Heaven's Light is Our Guide

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Rajshahi University of Engineering & Technology, Bangladesh

Realistic Activity Recognition using Sensors with Deep Convolutional Neural Network

Author

Md. Al Siam
Roll No. 1603008
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology

Supervised by

Abu Sayeed
Assistant Professor
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology

ACKNOWLEDGEMENT

We would first like to express our gratitude to the Almighty Allah for providing us with the chance to complete our thesis work. I would like to express my profound gratitude, appreciation, and respect to my supervisor Abu Sayeed, Assistant Professor, Department of Computer Science and Engineering at Rajshahi University of Engineering and Technology, Rajshahi. He has not only provided me with the technical instructions, guidance, and papers I've needed to finish the work; he has also continuously encouraged me and offered support whenever he saw I needed it. His cordial supports helped me to successfully run the analyses, experiments and all related things. Whenever we were stuck in any complex problems or situation, he was there for us at any time of the day. Without his sincere care, this work not has been materialized in the final form that it is now at the present.

I am also grateful to all the respective teachers of Computer Science and Engineering, Rajshahi University of Engineering and Technology, Rajshahi for good valuable suggestions and inspirations from time to time.

Finally, I convey my thanks to my parents, friends, and well-wishers for their constant inspirations and many helpful aids throughout this work.

October 27, 2022
RUET, Rajshahi

Md. Al Siam

Heaven's Light is Our Guide

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Rajshahi University of Engineering & Technology, Bangladesh

CERTIFICATE

*This is to certify that this thesis report entitled **"Realistic Activity Recognition using Sensors with Deep Convolutional Neural Network"** submitted by **Md. Al Siam, Roll:1603008** in partial fulfillment of the requirement for the award of the degree of Bachelor of Science in Department of Computer Science & Engineering of Rajshahi University of Engineering & Technology, Bangladesh is a record of the candidate own work carried out by him under my supervision. This thesis has not been submitted for the award of any other degree.*

Supervisor

Abu Sayeed
Assistant Professor
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology
Rajshahi-6204

External Examiner

Emrana Kabir Hashi
Assistant Professor
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology
Rajshahi-6204

ABSTRACT

The goal of sustainable work is to enhance working circumstances so that employees can significantly extend their working lives. In this environment, workplace safety and health are crucial issues, particularly in labor-intensive industries like construction-related professions. The Internet of Things and wearable sensors offer unobtrusive technologies that could improve safety by detecting human movement and potentially better working conditions and health. The research community does not, however, have access to widely accepted standard datasets that support realistic and varied user activity performed in working environments. In this work, we have taken a time series sensor dataset, namely VTT-ConIoT, which includes 16 different activities generally performed in construction sites performed by 13 different subjects. We have processed the data in such way that it can be classified with 2D-CNN based deep learning model. We have also experimented with a 2D-CNN based deep learning model to classify our processed dataset and gained a classification accuracy of 75.43% using our deep learning based network, which is comparable with the performance of the classic machine learning based methods. We depict that deep learning based 2D-CNN method is prospective to detect human activity with sensor data from real work environment which is important to ensure workplace safety.

CONTENTS

CHAPTER 4

CHAPTER 5

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Introduction

In this chapter, an overview of the entire undergraduate thesis is provided. The chapter begins with a brief explanation of the context and purpose for our research before defining the issue statement more precisely in terms of the scope of our study. The goals of our research project are then outlined, followed by a statement of the projected contribution. The structure of this thesis paper is then described. The final section provides a summary of this chapter.

1.2 Background

We have been accustomed to connecting with the digital world in a variety of ways during the past ten years. People frequently connect to the internet via touchscreens and other technological gadgets. The additional hassle of carrying around and removing tangible objects from one's pocket to interact with them comes with using cellphones and other physical devices. Next-generation technologies' main objective will be to deal with the need for physical intermediary devices like cellphones [1]. Virtual and augmented reality show up to be paving the way for the next generation of such technology, with output projected directly into the eyes of the user(s) via specialized glasses [2]. Additionally, due to the low cost, small size, and low power consumption, almost all intelligent gadgets have been incorporated with inertial measurement units as microelectronics technology has improved. Modern smart devices have three primary sensors built in to detect motion: an accelerometer, gyroscope, and magnetometer.

Sustainable work is defined by EUROFUND [3] *as a set of practices that aims at achieving living and working conditions that meet the needs of the workers in a durable and lasting way that does not compromise their current or future working life.* In this scenario, it is necessary to change the working environment in order to get rid of the things that discourage people from entering or staying in the job. Workplace safety and wellbeing are high concerns in the labor-intensive construction sector in order to establish fully sustainable work. Nonfatal and fatal accidents at construction sites [4] and musculoskeletal conditions that impair workers' ability to do their jobs properly [5] are two of the main causes of this concern. They have highly apparent impacts that lead to extended absences and even early retirement.

The negative impacts might be severe for the employer and society in addition to the decreased well-being of specific employees. The current situation clearly affects the work's potential to be sustained on an economic and social level. Each year, construction-related accidents cost the global economy hundreds of billions of euros [6]. For instance, it was determined that the economic expenses of occupational illnesses and injuries in the United Kingdom in 2018/2019 were approximately GBP 1.2 billion [7]. Given the increasing demographic and technological developments, employee well-being and safety has consequently become a strategic concern in all enterprises [8]. In consideration of the escalating levels of global competitiveness, the accelerated speed of work, and the extended working life, businesses in the construction industry regard, healthy, knowledgeable, and motivated professionals as their most valuable resource. Organizations invest in a variety of occupational safety programs to maintain sustainable employment, but these programs are not enough capable of support, evaluate, and quantify the effects of their guidelines and recommendations on an individual employee basis [9].

It is considered that the majority of absence-related factors are directly tied to employees' preferred methods for carrying out their tasks [10, 11]. However, implementing elaborate ergonomic solutions on building sites takes time and involves a lot of diverse players [12]. As a result, these solutions are not always simple to implement. Construction companies are searching for widely accepted solutions to continuously monitor staff safety and ergonomics, to reduce health risks, and to prevent negative outcomes [13].

Innovative sensor-based well-being technologies may make it easier to accurately monitor the activities and ergonomics of workers in real time. However, due to the serious effects on workplace safety and well-being as well as the ongoing issues, automatic detection and monitoring techniques are required. In this context, the proliferation of wearable technology and Internet of Things (IoT) devices offers a relatively unobtrusive technology that could improve safety by assisting in the observation of workers' actions and their compliance with ergonomic and risk-avoidance recommendations. Additionally, examining unusual behaviors like strolling slowly or maneuvering around objects could shed light on the conditions at the building site and enable the implementation of crucial safety precautions [11, 10].

A research area with several applications, human activity recognition (HAR) from wearable sensor data might give solutions that can be used in professional contexts like construction work. HAR primarily focuses on identifying movements or actions in areas that are generally unrestricted, using data collected from sensors worn by people while engaging in various activities. However, the scarcity of datasets that are sufficiently representative of the problem to examine makes HAR a difficult challenge in professional settings. The unique characteristics of construction sites—highly controlled, dynamic environments—present extra difficulties for the autonomous deployment and data collecting of HAR.

1.3 Motivation

Construction work safety might improve if IoT-based data-driven safety solutions replace conventional safety approaches and tools. Sensors, predictive analytics, high capacity communication infrastructures, and cloud computing are examples of IoT-based technologies that are currently becoming more prevalent across a variety of industries. IoT-based systems for enhancing workplace safety in labor-intensive industries like construction, however, face a few unique problems. These can be related to the complexity of building sites and a lack of understanding of the requirements that are unique to each site for IoT-based solutions.

In order to increase workplace safety and promote more environmentally friendly construction practices, accurate data must be gathered and used in a way that reveals how commonly accepted rules and guidelines for safety, security, and ergonomics are actually implemented. Therefore, it is crucial to present a comprehensive picture of the safety circumstances and how they have changed over time as the construction process advances.

The information that must be gathered can be categorized into three categories: The first step is the collection and real-time processing of sensor data that can produce personal alarms in real-time for potentially dangerous or unwise activities. The second step is the gathering and abstraction of actions for statistics that track compliance with established security, safety, and ergonomic guidelines in an anonymous manner. The utilization of recorded sensor data for accident forensic investigations, which enables a better understanding of the causes, comes in third [14].

Construction workers' actions may be automatically and precisely classified using inertial motion units (IMUs), resulting in continuous statistics at a level of granularity that is not yet achievable. Moreover, there are still a number of difficult issues to accurately and effectively detect and analyze the actions carried out in a building site. The sorts of activities—the activity classes are described in the context of the use of construction activity recognition. As a result, it can be challenging to identify specific activities because there are a wide range of options, even for single courses, on the conceivable list. Simple chores like painting or cleaning are made up of a variety of smaller tasks that are shared with other tasks like pushing things, walking, or climbing and descending stairs. Furthermore, such activities have unbalanced characteristics, such as the frequency with which they occur throughout the day, their normal length, and their starting timings. The method in which accuracy varies when we consider such imbalances is unknown because the standard approach often assumes that activity classes have similar probabilities of being performed, similar probabilities at all time of the day, and similar duration.

The activity classes are described in the context of the use of construction activity recognition. As a result, it can be challenging to identify specific activities because there are a wide range of options, even for single class, on the potential list. Simple activities like painting or cleaning are made up of a variety of smaller tasks that are shared with other tasks like pushing things, walking, or climbing and descending stairs. Furthermore, such activities have unbalanced characteristics, such as the frequency with which they occur throughout the day, their normal length, and their starting timings. The method in which accuracy varies when we consider such imbalances is unknown because the standard approach often assumes that activity classes have similar probabilities of being performed, similar probabilities at all time of the day, and similar duration.

1.4 Internet of Things

The term “Internet of things” (IoT) refers to physical objects (or groups of such objects) equipped with sensors, computing power, software, and other technologies that communicate with one another and exchange data through the Internet or other communications networks. The term ”internet of things” has been critiqued because devices only need to be individually addressable and connected to a network, not the whole internet. The fusion of numerous technologies, such as ubiquitous computing, widely available sensors, sophisticated embedded systems, and machine learning, has caused the sector to advance. IoT can provide provides businesses with a real-time look, enables companies to automate processes, reduces labor costs, cuts down on waste and improves service delivery and can give many other utilities. IoT is one of the most significant modern technologies, and it will only gain momentum as more companies see how linked gadgets can help them stay competitive.

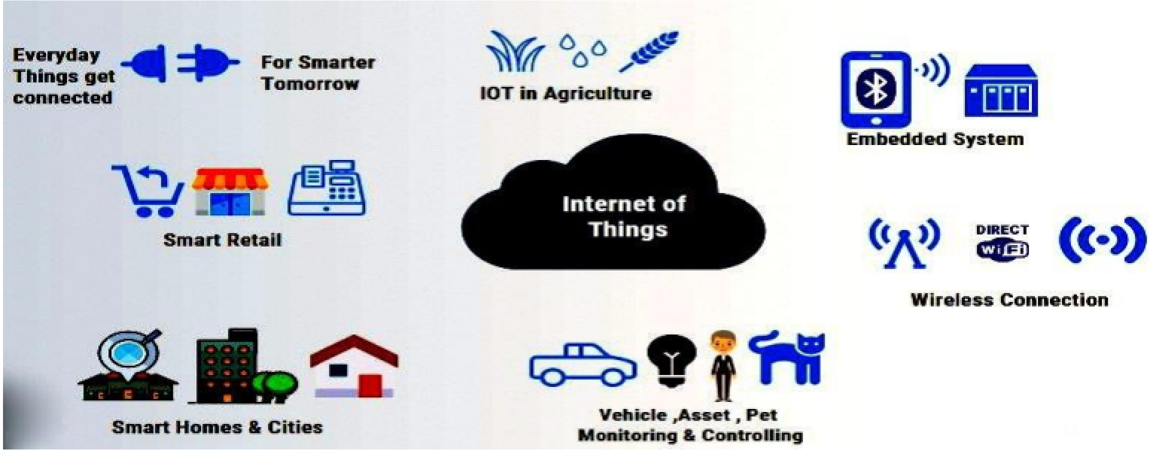


Figure 1.1: Exampes of IoT in Daily Life

1.5 Sensor Data

A sensor is a tool that detects events or changes in its surroundings and transmits that data to other electronics, often a computer processor. Always utilize sensors in conjunction with other electronics. Sensor data is the output of a device that detects and responds to some type of input from the physical environment. Sensor data may typically be displayed in tabular form since they typically consist of just a few values that indicate various physical characteristics. Smartphones with built-in sensors are increasingly commonplace today. By developing a mobile application for the data collection process, sensor data can be collected and preprocessed in accordance with user requirements. Hence, the domain of sensor datasets created by cellphones is increasing greater day by day.

