HUMAN-COMPUTER INTERACTION PLATFORM FOR THE HEARING IMPAIRED IN HEALTHCARE AND FINANCE APPLICATIONS

by

Necati Cihan Camgöz

B.S., in Computer Engineering, Yıldız Technical University, 2013

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2016

HUMAN-COMPUTER INTERACTION PLATFORM FOR THE HEARING IMPAIRED IN HEALTHCARE AND FINANCE APPLICATIONS

APPROVED BY:

Prof. Lale Akarun .....
(Thesis Supervisor)

Assist. Prof. Bert Arnrich .....

Assist. Prof. Berk Gökberk .....

DATE OF APPROVAL: 18.01.2016

ACKNOWLEDGEMENTS

First and foremost, I would like to thank Prof. Lale Akarun for her endless support, guidance and wisdom that she has provided me throughout my masters education. Without her, I wouldn't be the researcher I am today.

I would like to thank Berk Gökberk for all the invaluable advice he has given and the patience he has shown during our collaborations. I also would like to thank Bert Arnrich and Berk Gökberk for accepting to be on my thesis committee and for their insightful comments on my thesis.

I thank Prof. Sumru Özsoy, Meltem Kelepir and Serpil Karabüklü for sharing their extensive knowledge on Turkish Sign Language. I would also like to thank Elvan Tamyürek Özparlak for her dedication and understanding during the collection of BosphorusSign.

I would like to thank Ali Salah for believing in me and creating opportunities for me by which I had the chance to grow my academic network. I would also like to thank Vitomir Struc who hosted me in University of Ljubljana and guided me in my research.

I thank all my friends and colleagues in Perceptual Intelligence Laboratory for their support and endless understanding, namely, Alp Kındıroğlu, Alper Bozkurt, Barış Evrim Demiröz, Barış Kurt, Burak Kurutmaz, Burak Kadron, Çağatay Yıldız, Deniz Akyıldız, Doğa Siyli, Gül Varol, Hakan Güldaş, Hazal Koptagel, Heysem Kaya, Miraç Süzgün, Oğulcan Özdemir, Orhan Sönmez, Taha Çeritli, Umut Şimşekli and Yunus Emre Kara.

I would also like to thank all my friends in the computer engineering department, who motivated me to work harder by raising my spirit and providing me with their dearest friendship, namely, Bahar İrfan, Berkant Kepez, Binnur Görer, Gaye Genç,

Metehan Doyran, Nefise Yağlıkçı, Okan Aşık, Serhan Daniş, and Yiğit Yıldırım.

Last but not least, I would like to thank my family and my dearest friend Nimet Kaygusuz for their endless and unconditional patience, support and understanding.

This thesis has been supported by the M.Sc. Scholarship (2228) from the Scientific and Technological Research Council of Turkey (TÜBİTAK) and by the Industrial Thesis Program (SANTEZ) from the Republic of Turkey, Ministry of Science, Industry and Technology, Project No: 0341.STZ.2013-2.

ABSTRACT

HUMAN-COMPUTER INTERACTION PLATFORM FOR THE HEARING IMPAIRED IN HEALTHCARE AND FINANCE APPLICATIONS

In this thesis, we propose a human-computer interaction platform for the hearing impaired, that would be used in hospitals and banks. In order to develop such a system, we collected BosphorusSign, a Turkish Sign Language corpus in health and finance domains, by consulting sign language linguists, native users and domain specialists. Using a subset of the collected corpus, we have designed a prototype system, which we called HospiSign, that is aimed to help the Deaf in their hospital visits. The HospiSign platform guides its users through a tree-based activity diagram by asking specific questions and requiring the users to answer from the given options. In order to recognize signs that are given as answers to the interaction platform, we proposed using hand position, hand shape, hand movement and upper body pose features to represent signs. To model the temporal aspect of the signs we used Dynamic Time Warping and Temporal Templates. The classification of the signs are done using k-Nearest Neighbors and Random Decision Forest classifiers. We conducted experiments on a subset of BosphorusSign and evaluated the effectiveness of the system in terms of features, temporal modeling techniques and classification methods. In our experiments, the combination of hand position and hand movement features yielded the highest recognition performance while both of the temporal modeling and classification methods gave competitive results. Moreover, we investigated the effects of using a tree-based activity diagram and found the approach to not only increase the recognition performance, but also ease the adaptation of the users to the system. Furthermore, we investigated domain adaptation and facial landmark localization techniques and examined their applicability to the gesture and sign language recognition tasks.

ÖZET

SAĞLIK VE FİNANS UYGULAMALARINDA İŞİTME ENGELLİLERİN İŞARET DİLİ İLE BİLGİSAYAR ETKİLEŞİM PLATFORMU

Bu tezde, işitme engellilerin hastane ve bankalarda kullanmaları amacıyla tasarlanmış bir insan-bilgisayar etkileşim platformu önerilmektedir. Söz konusu sistemin geliştirilmesi için öncelikle sağlık ve finans alanlarında işaretler içerin BosphorusSign Türk İşaret Dili veritabanı toplanmıştır. Veritabanı için dil bilimcilere, Türk İşaret Dili kullanıcılarına ve ilgili alanların uzmanlarına danışılmıştır. Toplanan veritabanının bir alt kümesi kullanılarak hastanelerde işitme engellilerin iletişimine yardımcı olacak HospiSign sistemi tasarlanmıştır. HospiSign platformu kullanıcılarına önceden belirlenmiş soruları ve bu sorulara verebileceği cevapları sunarak, ağaç tabanlı bir etkinlik diyagramı ile kullanıcıları yönlendirmektedir. HospiSign'a cevap olarak verilen işaretleri tanıyabilmek için ellerin şeklini, pozisyonunu, hareketini, ve üst vücudun duruşunu niteleyen öznitelikler kullanılmıştır. İşaretlerin zamansal özellikleri Dinamik Zaman Bükmesi ve Zamansal Şablonlar kullanılarak modellenmiştir. İşaretler k-En Yakın Komşu algoritması ve Rassal Karar Ormanları kullanılarak sınıflandırılmaktadır. Sistemin öznitelik, zamansal modelleme ve sınıflandırma yönlerinden değerlendirilmesi için BosphorusSign veritabanının bir alt kümesinde deneyler yapılmıştır. Deneyler sonucunda ellerin pozisyonu ve hareketini niteleyen özniteliklerin birlikte kulanımının en yüksek başarımı verdiği görülmüştür. Farklı zamansal modelleme ve sınıflandırma yaklaşımlarından ise birbirlerine yakın başarımlar elde edilmiştir. Yapılan deneyler sonucunda ağaç tabanlı etkinlik diyagramı kullanımının sistem başarımını arttırmanın yanı sıra kullanıcı adaptasyonunu da hızlandırdığı görülmüştür. Bu çalışmalara ek olarak, alan uyarlama ve yüzel nirengi noktası bulma yöntemleri incelenmiş, ve işaret dili ve işmar tanıma problemleri için kullanılabilirlikleri sınanmıştır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

b_k	Bias term in the k^{th} iteration
D_S	Source domain in transfer learning
D_T	Target domain in transfer learning
f_S	Features from the source domain
f_T	Features from the target domain
F	Number of features dimensions
I	Identity matrix
M	Number of mixture models
N	Number of hidden states
O_L	Resampled observation length
$\mathbb{R}$	Real numbers
R_k	Descent direction in the k^{th} iteration
S	Features space of F dimensions
$\breve{S}$	Augmented features space of $3F$ dimensions
T_{CTS}	Combined target and source domain data
T_{DATS}	Domain adapted target and source domain data
T_S	Only using source domain data
T_T	Only using target domain data
x_*	Ground truth facial landmark locations
x_0	Initial facial landmark locations
x_k	Facial landmark locations in the k^{th} iteration
X	Observations of the predictors
β_k	Ridge regression estimator in the k^{th} iteration
Γ	Regularization term
Δx	Distance between the ground truth and the current location
ΔX_k	Concatenated distance matrix in the k^{th} iteration
θ	Trajectory angle

λ_k	Regularization term in the k^{th} iteration
ϕ_k	Local features of the landmarks in the k^{th} iteration
Φ_k	Concatenated observation matrix in the k^{th} iteration
Ψ^S	Source domain mapping function
Ψ^T	Target domain mapping function

LIST OF ACRONYMS/ABBREVIATIONS

*.EAF	ELAN Annotation Files
#LM	Number of Landmarks
1080p	1920*1080 pixels video resolution
2D	Two Dimensional
3D	Three Dimensional
AFS	Adaptive Feature Size
API	Application Program Interface
ASL	American Sign Language
ASPC	Asymmetry Patterns Shape Contexts
BSL	British Sign Language
CSE	Czech Sign Language
CSL	Chinese Sign Language
CSV	Comma Separated Values
CV	Computer Vision
DGS	German Sign Language
DTW	Dynamic Time Warping
ELAN	European Language Activity Network
FFMPEG	Fast Forward Moving Pictures Expert Group
FFS	Fixed Feature Size
FSN	Finite State Network
GB	Gigabyte
GSL	Greek Sign Language
HamNoSys	Hamburg Notation System
HMM	Hidden Markov Model
HOG	Histogram of Oriented Gradients
HOG-H	High Detailed Histogram of Oriented Gradients
HOG-L	Low Detailed Histogram of Oriented Gradients
HOG-M	Medium Detailed Histogram of Oriented Gradients

ID	User identification
IS	Interval Steps
k-NN	k-Nearest Neighbors
KB	Kilobyte
LDA	Linear Discriminant Analysis
LGP	Portuguese Sign Language
LSE	Spanish Sign Language
LSF	French Sign Language
LSR	Least Squares Regression
MB	Megabyte
MP4	Moving Picture Expert Group Layer-4
N/A	Not available
PaHMM	Parallel Hidden Markov Models
PC	Personal Computer
PCA	Principle Component Analysis
RDF	Random Decision Forests
RDT	Random Decision Trees
RGB	Red Green Blue
RR	Ridge Regression
RSL	Russian Sign Language
SASS	Size and Shape Specifiers
SDK	Software Development Kit
SDM	Supervised Descent Method
SIFT	Scale Invariant Feature Transform
SL	Sign Language
SLR	Sign Language Recognition
SRD	Supervised Ridge Descent
TESSA	Text and Sign Support Assistant
TİD	Türk İşaret Dili, Turkish Sign Language
TS	Template Size
TT	Temporal Templates

v2	Version 2
VC	Video Camera
W/O	Without
WPF	Windows Presentation Foundation
Z-Norm	Z-Normalization

1. INTRODUCTION

According to the 2000 Turkish Statistical Institute census, there are 109.000 people with total hearing disability in Turkey. In their daily routines, the hearing impaired are forced to use either written materials or the aid of an accompanying Turkish Sign Language (Türk İşaret Dili, TİD) interpreter to establish basic communication since they are unable to use speech as a medium of communication. The staggeringly low literacy rate among the hearing impaired greatly reduces the integration of this population, thus creating both a social and an economic disadvantage.

