-art by more than 10% on the R^2 for the x , y , and z components of the robot’s position on a separate test set.

Table 1. Comparison of the models’ parameters and characteristics.

| Model | # of parameters | MACs $\times 10^6$ | Receptive field [px] |
|---------|-----------------|--------------------|----------------------|
| Model S | 7,737 | 38.43 | 40×40 |
| Model M | 18,593 | 64.56 | 43×43 |
| Model L | 116,649 | 527.46 | 40×40 |

2 Method and Data

We consider the problem of peer-to-peer localization of nano-drones, using as input the feed of a grayscale monocular camera. In this context, we learn a fully convolutional network model [6], which predicts the position of the robot inside of the image $(u, v) \in \mathbb{R}^2$, its distance from the camera $d \in \mathbb{R}^+$ and the status of its onboard LEDs $l \in \{\text{OFF}, \text{ON}\}$. The FCN model consists only of convolutional blocks; each block is composed of the following operators in order: convolution, batch normalization, and ReLU activation. Each of the initial 3 blocks in the network is also followed by a 2×2 max-pooling operation. The model predicts three different maps: the image-space position map $\hat{\mathcal{P}} \in \mathbb{R}^{n \times m}$, the distance map $\hat{\mathcal{D}} \in \mathbb{R}^{n \times m}$, and the LED state map $\hat{\mathcal{L}} \in \mathbb{R}^{n \times m}$, where $n \times m$ are the dimensions of each map. In an FCN, the value of each pixel in an output map depends on a different region of the input image. In our model we set this size, known as *receptive field*, to 40×40 pixels, matching the expected size of the drone in the input images. During inference, each map has to be processed to obtain the respective scalar values $\hat{u}, \hat{v}, \hat{d}$ and \hat{l} .

We consider two methods to process the position map $\hat{\mathcal{P}}$ into the drone position (\hat{u}, \hat{v}) named argmax and barycenter. In the first method, the pixel with the highest activation value in $\hat{\mathcal{P}}$ is chosen as the most likely position for the drone, and its coordinates are used as the predicted pair (\hat{u}, \hat{v}) . The barycenter strategy instead, assigns a weight to each pixel in the output map, proportionally to that pixel’s value. We compute a weighted mean of the map coordinates using the weight assigned to each pixel, resulting in (\hat{u}, \hat{v}) . To obtain scalar predictions

for the robot’s distance \hat{d} and LED state \hat{l} , we combine the respective maps with the predicted position map that acts as a weighting function; this choice reflects the expectation that properties of the robot such as its distance and LED state are estimated accurately only on the portion of the image where the robot is depicted. Specifically, we compute

$$\hat{d} = \sum_{i,j} \left((\hat{\mathcal{D}} \circ \text{norm}(\hat{\mathcal{P}}))_{i,j} \right) \quad \hat{l} = \sum_{i,j} \left((\hat{\mathcal{L}} \circ \text{norm}(\hat{\mathcal{P}}))_{i,j} \right) \quad (1)$$

where \circ indicates a matrix element-wise product and norm is a function that re-scales a matrix’s elements such that they are all positive and sum to 1.

During training, the position, distance, and led scalar ground truth labels are converted into maps. The robot’s position in the image is used to create a ground truth map: we start from a zero-valued $n \times m$ matrix in which we set to 1 all elements within a certain Euclidean distance from (u, v) ; while distance and LED state are converted into ground truth maps by filling each map with the respective scalar value. Each task is learned by minimizing the respective loss; specifically, the loss for the position task is defined as:

$$\mathcal{L}_{\text{pos}} = 1 - \sum_{i,j}^{n,m} (\text{norm}(\hat{\mathcal{P}})_{i,j} \cdot \text{norm}(\mathcal{P})_{i,j}) \quad (2)$$

where $\mathcal{P} \in \mathbf{R}^{n \times m}$ is the ground truth image-space position map. At the beginning of the training process, the predicted robot position $\hat{\mathcal{P}}$ is noisy and negatively affects the learning of distance and LED tasks. As a consequence, the map $\hat{\mathcal{P}}_{\text{int}}$ is computed as $\hat{\mathcal{P}}_{\text{int}} = \hat{\mathcal{P}} \circ \text{norm}(\mathcal{P})$. This map is introduced in (3) to compute the losses for both the distance and LED tasks. The distance task loss is defined as $\mathcal{L}_{\text{task}}(d, \hat{\mathcal{D}})$, while the LED one is defined as $\mathcal{L}_{\text{task}}(l, \hat{\mathcal{L}})$.

$$\mathcal{L}_{\text{task}}(y, \hat{Y}) = \left(y - \sum_{i,j} (\hat{Y} \circ \text{norm}(\hat{\mathcal{P}}_{\text{int}}))_{i,j} \right)^2 \quad (3)$$

Dataset: We collected data in 72 recording sessions. Each session consists of two nano-drones flying inside our laboratory; one being the target drone while the other, called the observer, is tasked to locate its peer. During these flights, we collect: the target drone’s $SE(3)$ pose relative to the observer drone’s camera, as given by a motion capture system; and the state of the target drone’s onboard LEDs, as broadcasted by the drone itself. The LEDs are switched on and off at regular intervals to have balanced labels for the LED state estimation task. In total, we collected 57k examples with grayscale images of 320×320 pixels and divided them into 27k for the training set \mathcal{T} , and 30k for the test set \mathcal{Q} .

Training Approach: We consider three different variants of the same FCN architecture, as shown in Table 1, which vary in the number of channels for each block, having an increasing number of Multiply and ACCumulate (MACs) operations. During training, we adopt the Adam optimizer with a cosine annealing

learning rate scheduler starting from $1e^{-3}$ and decaying at $2e^{-5}$ over 100 training epochs.

3 Experimental Results

In Fig. 1a and 1b, we compare the three models on the error distribution for the localization task computed on the test set \mathcal{Q} . The three models have similar error distributions, regardless of the inference method, with a tendency to make more mistakes as the model size decreases. While Model L has a slightly higher accuracy than the others, it requires drastically more computations for the inference, achieving only 9 FPS when deployed on the nano-drone. Consequently, we opt for Model S for our experiments, achieving 39 FPS on the nano-drone.

In Fig. 1a and 1b, we compare the performance of Model S when using the argmax or barycenter inference methods. Argmax tends to be more precise than barycenter by both having smaller first and third-quartile error values, 9.5 vs 10.9 cm and 25.6 vs 30.5 cm respectively. Our model can process different image resolutions, allowing one to trade latency with performance by using a smaller image resolution. In our experiment, we use 160×160 pixel images (half the size of the dataset) to train and test a Model S architecture, comparing it with one trained and tested on 320×320 pixel images. We show the model's performance on the 3D relative position task in Fig. 1c. The results confirm that using smaller images trades model performance with on-board inference speed, which we measured using 160×160 images to be $25.7 \mu\text{s}$ (39 FPS), while images of larger size do not fit in the fast memory cache, making this strategy impractical for real-time applications. Additionally, we report R^2 scores for the predicted x , y , and z of 56%, 63%, and 59% on the test set, while a state-of-the-art approach [1] only scores 29%, 53%, and 45%, respectively.

Additionally, we report the performance of the LED status recognition task by different model sizes. The performance is quantified by the AUC score of each model obtained on the test set \mathcal{Q} : Model L, M, and S obtained a score of 97.7%, 94.2%, and 92.4% respectively, indicating that each model can solve the task effectively. These results strongly suggest that a visual communication protocol can be implemented on top of these models.

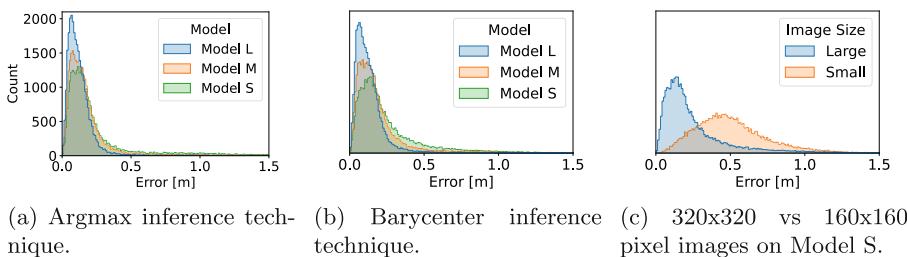


Fig. 1. Relative 3D position error distributions of different models.

4 Conclusions

We presented a relative visual localization and LED state estimation approach for nano-drones. It features a lightweight FCN model, designed to be executed onboard a tiny quadrotor in real-time. Results show that our approach is capable of localizing a peer robot with a median error of 0.18 m in the 3D space, while consistently producing estimates at a rate of 39 FPS on the onboard hardware.

References

1. Bonato, S., Palossi, D.: Ultra-low power deep learning-based monocular relative localization onboard nano-quadrotors. In: IEEE ICRA, pp. 3411–3417 (2023)
2. Coppola, M., McGuire, K.N., Schepers, K.Y.W., de Croon, G.C.H.E.: On-board communication-based relative localization for collision avoidance in micro air vehicle teams. Aut. Robots (2018)
3. Dias, D., Ventura, R., Lima, P., Martinoli, A.: On-board vision-based 3D relative localization system for multiple quadrotors. In: IEEE ICRA, pp. 1181–1187 (2016)
4. Güler, S., Abdelkader, M., Shamma, J.S.: Peer-to-peer relative localization of aerial robots with ultrawideband sensors. IEEE Trans. Control Syst. **29**(5), 1981–1996 (2021)
5. Krajník, T., et al.: A practical multirobot localization system. J. Intell. Robot. Syst. **76** (2014)
6. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: IEEE/CVF CVPR, pp. 3431–3440 (2015)
7. Palossi, D., Conti, F., Benini, L.: An open source and open hardware deep learning-powered visual navigation engine for autonomous nano-UAVs. In: DCOSS, pp. 604–611 (2019)
8. Saska, M., et al.: System for deployment of groups of unmanned micro aerial vehicles in GPS-denied environments using onboard visual relative localization. Aut. Robots **41**(4), 919–944 (2017)
9. Van der Helm, S., Coppola, M., McGuire, K.N., de Croon, G.C.H.E.: On-board range-based relative localization for micro air vehicles in indoor leader-follower flight. Aut. Robots (2020)
10. Vrba, M., Heřt, D., Saska, M.: Onboard marker-less detection and localization of non-cooperating drones for their safe interception by an autonomous aerial system. IEEE RA-L **4**(4), 3402–3409 (2019)
11. Wang, C.Y., Bochkovskiy, A., Liao, H.-Y.M.: YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: IEEE/CVF CVPR, pp. 7464–7475 (2023)



Effect of Optimizer, Initializer, and Architecture of Hypernetworks on Continual Learning from Demonstration

Sayantan Auddy^{1(✉)}, Sebastian Bergner¹, and Justus Piater^{1,2}

¹ Department of Computer Science, University of Innsbruck, Innsbruck, Austria

{sayantan.uddy, justus.piater}@uibk.ac.at,

sebastian.bergner@student.uibk.ac.at

² Digital Science Center, University of Innsbruck, Innsbruck, Austria

Abstract. In *continual learning from demonstration* (CLfD), a robot learns a sequence of real-world motion skills *continually* from human demonstrations. Recently, hypernetworks have been successful in solving this problem. In this paper, we perform an exploratory study of the effects of different optimizers, initializers, and network architectures on the continual learning performance of hypernetworks for CLfD. Our results show that adaptive learning rate optimizers work well, but initializers specially designed for hypernetworks offer no advantages for CLfD. We also show that hypernetworks that are capable of *stable* trajectory predictions are robust to different network architectures. Our open-source code is available at <https://github.com/sebastianbergner/ExploringCLFD>.

Keywords: Learning from Demonstration · Continual Learning

1 Introduction

Learning from Demonstrations (LfD) [1] is an intuitive way for humans to train robots without explicit programming. While the majority of research on LfD addresses single skill acquisition, some recent methods investigate continual learning from demonstration (CLfD) [2, 3], i.e. learning multiple LfD motion skills sequentially in an open-ended way. Auddy et al. [2] propose a system of hypernetwork-generated neural ordinary differential equation solvers (NODEs) for continually learning a sequence of real-world 6-DoF trajectory learning tasks from human demonstrations. More recent work [3], shows that by enforcing stable trajectory predictions through hypernetwork-generated *stable* NODEs [4], the continual learning performance is greatly enhanced in addition to the expected guarantee of non-divergent and safe trajectory predictions. Hypernetworks have also been utilized for continual reinforcement learning with robots [5]. The popularity of hypernetworks for robotic continual learning is mainly due to desirable features such as not having to store and retrain on data of past tasks, negligible parameter growth with additional tasks, and low catastrophic forgetting [6].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2024

C. Secchi and L. Marconi (Eds.): ERF 2024, SPAR 32, pp. 315–320, 2024.

https://doi.org/10.1007/978-3-031-76424-0_56

In any deep learning system, many decisions related to the architecture and training need to be taken. In previous works on CLfD [2,3], the effect of different deep learning components on the continual learning performance of hypernetworks remains unexplored. These past works have either followed accepted best practices (e.g. Adam is the optimizer) or followed the defaults from prior work (e.g. hypernetwork architecture). In this paper, we conduct an exploratory study in which we evaluate the effect of three key deep learning factors on the performance of hypernetworks for CLfD: (i) *optimization algorithms*, (ii) *initialization schemes*, and (iii) *hypernetwork and target network architectures*. We adopt the RoboTasks9 dataset of real-world LfD tasks [3] as a benchmark, and train hypernetworks and chunked hypernetworks continually on the 9 tasks of this dataset. Additionally, we evaluate two kinds of target networks (generated by the hypernetworks): NODE [2], and stable NODE (*s*NODE) [3].

