

T.R.
GEBZE TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

CONTINUOUS VS FIXED-WINDOW TEMPORAL DATA
PROCESSING FOR HAND MOVEMENT ANALYSIS

BEYZA TÜRKMEN
A THESIS SUBMITTED FOR THE DEGREE OF
MASTER OF SCIENCE
DEPARTMENT OF COMPUTER ENGINEERING

GEBZE
2021

T.R.
GEBZE TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

CONTINUOUS VS FIXED-WINDOW
TEMPORAL DATA PROCESSING FOR
HAND MOVEMENT ANALYSIS

BEYZA TÜRKMEN
A THESIS SUBMITTED FOR THE DEGREE OF
MASTER OF SCIENCE
DEPARTMENT OF COMPUTER ENGINEERING

THESIS SUPERVISOR
ASSIST. PROF. DR. YAKUP GENÇ

GEBZE
2021

T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

EL HAREKETİNİN ANALİZİ İÇİN
ZAMANSAL VERİNİN SÜREKLİ VE SABİT
PENCERELİ OLARAK İŞLENMESİ

BEYZA TÜRKMEN
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

DANIŞMANI
DR. ÖĞR. ÜYESİ YAKUP GENÇ

GEBZE
2021

GEBZE TEKNİK ÜNİVERSİTESİ	YÜKSEK LİSANS JÜRİ ONAY FORMU
----------------------------------	--------------------------------------

GTÜ Fen Bilimleri Enstitüsü Yönetim Kurulu’nun 05/02/2021 tarih ve 2021/06 sayılı kararıyla oluşturulan jüri tarafından 17/05/2021 tarihinde tez savunma sınavı yapılan Beyza Türkmen’in tez çalışması Bilgisayar Mühendisliği Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

JÜRİ

ÜYE

(TEZ DANIŞMANI) : Dr. Öğr. Üyesi Yakup GENÇ

ÜYE

: Prof. Dr. Yusuf Sinan AKGÜL

ÜYE

: Dr. Öğr. Üyesi Ayşe Betül OKTAY

ONAY

Gebze Teknik Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu’nun

...../...../..... tarih ve/..... sayılı kararı.

SUMMARY

The analysis of hand movements is a prominent computer vision problem appeared in diverse applications such as gesture analysis, and human-computer interaction. The primary target of this thesis is a hand movement analysis in videos. We address the issue within two different contexts which are handwriting recognition from pen top movements and hand sanitization monitoring. Both applications involve action recognition in videos including direct or indirect motion of the hands.

We first introduce three unique datasets for these two problems. Then a two-stage framework, including pen tracking and classification, is proposed for the first problem in which the pen top movement is observed instead of the resulting scribed text. For pen tracking, we use a marker-based and a marker-less tracking method, whose results are in turn fed to various deep learning classifiers based on convolutional neural networks (CNN) or long short-term memory (LSTM) architectures. Among several new alternatives to the classification task in the second stage, exploring the encodings of the temporal motion, LSTMs provide the best solution while the single image encoding provides a novel alternative. Of course, marker-based tracking for the first stage yields the best tracking results, hence the classification accuracy of 94% compared to 73% accuracy for the marker-less tracking.

For the second problem, we classify hand movements performed during the handwashing process from monocular RGB videos. The solutions to this problem involve analysis of a single image, multiple images representing a short time period or multiple images for varying temporal periods. Our experiments show that LSTM-based model tends to better capture temporal interaction of hands during the washing action while CNN-based models capture spatial interaction of the hands (90%). The proposed method has been successfully used in a video-based commercial hand-sanitization monitoring system with a very small GPU. Related analysis provided in this thesis can be useful for the researchers addressing this problem in the future.

Keywords: Deep Learning, Computer Vision, Convolution Neural Network (CNN), Long Short Term Memory (LSTM), Multi-Task Learning (MTL).

ÖZET

El hareketlerinin analizi, jest analizi ve insan-makine etkileşimi gibi çeşitli uygulamalarda ortaya çıkan yaygın bir bilgisayarla görme problemidir. Bu tezin öncelikli hedefi videolardaki el hareketlerinin analizidir. Biz bu konuyu kalemin tepesinin hareketinden el yazısı tanıma ve el sterilizasyonu takibi olarak iki farklı bağlamda ele aldık. Her iki uygulama da ellerin doğrudan veya dolaylı hareketi dahil olmak üzere videolarda hareket tanımayı içerir.

İlk olarak bu iki problem için üç eşsiz veri seti tanıtıyoruz. Ardından, sonuçta ortaya çıkan el yazısıyla yazılmış metin yerine kalem tepesini gözlemleyen birinci problem için kalem takibi ve sınıflandırılma olmak üzere iki aşamalı bir yapı önerilmiştir. Kalem takibi için, sonuçları sırasıyla Evrişimli Sinir Ağları (ESA) veya Uzun Kısa Süreli Bellek Ağları (UKSBA) tabanlı çeşitli derin öğrenme sınıflandırıcılarına beslenen işaret tabanlı ve işaret tabanlı olmayan takip yöntemleri kullanıyoruz. Zamana bağlı hareketin kodlanmasını inceleyen ikinci aşamadaki sınıflandırma görevine birkaç yeni alternatif arasında, tek görüntü kodlaması yeni bir alternatif sağlarken, UKSBA en iyi çözümü verir. Elbette, ilk aşama için işaretçi tabanlı takip en iyi takip sonuçlarını verir, dolayısıyla marker tabanlı olmayan takip için %73 doğrulukla kıyaslandığında %94 sınıflandırma doğruluğu elde edilir.

İkinci problem için, el yıkama işlemi sırasında gerçekleştirilen el hareketlerini, monoküler RGB videolardan sınıflandırıyoruz. Bu sorunun çözümleri, tek bir görüntünün, kısa bir zaman periyotunu temsil eden birden fazla görüntünün veya değişken zamansal periyotların analizini içerir. Deneylerimiz, ESA tabanlı modellerin ellerin uzamsal etkileşimini yakalarken UKSBA tabanlı modelin yıkama hareketi sırasında ellerin zamansal etkileşimini daha iyi yakalama eğiliminde olduğunu gösterir (90%). Önerilen yöntem, çok küçük bir GPU ile video tabanlı ticari bir el temizleme izleme sisteminde başarıyla kullanılmıştır. Bu tezde sağlanan ilgili analiz, gelecekte bu sorunu ele alan araştırmacılar için faydalı olabilir.

Anahtar Kelimeler: Derin Öğrenme, Bilgisayarla Görme, Evrişimsel Sinir Ağları (ESA), Uzun Kısa Süreli Bellek Ağları (UKSBA), Çok-Görevli Öğrenme (ÇGÖ).

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor Assist. Prof. Yakup Genç for his support, guidance, and encouragement through both my undergraduate and graduate education. I have gained a better research perspective thanks to him patiently sharing his extensive knowledge and experience. Besides his help in completing my thesis, he has always advised about the future of my journey in computer vision and machine learning research. It has been a pleasure for me to work with him and to learn from him. I also would like to thank members of my thesis committee for their suggestions in shaping the content of this work.

I am the most grateful to have a remarkably supportive family and close friends. I am thankful to Sümeyye Akın for sharing my excitement and difficulties in my academic career, supporting, and encouraging me in the process of writing this thesis. I thank Kubilay Göçer wholeheartedly for sharing my burden and offering help wherever he is and whenever he can. I am much obliged to my dear uncle Serdar Alkurt for being there for me whenever I needed to help. In particular, I thank my parents Zahide, Ali Türkmén for their unrelenting support in my education.

I would like to show my gratitude to the colleagues and friends in Gebze Technical University for their support and friendship. I thank Özgü Göksu for her companionship in the laboratory room that we share and her guidance in all matters. I would like to thank Melike İlteralp for having invaluable generative discussions, her very helpful insights and great cycling trips. I thank also Ferda Abbasoğlu for making us have enjoyable moments with the horror and comedy stories she told.

TABLE of CONTENTS

	<u>Page</u>
SUMMARY	v
ÖZET	vi
ACKNOWLEDGEMENTS	vii
TABLE of CONTENTS	viii
LIST of ABBREVIATIONS and ACRONYMS	x
LIST of FIGURES	xi
LIST of TABLES	xiii
1. INTRODUCTION	1
2. BACKGROUND	4
2.1. Video-based Hand Movement Analysis	4
2.1.1. Hand Tracking	5
2.1.2. Hand Gesture Recognition	6
2.2. DL Approaches for Gesture and Action Recognition in Videos	7
2.2.1. Single Frame CNN	8
2.2.2. Fixed-Window CNN	9
2.2.3. LSTM with CNN Features	10
3. INDIRECT HANDWRITING RECOGNITION	11
3.1. Dataset	12
3.2. Pre-processing	13
3.3. CNN-Based Approach	14
3.3.1. Temporal Data Encoding	14
3.3.2. Single and Multi-Tasking CNN Models	15
3.4. LSTM-Based Approach	17
3.5. Experimental Results	17
3.6. Generalization of the Problem	19
3.6.1. Marker-less Dataset	19
3.6.2. Feature Extraction and Classification	20
3.6.3. Experimental Results	21

4. HAND SANITIZATION MONITORING	24
4.1. Dataset	25
4.2. Single-Frame Classification	26
4.2.1. Single-Frame with Color-Based Skin Segmentation	29
4.3. Multi-Frame Classification	30
4.4. Feature Extraction followed by LSTM	31
4.5. Experimental Results	32
5. CONCLUSION AND FUTURE WORK	36
REFERENCES	39
BIOGRAPHY	42
APPENDICES	43

LIST of ABBREVIATIONS and ACRONYMS

<u>Abbreviations</u>	<u>Explanations</u>
<u>and Acronyms</u>	
2D	: 2 Dimensional
3D	: 3 Dimensional
AR	: Augmented Reality
CNN	: Convolutional Neural Network
ConvNet	: Convolutional Networks
DL	: Deep Learning
HCI	: Human-computer Interaction
HWR	: Handwriting Recognition
LSTM	: Long Short-Term Memory
MTL	: Multi-task Learning
RNN	: Recurrent Neural Network
ReLU	: Rectifier Linear Unit
WHO	: World Health Organization

LIST of FIGURES

<u>Figure No:</u>	<u>Page</u>
2.1: The background schematic of hand movement analysis.	4
3.1: Problem definition of indirect handwriting recognition.	11
3.2: The dataset was created for indirect handwriting recognition.	12
3.3: Pre-processing to extract location of markers.	13
3.4: Mapping of temporal location information into multi-channel image.	14
3.5: CNN model classifies 5-channel images generated through temporal encoding.	15
3.6: Multi-tasking CNN model carries out both classification of temporal data and the generation of handwritten digit image.	16
3.7: LSTM model classifies temporal data consist of the 2D location information of markers in each frame of video.	17
3.8: Experimental results of three models according to increasing the number of sample in the dataset.	18
3.9: a) The binary images converted from the images of digits written in the videos, ground truths. b) The binary images generated by multi-tasking model accordingly.	19
3.10: a) The frame of a video captured from the perspective one (P1). b) The frame of a video captured from the perspective two (P2). c) The image of the handwritten digit in the videos.	20
3.11: a) In the i th frame of a video, detection of 2D location of pen's top point using template matching algorithm. b) In the approach one (A1), the difference between consecutive frames are exploited to generate time-series data. c) In the approach two (A2), cosine-sine transformations are exploited for the same target.	21
3.12: The effect of the error in the pen tracking on model performance. The pen movement in marker-less dataset is within 2 pixels range whereas the movement of the pen in the marker-based dataset is 20 pixels. The perturbation amount for each case is equal in proportion to the range of pen movement.	22

3.13: The comparison of the confusion matrices belonging to the models M4 and M5.	23
4.1: Problem definition of hand sanitization monitoring. a) Video recording during handwashing process. b) There are frames extracted from a video and the predictions of them according to handwashing actions.	24
4.2: The illustration of some classes in the dataset.	25
4.3: The weights of 11 classes in the dataset.	26
4.4: The model architecture of SF_RGB.	28
4.5: The results of color-based hand segmentation in some frames.	29
4.6: The working principle of sliding window approach for multi-frame classification.	30
4.7: Feature extraction from consecutive frames to be fed to LSTM.	32
4.8: There are some video fragments shows the filtering by LSTM on the errors in single-frame classification results. All colors represent an action segment.	35

LIST of TABLES

<u>Table No:</u>	<u>Page</u>
3.1: The results obtained from different data generation approaches.	23
4.1: Proposed models with the architecture and input information.	27
4.2: The results of single-frame and multi-frame models.	33
4.3: The results of single-frame models and LSTM-based models.	34

1. INTRODUCTION

There have been significant advancements in technology improving the lives of people. As a result of these improvements, machines have been ubiquitously used to solve well-known problems in different fields. The recognition of hand movements is one of them because it has great importance in some fields such as human-computer interaction (HCI), communication, security, and health care. One primary purpose of such an analysis is to develop systems to recognize predetermined hand movements from temporal monocular video. While studied quite extensively in the field of computer vision, with the advancements in deep learning, potential improvements became possible.

In this thesis, we take a comprehensive approach to analysis of hand movements in monocular video. We explore obvious as well as novel approaches to apply deep learning models to two specific hand motion analysis problems. As far as we know, these problems are addressed the first time in this thesis.