Sign Languages are the main communication medium of the hearing impaired people. These languages differ from country to country. Turkish Sign Language is used by hearing impaired people of Turkish origin. Like other sign languages, TİD conveys meaning through combinations of hand shapes, hand movements, facial gestures and upper body postures that are unique to it. In 2005, as part of the European Union integration effort, the Turkish Social Services Law made it mandatory for all governmental organizations and offices to employ a TİD interpreter. In 2006, the use of TİD and the task of training TİD interpreters were introduced through regulations into the Turkish Social Services Law. Since it is neither practicable nor economically feasible to train and employ TİD interpreters for every government office, practical solutions like establishing sign language translation call centers that employ TİD interpreters are considered. The ideal solution to such an issue is the recognition and translation of sign languages to speech through the use of technology. Such a system would diminish the need for sign language interpreters, while fully integrating the hearing impaired. However, the technology needed for such a system is beyond the current state-of-the-art. Today's technology only allows for sign-to-speech translation systems with extremely limited corpora that are heavily user dependent.

In this thesis, we present a human-computer interaction platform that is designed to assist the Deaf in their hospital and bank visits. In order to develop the proposed system, we first developed a corpus collection software which records depth map, color

video, user mask and human body pose information that are provided by the Microsoft Kinect v2 sensor. By consulting with healthcare and finance professionals, TİD linguists and native TİD users, we have created a list of commonly used words and phrases that a deaf person would use in order to receive healthcare and finance services. Using our corpus collection software, we collected BosphorusSign, a Turkish Sign Language Recognition (SLR) corpus in health and finance domains. Using a subset of BosphorusSign, we created HospiSign, an interface aimed for healthcare services, that allows its user to communicate in sign language, by recognizing signed phrases. In HospiSign, we proposed a tree-based activity diagram interaction scheme that asks specific questions and requires users to answer from given options, thus easing the recognition task. Furthermore, the interface software was designed to be easily converted to other applications, such as banking applications.

We proposed using several features, temporal modeling techniques and classification methods for sign language recognition. As features, we extracted upper body pose, hand shape, hand position and hand movement features from the data provided by the Microsoft Kinect v2 sensor to model the spatial features of the signs. We model the temporal aspect of the signs by using Dynamic Time Warping (DTW) and Temporal Templates (TT). Finally, we classify spatio-temporal features extracted from the isolated sign phrases using k-Nearest Neighbors (k-NN) and Random Decision Forest (RDF) classifiers.

We evaluated the performance of the proposed recognition scheme on a subset of the BosphorusSign corpus, that contains a total of 662 samples of 33 signs, which were collected from three native TİD users. We investigated each feature's effect on the recognition performance and compared temporal modeling and classification approaches. In our experiments, combining hand position and hand movement features achieved the highest recognition performance while both of the temporal modeling and classification approaches yielded similar recognition results. Moreover, we inspected the outcome of using the tree-based activity diagram interaction scheme and came to the conclusion that this approach not only increases the recognition performance, but also fastens the users' adaptation to the system.

Furthermore, we examined the applicability of domain adaptation techniques to the hand gesture recognition problem. We proposed using a feature augmentation method, namely Frustratingly Easy Domain Adaptation [3], to improve hand drawn digit gesture recognition performance by transferring information from hand written digits. In addition, we studied facial landmark localization methods that are widely used in sign language recognition. We proposed an extension to the Supervised Descent Method (SDM) [1], which is the state-of-the-art facial landmark localization method in color images, and adapted the method to work on depth images. The proposed method, which we called Supervised Ridge Descent (SRD) [4], achieved state-of-the-art performance on frontal depth images.

The rest of this thesis is organized as follows. We review the state-of-the-art Sign Language Recognition methods, applications and available sign language corpora in Chapter 2. We present the BosphorusSign TİD corpus in Chapter 3 and specify the corpus collection procedure. In Chapter 4, we describe the HospiSign interface. We layout our proposed method in terms of extracted features, applied normalization techniques, proposed temporal modeling approaches and used classification methods in Chapter 5. We define our experimental setup and report our results in Chapter 6. We share our work on the application of domain adaptation techniques to the gesture recognition problem in Chapter 7 and the proposed facial landmark localization method, Supervised Descent Method, in Chapter 8. Finally, we discuss our findings and state the possible future work in Chapter 9.

2. SIGN LANGUAGE RECOGNITION METHODS, APPLICATIONS AND AVAILABLE CORPORA

Many hearing-impaired people cannot express themselves clearly in public since they are unable to use speech as a medium of communication, yet a large part of the hearing population cannot communicate with the deaf because they do not know sign language. In some cases, this challenge may be solved with either the use of an interpreter or through written material. However, many hearing-impaired people do not know how to read and write. In case of emergencies where the time is extremely valuable, such as when a Deaf person visiting a hospital with an urgent issue, the inability to communicate becomes a more pressing problem. A possible solution to this problem is using Sign Language Recognition systems to create a communication medium for the Deaf.

With the development of machine learning and computer vision algorithms and the availability of different sign language databases, there has been an increasing number of studies in Sign Language Recognition (SLR). A fundamental problem in sign language research is that signs are multimodal time series, meaning many signals are sent simultaneously to express meaning through hand and body movements, and therefore it is hard to spot and model these modalities in consecutive frames [5].

Among many methods, Hidden Markov Models (HMMs) [6] and Dynamic Time Warping (DTW) [7] based methods are still the most popular machine learning techniques to solve the modeling problem. Both of these methods are widely used in applications ranging from speech recognition to robot tracking. Starner and Pentland [8] introduced a real-time HMM-based system that can recognize American Sign Language phrases in sentence-level without any explicit model of the fingers. In a signer-dependent platform, Grobel and Assan [9] achieved a recognition rate of 94% on an isolated sign database that included 262 signs of the Sign Language of the Netherlands. Other approaches, such as Parallel Hidden Markov Models (PaHMMs) [10] and

HMM-based threshold model [11], are also used in gesture and sign language recognition systems. Chai et al. [12] used DTW based classifiers to develop a translation system similar to HospiSign, as it interprets Chinese Sign Language to Spoken Language and vice versa. In more recent studies, Pitsikalis and Theodorakis et al. [13, 14] used DTW to match subunits in Greek Sign Language for recognition purposes.

Prior to the release of consumer depth cameras, such as the Microsoft Kinect sensor [15], many computer vision researches had to use color and data gloves, embedded accelerometers and video cameras to capture a user's hand and body movements for sign language recognition [16]. However, the Microsoft Kinect sensor provides color image, depth map, and real-time human pose information [17], by which it diminishes the dependency to such variety of sensors.

In the rest of this chapter, we are going to discuss the methods and applications in sign language research in three categories: educational tools, translation and recognition systems, and community-aid applications. A summary of these sign language recognition applications can be seen in Table 2.1. Then we are going to examine the available corpora that is being used in sign language recognition research.

2.1. Educational Tools

Recently there have been studies on teaching sign language to non-native signers, including non-hearing-impaired people. Aran et al. have developed a sign language tutoring platform, SignTutor [20], which aims to teach sign language through practice. Using an interactive 3D animated avatar, the SignTutor enables its users to learn sign language by watching new signs and validate their performances through visual feedback. The system uses a left-to-right continuous HMM classifier for verification, and gives feedback on user's performance in terms of manual (handshape, motion and location, etc.), and non-manual (facial expressions and body movements) features for a selected sign. The performance of the SignTutor is evaluated on a dataset of 19 signs from American Sign Language (ASL) and reports the results for signer-depended and signer-independent scenarios in a real-life setting.

Table 2.1. A Survey of Sign Language Recognition Applications.

Study	**Year**	**Language**	**Goal**	**Input Sensor(s)**	**Method**
TESSA [18, 19]	2002	BSL	Community-aid	Wearable Sensors	FSN
SignTutor [20]	2009	ASL	Educational	Colored Gloves and Webcam	HMM
CopyCat [21]	2010	ASL	Educational	Wearable Sensors	HMM
SMARTSign [22]	2011	ASL	Educational	N/A	N/A
Hrúz et al. [23]	2011	Multilingual	Translation and Recognition	Color Camera (640x480)	k-NN, HMM
CopyCat [24]	2011	ASL	Educational	Wearable Sensors & Microsoft Kinect Sensor	4-state HMM
Dicta-Sign Wiki [25]	2012	Multilingual	Translation and Recognition	Microsoft Kinect Sensor	N/A
Karpov et al. [26]	2013	RSL & CSE	Translation and Recognition	N/A	N/A
VisualComm [27]	2013	CSL	Translation and Recognition	Microsoft Kinect Sensor	DTW
Kinect-Sign [12]	2014	LGP	Educational	Microsoft Kinect Sensor	N/A
LSESpeak [28]	2014	LSE	Community-aid	Microsoft Kinect Sensor	N/A
HospiSign (Ours)	2015	TİD	Community-aid	Microsoft Kinect v2 Sensor	DTW

On a database of 1,204 signed phrase samples collected from 11 deaf children playing the CopyCat, which is a gesture-based educational game for deaf children, Zafrulla et al. [21] have performed real-time ASL phrase verification using HMMs with a rejection threshold. During the game, a child is required to wear two different-colored gloves with embedded accelerometers on both hands. The child signs a particular phrase displayed on the screen to a (hero) avatar selected at each game and then the system determines whether (s)he has signed it correctly. If the child sign phrases correctly, (s)he gains points and progresses through the game. The authors achieved a phrase verification accuracy of 83% in their study even though many non-manual features were not included to reduce the complexity of their system.