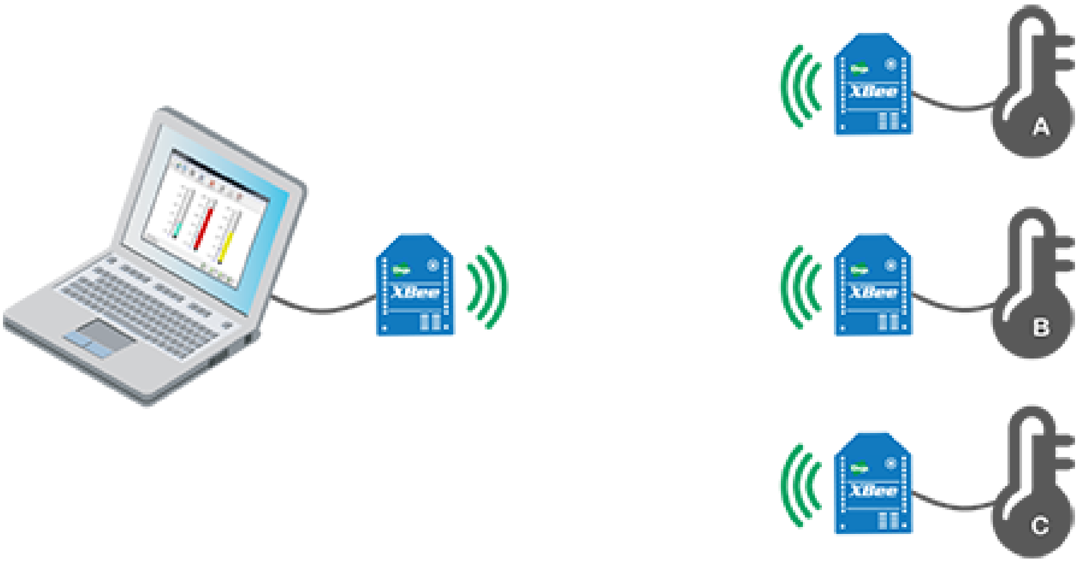


Figure 1.2: Example of Data Collection of IoT Sensors

1.6 Problem Statement

Construction sites have been considered to be one of the most risky working places. As the discussion on working site safety is emerging day by day, construction sites come in picture too. Internet of things (IoT) is being profoundly available in the day to day activities. The computation power is also massively increasing. We have taken the VTT-ConIoT dataset, which is consisted of activities which are generally performed in construction sites. This is claimed to be the first dataset created on the basis of construction site activities [14]. This is basically a time series dataset consisting of the accelerometer, gyroscope, magnotometer, barometer etc sensor signals. We aim to preprocess the dataset into activity segments, augment the data with overlapping techniques, and experiment on the performance of deep learning based model in the segmented activity data. The process is summarised in figure 1.3.

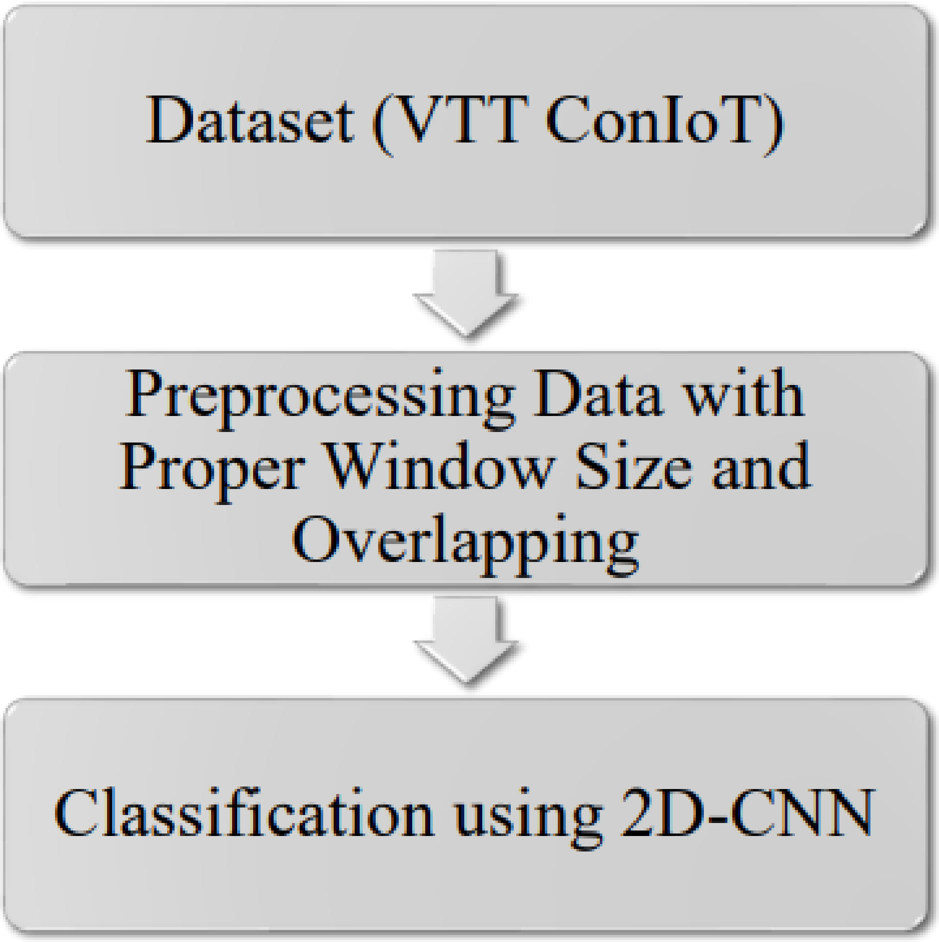

Dataset (VTT ConIoT)
Preprocessing Data with Proper Window Size and Overlapping
Classification using 2D-CNN

Figure 1.3: Summary of Experimentation Process

1.7 Objective of the Thesis

Basically, performing a preliminary study on Convolutional Neural Networks and it's implementation on IoT sensor dataset to build a system which can detect the type of human activity reducing unnecessary features of the data with the selection of a proper window size is the goal of this work. The objectives of this thesis work is as follows-

- Processing sensor data with proper windowing.
- Increasing the size of data with reasonable window overlapping.
- Constructing a deep 2D-CNN based model for classification.
- Using deep convolutional neural network on sensor data.
- Studying about the usability of sensor based dataset from real work environment.

1.8 Thesis Organisation

The rest of the thesis is organized as follows:

- **Chapter 2** is about the related works and background studies on sensor dataset, activity recognition problems with sensor data, activity recognition problems in professional contexts etc.

- **Chapter 3** contains the detailed dataset description, the curation methods of the dataset, and the details of the features of the dataset. The methods, concepts and ideologies behind the data preprocessing phase, model construction and classification are discussed later.

- **Chapter 4** is about the experiment results and analysis about the result. The results of validation and classification is depicted and compared to the results obtained from a classical machine learning based experiments. Later the analysis of the performance is discussed.

- **Chapter 5** is regarding the conclusion of this thesis work. The future scopes on the current work from various perspectives is depicted at the end.

1.9 Conclusion

As IOT sensors are becoming widely available day by day and the safety of the workers in their workplace is being an important concern, the use of IoT for ensuring the safety of the workers can be a great solution. The sensor data can be used to detect human activities. Human Activity Recognition (HAR) can be used to take safety measures in the workplace according to the activities performed by the people there.

Chapter 2

Background Studies

2.1 Introduction

Human activity recognition (HAR) from wearable sensor data has recently emerged as an area of study with a wide range of practical uses for both individuals and businesses. settings. HAR primarily focuses on identifying movements or activities in areas that are generally unrestricted utilizing data from wearable, sensor-based devices that individuals wear while engaging in various activities. These gadgets produce data about the user's physical activity using a combination of in-device processing and cloud services, offering them various context-adaptive services.HAR from sensors has demonstrated to deliver reasonably precise performance and great utility in specialized or semi-controlled conditions. However, creating reliable classifiers for spotting various activities is a difficult process that calls for a substantial amount of labeled training data gathered specifically for the context of interest.

In this chapter, the studies done regarding activity recognition are discussed. We have focused on the works of activity recognition using data from inertial sensor devices. Also we looked for the activity recognition problems studied in professional environments and similar context.

2.2 Related Works

2.2.1 Activity Recognition Using Inertial Sensors

The use of acceleration signals has been a major focus of earlier HAR studies based on inertial measurement units (IMUs) [15, 16]. Accelerometers are compact regarding their weight [17]. Rather, they are cheap to produce [18]. As they get integrated in many day-to-day use devices, they are widely available and affordable [19, 16]. Prior research in IMU-based sensors typically takes a multi-step approach that includes the collection and annotation of a subset of a sensor signal, the summarisation of the information in the subset using various signal features, and an instantaneous classification of the physical activity using machine learning [20, 21].

Although the approaches used across applications differ, they may generally be classified according on their response time (offline vs real time) and learning methodology (supervised vs. semi-supervised) [22] The feature extraction and summarization approaches determine whether Human Activity Recognition (HAR) systems are based on learned or handcrafted features. A wide range of statistical features, frequency-based features, or particular features based on human motion models, which are often referred to as physical features, are frequently included in the features of handicrafts that are arbitrarily selected [23]. While learnt features, which are typically based on feature selection strategies from a huge number of features, are the opposite, they are selection scheme based [23], even they can be on the applications of deep learning techniques [24]. Support vector machines, Gaussian mixture models, tree-based models like random forests, and hidden Markov models are examples of common classifiers which are based on Machine Learning [22]. The newest methods incorporate numerous processes in end-to-end systems [25].

Activity recognition with sensor data has been broadly studied with activities of writing characters. The gesture of human when writing in the air or free space is known as air-writing, which is widely studied in the sensor dataset literature. Liu et al. used accelerometer signal which was captured from a Wii remote and recognized a predefined set of eight gestures using a DTW-based method [26]. The DTW-based method is a well-studied approach to deal with time series data and air-writing recognition [27, 28, 29]. Chen et al. commented air-writing as better than virtual keyboards in typing accuracy [30, 31]. The authors also looked into identifying the beginning and end of each letter by the segments of the writing signal in a continuous data stream. Wii remote was also used by Xu and Xue where the users were given instructions about the order of movement for each of the air-written letters [32]. Li et al. used mobile phone captured motion signals performed by users and an LSTM-based deep neural network architecture to differentiate among twelve different handwritten characters consisting of six upper case letters and six digits [33]. Since users perform air-writing by hand, the signals received from palm-worn devices may lead it to be harder to recognize the activities [34]. However, several studies have depicted that it is possible to classify the gestures from palm-worn devices [35, 36]. Amma et al. recognized air-written letters with high accuracy using motion sensors positioned on the palm [37]. Xu et al. recognized textual input from wrist sensor data obtaining an accuracy of 98% [38]. Lin et al. investigated the orientations of the surfaces in which users were to write characters, the stabilization (support) point of the hand, and the rotation injection technique for data augmentation which uses a rotation matrix [39]. They obtained a remarkable high accuracy of 99.99% to recognize 62 characters by 10 subjects with a machine learning based approach.