Our approach explores several obvious uses of deep learning in image and video processing. First, we convert the given analysis problem into a single image (or a few frames) classification problem. Second, we address the problem as a temporal video analysis problem where recurrent networks, specifically long-short term memory (LSTM) networks, have shown remarkable performance. However, the problems we address differ from a straightforward application of LSTMs as we have a limited amount of labeled data. To alleviate this problem, we combine single frame approach which is easier to train with less data with LSTMs which is trained with the features obtained from the single frame models.

The proposed framework was applied to two different, but related problems with underlying problem as hand action recognition. These problems are 1) handwriting recognition from observation of pen top movement, and 2) hand sanitization monitoring from monocular RGB cameras.

In the conventional handwriting recognition (HWR) problem, images of the manuscripts are transformed into digital text. Further, HWR is an important research field in computer vision owing to the increase in the need to represent and transfer information with images[1]. Different from the classical HWR, we define a new problem where the recognition is done through the secondary observation, i.e., the

observed movement of the pen top while the writing takes place. When the direct observation of the handwritten document is not available, this indirect observation can be used to recognize handwriting with the analysis of hand movement. In addition to this, a solution to this problem can be beneficial for applications, including writing style prediction and signature verification. Furthermore, it may provide children who are just learning to write letters with the opportunity to be corrected quickly. The newly posed problem therefore is the probably the first in the computer vision field. Consequently, the first contribution of the thesis is to introduce a new perspective for a well-known problem.

Our solution to this problem is a two-stage approach where in the first stage the movement is monitored using marker-based or marker-less tracking in the video. Tracking techniques are well established and does not require any data to train. Once the coordinates of the tracked pen top are established using one of the existing tracking methods, the classification is achieved using a deep learning method. For this we offer two alternatives. In the first, we aggregate the temporal coordinate information in a single image and feed it to a convolutional neural network (CNN). In the second approach, we feed the temporal coordinate positions directly into an LSTM. Extensive experiments including perturbation analysis show that LSTM provides a better model as the tracked coordinates can be represented with subpixel accuracy while CNN can be equally powerful when sufficient resolution can be obtained with the aggregate image. We also show that LSTM model requires more data to train than the CNN model (especially when trained with pretrained networks).

In this thesis, the indirect handwriting recognition problem is accomplished by using videos recorded while someone is writing a digit. Several models based on deep learning methods are proposed to solve this problem. For this we created two datasets where the first one indicates the pen movement with the help of a set of markers, while the second one directly observes the pen top movement from two different perspectives (using two cameras simultaneously recording).

The first data set for indirect handwriting analysis consists of 2500 videos including 250 samples for each digit and the images of the handwritten digits. In the videos of marker-based dataset recorded from one perspective, a pen with 6 markers was utilized for the writing of digits. The second data set is created to recognize handwriting by tracking bare pen in a video. This dataset contains 800 couple videos captured from two different perspectives simultaneously and the images of the

handwritten digits. Therefore, indirect handwriting is carried out by leveraging vision-based pen tracking algorithm and deep learning-based classification models.

The second problem we address is hand hygiene monitoring using monocular cameras. Hand hygiene is one of the most critical conditions for staff in health services and food industry. Notably, poorly cleaned hands pave the way for customers to get severe infections in restaurants and health-care facilities. For this reason, monitoring of handwashing is a worthy problem to study. For monitoring hand sanitization, prior works benefit from electronic counters and RFID badges to measure soap consumption and remind the staff about washing hands[2]. These studies aim to increase the frequency of handwashing, but they do not check the handwashing process's quality.

We propose a solution to hand hygiene monitoring which recognizes specific movements performed during hand washing from a monocular RGB camera. There has been some work in the literature form similar problems, such as sign language and gesture recognition. The difference from prior works is that this problem requires a solution that can handle the interaction of two hands. We introduce several end-to-end deep learning approaches to handle action recognition problem based interacted hands. Another difficulty with this problem is that there is a limited amount of labeled data which has a significant amount of outlier (wrong labels). Experimental analysis shows that the proposed method works well on real-world data obtained from hospitals. Once again among multiple solutions an LSTM-based model working on tracked features via a CNN model provided the best solution. The final two stage CNN and LSTM-based solution is currently is utilized in a commercial solution with a low-end GPU card.

This thesis is organized as follows. Section 2 provides a background on the two problems we are addressing. We also introduce relevant image processing and deep learning methods applied to similar problems in this section. Section 3 discusses the first problem in detail and our solutions along with experimental findings and discussions. Similarly, Section 4 includes the second problem and our solutions for it along with our experiments and discussions. Finally, we conclude in Section 5 with some directions for some future research.

2. BACKGROUND

The main objective of this thesis is to analyze the movement of hands in the videos through two different applications. In this section, we present prior works belonging to video-based hand movement analysis which is the most comprehensive form of the problem. Afterwards, existing deep learning (DL) methods developed for this problem are introduced. Besides general solutions, recent works are reviewed in terms of two applications, handwriting recognition, and hand sanitization monitoring, respectively.

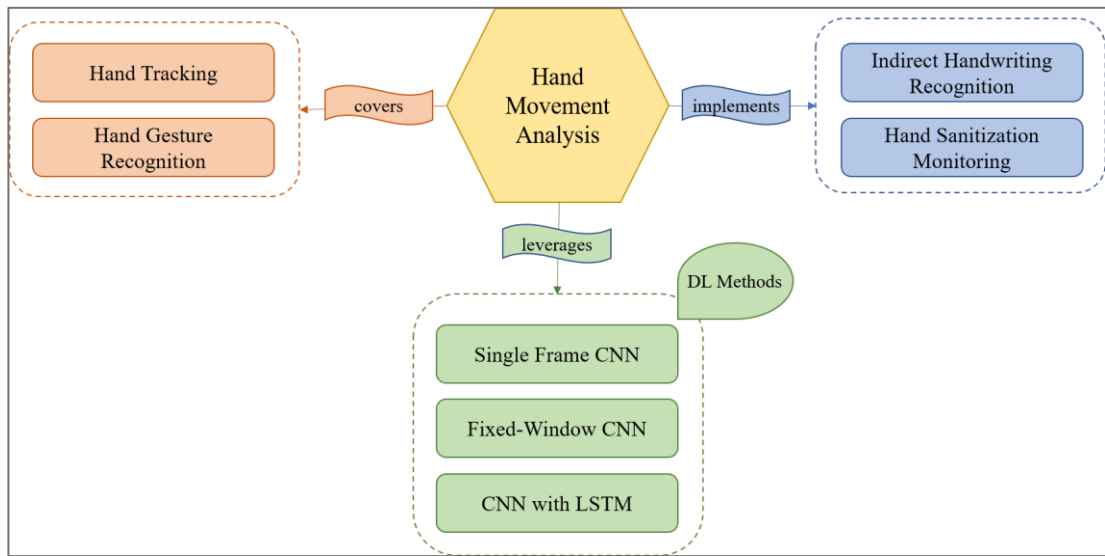


Figure 2.1: The background schematic of hand movement analysis.

2.1. Video-based Hand Movement Analysis

Hand movement analysis is a widespread problem in some fields such as computer vision, human-computer interaction, and augmented reality. There has been a considerable amount of study to analyze hand actions. In particular, they are considered as three types of research direction. First one is based on hand tracking and aims to detect hand position. The second is based on hand gesture recognition and aims to motion patterns of the hand. In this part, we present a literature review of these research directions.

2.1.1. Hand Tracking

A large part of research incorporates hand tracking methods. Formerly, these methods have been limited due to its high setup price. By the time, this limit is almost completely overcome with the advent of deep learning. Finally, recent methods address the bare-hand tracking problem without any additional equipment through simple cameras.

One of the most straightforward ways to follow an object is placing an indicator on it because there are many methods to detect specific indicators in an RGB image. In this case, it is aimed to track hand by detecting the indicator. This indicator may be a group of markers or a glove. In [3], a simple multi-color glove imprinted with a custom pattern is designed for the hand to be pointed out. A nearest-neighbor approach is employed to track hand. Another study [4] intends to decrease the cost of glove-based solutions. They suggest an algorithm to detect hand by tracking colored markers placed at palm and fingers.

Otherwise, image-based methods offer an attractive alternative to track hand motion because they require no external equipment such as markers and gloves. However, these methods may be incompetent to overcome ambiguity stemming from self-occlusion. In [5], Zhao *et al.* have been complimented the image-based system with markers to address this challenge. Further, this image-based system leverages RGB-D image data obtained by a Kinect camera. RGB-D data refers to the combination of color and depth images to bring about more accurate results. Before advanced computer vision and deep learning methods, this RGB-D data type tends to be chosen for hand tracking applications. In [6] and [7], Sridhar *et al.* propose articulated hand tracking and hand-manipulated object tracking methods using RGB and depth data, respectively. Another articulated 3D hand tracking approach [8] benefits from the segmentation of color-based hand skin and corresponding depth map to estimate hand pose.

The hand tracking is a challenging problem when two hands interact with each other because complicated inter-relations augment occlusions. In such a framework [2], the intended outcome is finding the right one in all possible configurations of two hands that best fits observation. Furthermore, Ballan *et al.* have addressed this problem using salient points in multi-camera setup even if an additional object shows up with the interacted two hands [9].

Besides these works, bare-hand tracking methods introduce an appealing alternative to hand tracking because they require no markers, gloves, or sensors. An article from Carnegie Mellon University [10] proposes a model-based hand tracking method benefiting from useful features extracted from raw images of the hand. In particular, bare-hand detection through web camera is essential for motion-based games. In this case, an algorithm consisting of two stages which are hand detecting with colour-based skin segmentation and analysis of gamers hand movements have been developed [11]. Augmented reality is another application field for this problem. The human hand is used as a distinctive pattern instead of a marker for these AR applications. In [12], [13], Lee *et al.* have developed a vision-based system to detect image features and track frame-to-frame computing optical flow. They have carried out the tracking of fingertips and hand to render a virtual object in the AR environment on the hand according to hand size and camera pose, respectively.

2.1.2. Hand Gesture Recognition

Video-based hand movement analysis is a critical problem in some branches, such as human-computer interaction (HCI), communication, security, and health care. First of all, hand gesture recognition provides users with the opportunity to interact with machines without any tool such as a touchpad, joystick, and button. Besides, it is possible to use recognized gestures as the input for a video game to introduce enhanced gaming experience thanks to a web camera placed in front of the work area[14].

In addition to HCI, hand gesture control is frequently encountered in augmented reality applications to manipulate virtual objects. It is more convenient than traditional input tools[15]. Lee *et al.* [16] have proposed a user interface for manipulating and interacting with virtual objects. There is a glove with a conducting material on the fingertips and the palm for gesture recognition. Also, visual markers are placed on the glove to track hand position and rotation.

On the other hand, human beings benefit from speech and body language for communication in daily life. In particular, the latter is crucial for deaf people. Besides body language, sign language paves the way to communicate easily with society for them. At this point, the automatic recognition of sign language, which can handle word by word or entire sentences is essential and required[17]. Besides the existence of hand movements in communication, these may be considered a sort of biometric

modalities. To illustrate, the way people perform a movement with their hands can expose their identity, which is a significant authentication issue. In this case, distinguishing different people can carry out using information obtained from a mobile phone accelerometer when they are performing a 3D movement while holding the phone[18][19]. In addition to this, a more appropriate way is to extract necessary information from videos which contain dynamic hand gestures[20]. Lastly, a hand action recognition system can improve the rehabilitation process's success when patients with limited capabilities to perform functional hand activities perform home-based physiotherapy exercises[21].

It is a common practice that hand gesture recognition is realized in two stages as feature extraction and recognition. In the former, some kind of features are extracted to be processed by an inference algorithm such as Hidden Markov Models and Support Vector Machine in the latter. However, recent studies benefit from deep learning methods for end-to-end hand gesture recognition.

2.2. DL Approaches for Gesture and Action Recognition in Videos

There have been numerous approaches that address problems of action and gesture recognition, analyzing the movement of human and hand respectively. With the increase in amount of data and computational power, deep learning (DL) based methods have become preferable to vision-based ones due to their high success in well-known problems. Therefore, a vast majority of recent works conducted for action and gesture recognition in image sequences exploit DL based methods. There are two types of neural networks including convolutional neural networks (CNN) and recurrent neural networks (RNN) tailored to specific tasks such as working with image and temporal data, respectively.

One of the most popular DL models for movement analysis is CNN conceived as a composition of multiple kinds of layers such as convolution, pooling, and fully connected layers. CNNs are capable of successfully capturing spatial dependencies within an image due to their convolutional layers transforming the input data by using filters before passing it to the next layer. For instance, DenseNet [22] is a type of CNN that densely connects each layer to all other layers to alleviate the vanishing gradient problem as networks get deeper. Thus, CNN model is one of the probable solutions

considering each frame of videos independently for action and gesture recognition problems.

In addition to the competence of CNNs in images, RNNs, interpreting time-series data, are another equally significant contribution to activity recognition in videos. A special kind of RNN is called long short-term memory (LSTM), which can learn long-term relationships in any type of temporal data such as videos. In LSTMs, there is a chain of repeating modules of the neural network including four interacting layers to carry adequate information. Consequently, the video-based activity recognition problem can be effectively solved with the help of these implementations, including CNN and LSTM which can capture both spatial and temporal information in the videos respectively.

2.2.1. Single Frame CNN

The most straightforward way for activity recognition in videos is considering each frame of video as an independent image. A vast majority of these methods performs some preprocessing stages such as hand segmentation to limit the area in which the recognition will be realized. For hand gesture recognition, the area covers only the pixels of hand. In contrast, Bao *et. al.* [23] propose a deep CNN-based system which can directly recognize hand gestures from image without any hand segmentation process. In this way, the system can effectively fulfil the recognition of hand gestures in any part of the image even if they occupy less space than the whole background.