Zafrulla et al. [24] made further improvements in their existing CopyCat system with a new approach to the automatic sign language recognition and verification tasks by using the Microsoft Kinect sensor. A total of 1000 ASL phrases were collected from two different platforms: CopyCat Adult and Kinect. For each of the 19 signs in their vocabulary, the samples in the classes were trained with HMMs. Using their previous work [29] as a baseline, the authors compared the performance of the Microsoft Kinect based system on two phases, recognition and verification. The Kinect-based system eliminates the need for color gloves and accelerometers, and gives comparable results to the CopyCat system. Similarly, Gameiro et al. [28] have developed a system that aims to help users to learn Portuguese Sign Language (LGP) through a game using the Microsoft Kinect sensor. The system has two modes: the school-mode and the competition mode. In the school mode, users learn new signs in classroom-like environment, whereas in the competition-mode, users experiment their sign language knowledge in a competitive game scenario (such as Quiz and Lingo).

In [22], Weaver and Starner introduced SMARTSign, which aims to help the hearing parents of deaf children with learning and practicing ASL via a mobile phone application. The authors share the feedback they received from the parents on the usability and accessibility of the SMARTSign system. Furthermore, they interviewed the parents in order to determine whether the SMARTSign can alleviate their problems and discuss the ways they can improve their system.

2.2. Translation and Recognition Systems

Hrúz et al. [23] have implemented an automatic translation system, which converts finger spelled phrases to speech and vice versa, in a client-server architecture. The goal of the study is not only to help a hearing-impaired person but also to assist a visually impaired person to interact with others. The system supports many spoken and sign languages, including Czech, English, Turkish and Russian, and the translation between these spoken languages are handled using the Google Translate API. The recognition of multilingual finger spelling and speech was done using k-Nearest Neighbors Algorithm (k-NN) and HMMs, respectively. In the fingerspelling synthesis model, a 3D animated avatar is used to express both manual and non-manual features of a given sign.

The Dicta-Sign [25] is a multilingual sign language research project that aims to make Web 2.0 applications accessible for Deaf people so that they can interact with each other. In their Sign Wiki prototype, the authors demonstrate how their system enables sign language users to get information from the Web. Like Wikipedia, in which users are asked to enter text as an input from their keyboard, sign language users can search and edit any page they want, and interact with the system via a Microsoft Kinect sensor in the Dicta-Sign Wiki. The Dicta-Sign is currently available in four languages: British Sign Language (BSL), German Sign Language (DGS), Greek Sign Language (GSL) and French Sign Language (LSF).

In a similar way, Karpov et al. [26] present their multimodal synthesizer system for Russian (RSL) and Czech (CSE) sign languages that uses a 3D animated avatar for synthesis. VisualComm [12, 27], a Chinese Sign Language (CSL) recognition and translation tool, aims to help the Deaf to communicate with hearing people using the Microsoft Kinect sensor in real-time. The system can translate a deaf person's sign phrases to text or speech and a hearing person's text or speech to sign language using a 3D animated avatar. Based on 370 daily phrases, VisualComm achieves a recognition rate of 94.2% and demonstrates that 3D sign language recognition can be done in real-time by using the modalities provided by the Microsoft Kinect sensor.

2.3. Community-Aid Applications

Community-aid applications are mainly designed to be used to help the deaf community in their daily life. One of the earliest tools was the TESSA (Text and Sign Support Assistant) [18, 19], which was developed for the United Kingdom Post Offices to assist a post office clerk in communicating with a Deaf person. The TESSA system translates a postal officer's (listener) speech into British Sign Language (BSL) and then displays the signs to the screen with an avatar to a Deaf customer at the post office. The authors used the entropic speech recognizer and performed semantic mapping on a "best match" basis to recognize the most phonetically close phrase. Using a subset of 155 out of the 370 phrases, the system achieved a 13.2% error in its best performance, whereas the language processor achieved an error rate of 2.8% on its semantic mapping to choose the most likely phrase on a given utterance.

Lopez-Ludena et al. [30] have also designed an automatic translation system for bus information that translates speech to Spanish Sign Language (LSE) and sign language to speech. The system works in real-time and achieves a sign error rate of 9.4%.

2.4. Available Sign Language Recognition Corpora

Sign language is an active and challenging topic for linguistics and Sign Language Recognition (SLR) research communities. Linguists are interested in analyzing sign languages' properties and rules, whereas computer scientists working on SLR aim to develop systems that can automatically recognize sign language. However, due to several factors such as the lack of high quality capture and annotation technology as well as the absence of common transcription systems, the creation of corpora suitable for sign language recognition and linguistic research only became feasible in the last 20 years [31].

In sign language research literature, numerous sign language corpora exist with different properties. Known sign language corpora can be grouped according to several criteria such as acquisition method, language, research domain, context of content, size and annotations.

One of the defining bottlenecks for the creation of sign language corpora was the quality of the acquisition methods. Especially in the field of SLR, where data was lost in the mapping from 3D world to 2D image space, meaningful capture of signs became achievable with advances in computing, processing, and sensing technologies. First efforts in the field involved instrumented gloves for data capture [32], while later efforts involved RGB colored [8, 33–35] and depth based segmentation of signer hands and body [36–38].

As can be seen in Table 2.2, the corpora used by linguistics and SLR communities have their own properties in correlation with respective research interests. Linguistically motivated corpora are often large vocabulary datasets. They usually have higher quality annotations to learn variation in sign performance, but fewer repetitions of signs or clauses due to difficulty of acquisition and annotation. Recent trends in corpora creation include creating datasets with large number of users from different regions / backgrounds to achieve widespread vocabulary coverage [39–41].

Contrary to linguistically motivated corpora, machine learning or sign language recognition motivated corpora are created with smaller vocabulary. SLR consists of a pipeline of subtasks such as human pose extraction, representation and statistical modeling. All of these tasks are open research questions, which makes large vocabulary SLR a challenging problem. Therefore, these corpora often contain few users, but a higher number of repetitions per user to improve recognition performance [5]. While linguistic corpora contain conversing people [48], recognition oriented corpora almost always belong to single users performing signs or clauses [37]. A large number of these corpora are recorded in constrained recorded environment settings such as dark [33] or monotone backgrounds [36] to allow easier segmentation of human body and hand.

Table 2.2. Existing Sign Language Corpora. VS: Vocabulary Size, NP: Number of Participants.

Study	Language	Research Field	Context	VS	Sample Size	NP	Acquisition Tool
[8]	American SL	SLR	General	40	478 Sentences	1	Video Camera (VC)
[32]	Chinese SL	SLR	General	208	4368 Samples	7	Data Glove
The NGT Corpus [42]	SL of the Netherlands	Linguistic	General	N/A	15 Hours	92	Video Camera
ATIS [43]	Multilingual	Linguistic	Flight Information	292	595 Sentences	N/A	Video Camera
RWTH-BOSTON [33]	American SL	Linguistic, SLR	General	483	843 Sentences	4	Video Camera
ASSLVD [44]	American SL	Linguistic, SLR	General	3000	12000 Samples	4	Video Camera
DGS Corpus [41]	German SL	Linguistic	General	N/A	2.25 million Tokens	328	Video Camera
SIGNUM [34]	German SL	SLR	General	450	33210 Sequences	25	Video Camera
AUSLAN [39]	Australian SL	Linguistic	General	N/A	1100 Videos	100	Video Camera
CopyCat [45]	American SL	SLR	Game	22	420 Phrases	5	Accelerometer & VC
RWTH-PHOENIX-Weather [35]	German SL	SLR	Weather	911	1980 Sentences	7	Video Camera
Dicta-Sign [46]	Multilingual	Linguistic, SLR	General	N/A	6-8 Hours (/Participant)	16-18 (/Language)	Video Camera
[38]	Swedish SL	SLR	Game	51	2550 Samples	10	Microsoft Kinect Sensor
BSL Corpus [40]	British SL	Linguistic	General	N/A	40000 Lexical Items	249	Video Camera
Montalbano [37]	Italian SL	SLR	Cultural Signs	20	13858 Samples	27	Microsoft Kinect Sensor
LSE-SIGN [47]	Spanish SL	Linguistic	General	2400	2400 Samples	2	Video Camera
DEVISIGN [36]	Chinese SL	SLR	General	2000	24000 Samples	8	Microsoft Kinect Sensor

Annotations of SLR oriented corpora are often composed of sign boundary information while annotations of linguistic oriented corpora are more various and detailed. Decades ago Stokoe defined sign language glosses as combinations of movements, hand shapes and location [49]. However, many studies developed their own gloss based annotations. The creation of sign transcription methods, such as HamNoSys [50] and SignWriting [51], together with the development and availability of time aligned annotation software, such as ELAN [52] and iLEX [53], started standardization across sign language corpora, reducing inconsistencies across studies.

3. BOSPHORUSSIGN: TURKISH SIGN LANGUAGE RECOGNITION CORPUS IN HEALTH AND FINANCE DOMAINS

Sign language linguists have been studying Turkish Sign Language (Türk İşaret Dili, TİD) in recent years by collecting large corpora and analyzing the aspects of the language [48]. However, there are no domain-specific TİD corpora for health and finance domains. In this thesis, we are presenting BosphorusSign, a Turkish Sign Language corpus in health and finance domains, collected by using the Microsoft Kinect v2 [15] sensor, that provides depth map, user mask, color video and human pose information. BosphorusSign consists of signs and phrases from three domains: The first is signs and phrases which would be used in a hospital or at a doctor's appointment; the second contains limited corpus in the finance domain and the third contains commonly used signs in everyday life. The signs and phrases that compose the BosphorusSign was chosen by consulting domain specialists, TİD linguists and native TİD users. We have collected *859* sign and phrase samples from multiple signers: *487* samples belonging to the health domain, *177* samples belonging to the finance domain and the remaining *195* samples comprising commonly used signs in everyday life. When completed, the corpus will have at least six repetitions of each sign performed by 10 signers, giving a wide variance to the data.