Our results show that adaptive learning rate optimizers exhibit the best empirical performance, but an initializer designed for hypernetworks (Principled Weight Initialization [7]) does not outperform a good default choice (Kaiming [8]) for CLfD. We also show that when stable NODEs (*s*NODEs) are used as the target network (i.e. the LfD trajectory predictions are non-divergent), the continual learning performance is mostly independent of the network architecture.

2 Background

Continual Learning from Demonstration: Learning from Demonstration (LfD) [1] is a robot training paradigm where a robot learns motion skills from human demonstrations. LfD can be performed via kinesthetic teaching where a human physically guides a robot and shows it how to perform a particular motion task. The trajectories demonstrated by the human are recorded and used to learn a vector field [4] which can then be used by the robot to perform a similar motion as the demonstration. While typical LfD approaches focus on learning a single motion skill, the objective of *continual* LfD is to learn and remember a sequence of different motion skills, one at a time, in an open-ended manner with a single model and without storing training data of past demonstrations. In the past, this has been achieved by generating parameterized dynamical systems called *Neural Ordinary Differential Equation solvers* (NODE) with Hypernetworks [2]. More recently, it has been shown that hypernetwork-generated *stable* NODEs (NODEs augmented with a stabilizing Lyapunov function) [4] produce stable, non-divergent trajectories and are more effective at continually learning sequences of real-world and high-dimensional LfD tasks [3].

Hypernetworks: A hypernetwork is a neural network that generates the parameters of another neural network called the *target network* [9]. A hypernetwork \mathbf{f} with parameters \mathbf{h} , takes as input a trainable task embedding vector \mathbf{e}^m and generates the target network parameters $\mathbf{f}(\mathbf{e}^m, \mathbf{h}) = \theta^m$ for the m^{th} task. A two-stage optimization process (see [2,3,9]) is employed to optimize \mathbf{h} and the task embedding vector \mathbf{e}^m . Once the m^{th} task is learned, the task embedding

\mathbf{e}^m is frozen and stored. For learning the $m + 1^{\text{th}}$ task, a new task embedding \mathbf{e}^{m+1} is initialized and the same two-step learning process is repeated. A regular hypernetwork generates all the parameters θ^m of the target network from the final layer, which can result in a large parameter size. Alternatively, *chunked* hypernetworks [9] generate the target network parameters in smaller segments called chunks, and consequently have a smaller size. See [2,3,9] for details.

3 Experiments and Results

We train hypernetworks (HN) and chunked hypernetworks (CHN), each with either a NODE or an sNODE as the target network, resulting in 4 kinds of hypernetworks (HN → NODE, CHN → NODE, HN → sNODE, CHN → sNODE). We compare the performance of 3 different optimizers: Stochastic Gradient Descent (SGD), RMSProp [10], Adam [11] and 3 different initializers: Kaiming [8], Principled Weight Initialization (PWI) [7], Xavier [12] when used to train the 4 kinds of hypernetworks. We evaluate 16 different architectures for each hypernetwork. Each model is trained continually on the 9 LfD tasks of RoboTasks9 [3]. We report the widely used *Dynamic Time Warping* (DTW) error metric [2,3]. Due to the large number of possible combinations, we perform our experiments in 3 stages to keep the number of runs manageable. To aid reproducability and further research, our code and experiment hyperparameters are available at <https://github.com/sebastianbergner/ExploringCLFD>.

Experiment 1 (Optimizers): We train each of the 4 kinds of hypernetworks with the 3 different optimization algorithms (SGD, RMSProp, Adam). We use a fixed architecture (same as [3]) and initializer (Kaiming) for all hypernetworks. After each task is learned during the continual learning process, we evaluate each model on the currently learned task and all previous tasks and repeat each run 5 times with independent seeds. Figure 1 (top row) shows the overall DTW errors

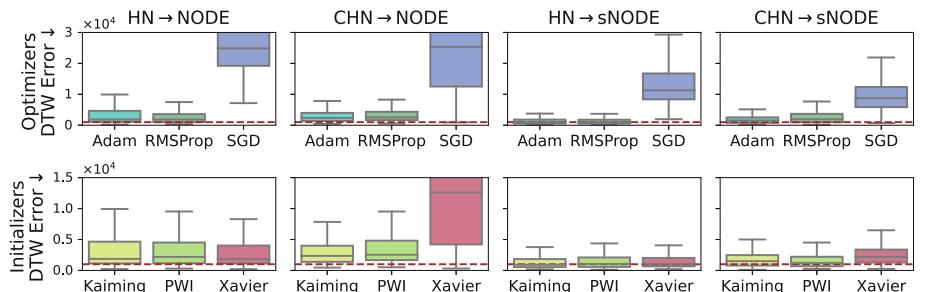


Fig. 1. DTW errors (lower is better) of different *optimizers* (top), and *initializers* (bottom) for 5 independent runs. For reference, the dotted brown line shows the best possible median DTW score from [3] (when each task is learned with a separate model).

during this evaluation. For all optimizers, $\text{HN} \rightarrow \text{sNODE}$ and $\text{CHN} \rightarrow \text{sNODE}$ outperform $\text{HN} \rightarrow \text{NODE}$ and $\text{CHN} \rightarrow \text{NODE}$. The overall performance of SGD is much worse than Adam and RMSProp, both of which achieve similarly good DTW errors.

Experiment 2 (Initializers): We use Adam as the optimizer since it is slightly better than RMSProp for the well-performing models (with sNODE) in the previous experiment. We compare 3 initializers (Kaiming, PWI, and Xavier) while training the 4 kinds of hypernetworks with fixed architectures (same as [3]). We follow the same training and evaluation steps as in experiment 1 and repeat each run 5 times. We report the DTW errors for these evaluations in Fig. 1 (bottom row). All initializers perform similarly, except for $\text{CHN} \rightarrow \text{NODE}$, where the Xavier initialization fails completely (very high DTW errors). For the other hypernetworks, all the initializers achieve similar results. All our hypernetworks use ReLU activations, and while Xavier is designed for *tanh/sigmoid*, we still include it in our experiments since it is used in a similar comparison in a prior work on hypernetwork initialization [7]. Additionally, though PWI is specially designed for hypernetworks, it does not outperform Kaiming for CLfD.

Experiment 3 (Architecture): In our final experiment, we fix Adam as the optimizer and Kaiming as the initializer, as they achieve marginally better median DTW scores than their respective alternatives. In this experiment, we evaluate 4 different network depths (2, 3, 4, or 8 layers) for both the hypernetwork and its generated target network, resulting in 16 different architectures for each of the 4 kinds of hypernetworks. We also modify the number of units in each layer such that the overall parameter size of the networks is roughly similar and comparable to the network sizes of the previous experiments. We follow the same training procedure as the previous two experiments and repeat each run 5 times with independent seeds. The median DTW results are shown in Fig. 2.

For $\text{HN} \rightarrow \text{NODE}$, the depth of the target network affects the overall performance much more than the depth of the hypernetwork, while for $\text{HN} \rightarrow \text{sNODE}$, almost all architectures achieve similar results irrespective of the depth of either network. Chunked hypernetworks ($\text{CHN} \rightarrow \text{NODE}$ and $\text{CHN} \rightarrow \text{sNODE}$) on the other hand, perform best when the hypernetwork is 3–4 layers deep and the target network is 4–8 layers deep. However, similar to $\text{HN} \rightarrow \text{sNODE}$, $\text{CHN} \rightarrow \text{sNODE}$ also performs similarly for almost all architectures. In summary, hypernetworks with sNODE as the target network perform well and are much less sensitive to the network architecture than hypernetworks with NODE.

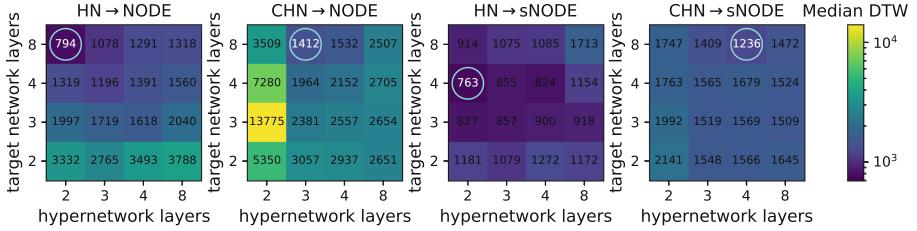


Fig. 2. Effect of hypernetwork depth (y-axis) and target network depth (x-axis) on continual learning from demonstration. Each heatmap corresponds to a different hypernetwork type. Circled numbers show the best DTW for each hypernetwork. Colors are scaled logarithmically. Median values over 5 independent runs are shown.

4 Conclusion

We demonstrated the effects of the optimizer, initializer, and network architecture on hypernetwork-based continual learning from demonstration. Our findings show that adaptive learning rate optimizers (Adam, RMSProp) are a good choice for CLfD. Kaiming is a good default initializer that performs as well as PWI which is specially designed for hypernetworks. We also showed that hypernetworks with *sNODE* as the target network are mostly independent of the network architecture. Our findings can help in making informed design decisions while developing hypernetwork-based methods for CLfD in the future.

References

1. Billard, A., Calinon, S., Dillmann, R.: Learning from humans. Springer Handbook of Robotics, 2nd ed. (2016)
2. Auddy, S., Hollenstein, J., Saveriano, M., Rodríguez-Sánchez, A., Piater, J.: Continual learning from demonstration of robotics skills. Robot. Auton. Syst. 165, 104427 (2023). <https://www.sciencedirect.com/science/article/pii/S0921889023000660>
3. Auddy, S., Hollenstein, J., Saveriano, M., Rodríguez-Sánchez, A., Piater, J.: Scalable and efficient continual learning from demonstration via hypernetwork-generated stable dynamics model. arXiv preprint [arXiv:2311.03600](https://arxiv.org/abs/2311.03600) (2023)
4. Kolter, J.Z., Manek, G.: Learning stable deep dynamics models. In: Advances in Neural Information Processing Systems, vol. 32, pp. 128–136 (2019)
5. Schöpf, P., Auddy, S., Hollenstein, J., Rodriguez-Sanchez, A.: Hypernetwork-ppo for continual reinforcement learning. In: Deep Reinforcement Learning Workshop NeurIPS 2022 (2022)
6. Parisi, G.I., Kemker, R., Part, J.L., Kanan, C., Wermter, S.: Continual lifelong learning with neural networks: a review. Neural Network 113, 54–71 (2019)
7. Chang, O., Flokas, L., Lipson, H.: Principled weight initialization for hypernetworks. In: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26–30, 2020. OpenReview.net (2020)

8. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1026–1034 (2015)
9. von Oswald, J., Henning, C., Sacramento, J., Grawe, B.F.: Continual learning with hypernetworks. In: International Conference on Learning Representations (ICLR) (2019)
10. Tieleman, T., Hinton, G.: 6.5-rmsprop coursera: neural networks for machine learning university of Toronto. Tech. Rep. (2012)
11. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2014)
12. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pp. 249–256 (2010)



Exploiting Foundation Models for Efficient Labeling of Deformable Linear Objects

Alessio Caporali¹(✉), Kevin Galassi¹, Matteo Pantano², and Gianluca Palli¹

¹ DEI Department, University of Bologna, Bologna, Italy
alessio.caporali2@unibo.it

² Department of Electrical and Computer Engineering, Technical University of Munich, Munich, Germany

Abstract. The integration of robotic solutions for manipulating Deformable Linear Objects (DLOs) faces significant challenges due to their complex perception. One way to solve this issue is to use deep learning algorithms. However, these require extensive training data. This paper introduces a method for efficiently labeling DLOs in images at the pixel level, starting from sparse annotations of key points. The method allows the generation of a real-world dataset of DLO images for segmentation purposes with minimal human effort. The approach comprises four main steps. First, a user operates a spatial sensor to record key points along the real-world DLOs in the scene. Second, a robot equipped with an eye-in-hand camera collects multiple images of the scene with an ellipsoidal trajectory. Third, a neural network, framed as a regression task, is employed to correct and align the input key points with the centerlines of the DLOs. Finally, a pre-trained foundation model, specifically the Segment Anything Model (SAM), transforms the sparse annotations into a comprehensive pixel-wise mask. The comparison with a baseline approach demonstrates that the proposed method can increase the intersection over the union score of circa 14% without requiring specific fine-tuning procedures. Therefore, the proposed method constitutes a groundstone for enabling low-effort DLO labeling and thus integrating deep learning perception of DLOs.