Chen et al. investigated real-time fingertip detection in frames captured from smart glasses. They built a synthetic dataset using Unity3D and proposed a modified Mask Regional Convolutional Neural Network. Their method could detect fingertip for air-writing in a minimal time for each frame [40]. Kim et al. experimented with the WiTA dataset which contains air-writing data for Korean and English alphabets collected by RGB camera [41]. Bastas et al. experimented with handwritten digits, ranging from 0 to 9, which are structured as a multidimensional time series data obtained via a Leap Motion Controller (LMC) sensor [42]. Tsai et al. suggested a reverse time-ordered algorithm to efficiently filter out unnecessary lifting strokes while writing in the air. To overcome the problem of different writing styles of different users, a tiered arrangement structure is presented by sampling the air-writing results with varied sample rates [43]. Arsalan et al. suggested an air-writing system based on a network of sparse radars and a 1D DCNN-LSTM-1D transposed DCNN architecture that can rebuild and identify the drawn character [44]. Moazen et al. attempted to recognize air-writing with a dataset containing 100 sets of samples of all 26 English letters collected from a single subject [45]. Uysal et al. proposed RF-Wri, a device-free machine learning-based air-writing recognition framework that can differentiate 26 capital letters [46]. Yanay et al. allowed the users to write with their hands in the air naturally while capturing the motion signals by smart-bands [34]. In this experiment, the accelerometer and gyroscope signals were collected from the smart-bands to create a dataset of 15 sets of English alphabet for 55 subjects each. Finally, an average accuracy of 83.20% with the user-independent method and 89.20% with the user-dependent method was obtained in their experiment.

2.2.2 Activity Recognition in Professional Contexts

Over the years, HAR has concentrated on recognizing common everyday behaviors in outdoor and indoor environments, such as walking, driving, sitting, or lying down. Sports environments, where the sort of action combined with utmost accuracy is of great interest, have received particular consideration. However, the acknowledgement of activities in professional contexts received very little attention up until recently. Tracking the actions of medical professionals, such as doctors and nurses, as well as, on a smaller scale, activities like cooking, appears to be gaining more and more attention [47, 48].

Only a few small-scale studies are known when examining activity recognition with regard to construction activities in particular [14]. In a lab and small-scale context, Joshua and Varghese used accelerometers to study brickwork operations. According to their study, in contexts with few constraints, classification accuracy can reach up to 80% [49]. Only two subjects were employed in Akhavian and Behzadan's simulation of a three-class construction activity setup that includes sawing, hammering, turning a wrench, and loading and unloading tasks. Their three-class approach achieved accuracy levels close to 90%, with significant user variability and professions [50]. The rest of the experiments regarding construction focused on complementary cases without direct human interaction, such as tracking the activity of particular equipment , or machinery [51, 52].

Mäkela et al. presented a novel dataset with the purpose of human activity recognition in construction sites. The subjects (n = 13) were given instructions to carry out a variety of tasks while dressed in sensorized clothing to simulate conditions found on a real construction site. The collection includes complimentary human posture and keypoint data from a fixed camera along with high resolution motion data from many IMU sensors. Each subject's data was meticulously annotated using both a broad six-class standard and a more detailed sixteen-class protocol, They also presented the first evaluation of the dataset with a machine learning based model [14].

2.3 Conclusion

Many works have been done regarding human activity recognition, daily activity recognition, medical activity recognition using IoT sensors and sensor data. Compared to that, activity recognition in real working environment is very low. Activity recognition in real costruction site is only claimed to be done by Makela et al. [14] with their self curated dataset namely VTT-ConIoT.

Chapter 3

Matereials and Methods

3.1 Introduction

In this chapter, we have briefly discussed about our used VTT-ConIoT dataset, along with the curation process and features. Then we have discussed about the preprocessing methods of the dataset such that the dataset fits our model. Later we have discussed about deep convolutional neural network model and about its constructive units. Furthermore, leaving one subject out (LOSO) technique, feature selection scheme and experimental setting are discussed.

3.2 Dataset Description

We used VTT-ConIoT dataset which is basically an activity recognition dataset where 16 activities were performed by 13 users. Raw sensor data from the VTT-ConIoT platform were preprocessed to assure measurement quality and coherence. In this situation, authors of the dataset applied an initial check of the sensor scales and orientations of the sensor inside the pocket (there are 4 possible orientations, with 2 orientations being considerably more likely). This makes sure that all sensors and recording sessions use the same data format. All sensor readings were calibrated to reflect the same axis and dynamic ranges based on the direction of the gravity vector for a static person and sensor, as seen in the Inertial Measurement Unit (IMU) signal and the value. This examination is especially relevant to the information gathered at the actual construction site.

Because of the characteristics of the hardware, the sampling rates between various signal modalities and the clocks of the sensors were not perfectly synchronized between those positioned in various areas of the body. Using timestamps and synchronization signals, dataset authors manually adjusted the clock offset between various sensors. In order to synchronize the data provided by the magnetometer (97 Hz) and gyroscope (97 Hz) to the accelerometer data (103 Hz), the authors of the dataset resampled the signals using linear interpolation to a increased sample rate of both of the first two signals. Despite the possibility of some minor impacts, the authors of the dataset claim that the impact of using frequential variables in classification is negligible for signals with a comparable sampling rate that are much greater than the significant repeating elements of human activity identification, seen as being in good under 6 Hz. The potential noise has just transient qualities and is not accumulated further as error because the IMU signals were used directly as acceleration vectors and not further integrated to produce absolute positions.

All of the subjects who took part in the data collection provided data for the continuous recordings made with the IMU sensors. As a result, the data files had to be divided into discrete sections for each person and activity. All of the activities shown in the VTT dataset begin with a control signal that consists of two consecutive steps, a feature that can be seen clearly in the movies as well as in the IMU signals as two spikes. The dataset authors synced the signals for each activity and user using these spikes, and eliminated the signal windows that contained such steps. A built GUI tool that utilised the annotations produced during the recording of the activities as an additional input was used to split data per user and per activity. The annotations were then precisely adjusted to match the precise start time positions of each activity segment using the GUI. The data were divided into CSV files based on these annotations.

Therefore, each of the 13 users performed 16 different activities. There has been produced a one minute signal per user per activity after the cleaning, resampling ans synchronising the dataset. The signal contains different sensor data which are depicted in tables 3.1, 3.2 and 3.3.Among the 13 participants, 10 were men and 3 were women, a gender distribution similar to what can be seen in typical construction sites in Europe. The subjects were wearing sensorized clothes that incorporated three (Inertial Measurement Unit) IMU sensors. One sensor located in the hip, and two other ones located near the shoulder of the non-dominant hand of each participant. The sensors are shown in figure 3.1. Figure 3.2 gives a representation of the first 15 activities as they were captured by the supplementary video data.

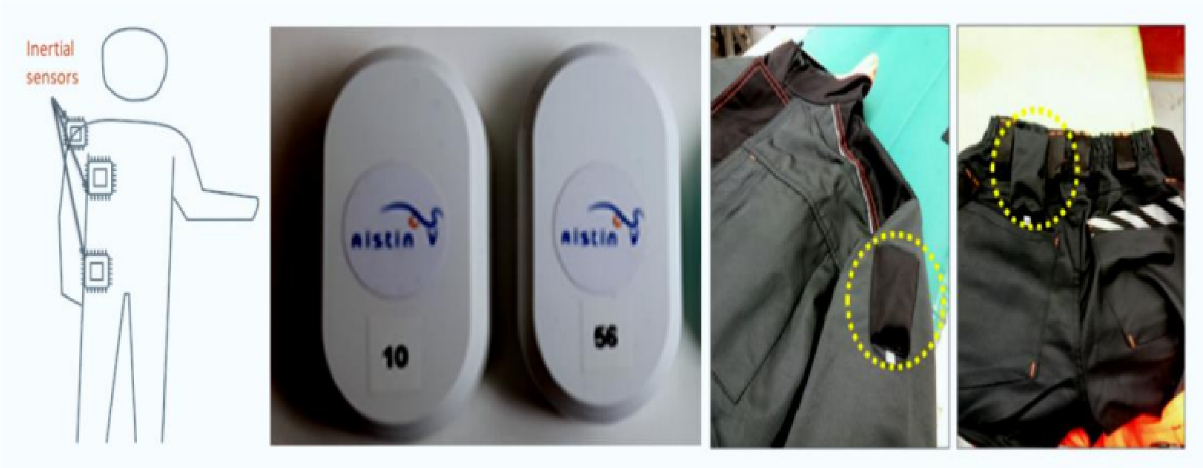


Figure 3.1: Depiction of the sensor locations. From left to right: planned locations, depiction of the actual sensors and example setup in the working clothes.

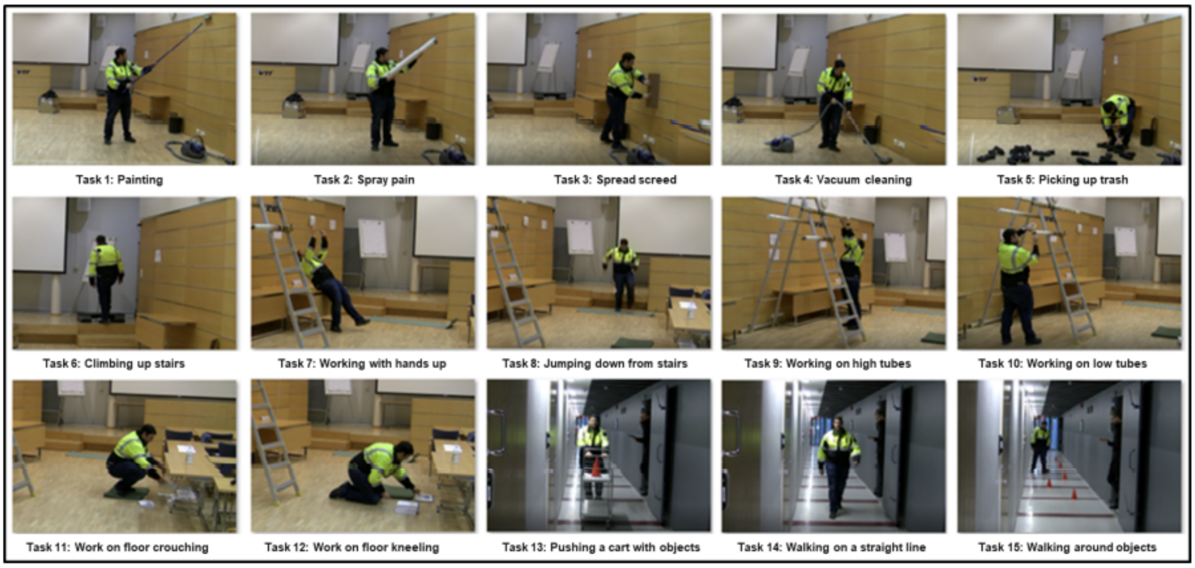

Task 1: Painting
Task 2: Spray pain
Task 3: Spread screed
Task 4: Vacuum cleaning
Task 5: Picking up trash
Task 6: Climbing up stairs
Task 7: Working with hands up
Task 8: Jumping down from stairs
Task 9: Working on high tubes
Task 10: Working on low tubes
Task 11: Work on floor crouching
Task 12: Work on floor kneeling
Task 13: Pushing a cart with objects
Task 14: Walking on a straight line
Task 15: Walking around objects

Figure 3.2: Example of the 16-activity setup from the VTT-ConIot dataset. Activity 16 (stairs) not depicted

The dataseet's objective was to portray the activities carried out at a construction site as realistically as possible and to replicate certain construction tasks that are important for construction safety. In this regard, the dataset authors have highlighted six of the most significant and usual duties carried out at a construction site in partnership with sectorial actors.There are a total of 16 actions spread over the 6 tasks. The representation of a few activities that might be rather

Table 3.1: Description of the Signal Data attributes Collected by the Sensors Set on User's Trousers