In order to streamline the recording procedure we have developed a recording software which records all the provided modalities of the Microsoft Kinect v2 sensor and allows online sign border annotation. In addition to the sign border annotations, the corpus will include gloss annotations rendered by linguists, thus making this corpus a valuable resource for sign language researchers both from the computer science and linguistics community. The developed acquisition software and the collected sign samples are currently available on the BosphorusSign website[1] . When completed the corpus will be accessible for academic purposes upon filling a license agreement.

[1] www.BosphorusSign.com

Furthermore, a subset of this corpus was used to develop Hospisign, a human-computer interaction platform that aims to assist the hearing impared in hospitals [54]. Further detail on HospiSign can be found in Chapter 4.

3.1. Recording Software and Setup

All of the recording sessions have been carried out in a controlled environment where all the signers are facing the Microsoft Kinect v2 sensor from a distance of 1.5 meters, in front of a green background. Although the Microsoft Kinect v2 sensor provides the user mask, the green background can be used for background subtraction by the researchers who would like to use color videos as their single modality. The recording setup can be seen in Figure 3.1 from the perspective of the signer and the recording person (user).

Figure 3.1. Recording setup from the perspective of the user (Left) and the signer (Right).

We have developed a data acquisition software for the Microsoft Kinect v2 sensor that is both user and signer friendly while enabling streamlined recording. The software was developed in the Visual Studio 2013 development environment using the C# (WPF) programming language. Emgu CV and FFMPEG external libraries were used for recording the color videos and for compressing them after each session.

The developed software consists of two windows, as it can be seen in Figure 3.2, one dedicated to the user while the other dedicated to the signer. At the beginning of each session the user provides a script that contains the sign names and their video samples. During the recording process the signer first sees a sign sample video playing that is surrounded by an orange bordered window (Figure 3.2). After the sample video finished playing, the user signals the signer to start performing the sign by clicking the *Start Sign* button, turning the borders to green. After the sign is performed by the signer, user clicks the *Stop Sign* button which turns the borders to gray, indicating that the recording of this sample is completed by the signer. Then the user clicks the *Next Sign* button, thus starting the recording procedure for the next sign in the given script. This procedure enables online annotation of the sign borders in the recorded sessions. In case of errors in performing the sign or timing of the online annotation, the sample can be re-recorded using the *Repeat* and *Invalid Sign* buttons, which would invalidate the previously annotated video segment.

The software records color video, depth map, user mask, body pose information and sign border annotations, and saves them into a folder, which is named according to the used script, the signer and the time of the recording session. In each session, signers are presented and asked to perform 30 to 70 signs. These signs are randomly sampled without replacement from the total set of signs. This makes each session unique by randomizing the temporal ordering of signs and reducing the statistical significance of the effects of co-articulation. At the end of each recording session, recorded Microsoft Kinect v2 modalities (color video, depth map, user mask and pose information) and sign border annotations are compressed and saved.

3.2. Annotation

There are two types of annotations provided by BosphorusSign: (1) The sign border annotations of each session, that are mainly for SLR researchers, and (2) The sign level annotations for each sign sample which will be available online for linguists and sign language enthusiasts. Sign level tagging will include content tagging, glosses, spoken language translations, lemmatization, parts of speech; such as classifier or buoy,

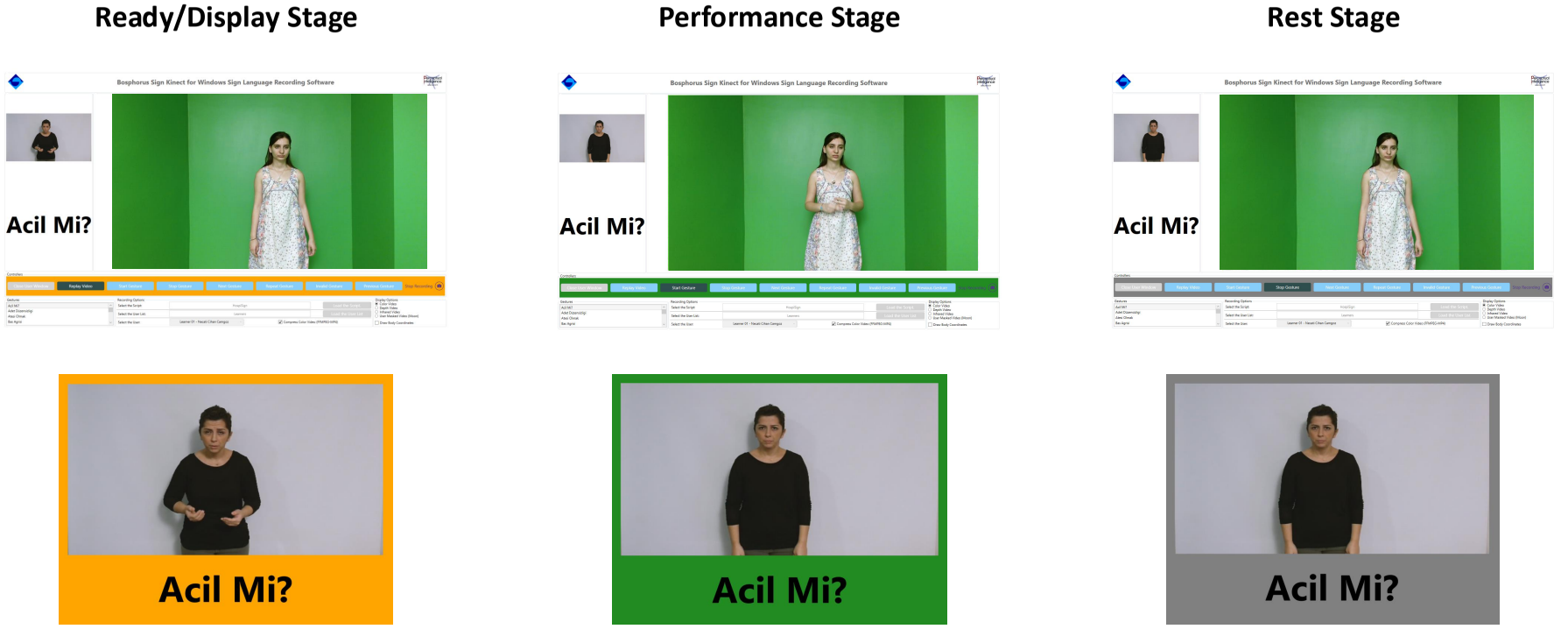

Figure 3.2. Recording software and its stages of recording a sample. Top Row: User Window, Bottom Row: Subject Window, From Left to Right: Getting Ready/Displaying the Sample Stage, Performance Stage and Rest Stage.

and non-manual marking [55]. The corpus is annotated using ELAN [56], which enables the search among the *.EAF files, according to the mentioned categories. The signs are provided with Turkish and English glosses alongside Turkish and English translations.

The corpus consists not only of words but also compounds and phrases. Considering this, the signs are lemmatized in order to facilitate searches among all the tokens in the corpus. Moreover, the ELAN files have tiers for mouth movements and other non-manuals. If a sign has mouthing, then the mouthing is annotated. If the sign has a mouth gesture, then the type of the gesture, such as puffed cheeks or tongue protruding, is given in this tier. This piece of information is expected to depict the differences among the signers.

In addition to the mouth tier, another non-manual tier is also given so that researchers would be able to look through the tokens consisting of a specific non-manual marker in the corpus. As a further step, the same motivation leads us to state the parts of speech information because the signs in the corpus contain classifiers or buoys, which can be crucial for future morphology research projects. The annotations also mark the types of classifiers [57] or buoys [58] such as SASS (Size and Shape Specifiers) or fragment buoy, respectively based on the studies in the literature. The linguistic annotations will be available on BosphorusSign website as ELAN Files.

3.3. Distribution

The collected corpus will be available for download for academic purposes upon filling a license agreement form on the BosphorusSign website. The provided data will include the Microsoft Kinect v2 modalities (1080p color video, depth map, pose information and user mask) and their sign border annotations.

In order to distribute the corpus we had to solve the file size issue. The Microsoft Kinect v2 sensor provides high definition color videos, which occupy large disk spaces. For example, an unprocessed one-minute long video has an approximate size of 12 GBs. The recording software we have developed does lossless compression at the end of each

session, thus shrinking the video's size. Nonetheless, these compressed videos are still not feasible for distribution as their sizes are approximately 5 GB/minute. In order to solve this issue, while preserving the video quality, we have conducted experiments using x264 compression algorithm and its parameters to lower the video size.

In the light of our experiments, we have chosen the lossy x264 parameters to be 23 for the Constant Rate Factor parameter and VerySlow for the preset parameter, which dropped the video size from 5 GB/minute to 16 MB/minute, making the video feasible to distribute while persevering the video quality (Mean pixel error rate of 2.7).

The provided data for each recording session, their formats and their mean sizes can be seen in Table 3.1.

Table 3.1. Contents of BosphorusSign Corpus.

Modality	File Type	Resolution	Content	Mean Size
Color Video	.MP4 Video File	960*1080 Pixels	24bpp Image Sequence	16 MB/minute
Depth Map	.RAR Binary File	512*424 Pixels	16bpp Image Sequence	235 MB/minute
User Mask	.RAR Binary File	512*424 Pixels	8bpp Binary Image Sequence	2 MB/minute
Pose Information	.CSV File	25 Joints	Joint Coordinates and Angles	6 MB/minute
Border Annotations	.CSV File	30-70 Signs	Sign Border Frames with Labels	5 KB/session

4. HOSPISIGN: A HUMAN-COMPUTER INTERACTION PLATFORM FOR THE HEARING IMPAIRED

In this chapter, we present HospiSign, a human computer interaction platform that is designed to assist the hearing-impaired in a hospital environment, which recognizes sign language phrases in order to interpret Turkish Sign Language. HospiSign proposes a possible solution to the communication problem between a Deaf patient and a doctor. By asking questions as sign videos and suggesting possible answers on a display, the system helps Deaf users to explain their problems. With the tree-based activity diagram interaction scheme, the system only looks for the possible answers in each level (step), instead of trying to recognize from all the signs in the dataset. At the end of the interaction, the system prints out a summary of the interaction and the users are guided to take this print out with their ID to the information desk, where they can be assisted according to their needs.