Keywords: Deformable Linear Objects · Spatial Labeling · Dataset Generation · Foundation Models

1 Introduction

Deformable Linear Objects (DLOs), encompassing wires, cables, ropes, catheters, and tubes, are ubiquitous in daily life. Their inherent elongated cylindrical shape, devoid of distinct features, poses a challenge for Computer Vision

This work was partially supported by the Horizon Europe project *IntelliMan* with grant number 101070136.

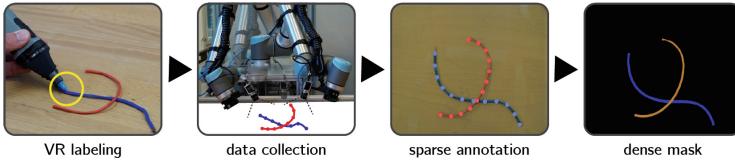


Fig. 1. Overview of the approach based on sparse labeling.

(CV) detection, impacting the feasibility of autonomous robotic manipulation [2, 3, 7]. Indeed, industrial tasks such as cabling and wire routing in sectors like automotive and aerospace predominantly rely on human intervention [9, 14].

To address this gap, new CV perception methodologies based on deep learning (DL) are crucial, specifically concerning the segmentation of DLOs [4, 6, 15]. However, those approaches require training data. The existing data-driven approaches face challenges related to dataset quality and size, particularly in the DLO domain [11], making the development of efficient data collection and labeling procedures desirable.

This paper expands upon the initial approach introduced in [5] to reduce domain knowledge. Specifically, it eliminates constraints related to knowledge of DLO diameter and the necessity for various basic image processing techniques to convert sparse annotation into a dense mask. Instead, within a zero-shot setting, a pre-trained Foundation Model [8] is employed, achieving improved results over [5]. This substitution enhances the method’s portability and efficiency by relaxing the overall setup requirements. To achieve this goal, this work leverages the Segment Anything Model (SAM) [10] for its proven capabilities as a multimodal foundation model [8]. For clarity, Fig. 1 provides an overview of the approach.

2 Method

To achieve dense labeling of DLOs in real-world scenarios through user input, two key components are exploited: a sparse key points annotation approach from [5] (Sect. 2.1) and the use of SAM for dense mask generation, discussed in Sect. 2.2.

2.1 Dataset Collection and Sparse Key-Points Labeling

Recording of Images with a Robot. To collect the set of images, knowledge of images and the camera’s position in the world coordinate system is essential. A 2D RGB camera is mounted to the flange of a robotic arm in an eye-in-hand setup, see Fig. 1. The camera is intrinsically and extrinsically calibrated. An ellipsoidal robot trajectory ensures that the object remains at the trajectory center with inward-facing camera shots as previously described by [5, 12].

Spatial Labeling. A spatially tracked sensor method is chosen for labeling DLO instances. Specifically, the TracepenTM spatial sensor is used. This sensor

utilizes reflective photodiode sensors for pose calculation from an emitting station's infrared signal. The spatial sensor system, the robot, and the camera are all calibrated, allowing the transformation of the spatial sensor coordinates collected by the user to camera pixel coordinates as proposed in [12,13]. Thus, this approach takes images of real-world scenarios with different camera positions using a single labeled input. This procedure is repeated for each DLO instance to label.

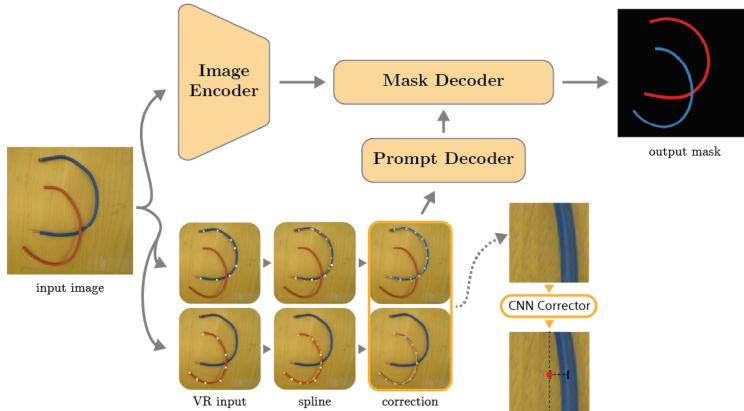


Fig. 2. Sparse key-point labeling (from [5])

Key-Points Correction via Neural Network Regressor. During real-world DLO labeling by a human operator, errors are expected, primarily arising from 1) calibration inaccuracies in the annotation tool and eye-in-hand camera, and 2) human operator labeling, with more pronounced issues on thin DLOs. To address these challenges, a fine-tuning step follows human input for each labeled DLO instance, outlined in Fig. 2. Firstly, the labeling points of a DLO instance undergo smoothing via an approximating spline curve in 2D pixel space (see Fig. 2(c)). Subsequently, a CNN-based approach computes a correction offset for each labeled point (see Fig. 2(d)). Given the source image, for each point, a vertically oriented crop is extracted. Then, the CNN model predicts the approximated local centerline of the DLO used to make the correction. Additional details about the CNN can be found in [5].

2.2 From Sparse to Dense Labels

Foundation models are networks trained on large datasets to be used straightforwardly in real-life applications. Recently, they have received significant attention, thanks to the exploitation of DL in everyday life [1] and robotics [8]. Typically, these models are tailored to *address* specific tasks such as object detection or

natural language processing. However, several multimodal models capable of taking different kinds of input data have recently emerged [8].

To tackle the challenge of obtaining dense labels from sparse labeling, we utilize the SAM model belonging to the class of multimodal models. SAM takes an RGB image as input and is prompted by key points (other possible promptable inputs, not exploited in this work, include bounding boxes and text). Specifically, the sparse set of key points obtained from Sect. 2.1 is leveraged to generate a segmentation mask for the input RGB image. To distinguish this approach from one of the previous work [5], we denote this as WSL-SAM.

3 Experiments

The experiments are performed by comparing the original approach [5], denoted as WSL in the following, with the proposed method based on SAM [10] (denoted as WSL-SAM) for obtaining the dense labels from the sparse annotation, see Sect. 2.2.

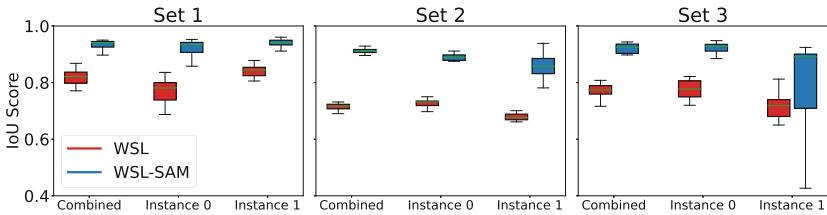


Fig. 3. Comparison between WSL ([5]) and proposed method WSL-SAM

Dataset Test and Metric. As test sets, three *Sets* obtained by exploiting the data collection approach detailed in Sect. 2.1 are employed. The *Sets* have the following structure: Set 1 (black and blue plastic tubes); Set 2 (red and blue electrical wires); and Set 3 (red and striped electrical wires). Notably, Set 3 is the hardest set due to the complex striped appearance of the DLO. Each set is composed of 25 images, each featuring two DLO instances in the scene. A manual labeling approach is performed on each of the 75 images to obtain ground truth data in the form of dense instance masks. As metric, the Intersection over Union (IoU) score is used to evaluate the performances, similarly to [5].

Discussion. The results across the different *Sets* are shown in Fig. 3. Concerning the *combined* mask, WSL-SAM achieves mean IoU scores of 93.3%, 91.1%, and 81.9% across *Sets* 1, 2, and 3, respectively. Conversely, the baseline WSL achieves mean IoU scores of 79.3%, 71.5%, and 75.1% on the same respective sets. Statistical significance is observed for *Sets* 1 and 2.

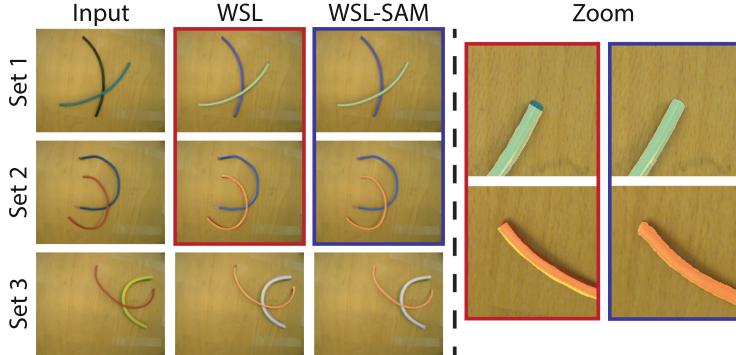


Fig. 4. Qualitative comparison of the methods with zoomed crops.

The results highlight how the dense annotation via WSL-SAM performs better than the original approach. Indeed, when prompted with high-quality input key points (obtained thanks to Sect. 2.1), the generalization capabilities of SAM allow accurate segmentation of the single DLO instances without relying on specific task knowledge like their diameter or crossing order, as is the case for [5]. A detailed view of the capabilities of SAM when segmenting DLOs from key points is provided in Fig. 4.

4 Conclusions

In conclusion, this paper addresses the challenge of labeling Deformable Linear Objects (DLOs) for generating a real-world, task-specific dataset for use in data-driven methods for perception. The proposed method offers an effective pixel-level labeling approach for DLOs in images, utilizing sparse annotations of keypoints as a starting point. The utilization of the Segment Anything Model (SAM) allows us to obtain accurate dense masks without the need for specific fine-tuning objectives with an improvement of circa 14% from our previous work. In the future, we will investigate the proposed method in more detail, especially analyzing its accuracy with different DLOs, such as ropes, and various backgrounds with cluttered scenes.

References

1. Bommasani, R., et al.: On the opportunities and risks of foundation models. arXiv preprint [arXiv:2108.07258](https://arxiv.org/abs/2108.07258) (2021)
2. Caporali, A., Galassi, K., Palli, G.: Deformable linear objects 3D shape estimation and tracking from multiple 2D views. IEEE Robot. Autom. Lett. **8**(6), 3852–3859 (2023)
3. Caporali, A., Galassi, K., Žagar, B.L., Zanella, R., Palli, G., Knoll, A.C.: RT-DLO: real-time deformable linear objects instance segmentation. IEEE Trans. Ind. Inf. **19**(11), 11333–11342 (2023)

4. Caporali, A., Galassi, K., Zanella, R., Palli, G.: Fast deformable linear objects instance segmentation. *IEEE Robot. Autom. Lett.* **7**(4), 9075–9082 (2022)
5. Caporali, A., Pantano, M., Janisch, L., Regulin, D., Palli, G., Lee, D.: A weakly supervised semi-automatic image labeling approach for deformable linear objects. *IEEE Robot. Autom. Lett.* **8**(2), 1013–1020 (2023)
6. Caporali, A., Zanella, R., De Greoglio, D., Palli, G.: Ariadne+: deep learning-based augmented framework for the instance segmentation of wires. *IEEE Trans. Ind. Inf.* **18**(12), 8607–8617 (2022)
7. Choi, A., Tong, D., Park, B., Terzopoulos, D., Joo, J., Jawed, M.K.: mbest: realtime deformable linear object detection through minimal bending energy skeleton pixel traversals. *IEEE Robot. Autom. Lett.* (2023)
8. Firoozi, R., et al.: Foundation models in robotics: applications, challenges, and the future. arXiv preprint [arXiv:2312.07843](https://arxiv.org/abs/2312.07843) (2023)
9. Galassi, K., Palli, G.: Robotic wires manipulation for switchgear cabling and wiring harness manufacturing. In: 2021 4th IEEE International Conference on Industrial Cyber-Physical Systems (ICPS). IEEE (2021)
10. Kirillov, A., et al.: Segment anything. arXiv preprint [arXiv:2304.02643](https://arxiv.org/abs/2304.02643) (2023)
11. Nguyen, H.G., Habiboglu, R., Franke, J.: Enabling deep learning using synthetic data: a case study for the automotive wiring harness manufacturing. *Procedia CIRP* **107**, 1263–1268 (2022)
12. Pantano, M., et al.: Simplifying robot grasping in manufacturing with a teaching approach based on a novel user grasp metric. In: 5th International Conference on Industry 4.0 and Smart Manufacturing (2023)
13. Pantano, M., Schmidt, M., Bolano, G., Schulenburg, E., Regulin, D., Saenz, J.: Analyzing the influence of self-defined trajectories on safety and task ownership: an empirical study. In: 19th IEEE International Conference on Automation Science and Engineering (CASE) (2023)
14. Trommnau, J., Kühnle, J., Siegert, J., Inderka, R., Bauernhansl, T.: Overview of the state of the art in the production process of automotive wire harnesses, current research and future trends. *Procedia CIRP* **81**, 387–392 (2019)
15. Zanella, R., Caporali, A., Tadaka, K., De Gregorio, D., Palli, G.: Auto-generated wires dataset for semantic segmentation with domain-independence. In: ICCCR (2021)