Signal Name	**Description**
datetime	timestamp starting from zero in format ”0 days 00:00.00.0000”. Generally not needed
trousers_index	trouser sensor original timestamp in index format. Generally not needed
trouser_Gx_dps	trouser sensor gyroscope/angular velocity in degrees per second, coordinate X
trousers_Gy_dps	trouser sensor gyroscopeangular velocity in degrees per second, coordinate Y
trousers_Gz_dps	trouser sensor gyroscopeangular velocity in degrees per second, coordinate Z
trousers_Ax_g	trouser sensor acceleration in (1/2) $\times Half\_gravity$ (coordinates relatives to sensor), coordinate X
trousers_Ay_g	trouser sensor acceleration in (1/2) $\times Half\_gravity$ (coordinates relatives to sensor), coordinate Y
trousers_Az_g	trouser sensor acceleration in (1/2) $\times Half\_gravity$ (coordinates relatives to sensor), coordinate Z
trousers_ts	trouser sensor timestamp in seconds. Generally not needed
trousers_Mx_uT	trouser sensor magnetometer in microTeslas, coordinate X
trousers_My_uT	trouser sensor magnetometer in microTeslas, coordinate Y
trousers_Mz_uT	trouser sensor magnetometer in microTeslas, coordinate Z
trousers_mbar	trouser sensor barometer data in milibars
trousers_tot_g	trouser sensor total acceleration computed as the norm of x,y,z coordinates of accelerometer
trousers_tot_dps	trouser sensor total angular velocity as the norm of x,y,z coordinates of gyroscope

Table 3.2: Description of the Signal Data attributes Collected by the Sensors Set on User's Back

Signal Name	**Description**
back_index	back sensor original timestamp in index format. Generally not needed
back_Gx_dps	back sensor gyroscopeangular velocity in degrees per second, coordinate X
back_Gy_dps	back sensor gyroscopeangular velocity in degrees per second, coordinate Y
back_Gz_dps	back sensor gyroscopeangular velocity in degrees per second, coordinate Z
back_Ax_g	back sensor acceleration in (1/2) $\times Half\_gravity$ (coordinates relatives to sensor), coordinate X
back_Ay_g	back sensor acceleration in (1/2) $\times Half\_gravity$ (coordinates relatives to sensor), coordinate Y
back_Az_g	back sensor acceleration in 1/2 $\times Half\_gravity$ (coordinates relatives to sensor), coordinate Z
back_ts	back sensor timestamp in seconds. Generally not needed
back_Mx_uT	back sensor magnetometer in microTeslas, coordinate X
back_My_uT	back sensor magnetometer in microTeslas, coordinate Y
back_Mz_uT	back sensor magnetometer in microTeslas, coordinate Z
back_mbar	back sensor barometer data in milibars
back_tot_g	back sensor total acceleration computed as the norm of x,y,z coordinates of accelerometer
back_tot_dps	back sensor total angular velocity as the norm of x,y,z coordinates of gyroscope

Table 3.3: Description of the Signal Data attributes Collected by the Sensors Set on User's Back

Signal Name	**Description**
hand_index	hand sensor original timestamp in index format. Generally not needed
hand_Gx_dps	hand sensor gyroscopeangular velocity in degrees per second, coordinate X
hand_Gy_dps	hand sensor gyroscopeangular velocity in degrees per second, coordinate Y
hand_Gz_dps	hand sensor gyroscopeangular velocity in degrees per second, coordinate Z
hand_Ax_g	hand sensor acceleration in (1/2)×$Half\_gravity$ (coordinates relatives to sensor), coordinate X
hand_Ay_g	hand sensor acceleration in (1/2) ×$Half_{g}ravity$ (coordinates relatives to sensor), coordinate Y
hand_Az_g	hand sensor acceleration in (1/2) ×$Half_{g}ravity$ (coordinates relatives to sensor), coordinate Z
hand_ts	hand sensor timestamp in seconds. Generally not needed
hand_Mx_uT	hand sensor magnetometer in microTeslas, coordinate X
hand_My_uT	hand sensor magnetometer in microTeslas, coordinate Y
hand_Mz_uT	hand sensor magnetometer in microTeslas, coordinate Z
hand_mbar	hand sensor barometer data in milibars
hand_tot_g	back sensor total acceleration computed as the norm of x,y,z coordinates of accelerometer
hand_tot_dps	hand sensor total angular velocity as the norm of x,y,z coordinates of gyroscope

common in informal or non-professional settings but are not advised on the construction site due to poor ergonomics or safety concerns is also included in this list. The six different tasks that group activities according to their tasks are as follows in table 3.5. Detailed descriptions of the 16 activities are as follows in table 3.4.

Each subject was given a brief introduction to the type of task they would be conducting when they arrived to the study setting by showing them a few sample pictures and videos of workers executing the activity. A carefully thought-out technique and system was used to sequentially gather the data from the 16 tasks for each of the participants (mock workers), as shown in Figure 3.3. This protocol made it easier to annotate the activities because it just needed a separate device to capture precise timestamps for the locations where the subject began the activities as directed. In the event that any a posteriori annotations were made in error, the complementary time-stamped video data allowed for possible adjustments.

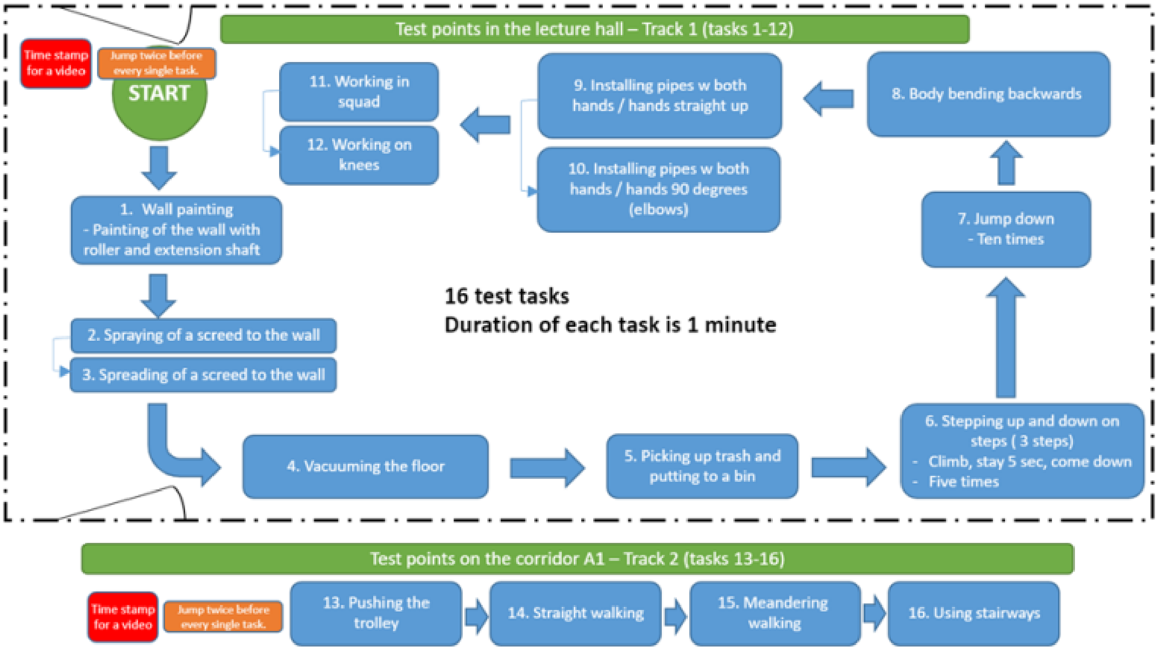

Test points in the lecture hall – Track 1 (tasks 1-12)
Time stamp for a video
Jump twice before every single task.
START
1. Wall painting - Painting of the wall with roller and extension shaft
2. Spraying of a screed to the wall
3. Spreading of a screed to the wall
4. Vacuuming the floor
5. Picking up trash and putting to a bin
6. Stepping up and down on steps (3 steps) - Climb, stay 5 sec, come down - Five times
7. Jump down - Ten times
8. Body bending backwards
9. Installing pipes w both hands / hands straight up
10. Installing pipes w both hands / hands 90 degrees (elbows)
11. Working in squad
12. Working on knees
16 test tasks
Duration of each task is 1 minute
Test points on the corridor A1 – Track 2 (tasks 13-16)
Time stamp for a video
Jump twice before every single task.
13. Pushing the trolley
14. Straight walking
15. Meandering walking
16. Using stairways

Figure 3.3: The data collection protocol, designed so the users can perform activities in a sequential manner

Table 3.4: Description of the Signal Data attributes Collected by the Sensors Set on User’s Back

Activity Name	**Description**
Roll painting	a subject uses a paint roll on a wall
Spraying paint	a subject uses a tube that mimics a machine to perform movements depicting the spraying of paint on
Leveling paint	a subject uses a tool to mimic the spreading of screed or paint on a wall
Vacuum cleaning	a subject uses a vacuum cleaner on the floor
Picking objects	a subject picks objects from the floor with their hands and throws them into a bin
Climbing stairs	a subject goes up 3 steps on a staircase, turns around and goes down 3 steps
Jumping down	a subject goes up 3 steps on a staircase, turns around and jumps down the 3 steps
Laying back	a subject mimics working with their hands up while laying back on a mid-level surface
HandsUp high	a subject mimics working on tubes with their hands up high above the head while laying back on a mid-level surface
HandsUp low	a subject mimics working on tubes with their hands up at the head or shoulder level
Crouch floor	a subject works on the floor, placing tiles while crouching
Kneel floor	a subject works on the floor, placing tiles while kneeling
Pushing cart	a subject walks on a corridor 20 m pushing a cart, turns around and pushes it back
Walk straight	a subject walks straight on a corridor 20 m, turns around and walks back
Walk winding	a subject walks winding around 7 cones, then 20 m, turns around, and walks back
Walk winding	a subject walks winding around 7 cones, then 20 m, turns around, and walks back
Stairs up-down	a subject climbs up stairs for 30 s, turns around and climbs them back down

Table 3.5: Activity Cluster Setup with Granular Activities

Activity Name	**Granular Activities**
Painting	Roll-Painting Spraying-Paint Leveling-Paint
Cleaning	Vacuum-cleaning Picking-objects
Climbing	Climbing-stairs Jumping-down Stairs-Up-Down
HandsUp	Laying-back HandsUp-high HandsUp-low
FloorWork	Crouch-floor Kneel-floor
WalkingDisplace	Walk-straight Walk-winding Pushing-cart

3.3 Data Preprocessing

In this work, we experimented with segmentation of sensor data with proper signal length so that the splitted signal has the proper properties of a single activity performed one time. Then we proceeded with the signals with deep learning based methods to identify if each of the segment can be classified properly.

3.3.1 Proper Signal Length Selection

As the datased is already interpolated at 103 Hz with Linear Interpolation and Makela et al. [14] found good results with the 5 seconds interval segments, we choose to use an one second interval for our experiment. As 5 second is a good interval to perform a single task from the activities we have taken into account.

Interpolation-based image processing techniques have been more popular recently because of their ability to improve even low-resolution photos while still keeping the image's properties. The interpolation algorithm that is used determines how well the image will turn out after processing [53]. There are numerous interpolation methods that have been created in the past. Most fields employ the nearest neighbor, bilinear, bicubic, and lanczos interpolation algorithms among them [54]. As our segmented signals will be a two dimensional matrix of size $signal\_length \times number\_of\_features$, we can consider our data segments as one dimensional image. Hence the interpolated data totally fits with our experimental setup.

3.3.2 Segmentation of Data

The data taken into account in this study is a total of 208 minutes, or roughly 16 minutes per subject, and are drawn from times when the users were engaged in the relevant activities. Here, each 1 minute signal is divided into sliding 5 seconds windows that are used as ”activity samples” in both the training and test sets. This equates to roughly 12,700 windows evenly dispersed over the 16 tasksactivities using a sliding period of one second. There are around 6000 (minimum 5558 and maximum 6000) signals per user per activity in our dataset.

3.3.3 Overlapping of Data

We used overlapping of data to increase our sample size. We used an one second overlapping. So the first sample was from 0 to 4 second, the second sample was from 1 to 5 second, the third sample was from 2 to 5 second, and so on. As our signal length was small and it was for only 1 minute or 60 seconds, there would be only 12 samples per activity per user, which would be pretty much low. Therefore overlapping or hopping came into picture. After implementing the hopping technique, data size was increased. Hence there were 260 samples per user per activity.