Despite the success of recognition within a frame, this outcome may not be sufficient when considering the entire action, as the action is spread across more than one frame. In the work [24], a two-stream convolutional networks (ConvNet) architecture for action recognition in videos incorporates the information about individual frames and the action between frames to complement the absence of a temporal aspect. Whereas spatial stream is fed by individual video frames, temporal stream exploits the input formed by stacking fields of optical flow displacement between several consecutive frames. The latter enables to significantly improve the classification accuracy thanks to the explicit description of the motion.

An alternative way of taking into account frame-based features in a video context is to compute the average feature over the frames in the video. Wang *et. al* [25] bring about a Siamese network with the two towers representing the precondition and effect

stages of an action. In this context, while the former refers to the state before the action happens, the latter considers the state after. Both tower of the ConvNet independently computes the feature for each frame in the set of precondition and effect of the video respectively and aggregates the features through average pooling. They aim to recognize action and divide the action into two according to precondition and effect at the same time. Thus, the inference for the whole video is acquired by exploiting the average of the frame-wise features.

2.2.2. Fixed-Window CNN

In hand gesture and action recognition applications, there are some challenging factors. One of them is that the action or gesture is unknown when it will be performed in real-time or in large-scale videos. For this reason, the sliding-window approach is preferred to catch the exact movement when performing these types of task. According to this approach, a window of fixed length moving over the video is fed to CNN-based models. In the work addressing a real-time hand gesture recognition task [26], there is two-stage model carrying out detection and classification simultaneously to implement a real-time system. They exploit 3D CNN architectures for both tasks whose latter is deeper than the former since it only classifies the presence of gesture in the window to increase the speed in computation. Hence, 3D CNNs make use of the temporal relation between frames in the window together with their spatial content.

There are some different techniques processing videos with various duration to be leveraged as a fixed-window input for 3D CNNs. One of them is re-sampling each gesture sequence to fixed length using nearest neighbor interpolation by dropping or repeating frames such as proposed in [27]. Thus, the hand gesture recognition system exploits depth and intensity channels with a 3D CNN classifier consisted of two sub-networks, which include high-resolution and low-resolution network, fused by multiplying outputs of their softmax layers element-wise. As another way of generation of fixed-length image sequence from varied-length videos, Al-Hammadi *et. al.* [28] leverage linear mapping to unify the length of the video when recognizing hand gestures with 3D CNN. In this way, whatever the length of an input video, it is able to be linearly mapped to specified number of frames sequence.

2.2.3. LSTM with CNN Features

Besides the way of processing temporal relation between frames with 3D CNN, there is a more effective approach for this issue, which employs LSTM. This approach addresses gesture or action recognition problem in two steps. Whereas the former extract the spatial information from frames with 2D CNN, the latter runs an LSTM network over the temporal data composed of the features extracted in the former from each frame of the video. Likewise, the action recognition framework [29] leverages this approach extracting deep features from every sixth frame of a video to reduce the redundancy and complexity via pre-trained AlexNet on the first stage. In the latter, they employ bi-directional LSTM with both forward and backward layers to learn sequence information in the features of video frames.

In addition to this two-step approach, Donahue *et al* [30] propose a end-to-end model consisting of CNN and LSTM layers for video recognition tasks. Unlike other studies mentioned before, this model carries out a many-to-many classification predicting each frame of the video to be applied to variable-length inputs which represent videos with different duration. LSTM can be beneficial in post-processing performed after the classification of each frame of video individually [31] as well as it is in classification itself like the previous work. The gaps caused by miss-prediction in the recognition stage, which appear in the video lead to fragmentation of the gestures with an adverse influence on the performance of gesture recognition. Therefore, exploitation of LSTM for post-processing can figure out this fragmentation problem by filling the gap to improve the task performance.

3. INDIRECT HANDWRITING RECOGNITION

Handwriting recognition via visual observation of hand movements is challenging due to the lack of primary hint, which is the handwritten document's image. Therefore, the proposed method benefits from the hand movements to handle it, unlike traditional ways. In this scope, the target is to develop an end-to-end deep neural network framework that enables the entire digitalization of handwriting someone is writing. However, there is no video data set in this concept, which is recorded from a perspective that does not cover the handwriting document but hand movement. Consequently, producing a sufficient amount of data is so costly to fulfil the end-to-end framework. This case paves the way to simplify the problem.

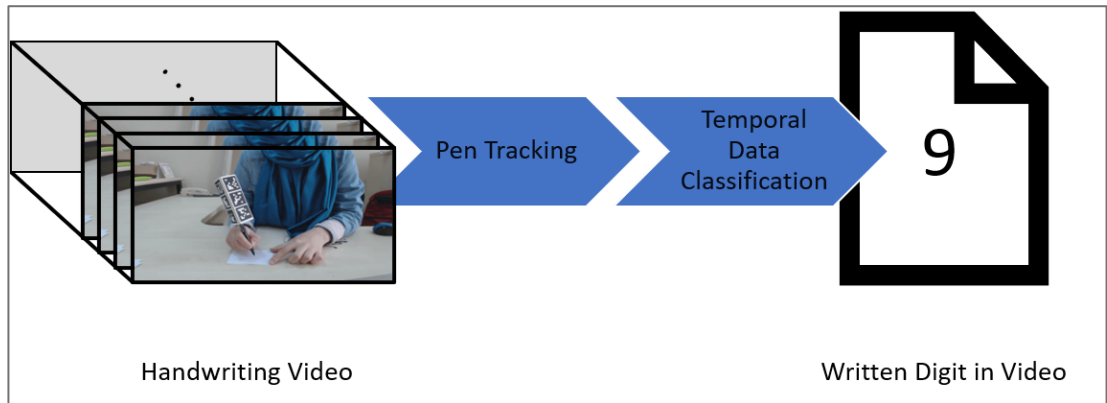


Figure 3.1: Problem definition of indirect handwriting recognition.

As a result of this simplification, in Figure 3.1, the objective is to recognize only one-digit written in a video. Accordingly, the problem is addressed in two main steps, pen tracking and digit recognition. In the first step, the pen is chosen to track because it is more straightforward than hand tracking. In this case, the movement of the pen manipulated by hand indirectly indicates the hand movement. Thus, the motion information is obtained from the pen. In latter, three different deep learning models are accomplished to leverage temporal data effectively. These models are based on successful approaches such as CNN and LSTM.

3.1. Dataset

There are several datasets for standard handwriting recognition problem in prior works. However, these are not convenient for our indirect approach. Therefore, a novel dataset was created in the scope of this thesis. The dataset contains 2500 videos equally distributed into ten digits. These videos were labelled according to the digit handwritten in the video. Besides, we captured the image belongs to the handwritten digit, as shown in Figure 3.2. In addition to this, all videos include the writing process realized by the same person.

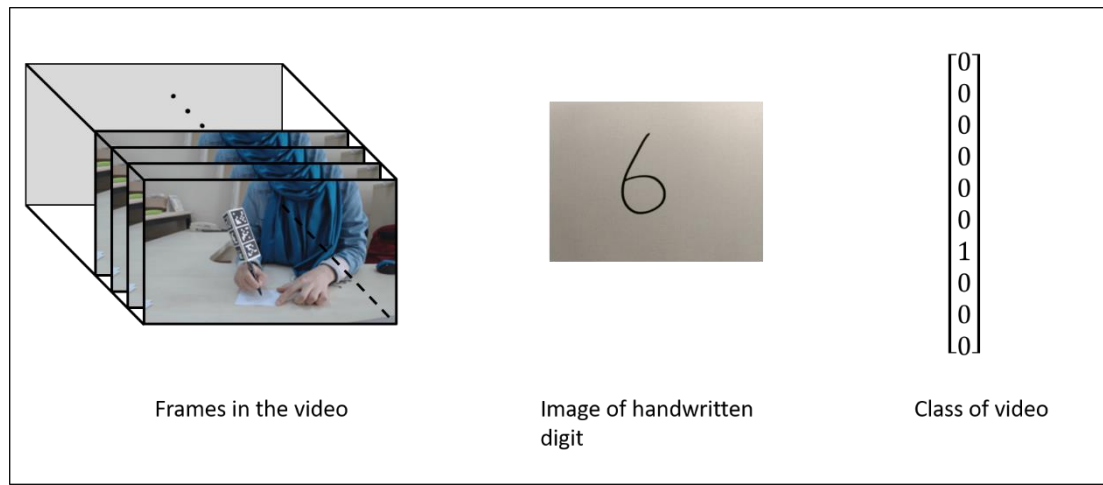


Figure 3.2: The dataset was created for indirect handwriting recognition.

A camera located the opposite of working area recorded videos while a digit was being written on paper. Afterwards, a photo of the paper with the size of 10.5x7.5cm was taken to be utilized with video if it is required. All videos take about two minutes, but they are not the same size. In other words, the number of frames is variable. Moreover, there are six markers located on the pen manipulated by the person during the writing. These markers make it easier 2D tracking of the pen thanks to Aruco library [32], a simple vision-based tracking algorithm. Because hand location detection is compulsory in each frame, more than one marker is located to prevent the markers' lost sight during the pen action. Besides, these are located in two groups of three, forming a 90-degree angle between them to reflect the depth.

3.2. Pre-processing

Video classification with deep learning methods is too costly because it requires a large amount of data and high computational power. In this case, only essential information must be extracted from data to classify them. Data pre-processing is the first stage of the problem, which is handled with the Aruco library [32] enabling tracking markers on an image. The Figure 3.3 illustrates markers in a frame and the center points of them. Following marker detection, these points' 2D location is acquired to form a vector representing a pen position. Afterwards, the coordinates x and y of these points are combined into a vector with twelve dimensions, as shown in Figure 3.3. Furthermore, this operation is carried out on each frame of each video in the dataset. Finally, vectors extracted from every frame of the same video are vertically concatenated along the axis of time. At first, the size of a sample with N frames in the dataset is $720 \times 1280 \times N$. Whereas, after preprocessing, the shape of the created matrix to be managed instead of the video anymore is $N \times 12$. In this case, the number of rows refers to the number of frames. Due to 2D positions of the six markers, the number of columns is twelve, as well.

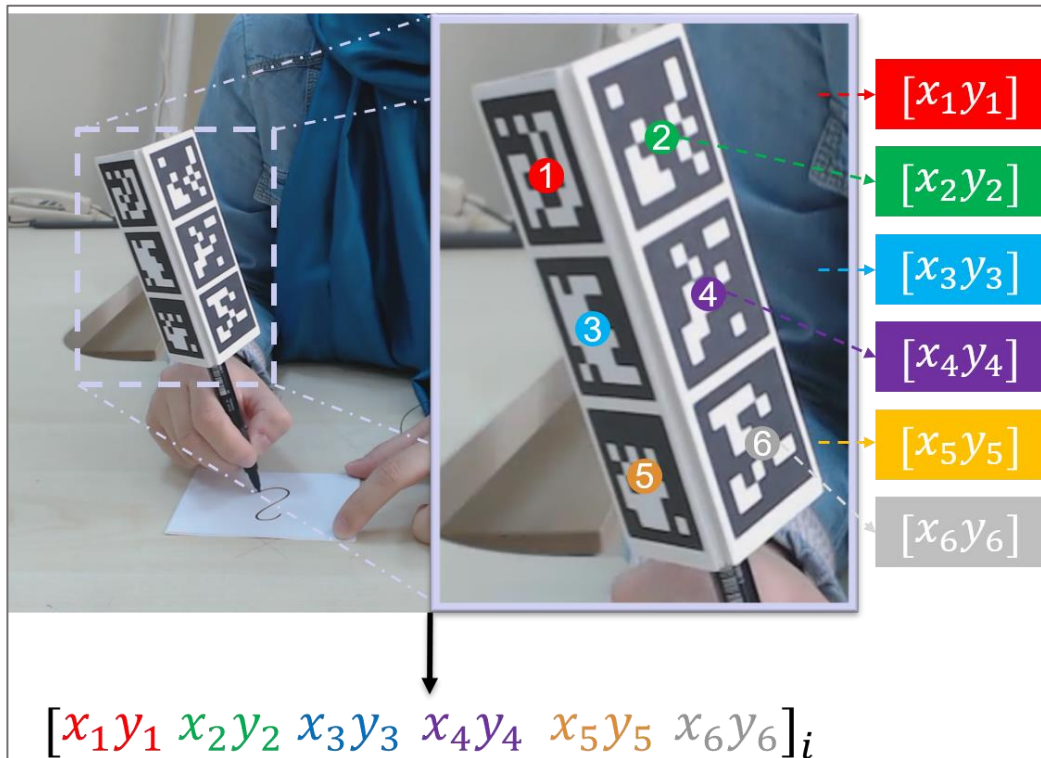


Figure 3.3: Pre-processing to extract location of markers.

3.3. CNN-Based Approach

In handwriting recognition context, two CNN-based models are introduced to handle the handwriting recognition problem observing the pen action. Firstly, a baseline model is designed to classify videos according to which digit is written. After that, a new task is combined with the baseline model to improve classification performance. With this extension, both video classification and image generation are addressed. The latter aims to generate images of handwritten digits, that contributes to the main task. The most significant part of CNN-based approach is temporal data encoding which is a novel method.