Approaches for Exploiting Neural Networks for Semi-supervised Myoelectric Control of Robot Hands

Roberto Meattini^(✉), Alessandra Bernardini, Alessio Caporali, Gianluca Palli,
and Claudio Melchiorri

Department of Electrical, Electronic, and Information Engineering (DEI), University
of Bologna, 40136 Bologna, Italy
roberto.meattini@unibo.it

Abstract. Human-In-The-Loop (HITL) control strategies using surface electromyography (sEMG) face challenges with labeling in supervised learning. Unsupervised regression methods for sEMG signals have limitations in controlling multiple grasp motions. This paper presents two semi-supervised regression approaches using neural networks (NN) for sEMG-based robot hand control. The first approach uses soft-DTW divergence as a loss function for minimally supervised NN training. The second combines Non-Negative Matrix Factorization (NMF) with self-supervised NN regression. Offline tests show the soft-DTW NN performs similarly to a standard MSE-based NN, and the self-supervised regression outperforms traditional unsupervised methods, enabling multiple grasp actions.

Keywords: Multifingered hands · intention recognition · human factors and human-in-the-loop · neural networks · semi-supervised learning

1 Introduction

This paper focuses on Human-Robot interfaces (HRI) for robot hands in teleoperation, prosthetics, and learning by demonstration [1]. In robot hand control, surface electromyography (sEMG) is widely explored for decoding grasping intentions, mainly through supervised machine learning [2]. Classification-based HRI discriminates discrete commands but faces reliability issues with an increasing number of grasping actions. Regression-based approaches, like simultaneous and proportional (s/p) control, offer continuous regulation but suffer from training errors and user frustration due to complex instant-by-instant labeling procedures.

To address these limitations, our paper illustrates two possible semi-supervised regression methods based on NN. The first approach leverages a soft-DTW Neural Network (soft-DTW NN), that exploits a differentiable version of Dynamic Time Warping (DTW) in a neural network – the soft-DTW [3]

–, allowing minimally supervised regression without instant-by-instant labeling. In the second approach, a self-supervised regression combining Non-Negative Matrix Factorization (NMF) [4] with NN is illustrated, addressing the limitations of unsupervised regression methods for controlling robot hands with multiple grasping motions. Both methods are systematically evaluated and compared offline, highlighting their advantages over traditional techniques, and demonstrating their efficacy in realizing semi-supervised regression of sEMG signals for human-in-the-loop (HITL) multi-grasp robot hand control.

2 Methods

Soft-DTW Neural Network. Let us consider the vector of filtered root mean square (RMS) sEMG signals, denoted as $E(t) = [e_1(t), \dots, e_8(t)]^T$, applied to the input of the soft-DTW NN. Let us consider the network structure as composed by n hidden layers and an output layer. The output layer – the $(n+1)$ -th layer of the network – contains two neurons (i.e. $N_{n+1} = 2$) with linear transfer function $\mathcal{H}(\cdot)$ and, therefore, the output vector of the network $a^{(n+1)}(t) \in \mathbb{R}^{N_{n+1}}$ is described by

$$a^{(n+1)}(t) = \begin{bmatrix} a_1^{(n+1)}(t) \\ a_2^{(n+1)}(t) \end{bmatrix} = \mathcal{H}(W^{(n+1)}a^{(n)}(t) + b^{(n+1)}), \quad (1)$$

where $a_1^{(n+1)}(t)$ and $a_2^{(n+1)}(t)$ are the two scalar outputs of the network, and $a^{(n)}(t) \in \mathbb{R}^{N_n}$, $W^{(n+1)} \in \mathbb{R}^{N_{n+1} \times N_n}$ and $b^{(n+1)} \in \mathbb{R}^{N_{n+1}}$ are the input vector, weight matrix and bias vector of the output layer. Defining $c(t) := a_1^{(n+1)}(t)$ and $g(t) := a_2^{(n+1)}(t)$, for the robot hand control we impose $\alpha_{\text{PO}}(t) = \gamma_{\text{PO}}(g(t))c(t)$, $\alpha_{\text{TR}}(t) = \gamma_{\text{TR}}(g(t))c(t)$, $\alpha_{\text{UL}}(t) = \gamma_{\text{UL}}(g(t))c(t)$, where $\alpha_{\text{PO}}(t)$, $\alpha_{\text{TR}}(t)$ and $\alpha_{\text{UL}}(t)$ are the synergistic references for the power, tripodal and ulnar grasps [5], respectively, and γ_{PO} , γ_{TR} , γ_{UL} are gains with a trapezoidal profile such that $c(t)$ has the task of regulating the closure level of the power, tripodal and ulnar grasps, allowing a continuous transition among the different grasps. The network training was performed with the scaled conjugate gradient back-propagation algorithm. The loss function used was the soft-DTW divergence [3] between the network 2-dimensional output $A_T \in \mathbb{R}^{2 \times d}$, and the 2-dimensional target output $T \in \mathbb{R}^{2 \times d}$ with a length of d samples. The target output T is given once the sEMG training dataset E_T is provided. Specifically, the user is required to execute the following sequence of continuous grasping motions while the sEMG data is acquired and stored in $E_T \in \mathbb{R}^{8 \times d}$: (i) power grasp closure followed by power grasp opening; (ii) tripodal grasp closure followed by tripodal grasp opening; (iii) ulnar grasp closure followed by ulnar grasp opening. Then, the target output $T = [t_1 \ t_2]^T \in \mathbb{R}^{2 \times d}$ is defined as described in the following. t_1 presents a trapezoidal profile between -1 and 1, uniformly distributed along the d samples given by the sEMG training set, resembling the three closure levels of the power, tripodal and ulnar grasps, and t_2 indicates the grasp related to each of the three closure levels, where the values 1, 2 and 3 indicate the power, tripodal and ulnar grasps, respectively. Note that no instant-by-instant labelling is required.

3 Self-supervised Regression

Consider M distinct grasping actions, where a user performs a specific grasping action m ($1 \leq m \leq M$). For each action, we define a matrix \mathbf{A}_m , which contains the sEMG signal samples captured during the execution of the grasping motion m . Thus, we have M matrices: $\mathbf{A}_1 \in \mathbb{R}^{8 \times l_1}, \dots, \mathbf{A}_M \in \mathbb{R}^{8 \times l_M}$. Next, Non-negative Matrix Factorization (NMF) [4] is applied to each matrix \mathbf{A}_m to estimate the relative muscular synergy matrix \mathbf{B}_m . We use the M synergy matrices $\mathbf{B}_1, \dots, \mathbf{B}_M \in \mathbb{R}^{8 \times M}$ to compute the offline motor drive matrices $\mathbf{C}_1 \in \mathbb{R}^{M \times l_1}, \dots, \mathbf{C}_M \in \mathbb{R}^{M \times l_M}$ as

$$\mathbf{C}_j = \begin{bmatrix} c_{e,j} \\ c_{f,j} \end{bmatrix} = \mathbf{B}_j^+ \mathbf{A}_j, \quad j = \{1, \dots, M\}. \quad (2)$$

We then derive the offline grasp closure vectors $\hat{\tau}_1 \in \mathbb{R}^{1 \times l_1}, \dots, \hat{\tau}_M \in \mathbb{R}^{1 \times l_M}$ as follows

$$\hat{\tau}_j^T = \frac{\gamma_{1,j}}{2} c_{e,j}^T - \frac{\gamma_{2,j}}{2} c_{f,j}^T + \gamma_{3,j}, \quad j = \{1, \dots, M\}, \quad (3)$$

where $\gamma_{1,j}$ and $\gamma_{2,j}$ are scaling factors for normalizing $c_{e,j}$ and $c_{f,j}$, and $\gamma_{3,j} = 1$. Moving from the general scenario of M grasping actions to the specific case studied here involving power (PW), tripodal (TR), and ulnar (UL) grasps, Eq. (3) simplifies to

$$\hat{\tau}_j^T = \frac{\gamma_{1,j}}{2} c_{e,j}^T - \frac{\gamma_{2,j}}{2} c_{f,j}^T + \gamma_{3,j}, \quad j = \{\text{PW, TR, UL}\}. \quad (4)$$

This allows the construction of an sEMG training set \mathbf{D}_T as

$$\mathbf{D}_T = [\mathbf{A}_{\text{PW}} \ \mathbf{A}_{\text{TR}} \ \mathbf{A}_{\text{UL}}] \in \mathbb{R}^{8 \times (l_{\text{PW}} + l_{\text{TR}} + l_{\text{UL}})}, \quad (5)$$

with corresponding labels $T \in \mathbb{R}^{3 \times (l_{\text{PW}} + l_{\text{TR}} + l_{\text{UL}})}$ defined as

$$T = [\omega_1 \ \omega_2 \ \omega_3]^T = [\beta_{\text{PW}} \hat{\tau}_{\text{PW}} \ \beta_{\text{TR}} \hat{\tau}_{\text{TR}} \ \beta_{\text{UL}} \hat{\tau}_{\text{UL}}], \quad (6)$$

where $\beta_{\text{PW}}, \beta_{\text{TR}}, \beta_{\text{UL}} \in \mathbb{R}^{n_G=3}$ represent the synergy activations for the maximum closure level of power, tripodal, and ulnar grasps [5]. The training set \mathbf{D}_T and labels T are then used to train a neural network (NN) for multi-grasp control of a robotic hand based on sEMG. Note that the labels T are automatically derived from the sEMG calibration data. The network comprises n hidden layers with Rectified Linear Activation Unit (ReLU) activation functions and an output layer. The output layer is characterized by $N_{n+1} = 3$ neurons, a weight matrix $\mathbf{G}^{(j)} \in \mathbb{R}^{N_{n+1} \times N_n}$ and a sigmoid activation function $\mathcal{S}(\cdot)$. The output vector $\mathbf{a}^{(n+1)}(t)$ of the NN is given by

$$\begin{aligned} \mathbf{a}^{(n+1)}(t) &= \left[a_1^{(n+1)}(t) \ a_2^{(n+1)}(t) \ a_3^{(n+1)}(t) \right]^T \\ &= \mathcal{S}(\mathbf{G}^{(n+1)} \mathbf{a}^{(n)}(t) + \mathbf{b}^{(n+1)}) \end{aligned} \quad (7)$$

where $a_1^{(n+1)}(t)$, $a_2^{(n+1)}(t)$, and $a_3^{(n+1)}(t)$ are the network's three scalar outputs, and $\mathbf{a}^{(n)}(t) \in \mathbb{R}^{N_n}$ and $\mathbf{b}^{(n+1)} \in \mathbb{R}^{N_{n+1}}$ are the input and bias vectors of the output layer. The goal is to train the network so that its outputs can control the closure levels of power, tripodal, and ulnar grasps of the robotic hand. Training can be performed using the scaled conjugate gradient back-propagation algorithm with mean squared error (MSE) as the loss function, using the training set \mathbf{D}_T and the target outputs $T = [\omega_1 \ \omega_2 \ \omega_3]^T$ as defined in Eq. (6). Once training is complete, the robotic hand is controlled by setting the vector of grasp synergy activations [5] as follows

$$\beta(t) = \left[a_1^{(n+1)}(t) \ a_2^{(n+1)}(t) \ a_3^{(n+1)}(t) \right]^T. \quad (8)$$

4 Experiment and Results

Evaluation of Soft-DTW Neural Network. In an offline experiment, we systematically assessed the soft-DTW NN's ability to tolerate inaccurately labeled sEMG signals. Comparisons were made with a standard MSE-based NN under varying labeling circumstances. Ten subjects performed grasp motions, and target outputs for NN training were created with different degrees of desynchronization w.r.t. the sEMG data: synchronized, shrunk by 1/3, or shrunk by 2/3. Six nested cross-validations were conducted for each subject, considering soft-DTW and MSE-based NNs with three target output variations. The final result, obtained through averaging six NN performance values, utilized standard DTW as the metric, suitable for offline sEMG-based s/p control evaluations. In the offline experiment reported in Fig. 1(a), boxplots group results by target outputs for NN training (soft-DTW and MSE-based). Two-way ANOVA showed a significant interaction between NN type and target output. One-way ANOVA revealed no significant difference in performance between soft-DTW NNs and MSE-based NN trained with synchronized, shrunk by 1/3, or shrunk by 2/3 targets. This demonstrates the robustness of the proposed sEMG-based minimally supervised regression approach to labeling desynchronization. Conversely, MSE-based NN performance significantly degraded with desynchronization, as evident in Fig. 1(a).