3.4 Convolutional Neural Network Architecture

Convolutional Neural Network, abbreviated as CNN, is a type of deep neural network for processing raw visual data, inspired by the organization of the visual cortex of animals [55, 56] and made to learn spatial hierarchies of features, from low-level to high-level patterns, automatically and adaptively using convolutional and pooling layers and activation functions. CNNs are widely employed in computer vision tasks. Classification, object localization and detection, segmentation, pose estimation to name a few. Lately, they have gained popularity in the research area of human activity recognition [57, 58, 59]. It has also been used for the classification of time series data obtained from accelerometers, gyroscopes and other sensors [60, 61, 62, 63]. We have proposed a convolutional neural network following the best practices that adopts well for the air-writing recognition utilizing time-series data from various sensors. A basic simple convolutional neural network architecture is depicted in fig. 3.4.

3.4.1 Convolutional Layer

A CNN is made up of many layers of neurons, each of which performs a nonlinear operation on the outputs of the layer before it in a linear transformation. Convolutional and pooling layers make up the majority of the layers. The pooling layers use a fixed function to alter the activation, but the convolutional layers include weights that must be learnt.The parameters of the filters of a convolutional layer must be learned. The filters are lighter and smaller in height than the input volume. An activation map composed of neurons is computed by convolving each filter with the input volume. In other words, the filter is moved across the input's width and height, and at

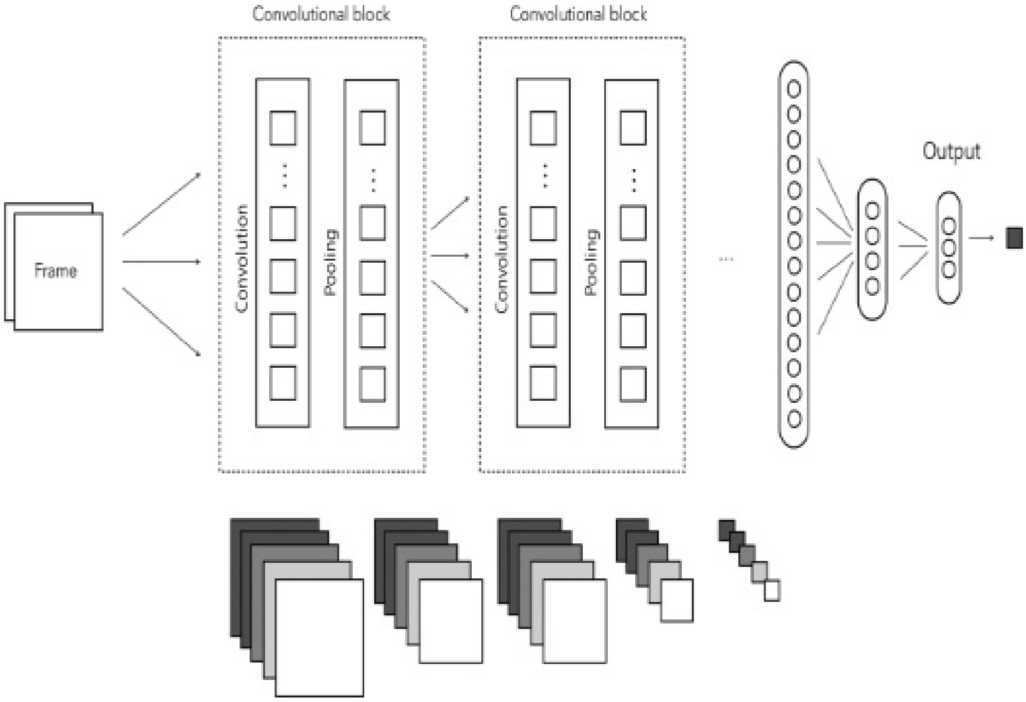


Figure 3.4: A Basic Simple Convolutional Neural Network Architecture

each spatial position, the dot product between the input and filter is computed. The activation maps of each filter are stacked along the depth dimension to create the convolutional layer's output volume. Each neuron in the activation map is only connected to a small local region of the input volume since the width and height of each filter is intended to be less than the input. In other words, each neuron has a small receptive field that is equal to the filter size. The architecture of the animal visual cortex , where the cell receptive fields are tiny, is what drives the local connection. Utilizing the spatial local correlation of the input, the convolutional layer's local connection enables the network to train filters that respond maximally to a local region of the input (for an input image, a pixel is more correlated to the nearby pixels than to the distant pixels). In addition, because the filter and input are convolutioned to create the activation map, the filter parameters are shared across all local positions. The number of criteria for effective communication, effective learning, and effective generalization are decreased through weight sharing. A sample convolutional layer is depicted in fugure 3.5.

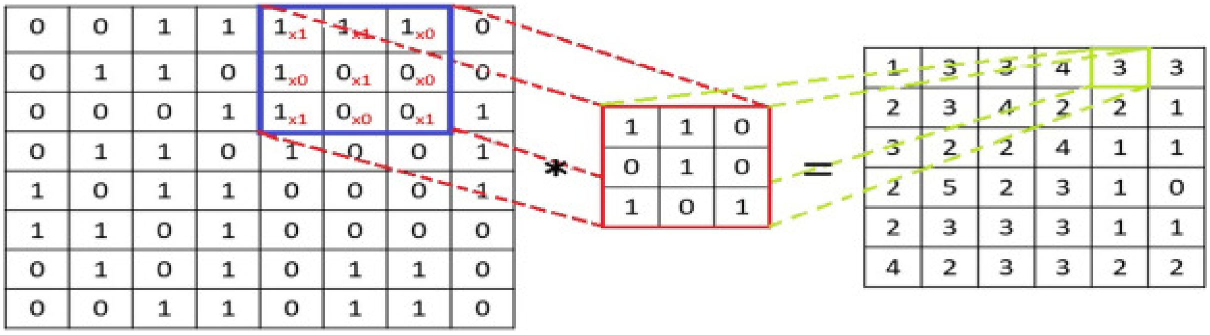


Figure 3.5: A Sample Convolution Process

3.4.2 Pooling Layer

To shrink the input's spatial size, the pooling layer is frequently added following a convolution layer. Each depth slice of the input volume receives an independent application of it. When pooling, volume depth is always maintained. Since the output volume of data depends on the values of the input volume of data, the pooling layer is not learned during the backpropagation of gradients. Different pooling layers come in different varieties.

Max Pooling: In this type of pooling, the maximum value of each kernel in each depth slice is captured and passed on to the next layer.

Min Pooling: In this type, the minimum value of each kernel in each depth slice is captured and passed on to the next layer.

L2 Pooling: In this type, the L2 or the Frobenius norm is applied to each kernel.

Average Pooling: In this type, the average value of the kernel is calculated.

An example of pooling is depicted in figure 3.6.

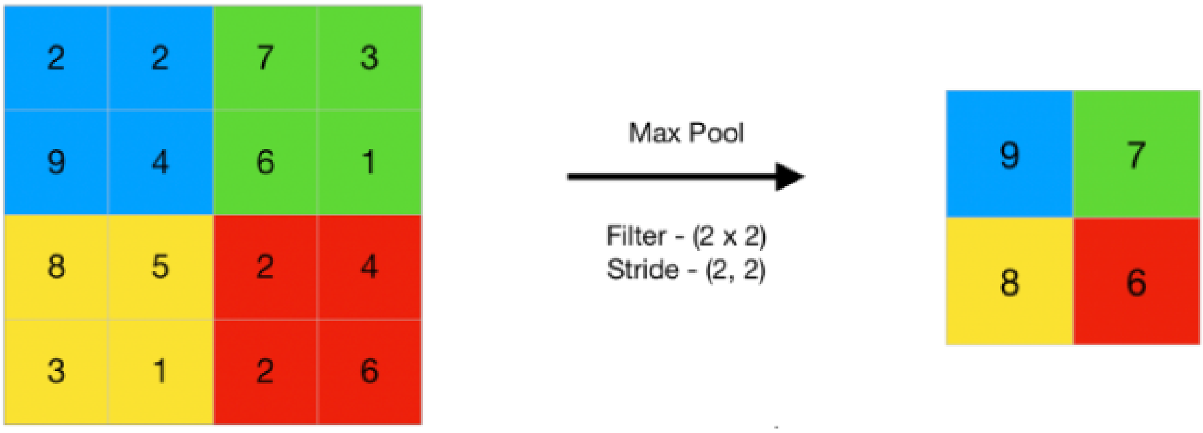


Figure 3.6: A Sample Pooling Process

3.4.3 Dropout Layer

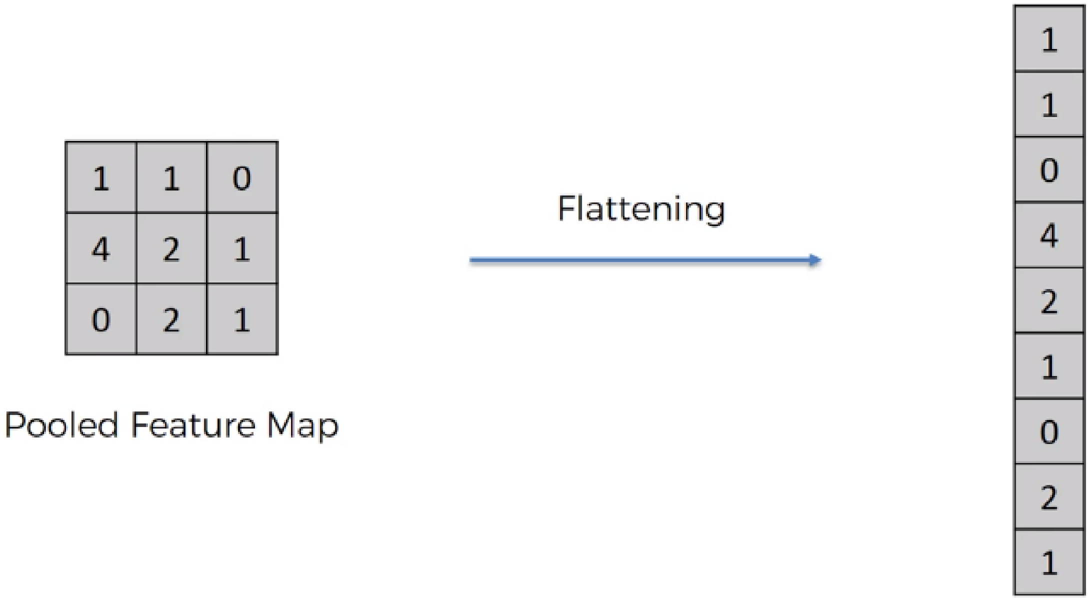
The Dropout layer is a mask that nullifies the contribution of some neurons towards the next layer and leaves unmodified all others. Dropout may be implemented on any or all hidden layers in the network as well as the visible or input layer. It is not used on the output layer. The term “dropout” refers to dropping out units (hidden and visible) in a neural network. Dropout is a simple way to prevent neural networks from overfitting. A new hyperparameter is introduced that specifies the probability at which outputs of the layer are dropped out, or inversely, the probability at which outputs of the layer are retained. The interpretation is an implementation detail that can differ from paper to code library. A common value is a probability of 0.5 for retaining the output of each node in a hidden layer and a value close to 1.0, such as 0.8, for retaining inputs from the visible layer. Dropout works well in practice, perhaps replacing the need for weight regularization (e.g. weight decay) and activity regularization (e.g. representation sparsity).

3.4.4 Flatten Layer

A flatten layer collapses the spatial dimensions of the input into the channel dimension. What happens after the flattening step is that we end up with a long vector of input data that you then pass through the artificial neural network to have it processed further. When we have many pooling layers, or we have the pooling layers with many pooled feature maps and then we flatten them. So, we put them into this one long column sequentially one after the other. And we get one huge vector of inputs for an artificial neural network. A sample flattening process is depicted in figure 3.7.