3.3.1. Temporal Data Encoding

We use tracking information for pre-processing, provides 2D location information of pen movement in time. However, CNN requires a single image in a few channels (a tensor). Therefore, a method is proposed to encode time-dependent location information into a multi-channel image to be used in CNN-based model training. The critical point of this encoding is the visual reflection of data without losing time and location information. As a result of the encoding, each sample of data is expressed as a multi-channel image. Data preprocessing provides location information of pen movement in time. However, CNN requires the visualization of preprocessed data. Therefore, a method is proposed to encode time-dependent location information into a multi-channel image. The critical point of this encoding is the visual reflection of data without losing time and location information. As a result of encoding, each sample of data is expressed as a multi-channel image.



Figure 3.4: Mapping of temporal location information into multi-channel image.

In detail, a blank image consisting of black-colored 27×48 pixels generated at first. The boundaries of the area containing pen action are determined considering the

maximum and minimum points of extracted positions. In the following steps, locations are mapped into blank image scaling according to these boundaries. Meanwhile, mapping means that extracted position information belonging to six markers is painted the corresponding pixel in the blank image in white. Therefore, this mapping creates an image including spatial information about the pen trajectory. However, time knowledge gets lost when the pen trajectory visualizes in one image. The pen movement is divided into five in the time axis to avoid this drawback. Next, each period is mapped to a different image, shown as Figure 3.4. Thus, these grayscale images are concatenated along the third axis as a channel in order. Lastly, a multi-channel image with the shape of $27 \times 48 \times 5$ comes out to visualize the pen movement in a video. In the images, black-coloured and white-coloured pixels have 0 and 255 intensity value, respectively.

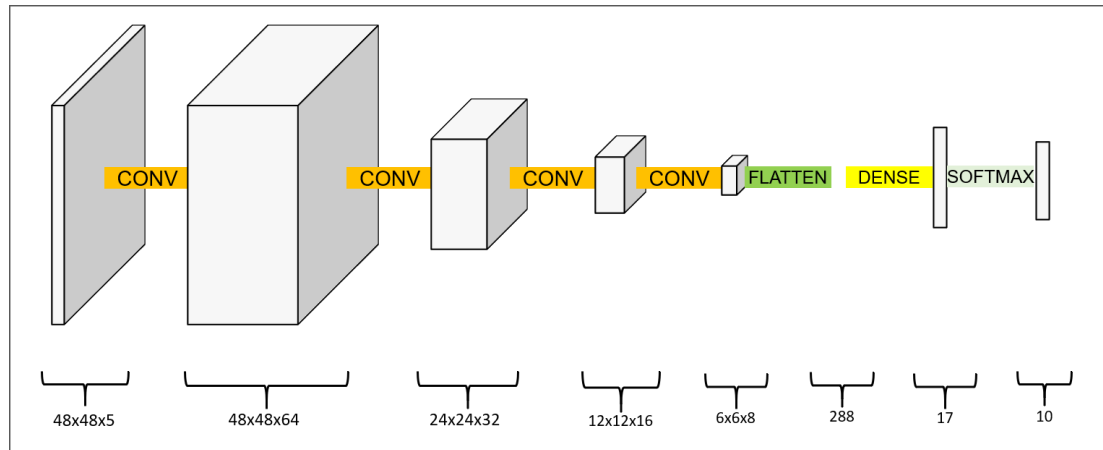


Figure 3.5: CNN model classifies 5-channel images generated through temporal encoding.

3.3.2. Single and Multi-Tasking CNN Models

There is a basic CNN architecture as single-tasking CNN model to classify an image with 5-channel according to which digit is indicated. Firstly, the range of pixel intensity values in the image generated by encoding is transformed from (0,255) into (0,1). Next, the model starts with a convolutional layer which increases the number of channels to 64 with the same image size. Also, three convolutional layers follow this that decreases both the number of channels and the image size to half in each step. In the last part, the final CNN layer's output is flattened to be fed to a dropout layer preventing overfitting before two fully connected layers. ReLU activation function is

preferred for all layers except the last one because it needs to Softmax instead to complete classification. Consequently, the model, including about 32K parameters, is trained with Adam optimizer in default settings using categorical-cross-entropy loss function.

In our multi-tasking model (MTL), there is an image generation task as an extension of basic CNN model. In this way, it handles both digit classification and handwritten image generation. The latter considers the binary version of handwritten digit images as labels. Image generation is not the primary task, but it has a supportive role in classification. In addition to basic CNN model, the third convolutional layer follows three deconvolutional layers which double up both the number of channels and the image size. Lastly, another deconvolutional layer doubles the image size when reducing the channel number to generate a grayscale image. Whereas categorical-cross-entropy as loss function are preferred for the classification task, image generation task uses binary-cross-entropy loss function in training. Besides, the multi-tasking model contains about 58K parameters.

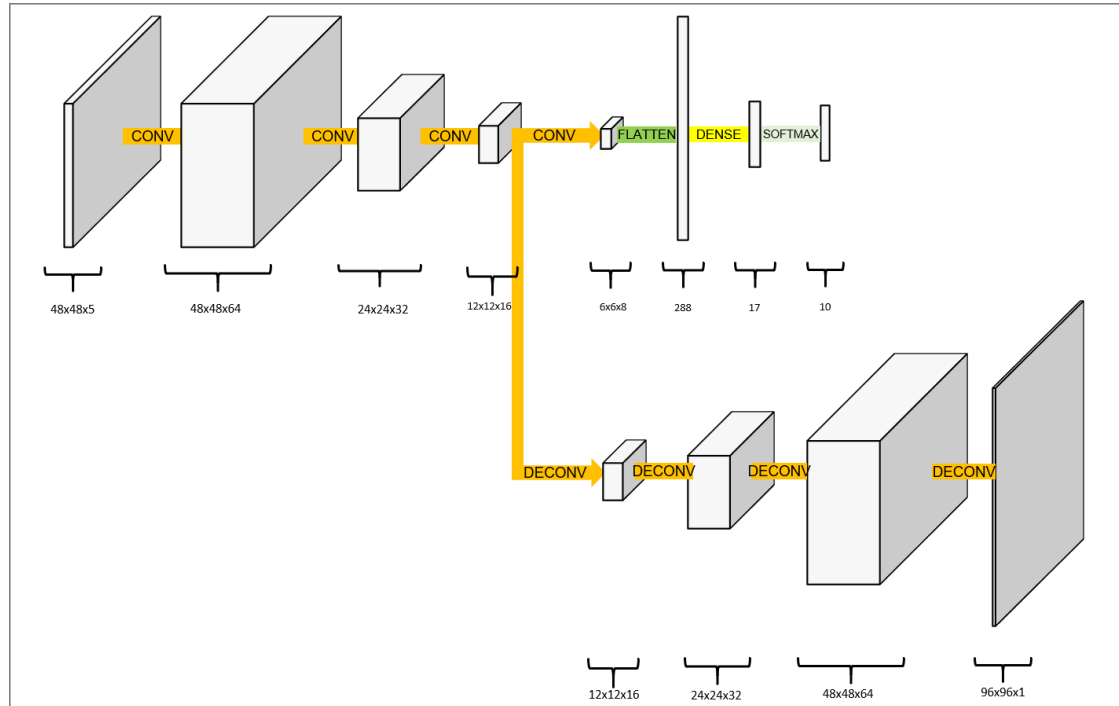


Figure 3.6: Multi-tasking CNN model carries out both classification of temporal data and the generation of handwritten digit image.

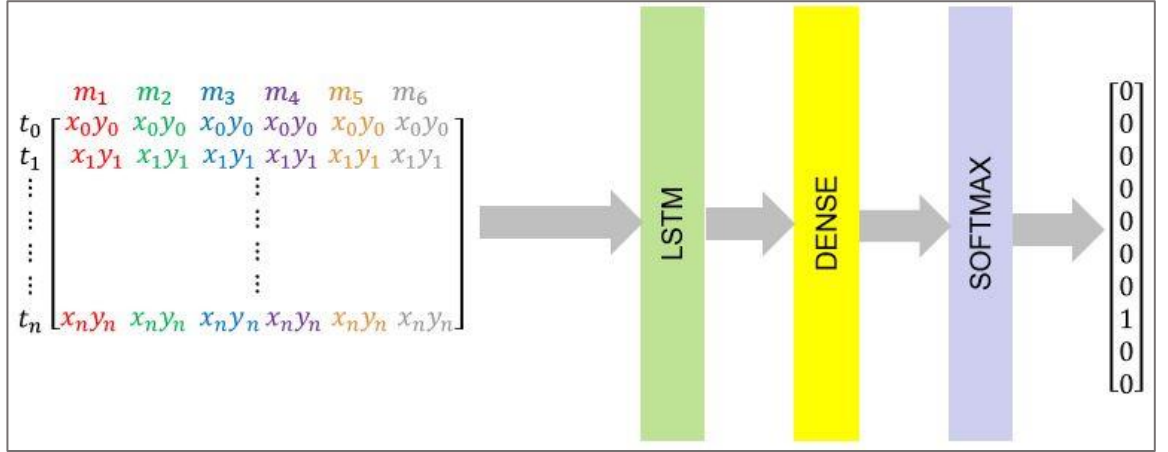


Figure 3.7: LSTM model classifies temporal data consist of the 2D location information of markers in each frame of video.

3.4. LSTM-Based Approach

As another approach, a simple LSTM model is developed to classify 2D temporal data extracted with pre-processing. In the previous method, encoding temporal data to an image causes the loss of most of the time information. Therefore, the advantage of LSTM over CNN-based models because the former leverages entire time-series data. Further, LSTM model carries out a many-to-one classification, as shown in Figure 3.7. Although the pre-processing step provides the location information in time, this data should be normalized to restrict the action area. Thus, we scaled the positions into a specific range. Besides this normalization step, the model architecture starts with a masking layer to eliminate the variation between sample lengths. Next, an LSTM layer with 64 neurons takes place, followed by a dense and dropout layer. Lastly, there is another dense layer for classification with Softmax activation function. We used ReLU activation function in other layers. The loss function is also categorical-cross-entropy to train this model with 22K parameters optimizing Adam.

3.5. Experimental Results

In the thesis's scope, three different models are proposed to classify videos according to the handwritten digit. Figure 3.8 illustrates the accuracy rates of these models in the case the dataset is expanded step by step. According to these results, the

LSTM model is the most successful with %96 accuracy thanks to the ability to process time-series data effectively. However, the CNN model has limited competence to handle temporal information represented in images. On the other hand, the MTL model's image generation task provides additional clues to increase its success rate. Besides its improvement in the classification accuracy, the outputs of the image generation task of MTL are satisfactory, although there is a limited amount of data. When Figure 3.9 is examined, the number of holes belongs to digits, 6 and 8, reflect the ground truth image. With the increase of dataset size, accuracy rates also increase in general because it brings about better generalization of models.

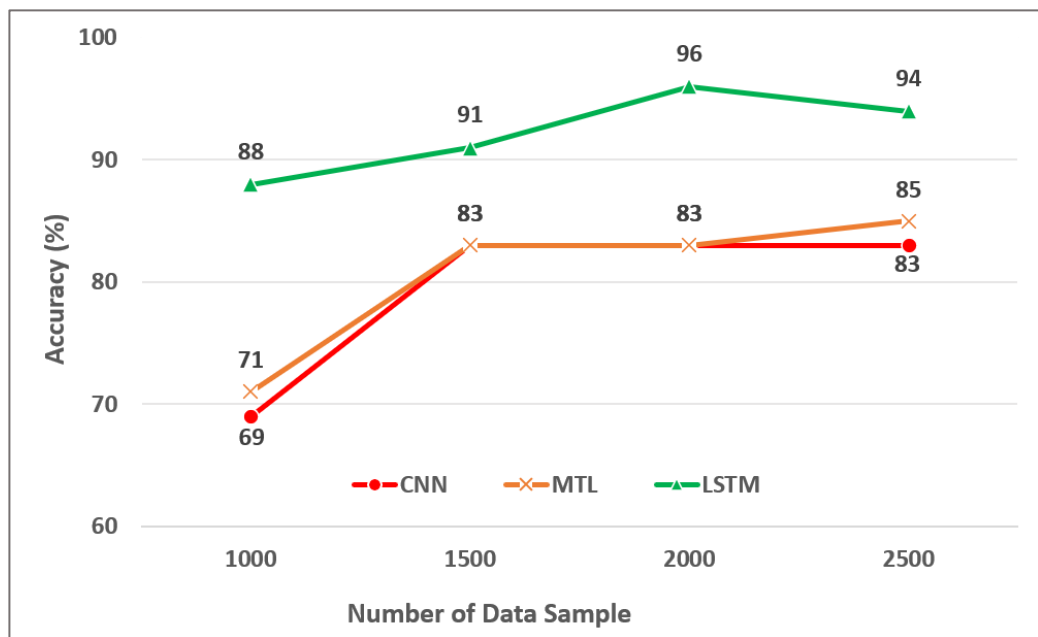


Figure 3.8: Experimental results of three models according to increasing the number of sample in the dataset.