Evaluation of Self-supervised Regression. The offline evaluation included ten participants replicating a series of hand movements. Three nested cross-validations (nCV) were independently conducted for three regression techniques: single-NMF [5], concatenated-NMF [4], and self-supervised regression. The nCV process aimed to reduce biases and overestimations by partitioning the sEMG dataset, performing grid-search for hyper-parameters, and assessing model performance using Dynamic Time Warping (DTW) on a test subset. The mean DTW score represented the nCV outcome for each regression technique and participant. The findings showed that self-supervised regression outperformed concatenated-NMF in offline systematic evaluations, as verified visually and

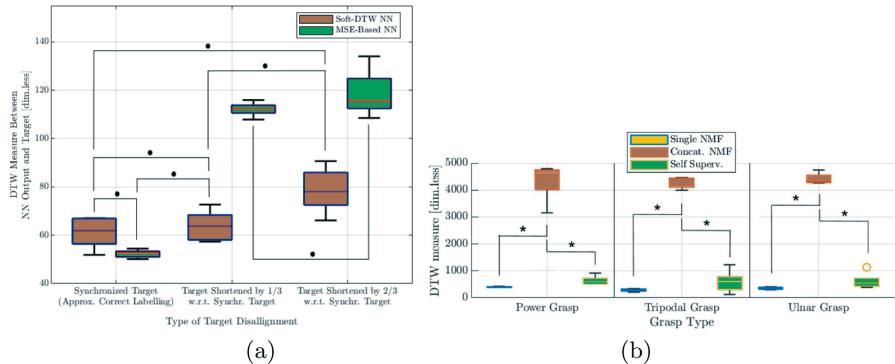


Fig. 1. Outcomes of (a) the assessment of the soft-DTW neural network (the symbol “●” indicates no statistically significant difference) and (b) self-supervised regression (the symbol “*” denotes a statistically significant difference).

statistically in Fig. 1(b). A two-way repeated measures ANOVA indicated a significant effect of Regression Technique, and pairwise comparisons using the Tukey Test revealed a significant difference only between concatenated-NMF and the other two methods, highlighting the superior performance of self-supervised regression.

5 Conclusions

This study proposed two effective semi-supervised methods based on NN for myoelectric robot hand control: soft-DTW and self-supervised regression. These findings offer practical solutions and avenues for future research.

Acknowledgement. This work was partially supported by European Commission’s Horizon Europe Framework Programme with the project IntelliMan under Grant 101070136, by MUR with the project “Sustainable Mobility Center” under Grant CN00000023-CUP J33C22001120001, and by MICS (Made in Italy – Circular and Sustainable) Extended Partnership and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.3 – D.D. 1551.11-10-2022, PE00000004).

References

1. Melchiorri, C.: Robot Teleoperation, pp. 1–14. Springer, London (2013)
2. Ameri, A., Kamavuako, E.N., Scheme, E.J., Englehart, K.B., Parker, P.A.: Support vector regression for improved real-time, simultaneous myoelectric control. *IEEE Trans. Neural Syst. Rehabil. Eng.* **22**(6), 1198–1209 (2014)
3. Blondel, M., Mensch, A., Vert, J.-P.: Differentiable divergences between time series. In: International Conference on Artificial Intelligence and Statistics, pp. 3853–3861. PMLR (2021)

4. Jiang, N., Englehart, K.B., Parker, P.A.: Extracting simultaneous and proportional neural control information for multiple-DOF prostheses from the surface electromyographic signal. *IEEE Trans. Biomed. Eng.* **56**(4), 1070–1080 (2008)
5. Meattini, R., Benatti, S., Scarcia, U., De Gregorio, D., Benini, L., Melchiorri, C.: An sEMG-based human-robot interface for robotic hands using machine learning and synergies. *IEEE Trans. Compon. Packag. Manuf. Technol.* **8**(7), 1149–1158 (2018)



Closing the Sim-to-Real Gap for Dynamics-Static Friction and Inertial Parameters: A Franka Robot Case Study

Davide Bargellini^(✉), Andrea Govoni, Riccardo Zanella, and Gianluca Palli

DEI, Department of Electrical, Electronic, and Information Engineering, Alma Mater
Studiorum - Università di Bologna, Bologna, Italy

Abstract. This exploration addresses challenges in transitioning robotic capabilities from simulation to real-world applications, focusing on disparities between simulated and real environments. It tackles issues like dynamic variations and parameter sensitivity with innovative solutions, introducing the SIPE benchmark for evaluating parameter estimation algorithms. Cutting-edge approaches such as Auto-Tuned Sim-to-Real Transfer and Adapting Simulation Randomization are examined, emphasizing the importance of experience and precise parameter estimation. The article thoroughly investigates dynamic and static friction, proposing practical solutions. The conclusion offers a comparative analysis of simulators and strategies to minimize disparities, contributing valuable insights to the simulation-to-real transfer discourse.

Keywords: Sim-to-Real Transfer · Reality Gap · Parameter Estimation

1 Introduction

In the realm of robotics, the efficacy of simulation-to-real transfer has emerged as a pivotal challenge. Bridging the gap between simulated environments and the real world is essential for deploying robotic systems that exhibit robust and reliable behaviours across diverse scenarios. This article delves into the nuances of sim-to-real transfer, drawing insights from recent research.

Simulation environments offer a controlled and cost-effective avenue for training robotic models, yet the transition to the real world often reveals discrepancies that impede seamless deployment. One prominent strategy to enhance the generalization capabilities of simulated models involves the application of *Domain Randomization*. This technique introduces variations in simulated parameters, encompassing both visual and dynamic aspects of the environment. The aim is to provide the model with a diverse training dataset that spans a wide range of conditions, preparing it for the unpredictability inherent in real-world settings. However, the effectiveness of domain randomization is contingent on judicious parameter selection, requiring expert knowledge to balance robustness without hindering the learning process.

Our exploration begins by delving into the challenges posed by simulation limitations. The inherent discrepancies between simulated and real-world dynamics, inaccuracies in model representations, and the sensitivity of simulation outcomes to parameter variations are among the key impediments. Recognizing these challenges, recent research proposes innovative methods to address simulation limitations and facilitate a more effective sim-to-real transfer.

One noteworthy contribution lies in the proposition of the *Simulation Parameter Estimation (SIPE) benchmark* [1]. This benchmark offers a standardized framework for evaluating algorithms in the space of parameter estimation, system identification, and simulator calibration. By dividing the parameter estimation process into distinct phases—trajectory collection, algorithm training, and real robot trajectory utilization—SIPE aims to provide a comprehensive means of assessing and comparing the performance of various parameter estimation techniques.

This article also delves into the intricacies of dynamic and static friction estimation, a crucial aspect of model calibration. By employing penalty-based optimization, researchers propose a method for identifying dynamic parameters, including joint friction parameters, for a specific robotic platform—the Panda robot. The significance of physically consistent dynamic parameters is emphasized, and the study validates the estimated parameters through comparative analyses and validation tests. In addressing static friction nuances, we adopt a practical approach. By incrementally applying torque ramps to individual joints in the physical robot, we estimate static friction empirically. This method provides specific insights into real-world friction characteristics, enhancing the accuracy of simulation models.

In the subsequent sections, we delve deeper into each of these facets, exploring the methodologies, results, and implications outlined in the aforementioned abstracts. Our goal is to contribute to the ongoing discourse on sim-to-real transfer, shedding light on the challenges, innovations, and promising avenues for improving the deployment of robotic systems in the real world.

1.1 Typical Solutions

In the realm of parameter estimation for robotic systems, recent advancements showcase diverse approaches to address challenges such as the reality gap and sim-to-real transfer. The paper [2] discusses the limitations of existing reality gap mitigation methods like domain randomization, emphasizing the need for prior knowledge. A comparative analysis with related works demonstrates its efficacy, particularly in image-based sim-to-real tasks.

Meanwhile, [3] presents a data-driven strategy using real-world data to enhance simulation randomization. The focus is on learning simulation parameter distributions conducive to successful policy transfer, minimizing the necessity for exact real-world replication. The article delves into closed-loop policy rollouts, comparing them with trajectory-based parameter learning and underlining the potential of physics simulations for parameterized models. Experimental validations demonstrate successful policy transfers for complex robotic tasks,

emphasizing the importance of accurate parameter estimation in achieving real-world applicability.

Shifting the focus to the Franka Emika Panda Robot, the paper [4] further underscores the importance of parameter estimation for accurate modelling and control of the Panda robot. It introduces a penalty-based optimization algorithm for retrieving feasible dynamic parameters, discussing the role of nonlinear and conditional constraints. Validation against measured torques reinforces the need for accurate parameter estimation, emphasizing its impact on control and planning strategies. A comparison with classical methods corroborates the reliability of the proposed penalty-based optimization approach.

2 Franka Simulation

To address the reality gap specific to the Franka Emika Panda Robot, a comprehensive set of solutions is proposed, encompassing dynamic friction and static friction parameters. The primary parameters influencing the reality gap are identified as the robot's inertias and static/dynamic frictions. The paper [5] introduces a solution for dynamic friction by presenting a formula specifically addressing link-side friction (Table 1).

$$\tau_{f,j} = \frac{\varphi_{1,j}}{1 + e^{-\varphi_{2,j}(\dot{q}_j + \varphi_{3,j})}} - \frac{\varphi_{1,j}}{1 + e^{-\varphi_{2,j}\varphi_{3,j}}}, \quad j \in [1, \dots, 7] \quad (1)$$

Table 1. Estimated Friction Parameters

| | Joint 1 | Joint 2 | Joint 3 | Joint 4 | Joint 5 | Joint 6 | Joint 7 |
|-------------|---------|---------|---------|---------|---------|---------|---------|
| φ_1 | 0.5462 | 0.8722 | 0.6407 | 1.2794 | 0.8390 | 0.3030 | 0.5649 |
| φ_2 | 5.1181 | 9.0657 | 10.1360 | 5.5903 | 8.3469 | 17.1330 | 10.3360 |
| φ_3 | 0.0395 | 0.0259 | -0.0461 | 0.0362 | 0.0262 | -0.0210 | 0.0036 |

However, it's crucial to note that the proposed dynamic friction solution is tailored for link-side friction, leaving motor-side friction unaddressed. The library libfranka internally runs a friction observer compensating for most of the motor-side friction, recognizing its dominance compared to link-side friction.

Moving on to static friction, a viable approach is suggested to introduce it into the simulation. By determining the minimum torque required for each joint in the real robot to initiate motion, it is possible to reproduce the effect of static friction in simulation. To do so, the joint torques are set as 0 when the joint velocity is within a threshold close to 0 and the joint torque is lower than the one found.

These values found through experimentation on the Franka Emika Panda are:

- Joint₀ = 0.97 Nm
- Joint₁ = 0.98 Nm
- Joint₂ = 1.02 Nm
- Joint₃ = 0.99 Nm
- Joint₄ = 0.94 Nm
- Joint₅ = 0.49 Nm
- Joint₆ = 0.71 Nm

After obtaining these approximate friction values, an improvement strategy involves generating trajectories in the real robot and comparing them with simulated trajectories. Furthermore, parameter randomization is employed to iteratively refine simulation parameters, aiming for simulated trajectories that closely align with real-world counterparts. This iterative process allows for a more accurate simulation of the robot's behaviour, effectively mitigating the reality gap for the Franka Emika Panda.

3 Differences Between MuJoCo and Gazebo

MuJoCo is identified as a physics engine, primarily designed for simulating the physical interactions of rigid-body structures. In contrast, Gazebo falls under the category of system simulators, encompassing not only the physics engine but also simulating sensors and robotic interfaces. MuJoCo represents joints as constraints between bodies, while Gazebo utilizes a modular structure that allows for extension by adding plugins. Gazebo's architecture is designed to facilitate the simulation of multi-contact scenarios and enables writing the same code for both real and simulated robots. In MuJoCo, the simulation time is governed by a control frequency that dictates the duration of each control timestep. Within each timestep, the simulation progresses through n substeps, and after each substep, an observation array is generated. This approach provides a direct control mechanism over the simulation timing.