3.4.5 Fully Connected Layer

Convolutional Neural Networks (CNNs), which have been demonstrated to be particularly useful in detecting and classifying pictures for computer vision, must include fully linked layers. Convolution and pooling, which divide the image into features and perform independent analyses of each, are the first steps in the CNN process. The result of this process feeds into a fully connected neural network structure that drives the final classification decision. The clas-



1	1	0
4	2	1
0	2	1

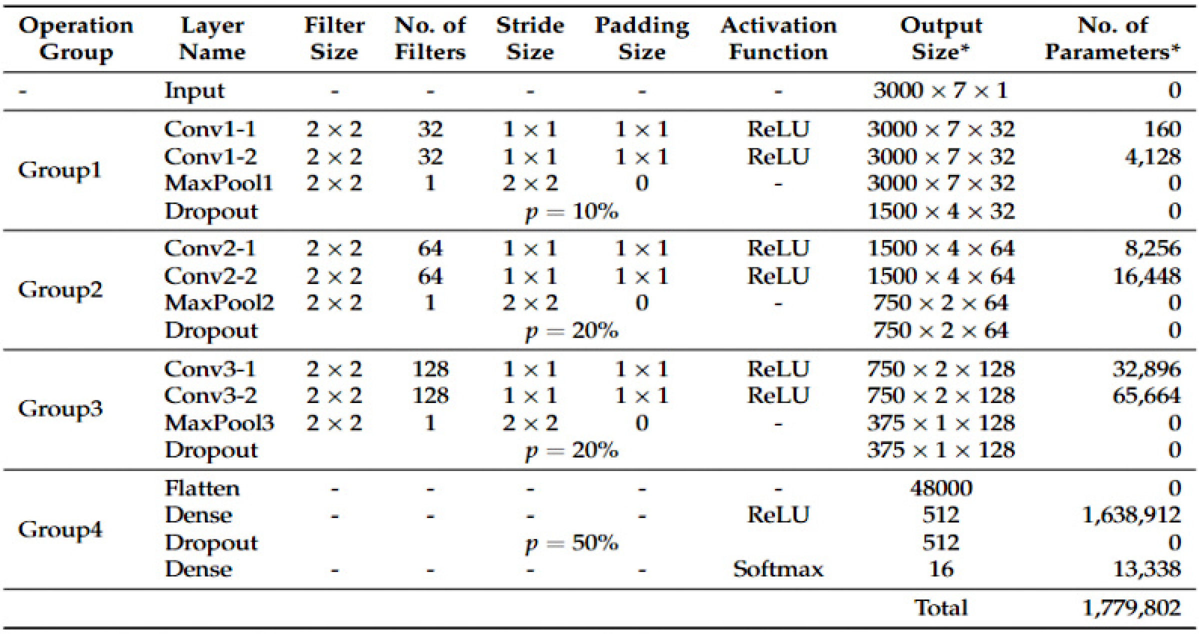
1
1
0
4
2
1
0
2
1

Figure 3.7: A Sample Flattening Process

sic neural network architecture was found to be inefficient for computer vision tasks. Images represent a large input for a neural network (they can have hundreds or thousands of pixels and up to 3 color channels). In a classic fully connected network, this requires a huge number of connections and network parameters. As part of the convolutional network, there is also a fully connected layer that takes the end result of the convolution/pooling process and reaches a classification decision. Fully connected layers in the neural network are used to perform high-level reasoning using several convolutional and max pooling layers continuously. As in conventional neural networks, neurons in a fully connected layer are coupled to every activation in the layer below. Activations can be calculated by applying a matrix multiplication followed by adding a bias.

3.4.6 CNN Architecture for Classifying VTT-ConIoT Dataset

Our proposed convolutional neural network architecture is composed of four groups of layers other than the input layer, where the first three groups consist of a couple of 2-dimensional convolution, maxpooling and dropout layers for feature extraction. We have flattened the output from the third convolutional group and a dense layer accompanying with dropout is employed



Operation Group	Layer Name	Filter Size	No. of Filters	Stride Size	Padding Size	Activation Function	Output Size*	No. of Parameters*
-	Input	-	-	-	-	-	$3000 \times 7 \times 1$	0
Group1	Conv1-1	2×2	32	1×1	1×1	ReLU	$3000 \times 7 \times 32$	160
	Conv1-2	2×2	32	1×1	1×1	ReLU	$3000 \times 7 \times 32$	4,128
	MaxPool1	2×2	1	2×2	0	-	$3000 \times 7 \times 32$	0
	Dropout			$p = 10\%$			$1500 \times 4 \times 32$	0
Group2	Conv2-1	2×2	64	1×1	1×1	ReLU	$1500 \times 4 \times 64$	8,256
	Conv2-2	2×2	64	1×1	1×1	ReLU	$1500 \times 4 \times 64$	16,448
	MaxPool2	2×2	1	2×2	0	-	$750 \times 2 \times 64$	0
	Dropout			$p = 20\%$			$750 \times 2 \times 64$	0
Group3	Conv3-1	2×2	128	1×1	1×1	ReLU	$750 \times 2 \times 128$	32,896
	Conv3-2	2×2	128	1×1	1×1	ReLU	$750 \times 2 \times 128$	65,664
	MaxPool3	2×2	1	2×2	0	-	$375 \times 1 \times 128$	0
	Dropout			$p = 20\%$			$375 \times 1 \times 128$	0
Group4	Flatten	-	-	-	-	-	48000	0
	Dense	-	-	-	-	ReLU	512	1,638,912
	Dropout			$p = 50\%$			512	0
	Dense	-	-	-	-	Softmax	16	13,338
							Total	1,779,802

Figure 3.8: Network architecture of the 2D-CNN model for construction site activity recognition based on VTT-ConvIoT dataset

$*$ No. of features and signal length, l, depending upon the dataset under consideration.

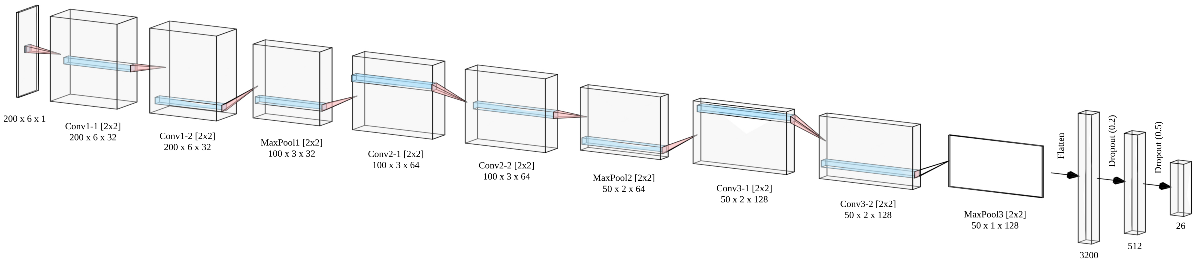


Figure 3.9: Visual Representation of CNN Architecture

with softmax activation function to get the prediction. Except for the prediction layer, we have used Rectified Linear Units (ReLU) as the activation function throughout the network.

The input of the network is the tensor of format: $l \times f \times 1$, where l is the signal length, f is the number of features (time-series signals) in the dataset. This tensor is therefore propagated through the convolutional layers. Each of the convolutional groups is constructed using conv-conv-maxpool-dropout layers, sequentially. The core objective for consecutive convolutional layers without pooling layers is to replace a single layer with a larger receptive field rather than skipping any pooling. It is a widely used construct for developing convolutional neural network [64, 65, 66, 67]. We incorporate two non-linear convolutional layers instead of a single one with a larger filter size to make the decision function more discriminative. Additionally, this approach decreased the number of trainable parameters [65].

Dropout has been an integral part of deep neural networks since its inception [68]. Wu and Gu studied the effects of dropout on different layers of CNN and showed that the dropout of maxpooling and fully-connected layers performed best [69]. Therefore, we have used dropout after every maxpooling layers and in the fully-connected layer where the percentage, p values of the dropouts was chosen according to the suggestions given by Park and Kwak [70]. The network architecture specification is provided in figure 3.8, considering the signal length is 200 and number of features is 6. The layers that construct the network and the attributes remain the same for all number of signals and number of features. A visual representation of CNN architecture is given in figure 3.9.

3.4.7 Activation Function

A neuron's activation status is determined by an activation function. By employing simpler mathematical procedures, it will determine whether or not the neuron's input to the network is significant during the prediction process. The activation function's objective is to add non-linearity to a neuron's output. Examples of different activation functions are linear activation funnction, sigmoid activation function, tanh activation function, softmax activation function.

We have used Rectified Linear Unit (ReLU) as the activation function in the convolutional layers of our deep CNN. If the input is positive, the rectified linear activation function, or ReLU for short, will output the input directly; if it is negative, it will output zero. Because a model that utilizes it is simpler to train and frequently performs better, it has evolved into the standard activation function for many different kinds of neural networks. The equation of ReLU activation function is defined as follows in equation 3.1 and a sample graph of a ReLU function is depicted in figure 3.10.

$$f(x) = max(0, x) \tag{3.1}$$

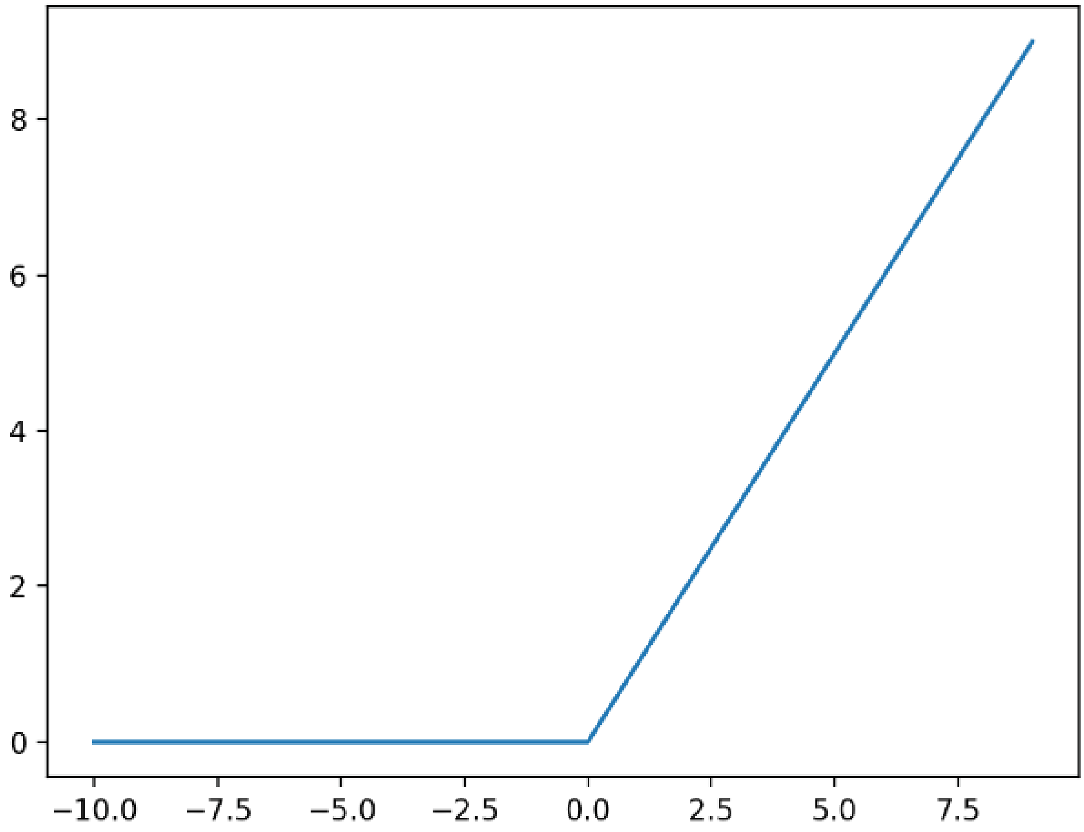


Figure 3.10: Rectified Linear Unit (ReLU) Activation Function

3.5 Leaving One Subject Out Method

Leave-one-subject-out (LOSO) cross-validation divides a dataset into a training set and a testing set, using all excluding one observation as part of the training set. Then it builds the model using only data which are from the training set. The model is used to predict the result value of the one observation that was left out. Despite the relative simplicity of the method, our baseline results, which are based on a LOSO evaluation scheme, show that generalization to various people is feasible. We test all of our models using a cross-validation method based on LOSO, where N distinct models are trained for N subjects. All subjects' data are utilized to train each model, with the exception of one, which is subsequently used for testing and calculating the model's performance.Thirteen models per classifier and modality was evaluated.