The HospiSign platform consists of a personal computer (PC), a touch display to visualize the sign questions and answers to the user, and a Microsoft Kinect v2 sensor. Since it is necessary to track the users' hand motions in order to recognize the performed signs, the Microsoft Kinect v2 sensor plays an essential role as it provides accurate real-time human body pose information.

4.1. Interaction Design

While designing the interface, the focus were on two criteria: functionality and usability. Therefore, the interaction scenarios were prepared based on the advice of the the family medicine clinic doctors, TİD linguists and native TİD users.

On the visual design of the interface, the question sign video is placed at the top-center of the screen to attract the user's attention. Then, the answer sign videos are displayed at the bottom of the screen with a smaller size than the size of the question

sign video. A sample user interface of the HospiSign platform can be seen in Figure 4.1. Since there are some questions that have more then three answers, the timing is adjusted for each question accordingly so that users would be able to view all the answers.

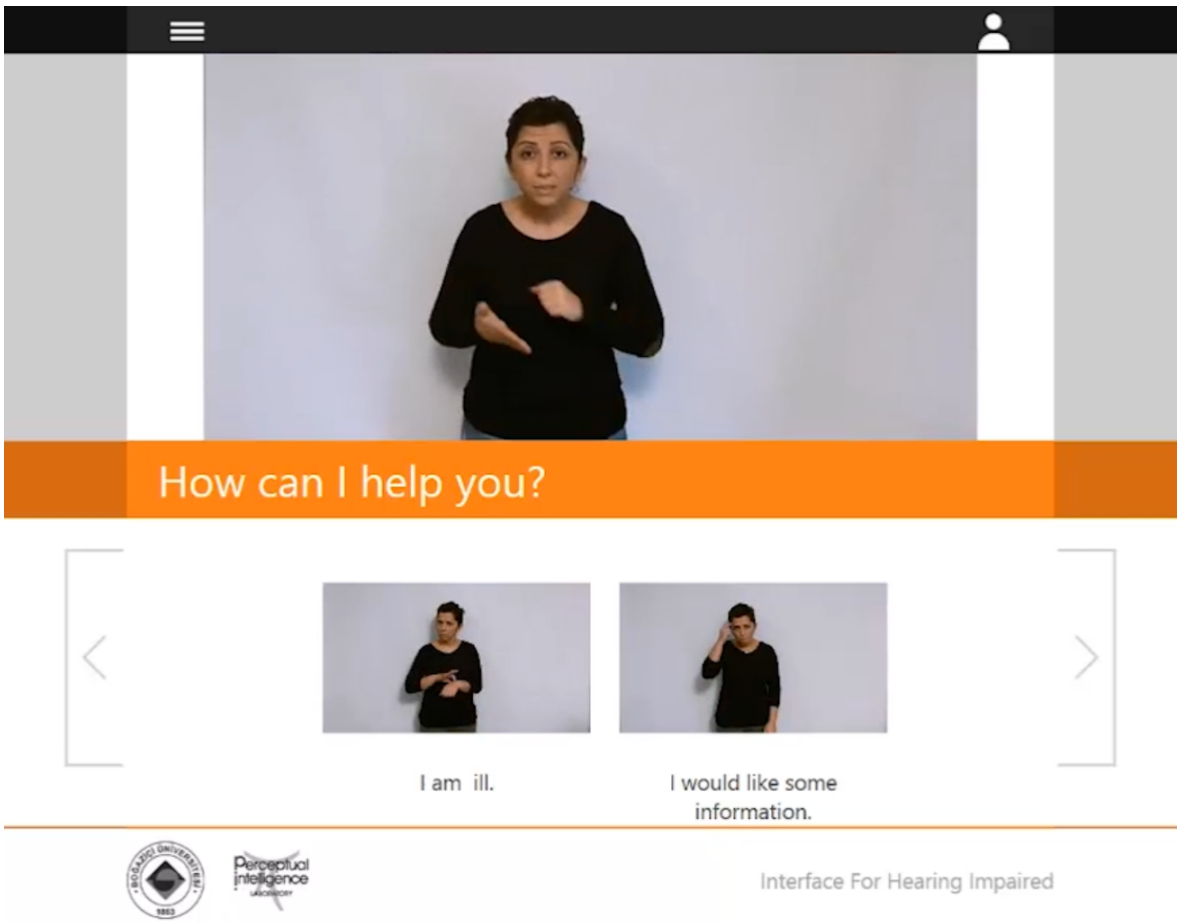

Figure 4.1. The HospiSign Interface.

The HospiSign system follows three stages to move from one question to another in the tree-based activity diagram interaction scheme: (1) display of the question; (2) display of the possible answers to that question; and (3) the recognition of the answer (sign). The user first watches the question displayed on the top-center of the screen; then performs a sign from the list of possible answers displayed at the bottom of the screen (See Figure 4.1); and then moves to the next question. This process is repeated until the system gathers all the necessary information from the user. After the user answers all the required questions, the system prints out a summary report to be given

to the information desk or the doctor at the hospital. This summary contains the details of user's interaction with HospiSign.

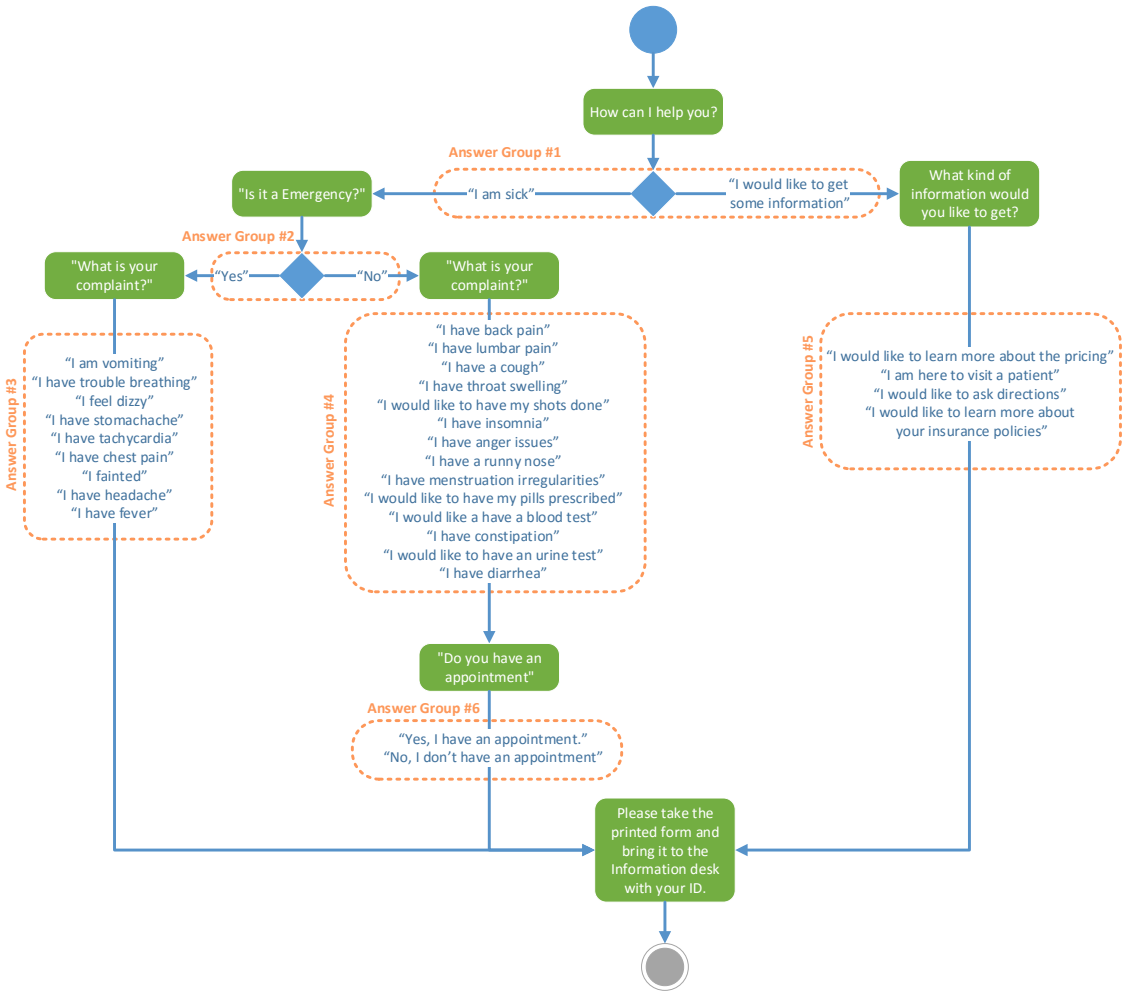

Figure 4.2. Tree-based activity diagram of HospiSign.

To make classification task easier, the questions are placed into a tree-based activity diagram in such a way that each question will lead to another sub-question with respect to the answer selected by the user. With categorization of possible answers to each question, it is intended to help the users to easily describe their illness or intention of their visit. The proposed tree-based activity diagram can be seen in Figure 4.2.

One of the most important advantages of using such a tree-based scheme is that it makes the system more user-friendly and easy-to-interact. The tree-based activity diagram interaction scheme also increases the recognition speed and performance of the system as the task of recognizing a sign from possible answers to each question is much easier and faster then recognizing a sign from the all possible answers. Extensive experiments on the effectiveness of tree-based activity diagram can be found in Chapter 6.