On the contrary, Gazebo adopts a more indirect approach. Users must set the simulation frequency for Gazebo, while the observation array is managed through ROS topics, though not all topics support high-frequency updates. Additionally, actions are published through a separate topic, introducing the possibility of delays in information transfer. In our study, we found out that setting the MuJoCo control frequency at 250 Hz with 4 substeps and ROS joint_state_publisher at 500 Hz is a good compromise between control reactivity and system stability.

Another substantial difference lies in how simulators handle contact forces and friction. To mitigate discrepancies, it is recommended to utilize the previously presented equation for dynamic friction (Fig. 1).

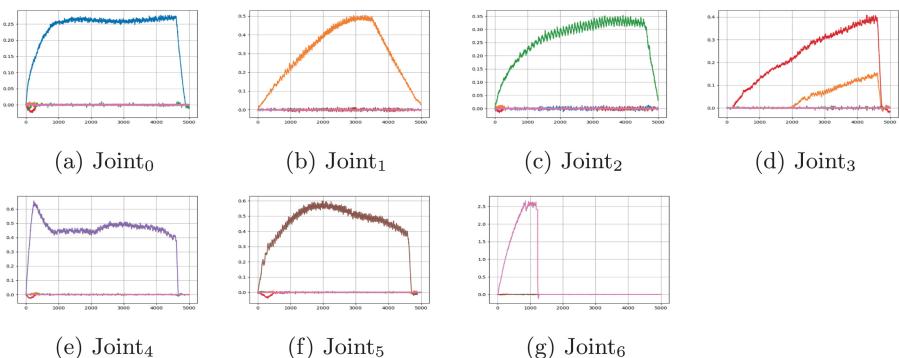


Fig. 1. Real joint trajectories

References

1. Mehta, B., et al.: A user's guide to calibrating robotic simulators. In: Kober, J., Ramos, F., Tomlin, C. (eds.) Proceedings of the 2020 Conference on Robot Learning, vol. 155. Proceedings of Machine Learning Research, pp. 1326–1340. PMLR (2021). <https://proceedings.mlr.press/v155/mehta21a.html>
2. Du, Y., et al.: Auto-tuned sim-to-real transfer. In: 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 1290–1296 (2021). <https://doi.org/10.1109/ICRA48506.2021.9562091>
3. Chebotar, Y., et al.: Closing the sim-to-real loop: adapting simulation randomization with real world experience. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 8973–8979 (2019). <https://doi.org/10.1109/ICRA.2019.8793789>
4. Gaz, C., et al.: Dynamic identification of the Franka Emika panda robot with retrieval of feasible parameters using penalty-based optimization. IEEE Robot. Autom. Lett. 4(4), 4147–4154 (2019). <https://doi.org/10.1109/LRA.2019.2931248>
5. Gaz, C., et al.: Dynamic identification of the Franka Emika panda robot with retrieval of feasible parameters using penalty-based optimization. Supplementary Material (2019). <https://inria.hal.science/hal-02265294>



Learning Passive Policies

Riccardo Zanella¹(✉), Federico Califano¹, Cristian Secchi², and Stefano Stramigioli¹

¹ Robotics and Mechatronics (RaM) Group, University of Twente, Enschede, The Netherlands
r.zanella@utwente.nl

² Department of Sciences and Methods of Engineering, University of Modena and Reggio Emilia, Modena, Italy

Abstract. We merge the techniques of *passivity-based control* (PBC) and *reinforcement learning* (RL) in a robotic context, with the goal of learning passive control policies. We frame our contribution in a scenario where PBC is implemented by means of *virtual energy tanks*, a control technique developed to achieve closed-loop passivity for any arbitrary control input. The use of RL in combination with energy tanks allows to learn control policies which, under proper conditions, are structurally passive. Simulations show the validity of the approach, as well as novel research directions in energy-aware robotics.

Keywords: passivity-based control · energy tanks · reinforcement learning

1 Introduction

The field of robotics is placing more emphasis on creating control frameworks that enable robots to move from controlled industrial environments to unstructured environments. This transition carries the ambitious objective of stable and safe execution of complex tasks where robots coexist with other robots or possibly humans. The practical impossibility of careful dynamic modeling of the environment along those tasks makes usual stability objectives very challenging, and model-based approaches poorly suitable.

Passive controllers have been presented as a feasible solution to tackle this problem as the stability of the closed-loop system is in principle independent on the external environmental interaction [1,2]. A powerful technique allowing the passivization of any control action is represented by *virtual energy tanks* [3–5]. The main drawback of passivity-based control methods is the lack of optimization over a task performance metric along the design of the controller [6]. On the other side of the control theoretic spectrum, recent advancements in the machine learning community are leading to robots with an outstanding awareness of complex environments and tasks. The drawback of these families of approaches is the difficulty in guaranteeing system theoretic properties such as passivity and stability for the controlled system [7].

In this work, mainly discussing the results in [8], we merge the passivity-based control (PBC) design involving energy tanks and reinforcement learning framework

The research leading to these results has received funding from the European Union's Horizon Europe Framework Programme under grant agreement No 101070596 (euROBIN).

(RL), combining system theoretic properties of the closed-loop system induced by the ultimately passive design, and high-performance achievement peculiar of the RL framework.

2 Background

Reinforcement Learning

RL [9] is a model-free framework, consisting of an agent interacting with an environment, to solve optimal control problems stated as Markov decision processes (MDPs). The goal of an RL algorithm is to find an optimal policy π^* that maximizes the expected γ -discounted sum of future rewards (that we define as return).

Energy Tanks

A virtual energy tank is a dynamical system that constitutes an energy-storing element and it can be represented as the 1D system, $\dot{x}_c = u_t, y_t = \frac{\partial V_c(x_c)}{\partial x_c}$ where the energy function is simply $V_c(x_c) = \frac{1}{2}x_c^2$. At the core of the energy tank algorithm [3], a special power-preserving interconnection between the robot and the tank allows to i) implement a desired control action w on the space of generalised forces, necessary for task execution of the robot, and ii) use the sum of the energy of the tank and the robot as storage function to prove passivity of the closed-loop system as long as the energy in the tank does not deplete. To meaningfully implement the energy tank algorithm, it is necessary to constantly observe the energy in the tank in order to implement a control action \bar{w} , rather than w , that is equal to w if $V_c(x_c) \geq 0$ and 0 otherwise (empty tank). This way, formal passivity of the closed-loop system can be formally proven since the system is just detached from the controller at the moment in which no energy budget is left.

The arbitrariness of the control input w and its disembodiment from any physical dynamics are tempting motivations to choose it as a decision variable in an optimization framework, which motivates the merging of RL and energy tank algorithm.

3 Energy Tanks Meet RL

Task Energy

Let us introduce the *task energy*, that we will denote by e^* , firstly defined in [10], as the minimum energy in the tank necessary to fulfill passively some task. In other words, if $V_c(x_c(t))|_{t=0} > e^*$, the tank never depletes along the whole task horizon. Note that the energy tank algorithm constitutes a clever way to disembody from any physical dynamics the implementation of a passive control action. In fact, there is no need to design the controller in a way that it mimics a physical system: as long as a task is achieved through a control input w and e^* is finite, a passive implementation is possible by means of the energy tank algorithm, just setting $V_c(x_c(t))|_{t=0} > e^*$.

Discrete-Time Energy Tanks

Since MDPs are discrete-time processes, we need to employ the energy tank algorithm at the discrete-time level. Let us denote by e_k the level of energy in the tank at the discrete time step k (that is, $e_k = V_c(kT)$), and likewise for generalised forces w_k and joint configuration q_k . The energy in the tank at the next time step, initialized by e_0 , is updated according to the rule $e_{k+1} = e_k - \Delta e_{k+1}$. As shown in [8], using the update rule

$$\Delta e_{k+1} := \begin{cases} w_k^\top (q_{k+1} - q_k) & \text{if } w_k^\top (q_{k+1} - q_k) \geq 0 \\ 0 & \text{otherwise} \end{cases}, \text{ passivity of the closed-loop system}$$

is achieved in a discrete-time setting, provided the energy in the tank does not deplete.

We define the *energy spent at step k* as the sum of the energies exiting the tank up to that step $\hat{e}_k := \sum_{l=0}^{k-1} \Delta e_{l+1}$. Since with the chosen update rule the sequence \hat{e}_k is monotonically non-decreasing, the task energy e^* over a task with N time steps can be simply calculated as $e^* = \hat{e}_N$.

Learning Passive Policies

In robotics it is often convenient to continuously achieve a passive closed-loop system while learning the task-dependent control policy. This is desirable for instance when the training takes place in real life, and not in simulations, such that initial exploration phases are guaranteed not to undergo hazardous unbounded energy generation. Furthermore, the passivized training phase endows the agent with awareness of the metabolic spending encoded in the tank architecture, induced by the initial energy budget e_0 . In particular, the learned policy will be influenced by the tank initialization e_0 , and in particular, the task energy e^* will be directly learned together with the control policy in a combined way, strengthening the significance of the resulting passivity property.

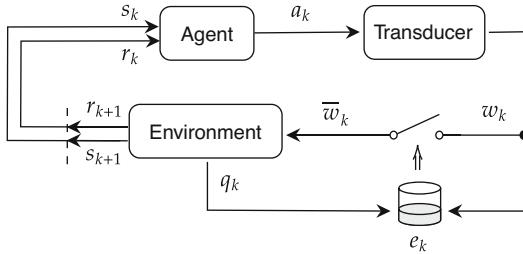


Fig. 1. RL scheme with energy tanks algorithm.

4 Simulations

The proposed framework is evaluated in an inverted-pendulum problem implemented in MuJoCo physics simulator [11]. In all the simulations presented in this work the agents are trained using SAC algorithm [12] implemented in PyTorch and trained with an NVIDIA GeForce GTX 1080 Ti on an Intel Core i7-7700K CPU clocked at 4.20 GHz.

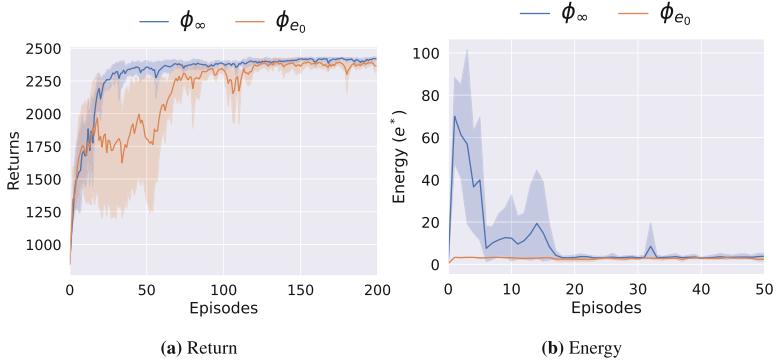


Fig. 2. In (a) and (b) respectively the average returns obtained and the energy spent in training.

Two agents ϕ_∞ and ϕ_{e_0} are trained in the pendulum environment sharing the same hyperparameter configuration, where ϕ_{e_0} is trained in an environment which includes energy tanks as in Fig. 1, while ϕ_∞ is a standard RL agent. In training ϕ_{e_0} , we initialise the tank with e_0 as the task energy estimated from the trained agent ϕ_∞ . As visible in Fig. 2, the average return of the two agents during the training converges to the same value, while ϕ_∞ can arrive to spend a level of energy that is almost 50 times greater than the one spent in ϕ_{e_0} , see Fig. 2.

5 Discussion

Benefits of the Proposed Approach

The initial energy assignment is a degree of freedom in the algorithm, whose implications are often naively addressed. In fact, a very high (yet bounded) energy initialization in the tank would technically still fulfill the PBC objective, yet creating so-called “practically unstable behaviors” [3, 13]. As a consequence, a naive tank design *de facto* makes robust stability a property that is not connected to any safety guarantee. In fact, notice that till the energy in the tank is not depleted, the control input w is completely transparent to the tank algorithm, which reduces to a trick to formally prove passivity, with limited significance in the context in which tasks need to be performed in unstructured scenarios. The fact that the mechanical power flow is undefined in sign worsens this criticality, which is sometimes addressed with empirical tank saturation arguments. Both the task energy e^* and the control action w are often difficult to be determined a priori, and these are not independent variables. For complex task executions, it is reasonable to take advantage of simulations to optimize for both variables *in a combined way*. If the PBC objective can be naively achieved just by initializing the energy in the tank to a sufficiently high value, what we claim to be a useful system theoretic property in the energy tank context is the achievement of the PBC objective *combined* with an estimation of the task energy. In fact, the knowledge of the latter leads to a meaningful energy tank initialization, so that a depletion of the tank can be used as a diagnostic tool to detect an important divergence from nominal task execution, beyond formally achieving a passive closed-loop system.