We measured the performance of the model by recognition accuracy (see Eq. 3.2) in LOSO principle. The evaluation metric is considered for multiclass classification as we have a various types of activities in our dataset. Note that, the dataset is mostly balanced regarding the number of samples per activity per subject where applicable. Therefore, evaluation of classification performance using only accuracy is justified.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \tag{3.2}$$

3.6 Feature Selection

It is clear that all of the features of the dataset is not important, such as, the datetime is not important to classify the activities. Studies show that the accelerometer, gyroscope are mostly used for human activity recognition. They provide measurement in three axis - X, Y and Z. Magnetometer is also important in that sense, as it provides three axis data. But magnetic resonance cannot classify human activities. Hence, we took the 21 features came from the accelerometer and gyroschope, 7 features from each sensor set - placed the trouser, back and shoulder.

3.7 Experimental Settings

Colaboratory, sometimes known as ”Colab,” is a Google Research product. Colab is particularly well suited to machine learning, data analysis, and education. It enables anyone to create and execute arbitrary Python code through the browser. Technically speaking, Colab is a hosted Jupyter notebook service that offers free access to computer resources, including GPUs, and requires no setup to use. To accelerate our training procedure, NVIDIA Tesla T4 GPU and 12 GB of RAM were used provided by Google Colaboratory free of cost [71]. We have used OpenCV library [72] to interpolate the time-series data and Keras API over TensorFlow backend [73] to create the CNN model.

3.8 Conclusion

VTT-ConIot is a sensor based dataset curated from real building construction working environment. The dataset needs to be preprocessed with suitable preprocessing method. Two dimensional deep convolutional neural network should be a suitable method for classification of sensor data, as the segmented sensor data contains image-like features.

Chapter 4

Result and Performence Analysis

4.1 Introduction

In this chapter, the results of the experimental analysis is mentioned. An analysis of the performances is also discussed later. The comparison between the machine learning based experiment performed by Makela et al. [14] and our constructed deep learning based 2D-CNN model is depicted and discussed.

4.2 Result

We have evaluated our proposed model with VTT-ConIoT dataset. In that dataset, there are 13 users performed 16 activities each, which are generally performed in the construction sites. We evaluated our model in leaving one subject (LOSO) out approach. Here, we trained our model 13 times, leaving a subject out each time. Then the model was evaluated on the excluded subject. Makela et al. [14] did this with their machine learning, in which they used some features directly and some statistically calculated features. We have made samples with selectively important features and fed them directly to the artificial neural network. After the models were trained and evaluated for each of the subjects, the accuracy scores of each of the models are averaged out. We call it as an average accuracy which can roughly be a performance measure of our model. The validation accuracy of the individual models are depicted in table 4.1, and the classification accuracy of the individual models are shown in table 4.2. The average validation accuracy is 67.08% and the average classification accuracy is 75.43%.

Table 4.1: LOSO (per-subject) validation results

Subject	**Makela et al. [14]**	**Our Deep CNN Model**
S1	49.50%	61.94%
S2	46.10%	68.39%
S3	41.50%	80.00%
S4	46.60%	64.52 %
S5	49.30%	71.61%
S6	51.80%	65.52 %
S7	62.30%	67.74 %
S8	56.70%	49.03 %
S9	39.20%	73.55%
S10	58.90%	83.87%
S11	56.00%	59.35%
S12	58.20%	61.94%
S13	62.10%	64.52%
Mean	**52.10%**	**67.08%**

* In this table, the comparison of the subject-wise validation results between machine learning based method performed by Makela et al. [14] and our 2D-CNN model is shown.

Makela et al. [14] experimented with a classical machine learning based method namely Random Forest classifier. Whereas, we have experimented with a deep learning based 2D-CNN model.

Table 4.2: LOSO (per-subject) classification results with the left-out subject data

Subject	**Classification Accuracy**
S1	61.94%
S2	69.38%
S3	83.23%
S4	74.19%
S5	69.38%
S6	89.66%
S7	93.55%
S8	65.16%
S9	74.84%
S10	83.87%
S11	66.45%
S12	71.61%
S13	77.42%
Mean	**75.43%**

* In this table, the subject-wise classification results has been shown. For every subject, the model is trained and and tested with the left-out subject data. After that, the mean of the classification accuracy is shown.

Makela et al. [14] tested with different machine learning based methods, namely, Support Vector Machine (SVM), Random Forest (RF), Extra Trees (ET), Linear Discriminant Analysis (LTA), Logistic Regression (LR), XGBoost (XGB). They also applied leaving one subject out method for testing every of the methods.

The best testing accuracy among all the classification methods of Makela et al. [14] is reported **78.00%** which is from the SVM classification method. Wheras, our deep 2D-CNN method could show an accuracy of **75.43%**.

4.3 Performance Analysis

The performances are competitive with the results of the machine learning based approach performed in [14]. They accuracies of the work of [14] and our work is pretty similar. The dataset we used to test our model is small. There are 13 users who performed 16 activities each. There are around 6000 signals per subject per activity. Even after using the hopping technique, i.e, overlapping the signals with a move forward of beginning of the corresponding signals of the previous time frame, there was around 2060 samples per subject per activity. Hence, our dataset is so small. So there was not very high performance which deep learning model tends to do for larger datasets. The validation accuracies and classification accuracies are pretty similar. It is notable that, for some subjects, the deep learning model performed better which can be seen in table 4.1 and table 4.2.

4.4 Conclusion

Deep learning based approach can perform a competitive performance compared to classical machine learning based method, but there should be enough data to feed in deep learning model to generalise the classification problem properly. With a relatively small dataset with large number of classes, we could achieve a 75.43% of accuracy.

Chapter 5

Conclusion and Future Works

5.1 Introduction

In this chapter, the conclusion of the thesis work has been given. What attempts and works can be done further regarding this work has been discussed at the end of this chapter.

5.2 Conclusion

We evaluated VTT-ConIoT, a realistic IMU-based dataset for construction worker activity identification that intends to improve job safety, ergonomics, and well-being by representing actions conducted in both recommended and unrecommended ways. This is the first sensor-based dataset that focuses on specific building operations. Unlike other similar datasets for various varied professional settings, the use of VTT-ConIoT was evaluated on a real construction site, assuring the closeness of the activities portrayed in the dataset and in real situations [14]. Data from many sensor placements and modalities are included in the dataset. Despite the relative simplicity of the method, our baseline results, which are based on a LOSO evaluation scheme, show that generalization to various people is feasible. We achieved a classification accuracy of 75.43% with our deep learning based approach, which is very close to the machine learning based approch. This proves the prospects of deep learning based approaches in construction site based activity recognition problems.

5.3 Future Works

5.3.1 Experimentation with Larger Datasets

The dataset we used to test our model is small. There are 13 users who performed 16 activities each. There are around 6000 signals per subject per activity. Even after using the hopping technique, i.e, overlapping the signals with a move forward of beginning of the corresponding signals of the previous time frame, there was around 2060 samples per subject per activity. Hence, our dataset is so small. Deep learning models generally learn well when large dataset is fed to the network. If large dataset is fed to the network, there are good number of variant data which helps the model to generalise the characteristics of the classes. For example, Abir et al. [74] recognised air-writing [45] dataset with a good accuracy using a similarly made deep learning model which had 100 samples per subject per activity. The number of activities and number of subjects were also larger. Hence, there could be made more samples and they were fed to deep learning model.

We have mentioned that there are scarcity of dataset about activities which are performed in the working sites, let alone the construction sites. So attempts can be made to make large dataset in working sites or constructions sites. Hence, the impact and prospects of deep learning model in recognising activities in real working site. There can be done taking data from real activity sites, which might incorporate noises to the taken signals. Hence there can be experimentation with noise cancellation methods to time series datasets tno, which is important to implement deep learning based real site activity recognition in production level.

5.3.2 Experimentation with Deeper Models

The model we have experimented with has three groups of sequential convolutional-convolutional-maxpooling-dropout layers. Then there is flatten layer, dense layer, dropout layer and another output dense layer, respectively. The model is, however, not big enough. There are models in image data classifications like Inception, VGG, MobileNet etc. These models are very deep and there are very large number of learnable parameters. Also, these models are trained on gigantic datasets. Hence, there have been produced deep learning models which are robust and can be used in production levels with feasible accuracy in new live samples. Hence, deeper and larger models have always the possibility to perform better. There can be used deeper models, also with another conventions of making deep learning model can be followed rather than construction of our model.

5.3.3 Combination of Machine Learning and Deep Learning Based Approaches

We have used the dataset prepared by Makela et al. [14]. They used machine learning based approach in which they used the features from the dataset, as well as some statistical features like average, median, variance, 25th percentile, 75th percentile, minimum and maximum. We have tried deep learning model in the dataset. Machine learning and deep learning model can be used at the same time to detect any activity, which can be then ensembled based on some features, weights or logics considering and analyzing the nature of the data. This can be used in production level where ensuring the correctness of the detection matters a lot.

REFERENCES

[1] C. Amma and T. Schultz, "Airwriting: Bringing text entry to wearable computers," *XRDS: Crossroads, the ACM Magazine for Students*, vol. 20, no. 2, pp. 50–55, 2013.

[2] T. Yanay and E. Shmueli, "Air-writing recognition using smart-bands," *Pervasive and Mobile Computing*, vol. 66, p. 101183, 2020.

[3] "Sustainable work." `https://www.eurofound.europa.eu/topic/sustainable-work`, Sep 2022.

[4] S. Winge and E. Albrechtsen, "Accident types and barrier failures in the construction industry," *Safety science*, vol. 105, pp. 158–166, 2018.

[5] X. Wang, X. S. Dong, S. D. Choi, and J. Dement, "Work-related musculoskeletal disorders among construction workers in the united states from 1992 to 2014," *Occupational and environmental medicine*, vol. 74, no. 5, pp. 374–380, 2017.

[6] K. Yang, K. Kim, and S. Go, "Towards effective safety cost budgeting for apartment construction: A case study of occupational safety and health expenses in south korea," *Sustainability*, vol. 13, no. 3, p. 1335, 2021.

[7] Nations, "U. sustainable development goals.." `https://www.eurofound.europa.eu/topic/sustainable-work`, 2022.

[8] W. T. Chen, H. C. Merrett, Y.-H. Huang, T. A. Bria, and Y.-H. Lin, "Exploring the relationship between safety climate and worker shttps://www.eurofound.europa.eu/topic/sustainable-workafety behavior on building construction sites in taiwan," *Sustainability*, vol. 13, no. 6, p. 3326, 2021.

[9] B. Hoła and T. Nowobilski, "Analysis of the influence of socio-economic factors on occupational safety in the construction industry," *Sustainability*, vol. 11, no. 16, p. 4469, 2019.

[10] J.-M. Kim, K. Son, S.-G. Yum, and S. Ahn, “Analyzing the risk of safety accidents: The relative risks of migrant workers in construction industry,” *Sustainability*, vol. 12, no. 13, p. 5430, 2020.

[11] J. Kim, S. Youm, Y. Shan, and J. Kim, “Analysis of fire accident factors on construction sites using web crawling and deep learning approach,” *Sustainability*, vol. 13, no. 21, p. 11694, 2021.

[12] A. M. Dale, L. Jaegers, L. Welch, E. Barnidge, N. Weaver, and B. A. Evanoff, “Facilitators and barriers to the adoption of ergonomic solutions in construction,” *American journal of industrial medicine*, vol. 60, no. 3, pp. 295–305, 2017.