5. PROPOSED SIGN LANGUAGE RECOGNITION TECHNIQUES

Vision based sign language recognition methods generally consist of four main modules: Human Pose Estimation, Feature Extraction, Feature Normalization, and Temporal Modeling and Classification, as visualized in Figure 5.1. Taking this framework as our baseline, we propose using various features, normalization approaches, temporal modeling techniques and classification methods to represent and to recognize isolated sign language instances.

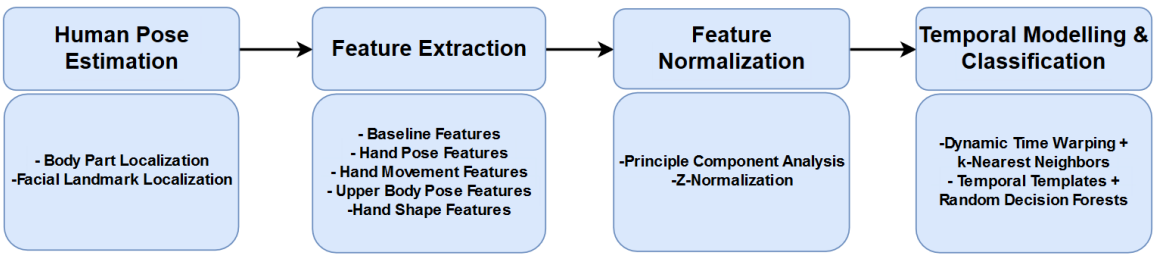

Figure 5.1. Four main modules of our sign language recognition framework.

5.1. Human Pose Estimation

Most of the vision based sign language recognition methods use color cameras as their means of capturing signs. However, human pose estimation in color videos is a challenging task due to the color ambiguity between the user and the background. Moreover, color images are highly effected by the illumination changes in the environment. With the emergence of depth sensors, researchers have moved towards using depth cameras as an alternative means of capturing signs, as the depth sensors are robust against lighting changes and provide depth information of the scene. Furthermore, depth information proved itself to be useful for human pose estimation as it makes distinguishing users from background a trivial task. In recent years, Shotton et al. proposed using depth pixel differences to estimate human poses in real-time, which is incorporated in most of the depth sensors' Software Development Kits (SDKs) [17].

Taking these facts into consideration, we have collected our sign samples using the Microsoft Kinect v2 sensor, as described in Chapter 3. By using the Microsoft Kinect v2 SDK, we were able to capture color videos, depth maps, user masks and body pose information of the signers. The body pose information consist of the world coordinates, orientations, and pixel coordinates (in depth and color images) of the 25 joints, which are visualized in Figure 5.2.

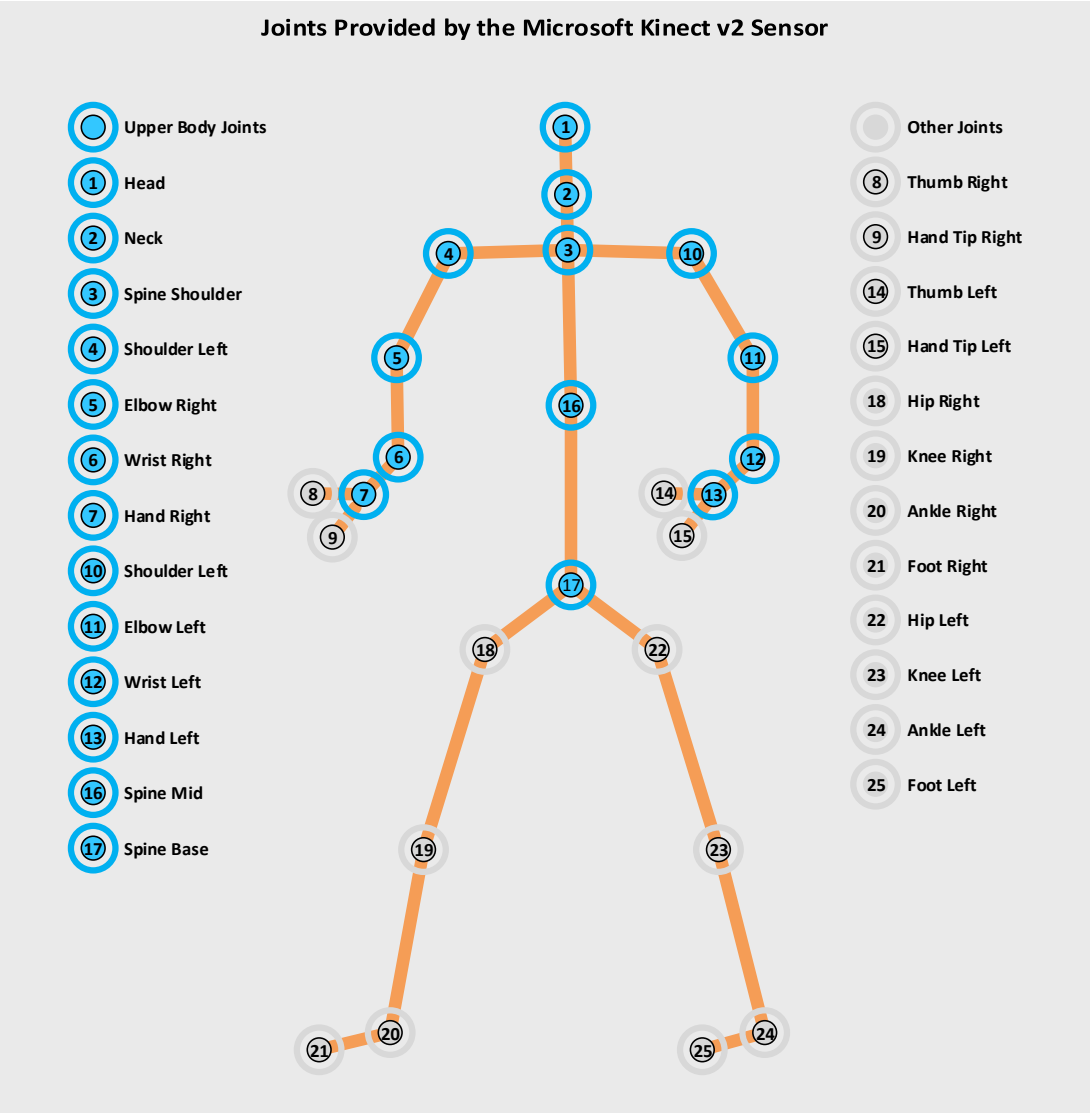

Figure 5.2. 25 Joints that are provided by the Microsoft Kinect v2 SDK.

Furthermore, we incorporated state-of-the-art facial landmark localization technique, Supervised Descent Method [1], to our human pose estimation module in order to localize the hands more precisely with respect to the face. A sample of the detected facial landmarks can be seen in Figure 5.3. Moreover, we have studied the application of these facial landmark localization methods in depth images and obtained state-of-the-art performance on frontal depth images, which is extensively explained in Chapter 8.

Figure 5.3. Facial landmarks that are localized by using the Supervised Descent Method [1].

5.2. Feature Extraction

As sign languages convey information through hand shape, upper body pose, facial gestures and hand trajectories, sign language recognition techniques extract features to represent each respective aspect of the signs. Taking the features proposed by Kadir et al. [2] as baseline, we have extracted hand position (*Baseline Hand Position*) and movement (*Baseline Hand Movement*) features from each video frame to represent our sign samples. In addition, we have extracted upper body pose (*Normalized World Coordinates, Normalized Pixel Coordinates, Upper Body Joint Orientations*), hand movement (*Hand Joint Movement*), and hand position (*Hand Joint Distance*) features using the provided body pose information. To represent hand shapes, we segmented the hand images using the signers' skin color and depth maps and extracted Histogram of Oriented Gradients [59] from the segmented hand patches for each frame. A list of our features and the aspects they represent in a sign can be seen in Table 5.1.

Table 5.1. Extracted features that are used to represent signs.

Feature Name	Represented Aspect
Baseline Hand Position	Hand Position
Baseline Hand Movement	Hand Movement
Normalized World Coordinates	Upper Body Pose
Normalized Pixel Coordinates	Upper Body Pose
Upper Body Joint Orientations	Upper Body Pose
Hand Joint Distance	Hand Position
Hand Movement Distance	Hand Movement
Histogram of Oriented Gradients	Hand Shape

5.2.1. Baseline Hand Position and Baseline Hand Movement Features

Kadir et al. [2] proposed hand position and hand movement features in order to achieve high sign language recognition performance using low number of training instances. Their features consists of two parts that are, hand positions and hand

movements, that are listed in Table 5.2 and Table 5.3 respectively. Hand movement and hand position features are extracted using the joint coordinates, that are provided by the Microsoft Kinect v2 sensor, and the facial landmark locations, which are extracted by using Supervised Descent Method [1].

Table 5.2. Baseline hand position features proposed by Kadir et al. [2].

Feature Name	Type	Length
Right Hand Raised	Binary	1
Left Hand Raised	Binary	1
Both Hands Raised	Binary	1
Hands are Together	Binary	1
Hands are Crossed	Binary	1
Closest Body Part	Categorical	12 x 2 (For both hands)

Table 5.3. Baseline hand movement features proposed by Kadir et al. [2].

Feature Name	Type	Length
Hands Move Apart	Binary	1
Hands Move Closer	Binary	1
Hands Move in Unison	Binary	1
Movement Direction	Categorical	4 x 2 (For both hands)

There are two types of features that are proposed by Kadir et al. [2]. First type is binary features, which are set to 1 if the requirement for the features are fulfilled. The second type is categorical features, in which only one of the possible categories of the features can be set in each frame. To be able to use categorical features in machine learning methods that measure euclidean distance, categorical features are represented with binary arrays with the length equal to the number of categories. When a category is set for a given frame, the respective binary value of that category is set to one while the rest is set to zero.

There are two categorical features: Closest Body Part (Hand Position) and Movement Direction (Hand Movement). Closest Body Part feature represents the closest body part to both of the hands. The body parts that are taken into consideration are: Face, Chin, Nose, Cheeks, Neck, Shoulders, Chest, Stomach and Hips. Movement Direction represents the movement direction of the hands. To extract the Movement Directions, world coordinates of the hand joints belonging to adjacent frames are used. The direction categories are: Up, Down, Left and Right.