Future Works

As a future work, we are processing a study of energy-aware robotics throughout the proposed neural framework. Our focus lies in leveraging the energy tank architecture to integrate crucial metabolic and safety metrics, intricately tied to the physical energy and power flows experienced by the robot. This extension goes beyond the confines of passive designs, representing a pivotal step forward, particularly for tasks demanding continuous energy injection, such as periodic locomotion. The incorporation of metabolic metrics allows us to delve into the intricate dynamics of the robot's energy consumption and utilization, providing valuable insights into its operational efficiency. Simultaneously, the integration of safety metrics is geared towards enhancing the overall reliability of the system.

To conclude, our aim is to bridge the gap between energy-aware robotics, safety considerations, and the intricacies of the energy tank framework, offering a more holistic approach to intelligent robotic systems.

6 Conclusions

We introduced a framework merging the energy tank algorithm, used as a tool to passivize arbitrary control schemes, and reinforcement learning, representing the most versatile method to learn control policies along complex tasks. The presented procedure allows us to learn constructively passive policies. After presenting simulations, we discuss the benefits of the proposed framework and future research directions.

References

1. Stramigioli, S.: Energy-aware robotics. In: Camlibel, M., Julius, A., Pasumarthy, R., Scherpen, J. (eds.) *Mathematical Control Theory I. Lecture Notes in Control and Information Sciences*, vol. 461, pp. 37–50 (2015)
2. van der Schaft, A.: *L2-Gain and Passivity Techniques in Nonlinear Control*, 3rd edn. Springer Publishing Company, Incorporated (2016)
3. Califano, F., Rashad, R., Secchi, C., Stramigioli, S.: On the use of energy tanks for robotic systems. In: Borja, P., Della Santina, C., Peternel, L., Torta, E. (eds.) *Human-Friendly Robotics*, vol. 2023, pp. 174–188. Springer, Cham (2022)
4. Secchi, C., Stramigioli, S., Fantuzzi, C.: Position drift compensation in port-Hamiltonian based telemanipulation. In: IEEE International Conference on Intelligent Robots and Systems, pp. 4211–4216 (2006)
5. Duindam, V., Stramigioli, S.: Port-based asymptotic curve tracking for mechanical systems. *Eur. J. Control.* **10**(5), 411–420 (2004)
6. Dimeas, F., Aspragathos, N.: Online stability in human-robot cooperation with admittance control. *IEEE Trans. Haptics* **9**(2), 267–278 (2016)
7. Buşoniu, L., de Bruin, T., Tolić, D., Kober, J., Palunko, I.: Reinforcement learning for control: performance, stability, and deep approximators. *Annu. Rev. Control.* **46**, 8–28 (2018)
8. Zanella, R., Palli, G., Stramigioli, S., Califano, F.: Learning passive policies with virtual energy tanks in robotics. *IET Control Theory Appl.* **00**, 1–10, (2024). <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/cth2.12558>
9. Sutton, R.S., Barto, A.G.: *Reinforcement Learning: An Introduction* (2018)

10. Schindlbeck, C., Haddadin, S.: Unified passivity-based Cartesian force/impedance control for rigid and flexible joint robots via task-energy tanks. In: Proceedings of the IEEE International Conference on Robotics and Automation, vol. 2015, pp. 440–447 (2015)
11. Todorov, E., Erez, T., Tassa, Y.: MuJoCo: a physics engine for model-based control. In: IEEE International Conference on Intelligent Robots and Systems, pp. 5026–5033 (2012)
12. Haarnoja, T., Zhou, A., Abbeel, P., Levine, S.: [SAC] Soft actor-critic. In: 35th International Conference on Machine Learning, ICML 2018, vol. 5, pp. 2976–2989 (2018)
13. Benzi, F., Ferraguti, F., Riggio, G., Secchi, C.: An energy-based control architecture for shared autonomy. *IEEE Trans. Robot.* **38**, 1–19 (2022)



Sim-to-Real Gap in RL: Use Case with TIAGo and Isaac Sim/Gym

Jaume Albardaner, Alberto San Miguel^(✉), Néstor García, and Magí Dalmau

Eurecat, Centre Tecnològic Catalunya, Robotics and Automation Unit,
Barcelona, Spain

{jaume.albardaner,alberto.sanmiguel,nestor.garcia,
magi.dalmau}@eurecat.org

Abstract. This paper explores policy-learning approaches in the context of sim-to-real transfer for robotic manipulation using a TIAGo mobile manipulator, focusing on two state-of-art simulators, Isaac Gym and Isaac Sim, both developed by Nvidia. Control architectures are discussed, with a particular emphasis on achieving collision-less movement in both simulation and the real environment. Presented results demonstrate successful sim-to-real transfer, showcasing similar movements executed by an RL-trained model in both simulated and real setups.

Keywords: Reinforcement Learning · Isaac Gym · Isaac Sim · TIAGo · Sim2Real

1 Reinforcement Learning Simulators

Reinforcement Learning (RL) techniques are especially useful for applications that require a certain degree of autonomy. In Robotics, this applies to tasks related to object manipulation and navigation, and the main difficulty lies on the gathering of data to train a model. An economically expensive approach to make it faster is to acquire multiple robots and run them simultaneously [1–3]. Models trained with that data are more likely to work on the real robot, since the data was obtained from the same platform the model runs on. However, performing RL on real robots still requires supervision to ensure that no unexpected scenario is met. This approach is not only expensive, it also requires a lot of time for data generation. Not only because of how slow the robot may move, but also because the environment may need to be prepared before every run.

Following this need of making the process faster and cheaper, several libraries for RL were developed such as OpenAI's Gymnasium [4] or Stable-baselines [5] and robot models were introduced into simulators, like Mujoco [6]. Although this process was definitely cheaper, it did not necessarily make the process faster. Following hardware development, simulators with GPU support appeared, among which both IsaacGym [7] and IsaacSim [8] from Nvidia stand out. Nonetheless, what happens inside the simulators is nothing more than a simplification of reality. Thereafter, an adaptation must be made to models if these are to be utilized

in a real environment. This is a very active branch of RL called sim-to-real (or Sim2Real), which has been the main focus of this paper.

2 TIAGo Use Case

2.1 Models

Simulating physics for objects that don't have simple meshes is computationally expensive. If possible, simulators simplify these meshes into convex shapes to speed up the simulation. Additionally, in most simulators the default settings for robots include not checking for self collisions.

Such is the case for the TIAGo mobile manipulator robot¹: the simulator settings had to be changed to allow self-collision and the model for the robot had to be modified because of the self-collisions caused by the mesh simplification.

2.2 Controls

The way the robot is controlled and how it reacts to control inputs differs in each simulator. In this section the control pipeline of the simulators that were used, as well as the real robot, are described.

Isaac Gym. Three types of joint controls can be applied: position, velocity and effort controls. As described in Fig. 1, when the reference value for the position or velocity is updated, a PD controller is used to track the new reference. The PD controller takes its k_p and k_d values from the *stiffness* and *damping* parameters of each joint. The resulting control is delivered to the robot, which ensures no joint limitations are violated and simulates its effects on each joint.

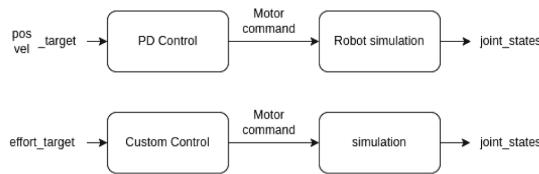


Fig. 1. Control pipeline for Isaac Gym and Isaac Sim.

In Isaac Gym there is no built-in controller for effort control. However, sample controllers are provided in Nvidia's IsaacGymEnvs GitHub repository².

¹ <https://pal-robotics.com/robots/TiaGo/>.

² <https://github.com/NVIDIA-Omniverse/IsaacGymEnvs>.

Isaac Sim. Since this simulator also relies on PhysX and has also been developed by Nvidia, the control pipeline that is followed is the same as Fig. 1. The only difference is that a wide variety of controllers are provided off-the-shelf. These controllers can operate the robot in joint and cartesian space.

TIAGo. The robot that has been used in this use case follows the control schematics established by the `ros_control` Robot Operating System (ROS) package³ (Fig. 2). Since it is community driven each step of the process can be customized and evaluated independently.



Fig. 2. Control pipeline for the real TIAGo.

In the case of TIAGo, the *JointGroupPosition* controller has been employed. This controller was selected because it's the most similar one to Isaac's PD: it is a PID.

3 Results

3.1 Simulator Responses

In order to evaluate the difference in responses in each simulator with the real robot, a step input was introduced to each joint of the arm. Although the individual response for each joint was similar in both simulators, when steps were introduced simultaneously to several joints responses differed.

Figure 3 shows that responses for the arm in Isaac Gym resemble better the results obtained in the real TIAGo than those obtained in Isaac Sim. Nonetheless, it seems to have a slower response, possibly due to prioritizing each joint based on their position in the hierarchical tree. This hypothesis is based on the evolution of the accumulated errors, where errors seem to converge slower on DOF 7 than in DOF 1. Additionally, accumulated errors are also smaller for Isaac Gym than for Isaac Sim, as seen in Table 1. On the other hand, the difference in response with respect to Isaac Sim is most likely due to velocity constraints not being respected at a joint level, as seen in Fig. 4.

Table 1. Accumulated errors (rad) for each DOF after steady state has been achieved. The lowest values are highlighted in bold.

| | DOF 1 | DOF 2 | DOF 3 | DOF 4 | DOF 5 | DOF 6 | DOF 7 | Σ |
|-----------|---------------|---------------|----------------|---------------|----------------|----------------|----------------|-----------------|
| Isaac Gym | 8.4202 | 7.5179 | 26.4569 | 9.0018 | 43.6909 | 6.46288 | 12.7499 | 114.3006 |
| Isaac Sim | 6.6786 | 4.6277 | 16.5728 | 9.1713 | 11.9117 | 46.1073 | 57.0088 | 152.0785 |

³ https://github.com/ros-controls/ros_control.

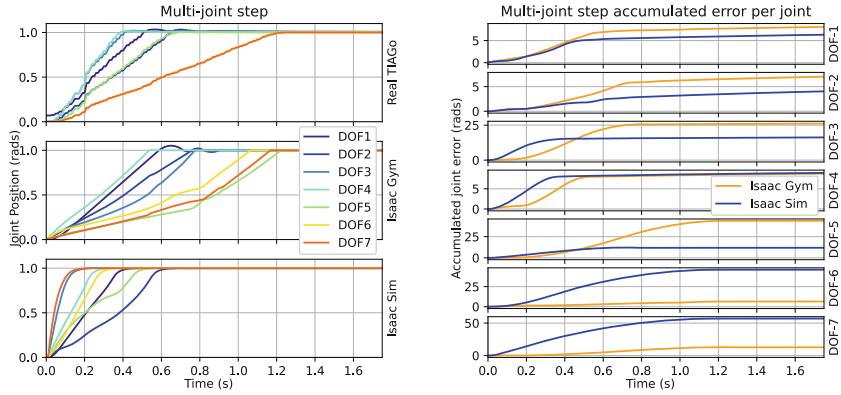


Fig. 3. Responses for each DOF in TIAGo’s arm to simultaneous step inputs on each joint in the real TIAGo, Isaac Gym and Isaac Sim (a) and the accumulated errors of each simulator w.r.t the real (b)

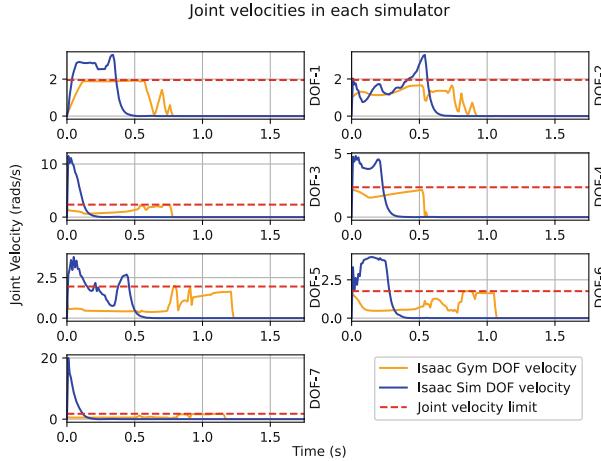


Fig. 4. Velocities for each DOF in TIAGo’s arm to simultaneous step inputs on each joint in Isaac Gym and Isaac Sim

3.2 Trained Model

The first model that was trained with this setup takes the TIAGo mobile manipulator from its “Home” position to a position where the arm is fully extended (Fig. 5), which corresponds to all joints being in the zero position.

In Fig. 5 it can be seen that although both models were trained using the same reward function ($\text{reward} = -|DOF_{value}|$) for the same number of training epochs (100K), the movement differs.