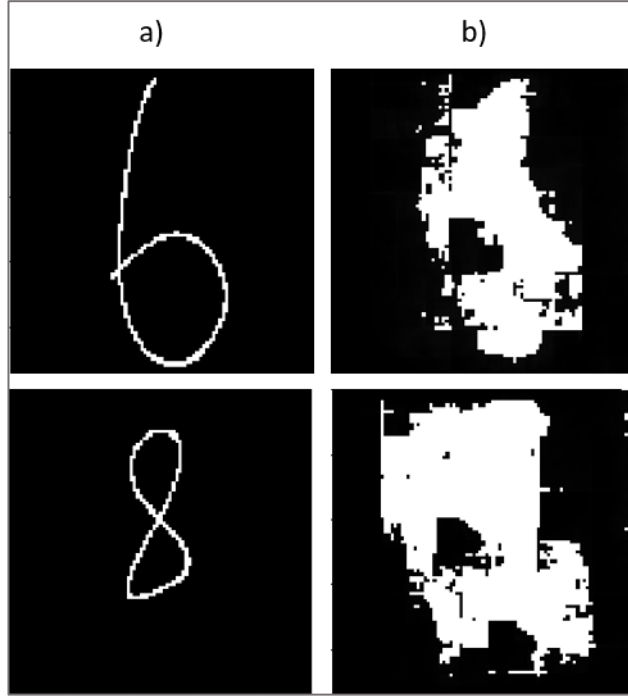


Figure 3.9: a) The binary images converted from the images of digits written in the videos, ground truths. b) The binary images generated by multi-tasking model accordingly.

3.6. Generalization of the Problem

In real world, tracking the pen movement through a 3D marker tracker is not practical. Instead, we have collected a set of data to realize tracking directly. For that case, 800 videos are recorded when a person is writing digits on a sketch pad. Thus, there is a dataset including both videos of writing movement and images of written digits. Afterwards, we have created a framework which extracts features from each frame and classifies videos. Whereas templated matching method is exploited for the feature extraction, the many-to-one LSTM model fulfils the video classification. Although the results are not better than our marker-based approach because of the limited performance of tracking method. More successful tracking algorithm may provide high performance in this problem.

3.6.1. Marker-less Dataset

Whereas digits were written in a larger font than standard handwriting in the marker-based dataset to be able to track the pen movement obviously, in this marker-

less dataset, the font size of the handwritten digit is so close to the exact handwriting font size. Furthermore, we record handwriting videos through two different perspectives simultaneously to get the required information about the pen location in the real world. The first camera records during the handwriting only the upper part of the pen (P1), without the hand from the side. However, in the latter perspective (P2), the entire handwriting process is visibly recorded, including hand, pen and the canvas in which it is written. There are 800 pairs of videos, take time about one second, with the size of 640x480 in the dataset. We also have the images of the handwritten digits because a digital drawing pad was utilized to write digits. As shown in Figure 3.10, there are two sample frame extracted simultaneous video pair and the image of digit which is handwritten during the videos.

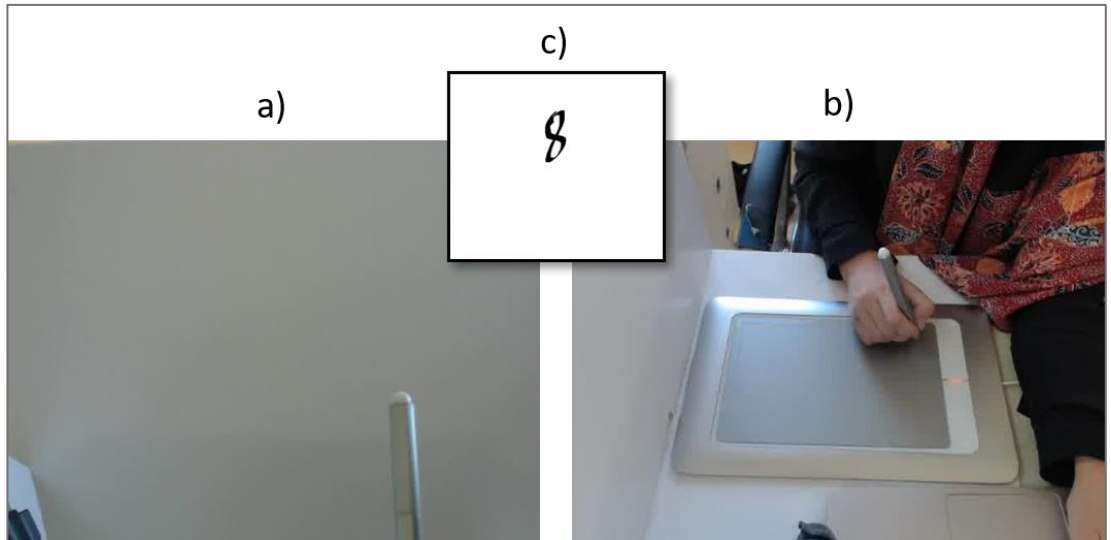


Figure 3.10: a) The frame of a video captured from the perspective one (P1). b) The frame of a video captured from the perspective two (P2). c) The image of the handwritten digit in the videos.

3.6.2. Feature Extraction and Classification

The aim is the same with the marker-based approach to classify videos according to handwritten digit excluding that in this problem, the font size of digit is smaller and there are no markers on the pen. In this case, we attempted various computer vision algorithms such as Scale-invariant Feature Transform (SIFT), Circle Hough Transform (CHT), and Template Matching to extract the features representing the top point of the pen. None of these algorithms except for Template Matching gave

satisfactory results because the features extracted by these algorithms do not belong to the same points in different frames. However, Template Matching [33] algorithm provides the information about pen's top point through the template image belonging to the upper part of the pen.

Whereas in the first step of the video classification, the pen's top point location is extracted from each frame of videos with Template Matching, in the latter, a simple LSTM network classifies the time-series data generated by concatenated extracted features. As shown in Figure 3.11, after the area shown with the yellow borders is detected by Template Matching algorithm, the location of the pen's top point marked by red, $[x_i, y_i]$, is extracted from i^{th} frame of the video, f_i . The pen locations extracted all frames are concatenated to be fed to LSTM network in two different approaches. Whereas in the former (A1), the difference in pen position between consecutive two frames is exploited, the latter (A2) benefits from sine and cosine transformations of the location. Finally, time-series data generated in one of the ways is fed to the LSTM network to classify according to handwritten digit in videos.

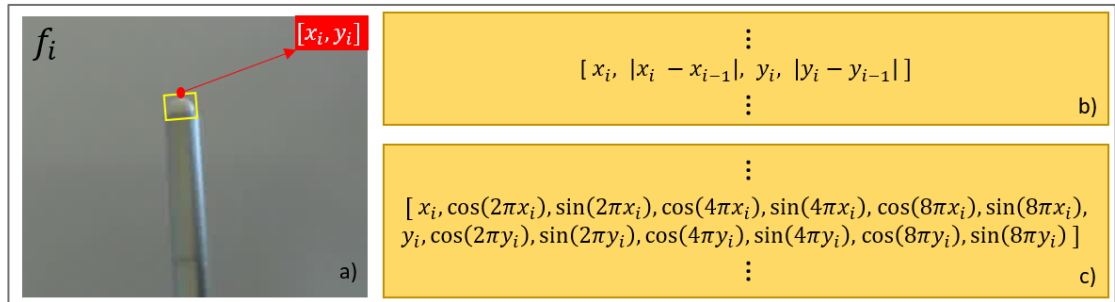


Figure 3.11: a) In the i^{th} frame of a video, detection of 2D location of pen's top point using template matching algorithm. b) In the approach one (A1), the difference between consecutive frames are exploited to generate time-series data. c) In the approach two (A2), cosine-sine transformations are exploited for the same target.

3.6.3. Experimental Results

In the marker-less dataset, two-step method is utilized for indirect handwriting recognition problem. The first step is the tracking of the pen throughout the video leveraging computer vision methods instead of markers for pen tracking in marker-less dataset. A single point is tracked throughout the video. This point is the topmost point of the pen. When the 2D location information was trained with a simple LSTM model, the classification is carried out with 50% accuracy. Whereas the performance

of LSTM model on marker-based dataset including 1000 sample is higher (88%, Figure 3.8) than the performance of marker-less dataset (including only P1) because the algorithm used for pen tracking does not yield as successfully as marker-based tracking.

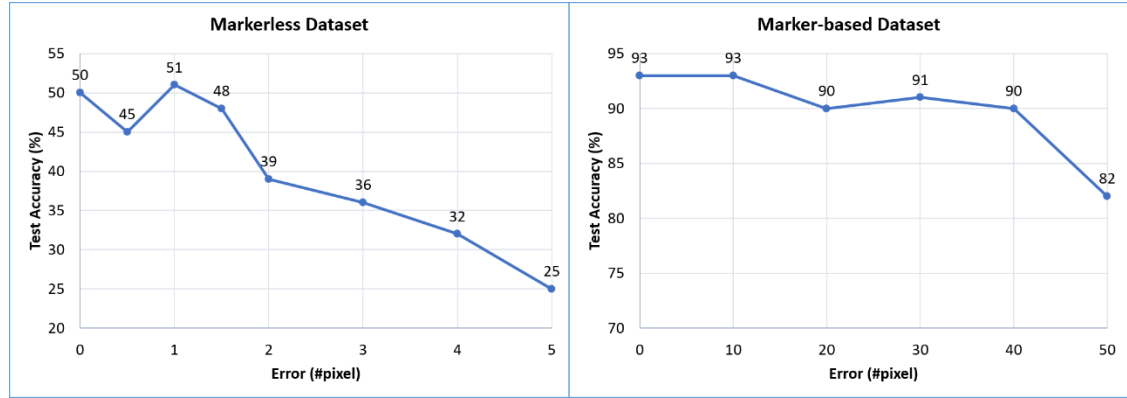


Figure 3.12: The effect of the error in the pen tracking on model performance. The pen movement in marker-less dataset is within 2 pixels range whereas the movement of the pen in the marker-based dataset is 20 pixels. The perturbation amount for each case is equal in proportion to the range of pen movement.

In this way, it is clear that errors in the tracking pen's top point decreases the model performance. In Figure 3.12, we added some pixel noise to the location information in both datasets gradually to show this relation between the tracking error and the classification performance. The amounts of noise expected to affect the performance are not same for both datasets. Whereas in the marker-based one, this amount is about 20 pixels since the size of the handwritten digit of the marker-based one is larger and the movement is spread over a wider area, in marker-less dataset, the amount of noise nearly 2 pixels because the movement covers a much smaller area than the former.

Instead processing only pen's top point 2D location, we trained the data with three different approach (A0, A1 and A2) using location difference between frames, using it with original position of pen, and cosine/sine transformations respectively. Moreover, the experiments are accomplished using only P1 videos or both P1 and P2 videos. They view only pen movement and all handwriting setup respectively. When Table 3.1 is examined, additional information utilized with the basic location information improves the classification accuracy from 50% up to 73% with different

approaches for input data generation. Data generation approach A2 gives better results compared with the others.

Table 3.1: The results obtained from different data generation approaches.

Model	Data Generation	Videos	Train Loss	Train Accuracy	Validation Loss	Validation Accuracy	Test Accuracy
M1	A0	P1	0.03	99%	2.40	60%	58%
M2	A0	P1&P2	0.02	99%	2.74	51%	57%
M3	A1	P1	0.03	99%	1.46	79%	61%
M4	A2	P1	0.13	97%	1.36	72%	73%
M5	A2	P1&P2	0.14	97%	1.31	66%	61%

With the same approach, the classification performance decreases when P1 and P2 videos are combined into input data, M4 and M5. Although the combination of two videos from different perspective decrease the performance, it is noticed that this way has a different benefit when confusion matrices of them are paid attention. When the average of class-based accuracy values is calculated, the average of M5 is higher than M4, which are 74% and 67%, unlike the total test accuracy values. Besides, M5 works better at discrimination of some pair of classes such as 0-6, 3-6, 8-6 and 4-7 because these classes are so similar that they can be difficult to distinguish when viewed from just a single perspective.



Figure 3.13: The comparison of the confusion matrices belonging to the models M4 and M5.

4. HAND SANITIZATION MONITORING

Successful handwashing performed applying certain hand movements specified by WHO is an essential procedure to maintain hygiene. In the scope of the thesis, we propose a hand sanitization monitoring framework to check whether this handwashing procedure is accomplished correctly. The framework aims to predict the actions performed during hand washing. Thus, it classifies each frame of video recorded when somebody is washing their hands in terms of the action class to evaluate the procedure. The classes are based on the hand hygiene norms of the WHO explained in the guideline [34]. In the development, we exploited a private dataset including videos collected in a hospital with the contribution of health-care staff. Hence, the framework is introduced to recognize handwashing actions in long untrimmed videos.

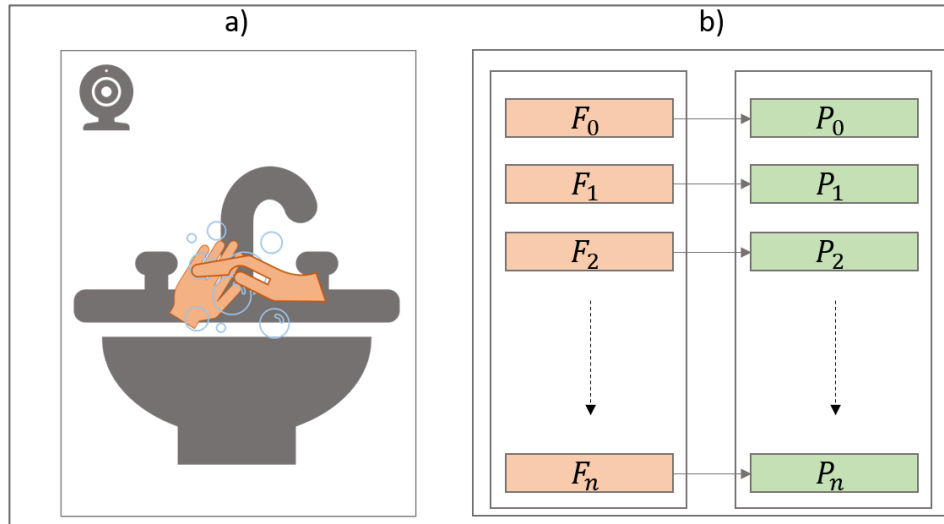


Figure 4.1: Problem definition of hand sanitization monitoring. a) Video recording during handwashing process. b) There are frames extracted from a video and the predictions of them according to handwashing actions.