5.2.2. Upper Body Pose Features

As upper body pose features, we used world coordinates (*Normalized World Coordinates*) , pixel coordinates (*Normalized Pixel Coordinates*), and orientations (*Upper Body Joint Orientations*) of the 12 upper body joints that are are visualized in Figure 5.2. We use the joint orientations as is. However, we normalize the world and pixel coordinates of the joints, to remove their user dependent location and scale factors. In order to normalize the coordinates in location, we move the *SpineBase* of all frames to the origin by subtracting its coordinate values from all the other joints' coordinate values. To normalize the coordinates in scale, we divide each coordinate by the height of the person, which is inferred by calculating the distance between the *SpineCenter* and the *SpineBase* in the y axis.

5.2.3. Hand Joint Distances

Using the 12 upper body joints that are visualized in Figure 5.2, we extracted the *Hand Joint Distance* feature, by calculating each joint's euclidean distance from the hand joints. Then each distance is divided by the sum of all the distances, by which we normalize the feature in scale while creating a discrimination between different body poses (as sum of the distances vary from one pose to another).

5.2.4. Hand Joint Movements

We extract the *Hand Joint Movement* feature by subtracting world coordinates of hand joints belonging to adjacent frames. The distance is then normalized in scale by being divided by the height of the person, which is inferred by calculating the distance between the *SpineCenter* and the *SpineBase* in the y axis.

5.2.5. Hand Shape Features

Hand shape is a crucial feature for sign language recognition as some signs have the same hand movement and body posture while hand shape is the only differentiating characteristic. Therefore, we used Histogram of Oriented Gradients (HOG) [59], that is commonly used to represent hand shapes in sign language recognition and hand gesture recognition applications [27].

Using the pixel coordinates provided by the Microsoft Kinect v2 sensor, we crop a 160x160 pixels patch around the hand. The size of the patch is chosen to be large, in order to make sure that the hands are fully in the cropped patches. Using an adaptive skin color model, we segment the skin colored regions in the patch. Finally, using the depth information, we omit any region that lies 20 cm behind the hand joint, giving us the segmented hand region. We proposed using three parameter setups while extracting Histogram of Oriented Gradients that are: High Detailed HOG (HOG-H), Medium Detailed HOG (HOG-M) and Low Detailed HOG (HOG-L), that are visualized in Figure 5.4. The parameters and feature sizes of all the three setups can be seen in Table 5.4.

5.2.5.1. Histogram of Oriented Gradients (HOG). First proposed by Dalal et al. [59] for human detection, HOG is a spatial descriptor that uses pixel gradient information. HOGs are extracted by dividing a given image into cells and calculating gradients of the pixels in each cell. Then, histograms of gradients are calculated from blocks of cells. Lastly, all histograms are normalized and concatenated to form the HOG descriptors.

Table 5.4. Proposed Histogram of Oriented Gradients parameter setups.

Parameters	HOG-L	HOG-M	HOG-H
Cell Size	80x80	40x40	20x20
Block Size	1x1	2x2	4x4
Feature Size	18	108	432

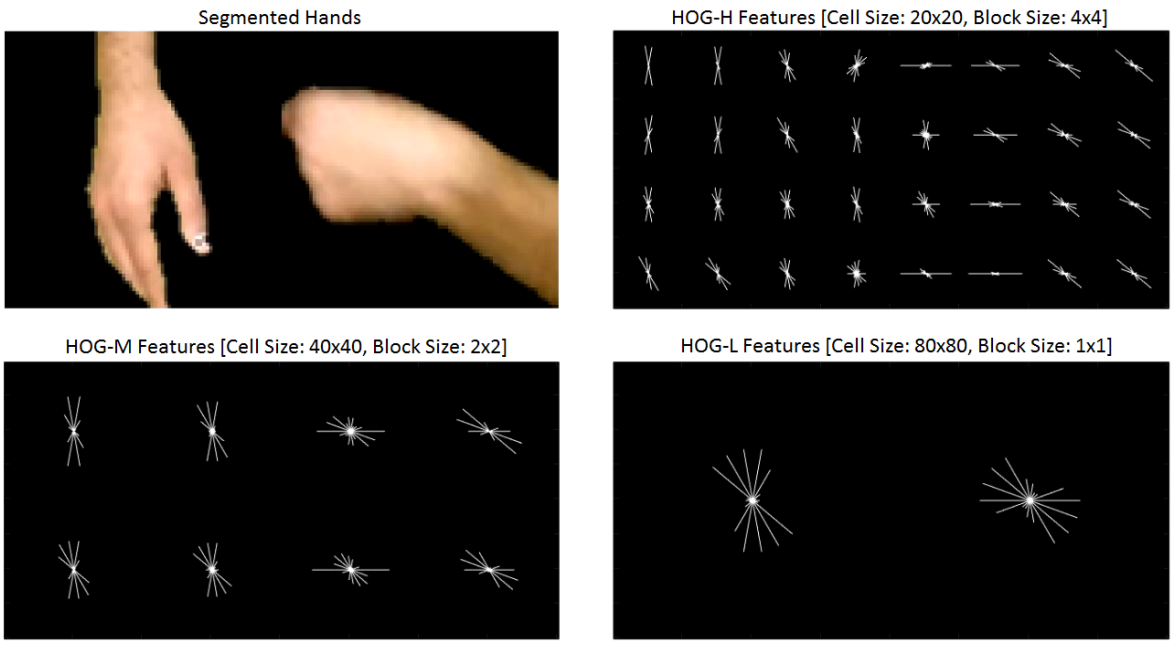

Figure 5.4. Segmented hands and extracted Histogram of Oriented Gradients with different parameter setups. Top Left: Segmented Hands, Top Right: HOG-H, Bottom Left: HOG-M, Bottom Right: HOG-L.

5.3. Feature Normalization

As different features come from different distribution, they require normalization before being used in temporal modeling and machine learning methods, which calculate euclidean distance to measure similarities. In order the normalize our features, we propose using three normalization strategies that are applying Principle Component Analysis (PCA) [60], applying Z-Normalization [61], and applying PCA that is followed by Z-Normalization.

We apply PCA to each group of features separately, as their scales might be different, which would cause PCA to choose the directions in which the scale is the most instead of the direction with the highest variance. Z-Normalization is applied to each feature in order to represent all the features in the standard score.

In addition to the three normalization strategies, while using Dynamic Time Warping, we give weights to each feature that are inversely proportional to their feature lengths, in order to make each feature equally effective on the measured distance.

5.4. Temporal Modeling and Classification

Same as all the natural languages, sign languages are series of utterances that are used in sequences to convey information. Therefore, temporal modeling is an essential step for sign language recognition, as any change in the arrangement of signs or its subunits may change the meaning of the sign sequence.

Commonly used temporal modeling techniques in Sign Language Recognition are Hidden Markov Models (HMMs) [6] and Dynamic Time Warping (DTW) [7]. While Hidden Markov Models are statistical models that are explicitly developed to model time series, Dynamic Time Warping is an algorithm to measure similarity between two time series that can vary in time and speed. Another approach for temporal modeling is to create Temporal Templates (TT) using the smallest time units (in our case the frames of the videos). We propose using Dynamic Time Warping and Temporal Templates to

model the temporal aspect of the sign sequences. For classification, we propose using k-Nearest Neighbors for Dynamic Time Warping based temporal modeling and Random Decision Forests [62] for Temporal Template based temporal modeling approaches.

In addition, we have conducted experiments using Hidden Markov Models [6] to evaluate the applicability of transfer learning methods for gesture recognition which is extensively explained in Section 7.

5.4.1. Temporal Templates based Random Decision Forests

Due to their lack of temporal mechanisms, spatial machine learning methods like Support Vector Machines and Random Decision Forests, are not suitable for recognizing time series, such as sign language sequences. In order to use powerful spatial classifiers for classifying time series, spatial features of each time step are concatenated with its neighboring steps, thus representing each step by a temporal window of features. In our framework, this is achieved through Temporal Templates (TT) that represent each frame with the concatenated features of its neighbors.

In template based temporal modeling, increasing template size enhances temporal representation. However, memory and computational power restrictions of development systems limit the feature vector size. To overcome this limitation, frame selection methods for creating templates can be altered. We propose selecting frames in intervals, in order to be able to represent larger temporal windows while using the same number of frames.

We classify the constructed temporal templates of each frame by using Random Decision Forests (RDFs). RDF is a supervised classification and regression technique that has become widely used due to its efficiency and simplicity. RDFs are an ensemble of random decision trees (RDT) [62]. Each tree is trained on a randomly sampled subset of the training data. This reduces over-fitting in comparison to training RDTs on the entire dataset; therefore increasing stability and accuracy.

During training, a tree learns to split the original problem into smaller ones. At each non-leaf node, tests are generated through randomly selected subsets of features and thresholds. The tests are scored using the decrease in entropy, and best splits are chosen, and used for each node [62]. Each tree ends with leaf nodes, that represent the probabilities of a given data to belong to the possible classes.

Classification of a frame is performed by starting at the root node and assigning the frame either to the left or to the right child recursively until a leaf node is reached. Majority voting is used on prediction of all decision trees to decide on the final class of the frame. Finally, signs are classified by taking the mode of its frames' classification results.

5.4.2. Dynamic Time Warping and k-Nearest Neighbors

Dynamic Time Warping (DTW) is a popular tool for finding the optimal alignment between two time series. The DTW algorithm calculates the distance between each possible pair of points out of the two series in terms of their spatial and temporal features.

DTW uses these distances to calculate a cumulative distance matrix and finds the least expensive path through this matrix using dynamic programming. This path represents the ideal synchronization of the two series with the minimal feature distance. Usually, the samples are normalized to zero mean and smoothed with median filtering before distance calculation. k-Nearest Neighbors algorithm is widely used in classifying time series that are modeled by Dynamic Time Warping and often achieve state-of-the art performance, in which the mode of the class labels belonging to the samples which have the least distance is chosen as the classification result.