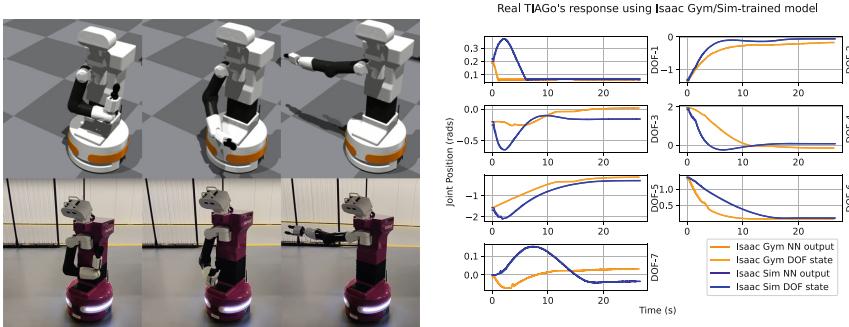


Fig. 5. Home-to-zero movement from the TIAGo in Isaac Gym (left-top), in the real setup (left-bottom) and the DOF positions through time for the models trained in Isaac Gym and Isaac Sim (right).

4 Conclusions

In this paper, policy learning approaches were brought to the context of sim-to-real transfer for robotic platforms, particularly using Isaac Gym and Isaac Sim simulators for the TIAGo mobile manipulator. While presented results showcase promising achievements, identified issues should be addressed in the future to reduce their sim-to-real gap for this case and evaluate their effect on other robotic platforms.

References

- Levine, S., Pastor, P., Krizhevsky, A., Ibarz, J., Quillen, D.: Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int. J. Robot. Res.* **37**(4–5), 421–436 (2018). <https://doi.org/10.1177/0278364917710318>
- Gu, S., et al.: Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE (2017)
- Kalashnikov, D., et al.: Scalable deep reinforcement learning for vision-based robotic manipulation. In: Conference on Robot Learning. PMLR (2018)
- Towers, M., et al.: Gymnasium (2023). <https://zenodo.org/record/8127025>
- Raffin, A., et al.: Stable-baselines3: reliable reinforcement learning implementations. *J. Mach. Learn. Res.* **22**(1), 12348–12355 (2021)
- Todorov, E., Erez, T., Tassa, Y.: Mujoco: a physics engine for model-based control. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE (2012)
- Makoviychuk, V., et al.: Isaac gym: high performance GPU-based physics simulation for robot learning. arXiv preprint [arXiv:2108.10470](https://arxiv.org/abs/2108.10470) (2021)
- NVIDIA. Isaac Sim Robotics Simulator. <https://developer.nvidia.com/isaac-sim>. Accessed 16 Apr 2024

Author Index

A

- Abe, Fumiaki 180
Abello, Carola 43
Abidi, Haider 31
Alameh, Mohamad 53
Albanese, Giulia A. 25
Albardaner, Jaume 344
Albrecht, Stefano V. 120
Alexiou, Dimitrios 247
Alirezaei, Mohsen 108
Alt, Benjamin 274
Ambriola, Vincenzo 87
Amesegher, Ismail 143
Anil Meera, Ajith 103
Arapakis, Ioannis 108
Arzel, Matthieu 143
Auddy, Sayantan 315

B

- Bacci, Davide 87, 108
Bahrami, Bahador 87, 108
Baizid, Khelifa 31
Barbosa, Nancy 8
Bargellini, Davide 333
Bartelt, Stefanie 37
Becchio, Cristina 108
Belcamino, Valerio 53
Benini, Luca 286, 292
Bergner, Sebastian 315
Bernardini, Alessandra 327
Berselli, Giovanni 25, 43
Bertонcelli, Filippo 158
Beschi, Manual 224
Bonetto, Edoardo 280
Brevi, Daniele 280
Brisse, Milan 37
Buono, Chiara 131
Buoso, Davide 148
Burrello, Alessio 304
Buzzetti, Giulia 81

C

- Caccavale, Riccardo 200
Califano, Federico 338
Caliskanelli, Ipek 180
Campoy, Pascual 92
Cancelliere, Francesco 120, 207, 269
Cannella, Ferdinando 31
Caporali, Alessio 321, 327
Carfi, Alessandro 53
Carlotti, Nicholas 310
Carlucio, Ignacio 120
Casadio, Maura 25
Casas, Jean Carlos Quito 148
Castellano, Ginevra 108
Castellini, Fabio 3
Ceni, Andrea 87
Cereda, Elia 298
Charchalakis, Periklis 180
Cherif, Amel 25
Chiocchetti, Ezio 280
Chopra, Nikhil 170, 175
Cianchetti, Matteo 48
Cimini, Sara 252
Coleman, Eric Nuertey 114
Conti, Francesco 286
Corbato, Carlos Hernandez 108
Cossu, Andrea 87
Cremasco, Simone 3
Crupi, Luca 298
Curado-Soriano, Javier 263

D

- D'Haro, Luis Fernando 108
D'Imperio, Mariapaola 31
Daghero, Francesco 304
Dalmau, Magí 344
De Caro, Valerio 87
De Gara, Laura 252
de Jong, Jan 64
De Momi, Elena 43
Del Bianco, Edoardo 58

Della Santina, Cosimo 87, 108

Deroy, Ophelia 87, 108

Di Stefano, Giovanni 224

Diguet, Jean-Philippe 143

Dimarogonas, Dimos V. 108

Divet, Yann 195

Djedilbaev, Tim 98

E

Egel, Robert 37

Ehrhardt, Robin 48

Elamrani, Aïda 108

Espinosa, Nahuel 126

Espinosa, Paul 8

F

Farinelli, Alessandro 195

Fassio, Davide 148

Fausti, Roberto 224

Fernandez-Cortizas, Miguel 92

Ferre, Manuel 8

Finzi, Alberto 200

Fiorini, Edoardo 3

Foggia, Pasquale 212

Fontana, Eleonora 31

Fradi, Hajar 143

G

Galassi, Kevin 321

Gallicchio, Claudio 87, 108

García, Néstor 344

García, Olmer 126

Gasulla, Manel 108

Gatti, Fabrizio 280

Gavilanes, Guido 280

Ghosh, Arabinda 108

Giakoumis, Dimitrios 247

Giusti, Alessandro 310

Goodliffe, Matthew 180

Govoni, Andrea 333

Grogan, Helen 48

Guastella, Dario C. 120, 207

Guastella, Dario Calogero 269

Guazzotti, Marco 25

Guglieri, Giorgio 148

Guidotti, Riccardo 87

H

Haesaert, Sofie 108

Hauert, Sabine 87, 108

Henkel, Christian 137

Heravi, Farshad Nozad 31

Hermann, Andreas 274

Herr, Gurtajbir Singh 175

Hidalgo, Michel 126

Hofmann, Michael 236

Hurtado, Julio 114

I

Iacca, Giovanni 81, 108

Ikeda, Markus 19, 218, 236

Indurkhyà, Bipin 185

J

Jäkel, Rainer 274

Jones, Simon 87

K

Kamal, Nabeel 43

Kaper, Floris 64

Karpus, Jurgis 87

Kartsch, Victor 292

Kashiri, Navvab 58

Katic, Darko 274

Katsatos, Dimitrios 247

Kchir, Selma 137

Keßner, Urs 274

Klauck, Michaela 137

Kontodina, Theodora 247

Kostavelis, Ioannis 247

Kuhlenkötter, Bernd 37

L

Lahoud, Marcel 31

Lamberti, Lorenzo 286

Lange, Ralph 137

Lanillos, Pablo 103, 108

Lanzarini, Matteo 258

Lauretti, Clemente 252

Lee, Suet 87

Leiva, Luis A. 108

Liard, Clémence 143

Lin, Xiao 230

Lippiello, Vincenzo 200

Liso, Adriano 224

- Liu, Jingyue 87
Liu, Tianchen 170
Lomonaco, Vincenzo 87, 114
Longo, Nicola 224
Lucci, Niccolo 241
- M**
- Macii, Enrico 304
Makhdoomi, Mohatashem Reyaz 164
Malascorta, Donato 13
Mantovani, Mattia 153
Manzi, Guido 108
Maragos, Petros 48
Marchello, Gabriele 31
Marconi, Lorenzo 258
Mariano, Federico 43
Marino, Francesco 148
Martinez, Carol 164
Mastrogeorgiou, Athanasios 48
Mastrogiovanni, Fulvio 53
Mattos, Leonardo S. 43
Meattini, Roberto 327
Meddahi, Amal 31
Meertens, Nadine 87
Melchiorri, Claudio 327
Meli, Daniele 195
Mendizabal-Arrieta, Iñigo 230
Messina, Claudio Giuseppe 148
Miguel, Alberto San 344
Miloch, Elías 37
Milner, Emma 87
Minervini, Alessandro 148
Minotti, Matteo 280
Mitteramskogler, Johann J. 19
Möhl, Philipp 236
Molina, Martin 92
Monguzzi, Andrea 13
Monreale, Anna 87
Montini, Elias 241
Morasso, Cristian 195
Motetti, Beatrice Alessandra 304
Mugnocco, Maddalena 25
Müller, Hanna 292
Muller, Rick 64
Muradore, Riccardo 3
Muralidharan, Vivek 164
Muscato, Giovanni 120, 207, 269

- N**
- Nakahara, Renan Picoreti 87
Nava, Mirko 310
Neumann, Gerhard 274
Nicola, Giorgio 224
Nitti, Massimiliano 224

- O**
- Ojer, Marco 230
Okken, Boi 64, 76
Olivares-Mendez, Miguel 164

- P**
- Pagliarani, Niccoló 48
Pagliari, Daniele Jahier 304
Palazzo, Simone 120, 207
Palli, Gianluca 321, 327, 333
Palmas, Matteo 137
Palossi, Daniele 298
Panicucci, Simone 224
Pantano, Matteo 321
Papadakis, Panagiotis 143
Papadopoulos, Evangelos 48
Parmiggiani, Alberto 70
Pastrone, Claudio 280
Paunovic, Ivan 126
Pedrochi, Nicola 224
Pérez-Grau, Francisco J. 263
Perez-Saura, David 92
Piatelli, Gianluca 43
Piater, Justus 315
Pichler, Andreas 19, 218, 236
Pippo, Irene 25
Poncino, Massimo 304
Porichis, Antonis 48
Pratheepkumar, Anish 218
Pratisoli, Federico 131, 153
Princiotto, Federico 280
Puga, Gerardo 126

- R**
- Ramirez, Mariano 87
Redondo, Violeta 8
Reitano, Luca 269
Renò, Vito 224
Risso, Matteo 304
Rocchi, Francesco 70
Rocco, Paolo 13, 241

Romata, Ileana 224
 Roozing, Wesley 76
 Rosa, Francesco 212
 Roveri, Marco 58
 Rutishauser, Georg 286

S

Sabattini, Lorenzo 131, 153, 158
 Sakue, Tomoki 180
 Sánchez-Urán, Miguel Á. 8
 Sandoval, Juan 164
 Santonico, Marco 252
 Sanz, Ricardo 92, 108
 Sarpietro, Riccardo E. 120, 207
 Scognamiglio, Vincenzo 200
 Secchi, Cristian 338
 Sessa, Salvatore 195
 Shahabi, Ebrahim 87
 Sienkiewicz, Barbara 185
 Sierra, Carles 108
 Sigg, Stephan 108
 Sisman, Burak 98, 108
 Skilton, Robert 180
 Soudjani, Sadegh 108
 Spampinato, Concetto 120, 207, 269
 Steels, Luc 108
 Stipidis, Elias 180
 Stölzle, Maximilian 87
 Stramigioli, Stefano 338
 Strano, Alessandro 120
 Stroeve, Sybert 108
 Sutera, Giuseppe 207, 269

T

Tamantini, Christian 252
 Tammaro, Antonio 230
 Taranovic, Aleksandar 274
 Tsagarakis, Nikos G. 58
 Tzafestas, Costas 48
 Tzovaras, Dimitrios 247

V

Vakalellis, Michalis 108
 van Ruitenbeek, Floris 76
 Vanderdonckt, Jean 108
 Vecchio, Giuseppe 120, 207
 Venezia, Antonio 224
 Vento, Mario 212
 Verschure, Paul 108
 Viguria, Antidio 263
 Vincze, Markus 218
 Visentin, Francesco 3

W

Weerakoon, Lasitha 170, 175
 Widmoser, Fabian 19

Z

Zanchettin, Andrea Maria 13, 241
 Zanella, Riccardo 333, 338
 Zappa, Isacco 241
 Zappetti, Davide 81
 Zenzeri, Jacopo 25
 Zollo, Loredana 252
 Zompanti, Alessandro 252