Accordingly, there are several models developed in the scope of this thesis to recognize handwashing actions in videos. These models leverage deep learning methods to solve the problem which aims to classify every frame in the video as shown in Figure 4.1. Since video data contain both spatial and temporal information, CNN and LSTM are preferred to process this type of data.

4.1. Dataset

The dataset in which all experiments are carried out is created from a private company. Videos in this dataset were recorded by IP cameras while someone were washing their hands with water and soap. There are eighty-five videos recorded when more than five people were washing hands in two different sinks. Thus, the data contains videos collected in different light conditions and different backgrounds. Besides, videos even if they belong to same hospital staff may not contain actions in the same order as there is no order of action sequence to wash hands. In addition to the variety in the order in which movements are performed, the time duration of every action varies according to person washing hands and the type of action performed during hand washing, that causes variable length videos.

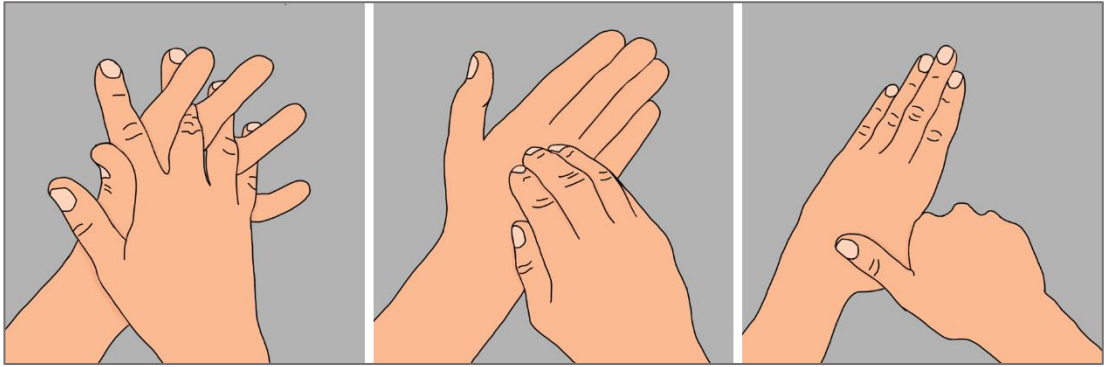


Figure 4.2: The illustration of some classes in the dataset.

After the video record, all of them have been labelled frame by frame. There are eleven classes representing handwashing actions, including mostly two hands interacting with each other in the dataset. These classes are defined according to which part of the hand, such as palm, fingertip, thumb of both hands is cleaned. Due to some issues such as the close interaction between the hands and fingers, and foam, the discrimination of actions is challenging even when viewed through the human eye. Some of these actions are illustrated in Figure 4.2. Moreover, recorded videos may contain a set of handwashing actions or all of them, the distribution of the classes in the dataset is unbalanced as shown in Figure 4.3. Because imbalance problem may influence the model achievement, we applied median frequency balancing [35] to overcome this problem.

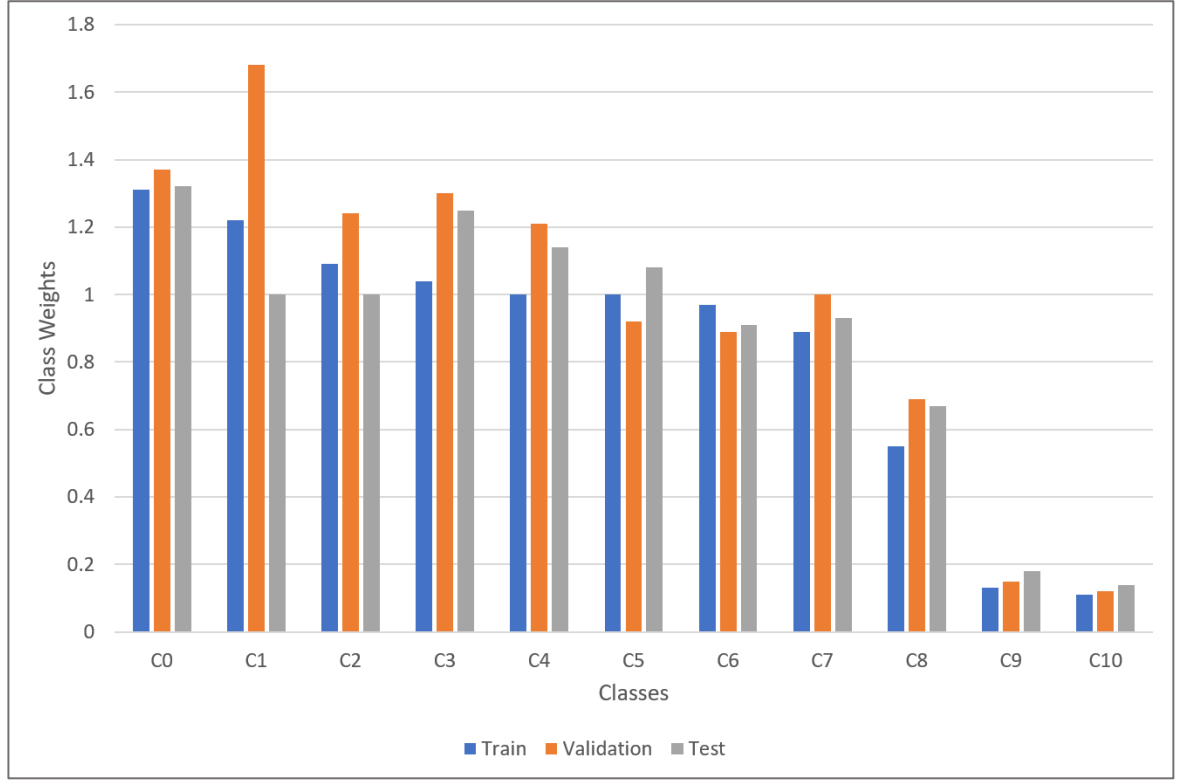


Figure 4.3: The weights of 11 classes in the dataset.

In the labeling stage, a tool developed to label recorded videos is exploited by some authorized people. The labeling tool enables to mark the start and end points of every action in the video. In this way, videos are divided into parts that contain only one action to label each frame of it in the same class. According to proposed models, each frame of the videos is classified as either independent from the video or with a certain number of neighbor frames. Therefore, frames are extracted from each video in the dataset. The total number of extracted frames is approximately 200K. Afterwards, the dataset was created by grouping entire frames extracted frames based on their labels. Thus, these frames can be exploited individually or a time-series image data.

4.2. Single-Frame Classification

The most straightforward way of hand washing action recognition in videos is considering each frame individually as regardless of the context. We propose a CNN-based model to classify each frame of the video according to the action. This model

can be considered as shallower version of DenseNet [22] architecture. Each layer in this model acquires additional input features from all prior layers to concatenate with its own input features and passes on its own feature-maps to all following layers. Thus, each layer obtains a collective knowledge from prior layers. Besides, this method enables to alleviate the vanishing gradient problem and substantially reduce the number of parameters.

There are two variables as number of dense layers in dense blocks and growth rate to tune adjust in the architecture. We tuned these variables to make the complexity of the model optimum to obtain the highest performance without overfitting. As a result of a set of experiments, three models with the highest performance score and required information about them are shown in Table 4.1. Color space can be RGB or YCbCr because images are fed to the model in these two different formats. Whereas RGB is the most widespread color space for this kind of models, YCbCr is exploited due to its ability to reflect human skin color more effectively.

Table 4.1: Proposed models with the architecture and input information.

Model	Color Space	Dense Layers	Growth Rate	Parameter Size
SF_RGB-1	RGB	[3,6,12,10]	32	~2.5M
SF_RGB-2	RGB	[6,12,24,16]	12	~980K
SF_RGB	RGB	[6,12,24,16]	18	~2.2M
SF_YCBCR	YCbCr	[6,12,24,16]	18	~2.2M

In Figure 4.4, the numbers under each volume represent the width and depth sizes, while those above represent the dimension of feature maps. Owing to the concatenation of new information with the prior volume, the volume following each dense block increases by the growth rate multiplied by the number of dense layers within that dense block. The proposed model consists of four dense blocks with the variable number of layers as 6,12,24, and 16 in order and its growth rate is 18. Besides, there is a transition layer between each two consecutive dense blocks to reduce by half the volume and the input size. Finally, there are approximately 2.2 million parameters in the model, fed to RGB image with the size of 224x224.

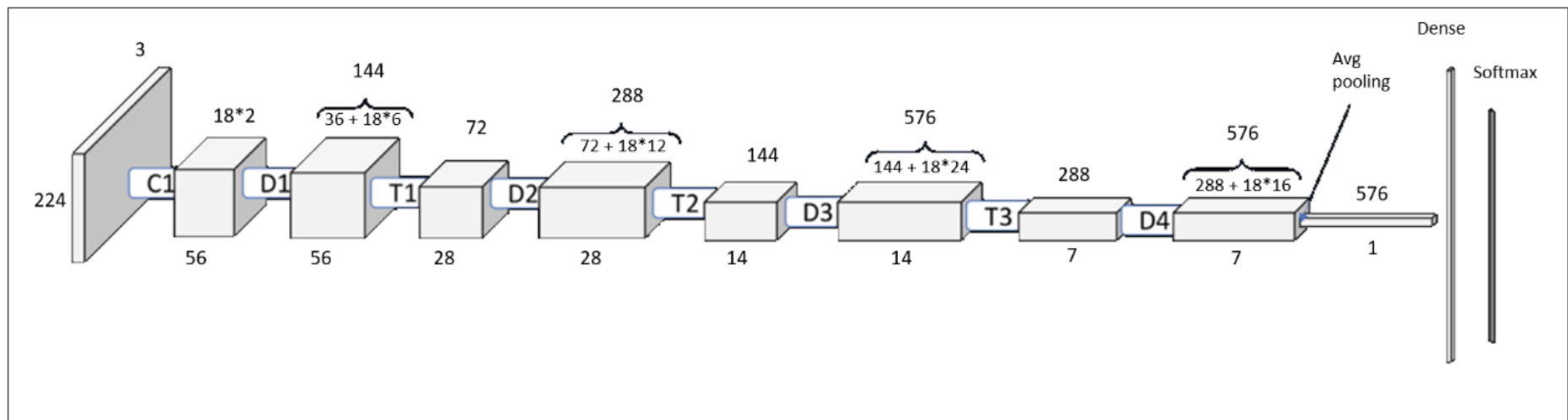


Figure 4.4: The model architecture of SF_RGB.

4.2.1. Single-Frame with Color-Based Skin Segmentation

There is a pre-processing stage before the main classification method. In this way, we applied color-based hand segmentation to improve the performance of single-frame classification model. Unlike the single-frame classification model, we concatenated 4th channel into standard RGB image to be fed as input. This additional channel represents the information acquired from hand segmentation stage. Whereas the first three channels belong to the frame in RGB color format, the 4th channel is a binary mask which represents hand area as white and background as black.

In color-based hand segmentation stage, input frames extracted from videos are pre-processed for background subtraction. In detail, we firstly convert the RGB image into YCbCr format before the classification of skin pixels. Let $\mathbf{x} = [Y, Cb, Cr]$ be a pixel in an input image. The classification rule is as follows: $\mathbf{x}$ is skin color if it satisfies the following condition: where Cb and Cr fall into the following ranges which are $[77,127]$ and $[133,179]$ respectively. The ranges are found experimentally to eliminate quickly non-skin colors. Afterwards, skin color pixels are represented as white color and the remainders as black color to generate binary hand mask. Therefore, an additional information is obtained to use with the input image of single-frame classification model. It is aimed to draw attention to the hand area during the recognition of handwashing action by concatenating the mask with the input image. Some hand masks are shown in Figure 4.5, acquired from frames belonging to different classes.



Figure 4.5: The results of color-based hand segmentation in some frames.

4.3. Multi-Frame Classification

Individually classification of each frame is not a comprehensive solution to the problem of action recognition in videos because the information originated from the sequential nature of the input is totally disregarded in this way. Therefore, we propose a multi-frame classification model which carries out the classification of frames considering prior and following frames as well. Whereas the input of the single-frame classification model is a 3-channel image, the input of this model is a 7-channel image as it consists of a set of sequential frames. With the alteration of the input, the model architecture was modified to cover the variation of the input data better. In this case, we updated the numbers of layers of the last two dense blocks and growth rate as 32 and 16 respectively to increase the complexity of the model.