[13] I. Park, J. Kim, S. Han, and C. Hyun, “Analysis of fatal accidents and their causes in the korean construction industry,” *Sustainability*, vol. 12, no. 8, p. 3120, 2020.

[14] S.-M. Mäkela, A. Lämsä, J. S. Keränen, J. Liikka, J. Ronkainen, J. Peltola, J. Häikiö, S. Järvinen, and M. Bordallo López, “Introducing vtt-coniot: A realistic dataset for activity recognition of construction workers using imu devices,” *Sustainability*, vol. 14, no. 1, p. 220, 2021.

[15] P. Gupta and T. Dallas, “Feature selection and activity recognition system using a single triaxial accelerometer,” *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 6, pp. 1780–1786, 2014.

[16] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, “Activity recognition using cell phone accelerometers,” *ACM SigKDD Explorations Newsletter*, vol. 12, no. 2, pp. 74–82, 2011.

[17] J. Petersen, D. Austin, R. Sack, and T. L. Hayes, “Actigraphy-based scratch detection using logistic regression,” *IEEE journal of biomedical and health informatics*, vol. 17, no. 2, pp. 277–283, 2013.

[18] W.-C. Cheng and D.-M. Jhan, “Triaxial accelerometer-based fall detection method using a self-constructing cascade-adaboost-svm classifier,” *IEEE journal of biomedical and health informatics*, vol. 17, no. 2, pp. 411–419, 2012.

[19] A. Matic, V. Osmani, and O. Mayora, “Speech activity detection using accelerometer,” in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 2112–2115, IEEE, 2012.

[20] J. Parkka, M. Ermes, P. Korpipaa, J. Mantyjarvi, J. Peltola, and I. Korhonen, “Activity classification using realistic data from wearable sensors,” *IEEE Transactions on information technology in biomedicine*, vol. 10, no. 1, pp. 119–128, 2006.

[21] M. Ermes, J. Pärkkä, J. Mäntyjärvi, and I. Korhonen, “Detection of daily activities and sports with wearable sensors in controlled and uncontrolled conditions,” *IEEE transactions on information technology in biomedicine*, vol. 12, no. 1, pp. 20–26, 2008.

[22] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, and Y. Amirat, “Physical human activity recognition using wearable sensors,” *Sensors*, vol. 15, no. 12, pp. 31314–31338, 2015.

[23] M. Zhang and A. Sawchuk, “A feature selection-based framework for human activity recognition using wearable multimodal sensors,” in *6th International ICST Conference on Body Area Networks*, 2012.

[24] F. Li, K. Shirahama, M. A. Nisar, L. Köping, and M. Grzegorzek, “Comparison of feature learning methods for human activity recognition using wearable sensors,” *Sensors*, vol. 18, no. 2, p. 679, 2018.

[25] M. M. Hassan, S. Ullah, M. S. Hossain, and A. Alelaiwi, “An end-to-end deep learning model for human activity recognition from highly sparse body sensor data in internet of medical things environment,” *The Journal of Supercomputing*, vol. 77, no. 3, pp. 2237–2250, 2021.

[26] J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan, “uwave: Accelerometer-based personalized gesture recognition and its applications,” *Pervasive and Mobile Computing*, vol. 5, no. 6, pp. 657–675, 2009.

[27] R. Ye and Q. Dai, “Implementing transfer learning across different datasets for time series forecasting,” *Pattern Recognition*, vol. 109, p. 107617, 2021.

[28] Y. Luo, J. Liu, and S. Shimamoto, “Wearable air-writing recognition system employing dynamic time warping,” in *2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)*, pp. 1–6, IEEE, 2021.

[29] L. MerlinLivingston, P. Deepika, and M. Benisha, "An inertial pen with dynamic time warping recognizer for handwriting and gesture recognition," *International Journal of Engineering Trends and Technology (IJETT)–Volume*, vol. 35, 2016.

[30] M. Chen, G. AlRegib, and B.-H. Juang, "Air-writing recognition—part i: Modeling and recognition of characters, words, and connecting motions," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 3, pp. 403–413, 2016.

[31] M. Chen, G. AlRegib, and B.-H. Juang, "Air-writing recognition—part ii: Detection and recognition of writing activity in continuous stream of motion data," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 3, pp. 436–444, 2016.

[32] S. Xu and Y. Xue, "Air-writing characters modelling and recognition on modified chmm," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 001510–001513, IEEE, 2016.

[33] C. Li, C. Xie, B. Zhang, C. Chen, and J. Han, "Deep fisher discriminant learning for mobile hand gesture recognition," *Pattern Recognition*, vol. 77, pp. 276–288, 2018.

[34] T. Yanay and E. Shmueli, "Air-writing recognition using smart-bands," *Pervasive and Mobile Computing*, vol. 66, p. 101183, 2020.

[35] H. Wen, J. Ramos Rojas, and A. K. Dey, "Serendipity: Finger gesture recognition using an off-the-shelf smartwatch," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 3847–3851, 2016.

[36] A. Levy, B. Nassi, Y. Elovici, and E. Shmueli, "Handwritten signature verification using wrist-worn devices," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 3, pp. 1–26, 2018.

[37] C. Amma, M. Georgi, and T. Schultz, "Airwriting: a wearable handwriting recognition system," *Personal and ubiquitous computing*, vol. 18, no. 1, pp. 191–203, 2014.

[38] C. Xu, P. H. Pathak, and P. Mohapatra, "Finger-writing with smartwatch: A case for finger and hand gesture recognition using smartwatch," in *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*, pp. 9–14, 2015.

[39] X. Lin, Y. Chen, X.-W. Chang, X. Liu, and X. Wang, “Show: Smart handwriting on watches,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 4, pp. 1–23, 2018.

[40] Y.-H. Chen, C.-H. Huang, S.-W. Syu, T.-Y. Kuo, and P.-C. Su, “Egocentric-view fingertip detection for air writing based on convolutional neural networks,” *Sensors*, vol. 21, no. 13, p. 4382, 2021.

[41] U.-H. Kim, Y. Hwang, S.-K. Lee, and J.-H. Kim, “Writing in the air: Unconstrained text recognition from finger movement using spatio-temporal convolution,” *arXiv preprint arXiv:2104.09021*, 2021.

[42] G. Bastas, K. Kritsis, and V. Katsouros, “Air-writing recognition using deep convolutional and recurrent neural network architectures,” in *2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pp. 7–12, IEEE, 2020.

[43] T.-H. Tsai, J.-W. Hsieh, C.-W. Chang, C.-R. Lay, and K.-C. Fan, “Air-writing recognition using reverse time ordered stroke context,” *Journal of Visual Communication and Image Representation*, vol. 78, p. 103065, 2021.

[44] M. Arsalan, A. Santra, K. Bierzynski, and V. Issakov, “Air-writing with sparse network of radars using spatio-temporal learning,” in *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 8877–8884, IEEE, 2021.

[45] D. Moazen, S. A. Sajjadi, and A. Nahapetian, “Airdraw: Leveraging smart watch motion sensors for mobile human computer interactions,” in *2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, pp. 442–446, IEEE, 2016.

[46] C. Uysal and T. Filik, “Rf-wri: An efficient framework for rf-based device-free air-writing recognition,” *IEEE Sensors Journal*, 2021.

[47] S. Inoue, N. Ueda, Y. Nohara, and N. Nakashima, “Mobile activity recognition for a whole day: Recognizing real nursing activities with big dataset,” in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 1269–1280, 2015.

[48] S. S. Alia, P. Lago, S. Takeda, K. Adachi, B. Benaissa, M. A. R. Ahad, and S. Inoue, “Summary of the cooking activity recognition challenge,” in *Human Activity Recognition Challenge*, pp. 1–13, Springer, 2021.

[49] L. Joshua and K. Varghese, “Accelerometer-based activity recognition in construction,” *Journal of computing in civil engineering*, vol. 25, no. 5, pp. 370–379, 2011.

[50] R. Akhavian and A. H. Behzadan, “Smartphone-based construction workers’ activity recognition and classification,” *Automation in Construction*, vol. 71, pp. 198–209, 2016.

[51] K. M. Rashid and J. Louis, “Times-series data augmentation and deep learning for construction equipment activity recognition,” *Advanced Engineering Informatics*, vol. 42, p. 100944, 2019.

[52] B. Sherafat, C. R. Ahn, R. Akhavian, A. H. Behzadan, M. Golparvar-Fard, H. Kim, Y.-C. Lee, A. Rashidi, and E. R. Azar, “Automated methods for activity recognition of construction workers and equipment: State-of-the-art review,” *Journal of Construction Engineering and Management*, vol. 146, no. 6, p. 03120002, 2020.

[53] H. A. Aly and E. Dubois, “Image up-sampling using total-variation regularization with a new observation model,” *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1647–1659, 2005.

[54] R. Roy, M. Pal, and T. Gulati, “Zooming digital images using interpolation techniques,” *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, vol. 2, no. 4, pp. 34–45, 2013.

[55] D. H. Hubel and T. N. Wiesel, “Receptive fields and functional architecture of monkey striate cortex,” *The Journal of physiology*, vol. 195, no. 1, pp. 215–243, 1968.

[56] K. Fukushima and S. Miyake, “Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition,” in *Competition and cooperation in neural nets*, pp. 267–285, Springer, 1982.

[57] M. Zeng, L. T. Nguyen, B. Yu, O. J. Mengshoel, J. Zhu, P. Wu, and J. Zhang, “Convolutional neural networks for human activity recognition using mobile sensors,” in *6th international conference on mobile computing, applications and services*, pp. 197–205, IEEE, 2014.

[58] S. Duffner, S. Berlemont, G. Lefebvre, and C. Garcia, “3d gesture classification with convolutional neural networks,” in *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 5432–5436, IEEE, 2014.

[59] J. Yang, M. N. Nguyen, P. P. San, X. Li, and S. Krishnaswamy, “Deep convolutional neural networks on multichannel time series for human activity recognition,” in *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015* (Q. Yang and M. J. Wooldridge, eds.), pp. 3995–4001, AAAI Press, 2015.

[60] S. Ha and S. Choi, “Convolutional neural networks for human activity recognition using multiple accelerometer and gyroscope sensors,” in *2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 381–388, IEEE, 2016.

[61] S.-M. Lee, S. M. Yoon, and H. Cho, “Human activity recognition from accelerometer data using convolutional neural network,” in *2017 ieee international conference on big data and smart computing (bigcomp)*, pp. 131–134, IEEE, 2017.

[62] M. Panwar, S. R. Dyuthi, K. C. Prakash, D. Biswas, A. Acharyya, K. Maharatna, A. Gautam, and G. R. Naik, “Cnn based approach for activity recognition using a wrist-worn accelerometer,” in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 2438–2441, IEEE, 2017.

[63] T. Li, Y. Zhang, and T. Wang, “Srpm–cnn: a combined model based on slide relative position matrix and cnn for time series classification,” *Complex & Intelligent Systems*, vol. 7, no. 3, pp. 1619–1631, 2021.

[64] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[65] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[66] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv preprint arXiv:1704.04861*, 2017.

[67] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

[68] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[69] H. Wu and X. Gu, “Towards dropout training for convolutional neural networks,” *CoRR*, vol. abs/1512.00242, 2015.

[70] S. Park and N. Kwak, “Analysis on the dropout effect in convolutional neural networks,” in *Asian conference on computer vision*, pp. 189–204, Springer, 2016.

[71] T. Carneiro, R. V. Medeiros Da NóBrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque, and P. P. R. Filho, “Performance analysis of google colaboratory as a tool for accelerating deep learning applications,” *IEEE Access*, vol. 6, pp. 61677–61685, 2018.

[72] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.

[73] F. Chollet *et al.*, “Keras.” `https://github.com/fchollet/keras`, 2015.

[74] F. A. Abir, M. A. Siam, A. Sayeed, M. A. M. Hasan, and J. Shin, “Deep learning based air-writing recognition with the choice of proper interpolation technique,” *Sensors*, vol. 21, no. 24, p. 8407, 2021.