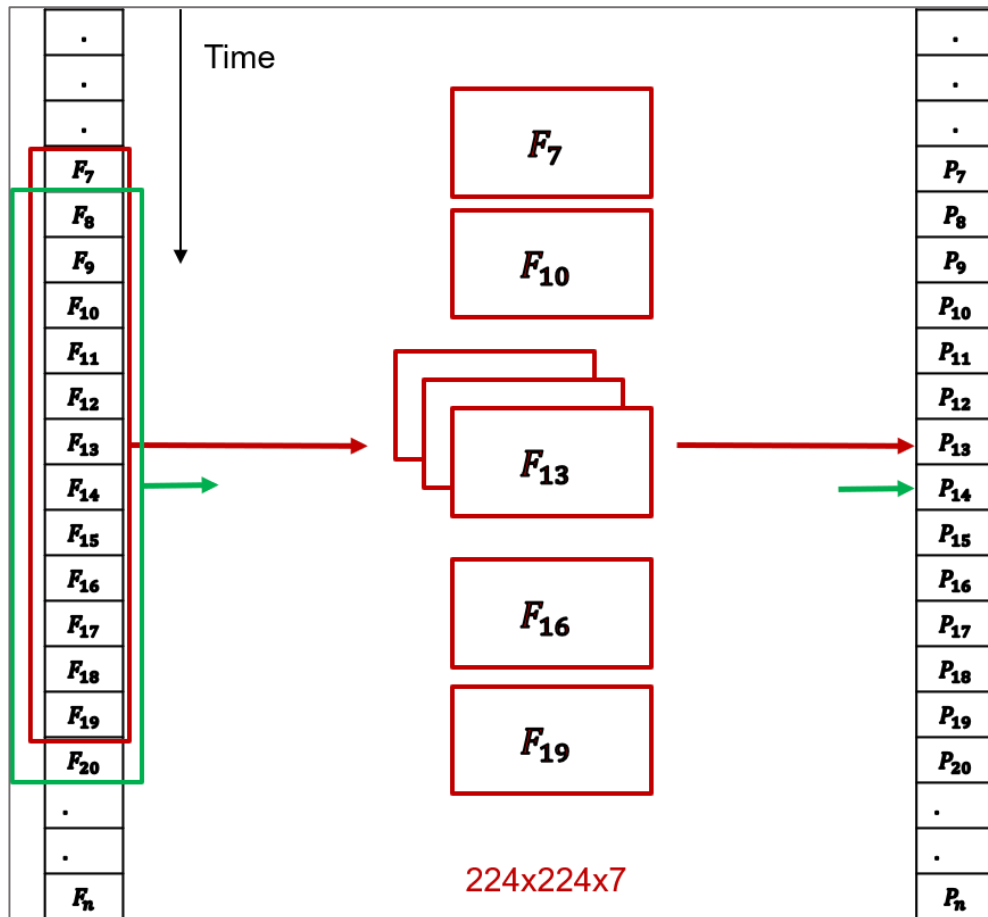


Figure 4.6: The working principle of sliding window approach for multi-frame classification.

When multiple frames are taken into account in the handwashing action recognition problem, the video is processed exploiting a sliding window approach with a stride of one. The sliding window is used over the entire video feeding the input frames to our model. As shown in Figure 4.6, the sliding window allocates thirteen frames in each step. However, the frames in the window are re-sampled to five to be used as input of the model by fetching the first one of every three frames because consecutive ones are almost the same. The model classifies the frame in the middle of the window with the help of its four neighbor frames. In addition to re-sampling, there is another preliminary process, decreasing the volume of input to prevent the negative effect of over-volumed structure. In this process, neighbor frames are converted grayscale to reduce the number of the channel to one third. In this way, the input of the model consists of seven channels, four of which come from the neighbors and three from the central frame because the central frame is in RGB format.

4.4. Feature Extraction followed by LSTM

Besides the multi-frame classification model, we propose LSTM-based model to process temporal information more efficiently due to CNN's limited ability in time-series data. This model consists of two sequential steps following feature extraction and LSTM. Therefore, both the ability of CNNs to process spatial information in images and the ability of LSTMs to process temporal information are exploited. In the feature extraction step, single-frame classification model is used except for the last classification layer. Thus, 576 dimensional vectors are extracted from each frames of the videos. On the other hand, a series of these feature vectors is fed to the simple LSTM network to classify the action performed in that frame series in the latter step, Figure 4.7. A sliding window approach like that it is in the multi-frame classification model is leveraged for this model. Unlike the multi-frame classification model, this model classifies the last frames of the window with the help of prior frames.

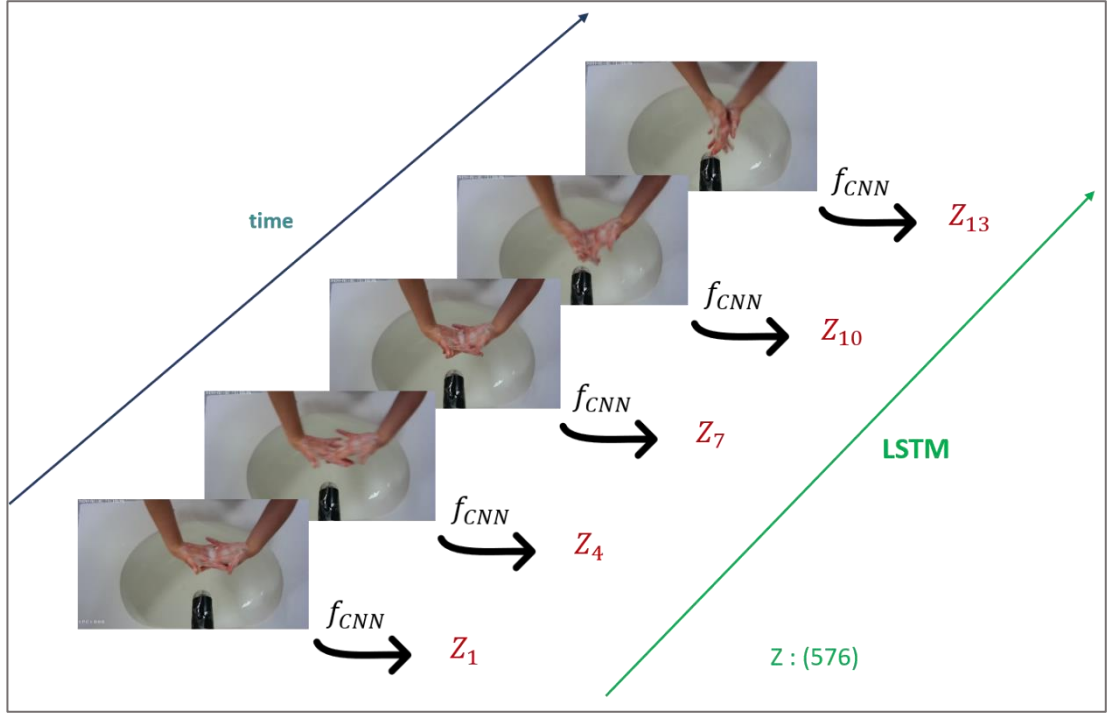


Figure 4.7: Feature extraction from consecutive frames to be fed to LSTM.

4.5. Experimental Results

There are some experiments to evaluate success of all models proposed for hand sanitization monitoring problem. We carried out all experiments in the same dataset including frames extracted handwashing videos. According the model, the input was prepared either any one frame in the dataset or the adjacency frames with the current frame. On the other hand, output of the model is the class of the current frame. Therefore, we first trained the frames with our models to classify the hand actions performed in there. We used Adam optimizer and Cross Categorical Loss function for training of all models.

In Table 4.2, there are the results of single-frame models with different inputs and multi-frame model. According to these results, processing frames in YCbCr color format instead of RGB improves the test accuracy from 77% to 85% due to its representing skin colors better. Adding the channel of binary mask, obtained from color-based segmentation to segment hands area, into RGB frames increases the accuracy 86%. However, adding mask into YCbCr frames is not successful as much as SF_RGB+Mask because it does not include the extra information coming from RGB format. Besides the improvement based on color format, using multi-frame in

classification also increases the test accuracy to 80% even if it is not better than SF_RGB+Mask because CNNs have limited ability to process multiple frames.

Table 4.2: The results of single-frame and multi-frame models.

Model			Loss	Accuracy	F1	Precision	Recall
SF_RGB		Train	0.39	94%	94%	95%	93%
Layers	[6, 12, 24, 16]	Validation	0.73	89%	89%	90%	88%
Growth Rate	18	Test	0.75	77%	76%	78%	75%
SF_YCbCr		Train	0.43	94%	94%	96%	93%
Layers	[6, 12, 24, 16]	Validation	0.68	90%	90%	91%	89%
Growth Rate	18	Test	0.69	85%	85%	87%	84%
SF_RGB+Mask		Train	0.44	94%	94%	95%	92%
Layers	[6, 16, 32, 16]	Validation	0.58	91%	91%	93%	89%
Growth Rate	16	Test	0.61	86%	86%	88%	84%
SF_YCbCr+Mask		Train	0.47	93%	93%	95%	92%
Layers	[6, 16, 32, 16]	Validation	0.64	90%	90%	93%	87%
Growth Rate	16	Test	0.65	84%	84%	86%	83%
MF_RGB		Train	0.59	92%	92%	94%	91%
Layers	[6, 12, 32, 32]	Validation	1.15	80%	81%	83%	79%
Growth Rate	16	Test	0.92	80%	80%	81%	79%

In Table 4.3, the results of single-frame models and LSTM-based models are listed. Unlike the multi-frame model, these LSTM-based models provide considerable improvement of the performance because they process temporal data efficiently. These models include feature extraction and classification stages. Whereas in SF_RGB+LSTM features are extracted after average pooling layer, in the post processing version of this model, the output of the Softmax layer is exploited as the input of LSTM. Therefore, post processing provides a simple filtering with the 86% accuracy while SF_RGB+LSTM increases the performance up to 90%. In Figure 4.8, there are ground truths and the prediction of SF_RGB and SF_RGB+LSTM model on the frames of video partition. In this way, it is clear that LSTM-based models function as a filter of miss-predicted frames.

Table 4.3: The results of single-frame models and LSTM-based models.

Model		Loss	Accuracy	F1	Precision	Recall
SF_RGB	Train	0.39	94%	94%	95%	93%
	Validation	0.73	89%	89%	90%	88%
	Test	0.75	77%	76%	78%	75%
SF_RGB + LSTM	Train	0.08	93%	93%	95%	91%
	Validation	0.32	91%	92%	92%	91%
	Test	0.39	90%	90%	90%	89%
SF_YCbCr	Train	0.43	94%	94%	96%	93%
	Validation	0.68	90%	90%	91%	89%
	Test	0.69	85%	85%	87%	84%
SF_YCbCr + LSTM	Train	0.10	93%	92%	95%	90%
	Validation	0.33	91%	91%	92%	90%
	Test	0.33	90%	91%	92%	90%

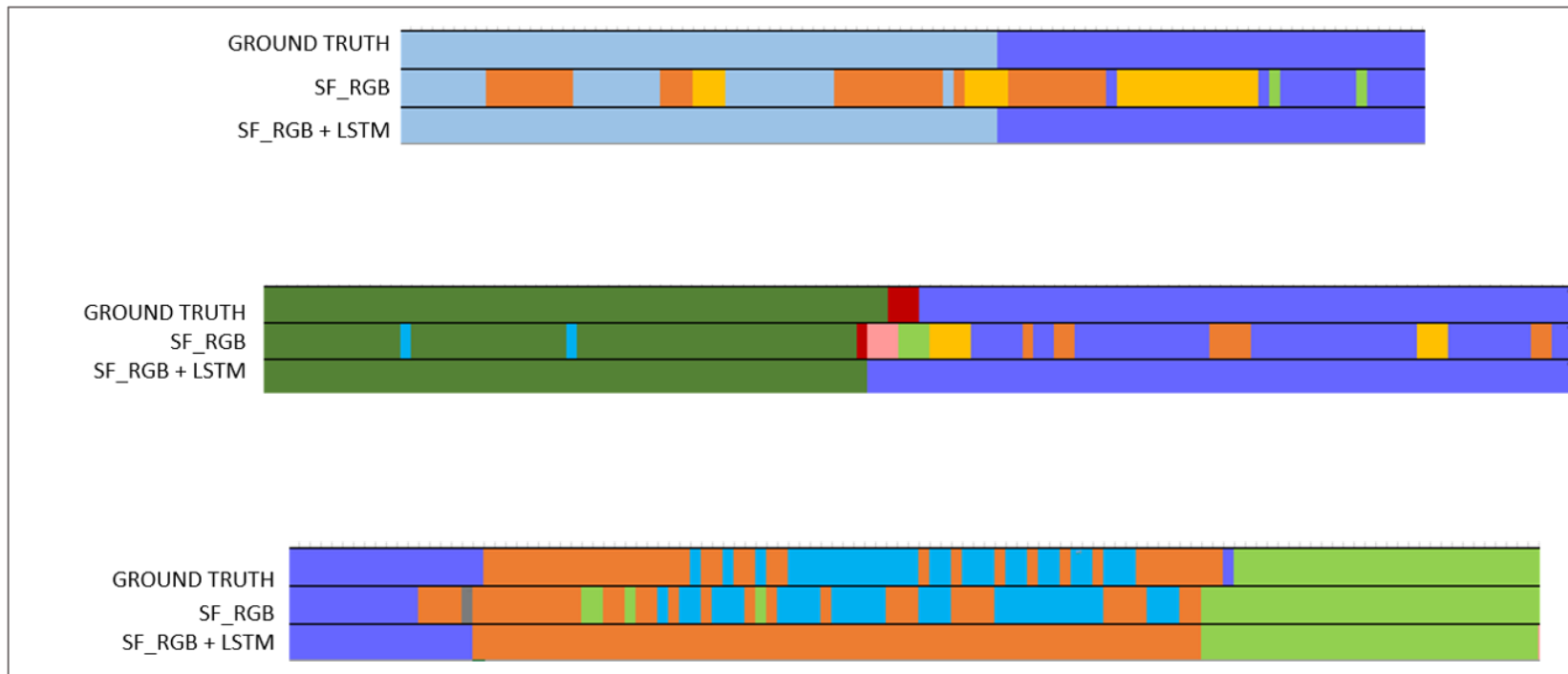


Figure 4.8: There are some video fragments shows the filtering by LSTM on the errors in single-frame classification results. All colors represent an action segment.

5. CONCLUSION AND FUTURE WORK

In this thesis, we address hand movement analysis in videos in terms of two different contexts as indirect handwriting recognition and hand sanitization monitoring. It is aimed in both contexts, that action recognition in videos including hand movement. In the former, we predict handwritten text while writing is in progress unlike the conventional methods which benefit from the image of handwritten text to recognize handwriting. Further, the latter recognizes hand actions performed during the handwashing process in videos. For the solution to these problems, a set of deep learning methods is developed that includes the ones exploiting spatial and temporal information in the videos recorded in the course of the handwriting or handwashing process is proposed. Because there is no comprehensive dataset for both problem, three different datasets are created in the scope of this thesis.

Essentially, prior works enable to interpret static handwritten texts in scanned documents. In contrast, the proposed approach dynamically fulfils handwriting recognition by observing hand and pen movement while just being written. In the latter context, an end-to-end model is developed to recognize hand actions performed during handwashing. The major challenge of this problem is to track two hands closely interacted with each other. However, our model directly carries out end-to-end hand actions classification without requiring overcoming this tracking challenge.

Firstly, there is no dataset to be evaluated our handwriting recognition solution. Therefore, some videos are recorded when someone is handwriting a digit. The aim is to predict which digit is written in the video. Since the created dataset has a limited sample and limited variety, the problem has been divided into two sequential steps to simplify the task. In the initial step, a marker tracking method [32] developed for AI applications has been used to track the pen through its markers. In this case, the pen movement indirectly represents the hand movement due to its being manipulated by hand.

In the next step of handwriting recognition, two CNN-based models have been proposed to classify videos according to digits. We introduce a novel method encoding temporal location data, extracted from a video, into a multi-channel image. One of both is formed up as basic CNN architecture for classification. However, multi-tasking one performs image generation task along with classification. This additional task

enables the model to classify more accurately using the image of the handwritten digit. Experimental results verify this improvement in these single and multi-tasking models whose accuracy rates are 83% and 85% in the most extensive dataset, respectively. Instead, the third model developed for this task leverages temporal location data directly. The last model overweighs the others with 94% accuracy considerably thanks to its basing on LSTM, one of the most successful ways to process temporal data.

Marker-based solution is not useful for indirect handwriting recognition because it is not practical in daily life. Thus, we propose another model to generalize the problem without markers that consists of two step including pen tracking and the classification of the time-series data obtained from pen tracking. Whereas the second is so similar with the marker-based LSTM model, in the first step, we leverage Template Matching algorithm to track pen. Location information extracted from each frames of videos through this algorithm are concatenated to generate time-series data for next step. When the generation of time-series data some handcrafted features including the position difference between frames and cosine-sine transformations are calculated to improve the performance. Thus, while the former increases the classification accuracy from 50%, the result obtained using only 2D pen position, to 60%, the latter increases it to 73%.

The second dataset is created for the generalization of indirect handwriting recognition problem by recording the handwriting with a simple pen without any marker. In the dataset, handwriting process recorded from two different perspective including observing only pen and all setup simultaneously. The combination of the both information coming from these videos contributes the discrimination of similar digits taking advantage of seeing the movement from different aspects. Because Template Matching algorithm is incompetent compared to marker-based approach for pen tracking, its performance is also lower accordingly. Therefore, the main task for this problem to be accomplished in the future is making the pen tracker more robust. In this way, a dataset containing more data may be created to track pen through deep learning methods. Besides, end-to-end solution needs to be developed to recognize all letters when somebody is writing.

Our second problem is hand sanitization monitoring. In this thesis, a system recognizing handwashing actions has been developed leveraging deep learning techniques. Three primary approaches have been developed for this hand action recognition problem. Firstly, each frame of videos is considered as an independent

image. The developed model for this aims to classify individual images according to the handwashing actions to which they belong. DenseNet-based models classify RGB images into eleven class with %83 accuracy rate in the test set. Besides, the model has been improved being altered the colour format to YCbCr. The high performance of YCbCr space in the representation of skin parts causes an increase from 83% to 87% in the model's success.

Our second approach is to classify frames by regarding them with their neighbours. Images are concatenated with previous frames and next frames in a multi-channel image to feed to DenseNet model. The expectation is to provide a better understanding of the movement over time. In this case, window size and the gap between frames affects the results of this model. Lastly, frame-based action recognition is so limited approach. Prediction of a movement requires to process an interval instead of only one frame. Therefore, an LSTM network is proposed to process temporal data extracted from each frame of a video. This model enables to removing miss-predicted frames within a series of frames with same labels. Convolutional LSTM can be an alternative for this model. However, to train this type network requires larger machines. As a result of this thesis, hand movement analysis has been fulfilled leveraging computer vision and deep learning methods. In both context, models with LSTM are more successful because of the success of temporal data processing.

The next work may be action segmentation to improve the performance of the hand actions recognition because it is difficult for this problem to recognize action segments, especially with various action classes, which end up with continuously fluctuating action class predictions. Detection of action boundaries is crucial when processing untrimmed videos to improve the action segmentation performance. To address this problem, Graph Convolutional Networks may be beneficial by representing the actions and their relations as nodes of the graph.

REFERENCES

- [1] Ghosh M. M. A., Maghari A. Y., (2017), “A comparative study on handwriting digit recognition using neural networks”, Proc. - 2017 Int. Conf. Promis. Electron. Technol. ICPET 2017, 77–81.
- [2] Oikonomidis I., Kyriazis N., Argyros A. A., (2012), “Tracking the Articulated Motion of Two Strongly Interacting Hands”, IEEE Conference on Computer Vision and Pattern Recognition, 1862–1869.
- [3] Wang R. Y., Popović J., (2009), “Real-time hand-tracking with a color glove”, ACM Transactions on Graphics, 28(3), 1–8.
- [4] Duca F., Fredriksson J., Fjeld M., (2007), “Real-Time 3D Hand Interaction : Single Webcam Low-Cost Approach”, Proceedings of the Workshop at the IEEE Virtual Reality 2007 Conference; Trends and Issues in Tracking for Virtual Environments, 1–5.
- [5] Zhao W., Chai J., Xu Y. Q., (2012), “Combining Marker-based Mocap and RGB-D Camera for Acquiring High-fidelity Hand Motion Data”, Proceedings of the ACM SIGGRAPH/eurographics symposium on computer animation, 33–42.
- [6] Sridhar S., Oulasvirta A., Theobalt C., (2013), “Interactive Markerless Articulated Hand Motion Tracking Using RGB and Depth Data”, Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2456-2463.
- [7] Sridhar S., Mueller F., Zollh M., Casas D., Oulasvirta A., Theobalt C., (2016), “Real-Time Joint Tracking of a Hand Manipulating an Object from RGB-D Input”, European Conference on Computer Vision, 294–310.
- [8] Oikonomidis I., Kyriazis N., Argyros A. A., (2011), “Efficient Model-based 3D Tracking of Hand Articulations using Kinect”, BmVC, 1(2), 3.
- [9] Ballan L., Taneja A., Gall J., Van Gool L., Pollefeys M., (2012), “Motion Capture of Hands in Action Using Discriminative Salient Points”, European Conference on Computer Vision, 640–653.
- [10] Rehg J. M., Kanade T., (1994), “Visual Tracking of High DOF Articulated Structures : an Application to Human Hand Tracking”, European conference on computer vision, 35-46.
- [11] Lee Y. J., Lee D. H., (2008), “Research on Detecting Face and Hands for Motion-based Game Using Web Camera”, 2008 International Conference on Security Technology, 7-12.
- [12] Lee T., Tobias H., (2007), “Handy AR : Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking”, 2007 11th IEEE International

- [13] Lee T., Höllerer T., (2009), “Multithreaded hybrid feature tracking for markerless augmented reality”, *IEEE transactions on visualization and computer graphics*, 15(3), 355–368.
- [14] Manresa C., Varona J., Mas R., Perales F. J., (2005), “Hand Tracking and Gesture Recognition for Human-Computer Interaction”, *ELCVIA Electronic Letters on Computer Vision and Image Analysis*, 5(3), 96-104,
- [15] Liang H., Yuan J., Thalmann D., Magnenat-Thalmann N., (2015), “AR in Hand: Egocentric Palm Pose Tracking and Gesture Recognition for Augmented Reality Applications”, *Proceedings of the 23rd ACM international conference on Multimedia*, 743-744.
- [16] Lee J. Y., Rhee G. W., Seo D. W., (2010), “Hand gesture-based tangible interactions for manipulating virtual objects in a mixed reality environment”, *The International Journal of Advanced Manufacturing Technology*, 51(9–12), 1069–1082.
- [17] Huang J., Zhou W., Zhang Q., Li H., Li W., (2018), “Video-based Sign Language Recognition without Temporal Segmentation”, *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).
- [18] Guerra-Casanova J., Sánchez-Ávila C., Bailador G., de Santos Sierra A., (2012), “Authentication in mobile devices through hand gesture recognition”, *International Journal of Information Security*, 11(2), 65–83.
- [19] Zhang X., Chen X., Li Y., Lantz V., Wang K., Yang J., (2011), “A framework for hand gesture recognition based on accelerometer and EMG sensors”, *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 41(6), 1064–1076.
- [20] Liu C., Yang Y., Liu X., Fang L., Kang W., (2020), “Dynamic-Hand-Gesture Authentication Dataset and Benchmark”, *IEEE Transactions on Information Forensics and Security*, 16, 1550-1562.
- [21] Ar I., Akgul Y. S., (2014), “A computerized recognition system for the home-based physiotherapy exercises using an RGBD camera”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(6), 1160–1171.
- [22] Huang G., Liu Z., Van Der Maaten L., Weinberger K. Q., (2017), “Densely Connected Convolutional Networks”, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4700-4708 .
- [23] Bao P., Maqueda A. I., Del-Blanco C. R., Garcíá N., (2017), “Tiny hand gesture recognition without localization via a deep convolutional network”, *IEEE Transactions on Consumer Electronics*, 63(3), 251–257.
- [24] Simonyan K., Zisserman A., (2014), “Two-Stream Convolutional Networks for

Action Recognition in Videos”, arXiv preprint arXiv:1406.2199.

- [25] Wang X., Farhadi A., Gupta A., (2016), “Actions $\leftrightarrow$ Transformations”, Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2658-2667.
- [26] Köpüklü O., Gunduz A., Kose N., Rigoll G., (2019), “Real-time hand gesture detection and classification using convolutional neural networks”, 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition, 1-8.
- [27] Molchanov P., Gupta S., Kim K., Kautz J., (2015), “Hand Gesture Recognition with 3D Convolutional Neural Networks”, Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 1-7.
- [28] Al-Hammadi M., Muhammad G., Abdul W., Alsulaiman M., Hossain M. S., (2019), “Hand Gesture Recognition Using 3D-CNN Model”, IEEE Consumer Electronics Magazine, 9(1), 95–101.
- [29] Ullah A., Ahmad J., Muhammad K., Sajjad M., Baik S. W., (2017), “Action Recognition in Video Sequences using Deep Bi-Directional LSTM with CNN Features”, IEEE Access, 6, 1155–1166.
- [30] Donahue J., Anne Hendricks L., Guadarrama S., Rohrbach M., Venugopalan S., Saenko K., Darrell T., (2015) “Long-term Recurrent Convolutional Networks for Visual Recognition and Description”, Proceedings of the IEEE conference on computer vision and pattern recognition, 2625-2634.
- [31] Naguri C. R. Bunescu R. C., (2017), “Recognition of dynamic hand gestures from 3D motion data using LSTM and CNN architectures”, 2017 16th IEEE International Conference on Machine Learning and Applications, 1130–1133.
- [32] Romero-Ramirez F. J., Muñoz-Salinas R., Medina-Carnicer R., (2018), “Speeded up detection of squared fiducial markers”, Image and vision Computing, 76, 38–47.
- [33] Brunelli R., (2009), “Template matching techniques in computer vision: theory and practice”, John Wiley & Sons.
- [34] Web 1, (2021),
http://apps.who.int/iris/bitstream/handle/10665/44102/9789241597906_eng.pdf;jsessionid=5FCA460BF882961840326820865681E8?sequence=1.
(Accessed: May 01, 2021).
- [35] Badrinarayanan V., Kendall A., Cipolla R., (2017), “Segnet: A deep convolutional encoder-decoder architecture for image segmentation”, IEEE transactions on pattern analysis and machine intelligence, 39(12), 2481-2495.

BIOGRAPHY

Beyza Türkmen was born in İstanbul in 1995. She took her bachelor's degree from the Computer Engineering Department of Gebze Technical University in February 2018. Since the following semester, she has been a master student at the Computer Engineering Department of Gebze Technical University. In 2019, she was awarded a scholarship by the Ministry of National Education to be sent abroad for graduate education. In addition to her academic career, she has been working on an Embedded AI project based on computer vision and deep learning as Computer Vision Engineer at ERG Controls for approximately two years.

APPENDICES

Appendix A: The Publications about the Thesis

Türkmen B., Genç Y., (2020), "Handwriting Recognition via Visual Observation of Pen Movements", 5th International Conference on Computer Science and Engineering (UBMK), Diyarbakır, Turkey, 9-11 Sept.