6. EXPERIMENTS AND RESULTS

We examined the performance of our methods in terms of the features, temporal modeling techniques and classification approaches. First, we conducted experiments to find the optimum parameters for Histogram of Oriented Gradients, which we used to represent our hand shapes. Then using the optimum HOG parameters we conducted experiments in order to find the combination of features that yields the highest recognition performance. In both of the feature selection experiments, we used all the three proposed normalization setups, and Dynamic Time Warping (DTW) to measure the distance between isolated sign phrases. Then we use k-Nearest Neighbors (k-NN) algorithm to classify the isolated signs by taking the mode if its k nearest neighbors' class labels.

Using the best performing feature combination and the normalization setup, we conduct experiments in order to find the optimum window size and interval steps for the Temporal Templates (TT). We classify the temporal templates using Random Decision Forest (RDF) that contains 100 trees. Then we compare the performance of DTW and TT based approaches.

Finally, we conclude our performance evaluation by conducting experiments using the proposed tree-based activity diagram, by which the recognition task is divided into six subtasks and handled separately. Each subtask aims to recognize the signs that are in its respective answer group, which can be seen in Figure 4.2. Best performing features, normalization and temporal template setups are used in these experiments.

All of our experiments were conducted on a subset of the BosphorusSign corpus, which is used in the development of HospiSign. The subset contains 662 sign phrase samples belonging to 33 phrase classes which are performed by three native TİD users in six to eight repetitions. In order to obtain user independent results we performed leave-one-user-out cross-validation and report the mean and standard deviation of recognition performance in all of our experiments.

6.1. Histogram of Oriented Gradients Parameter Optimization

In this experiment setup, we find the optimum Histogram of Oriented Gradients (HOG) parameters to represent the hand shapes in isolated sign phrases. We use three HOG parameter setups that are Low Detailed (HOG-L, Cell Size: [80x80] Block Size: [1x1]), Medium Detailed (HOG-M, Cell Size: [40x40] Block Size: [2x2]), High Detailed (HOG-H, Cell Size: [20x20] Block Size: [4x4]). Examples of all the three parameter setups can be seen in Figure 6.1.

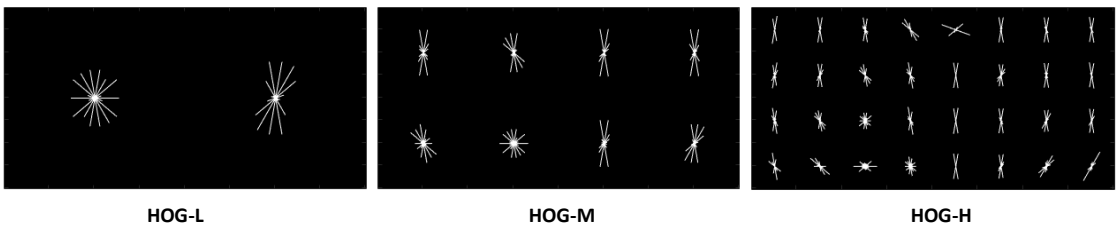

Figure 6.1. HOG Parameter Setups. From left to right: Low Detailed HOG (HOG-L), Medium Detailed HOG (HOG-M), High Detailed HOG (HOG-H).

We apply all the three normalization methods on the HOG features. The temporal aspects of the signs were modeled using Dynamic Time Warping and k-Nearest Neighbors algorithm was used to classify signs. The experiments were conducted on a subset of BosphorusSign, which contains 662 samples of 33 sign classes belonging to three native TİD users. All of the experiments were conducted by doing leave-one-user-out cross-validation, and we report the mean and standard devation of each parameter and normalization setup in Table 6.1.

As it can be seen from Table 6.1, HOG-M performs the best while normalized using PCA. Taking this into consideration, in the rest of our experiments we used HOG-M as our hand shape representation.

Table 6.1. Histogram of Oriented Gradients parameter optimization results.

Normalization Setup	HOG-L	**HOG-M**	HOG-H
No Normalization	52.48 ± 11.13	81.15 ± 6.02	78.16 ± 13.03
PCA	51.85 ± 10.80	$\mathbf{81.78 \pm 5.69}$	78.79 ± 12.71
Z-Norm	50.26 ± 10.11	79.56 ± 5.77	65.85 ± 17.72
PCA + Z-Norm	28.61 ± 8.02	22.52 ± 9.83	11.57 ± 6.64

6.2. Comparing and Combining Features

Using the best performing HOG parameter setup (HOG-M), we conducted experiments in order to find the best feature combination from the eight proposed features. A list of these features and the sign modalities they represent can be seen in Table 6.2.

Table 6.2. Eight features and the modalities they represent in a sign.

ID	Feature Type	Modality
FT1	HOG-M (C:40x40; B:2x2)	Hand Shape
FT2	Baseline Hand Movement	Hand Movement
FT3	Baseline Hand Position	Hand Position
FT4	Hand Movement Distance	Hand Movement
FT5	Hand Joint Distance	Hand Position
FT6	Normalized World Coordinates	Upper Body Posture
FT7	Normalized Pixel Coordinates	Upper Body Posture
FT8	Upper Body Joint Orientations	Upper Body Posture

To find the best combination of features, we first find the four features that yield the highest recognition performance. We apply all the three normalization setups on all of the features and model their temporal aspects using Dynamic Time Warping. The classification is done using the k-Nearest Neighbors algorithm. The experiments were conducted on the BosphorusSign subset, which contains 662 sign samples belonging to

33 sign classes that are performed by three native TİD users. All of the experiments were conducted by doing leave-one-user-out cross-validation, and we report the mean and standard deviation of each parameter and normalization setup in Table 6.3

Table 6.3. Performance evaluation of all the nine features.

ID	Raw Data	PCA	Z-Norm	PCA + Z-Norm
FT1	81.15 ± 6.02	81.78 ± 5.69	**79.56 ± 5.77**	22.52 ± 9.83
FT2	69.20 ± 9.70	69.03 ± 9.64	**69.12 ± 9.91**	67.86 ± 9.44
FT3	73.15 ± 4.26	72.98 ± 5.55	3.02 ± 0.01	71.00 ± 5.22
FT4	**85.19 ± 0.30**	**85.19 ± 0.30**	**83.59 ± 1.97**	**82.80 ± 2.75**
FT5	**94.46 ± 0.51**	**93.87 ± 0.64**	**93.62 ± 2.05**	**91.90 ± 0.55**
FT6	**91.57 ± 2.51**	**91.02 ± 1.90**	45.86 ± 18.80	**77.12 ± 7.07**
FT7	**92.99 ± 0.95**	**92.57 ± 1.05**	47.75 ± 12.90	**87.12 ± 3.21**
FT8	33.98 ± 5.12	33.98 ± 5.61	3.02 ± 0.01	39.81 ± 9.51

As it can be seen from Table 6.3, Hand Movement Distance (FT4), Hand Joint Distance (FT5), Normalized World Coordinates (FT6) and Normalized Pixel Coordinates (FT7) generally yield the four highest recognition performances. To find the best combination of features, we conducted experiments using the combination of these four features using the same experiment setup. We report the results of our experiments in Table 6.4.

As it can be seen in Table 6.4, combining Hand Joint Distances with Hand Movement Distances, and normalizing them using Principle Component Analysis yields the highest recognition performance. Therefore, for the rest of our experiments, we used this combination of features and normalization setup while evaluating other aspects of our proposed system.

Table 6.4. Performance evaluation of feature combinations. (FT4: Hand Movement Distance, FT5: Hand Joint Distance, FT6: Normalized World Coordinates, FT7: Normalized Pixel Coordinates)

Combinations	Raw Data	**PCA**	Z-Norm	PCA + Z-Norm
FT4	85.19 ± 0.30	85.19 ± 0.30	83.59 ± 1.97	82.80 ± 2.75
FT5	94.46 ± 0.51	93.87 ± 0.64	93.62 ± 2.05	91.90 ± 0.55
FT6	91.57 ± 2.51	91.02 ± 1.90	45.86 ± 18.80	77.12 ± 7.07
FT7	92.99 ± 0.95	92.57 ± 1.05	47.75 ± 12.90	87.12 ± 3.21
FT4 + FT5	94.84 ± 1.87	**96.72 ± 2.92**	92.70 ± 1.12	92.24 ± 0.83
FT4 + FT6	93.03 ± 3.16	92.07 ± 3.28	71.04 ± 9.37	82.92 ± 1.42
FT4 + FT7	93.11 ± 2.55	93.58 ± 1.02	76.75 ± 1.37	89.55 ± 1.06
FT5 + FT6	92.83 ± 2.63	92.74 ± 2.96	79.89 ± 10.87	90.34 ± 5.20
FT5 + FT7	94.00 ± 0.45	93.20 ± 0.27	86.32 ± 3.66	92.07 ± 2.20
FT6 + FT7	93.24 ± 0.98	93.28 ± 1.18	47.32 ± 19.17	87.91 ± 1.33
FT4 + FT5 + FT6	93.66 ± 2.85	93.41 ± 3.56	84.17 ± 7.03	91.10 ± 2.60
FT4 + FT5 + FT7	95.26 ± 1.98	94.04 ± 0.64	87.20 ± 2.31	92.86 ± 1.69
FT4 + FT6 + FT7	94.04 ± 2.44	93.58 ± 1.44	66.30 ± 9.29	89.51 ± 0.63
FT5 + FT6 + FT7	93.24 ± 0.99	93.41 ± 1.30	76.41 ± 10.62	91.69 ± 2.63
All Features	94.50 ± 2.09	94.16 ± 1.51	81.07 ± 5.98	92.19 ± 2.24

6.3. Optimizing Temporal Template Size and Interval Steps

Using the best performing feature combination (Hand Joint Distance and Hand Movement Distance) and the normalization setup (Principle Component Analysis) we conducted experiments in order to find the optimum temporal template size and interval steps. We did a grid search for parameter optimization in which we searched templates size from $\{9, 11, 13, 15, 17, 21\}$ and interval steps from $\{1, 2, 3, 5\}$. The constructed temporal templates of each frame was classified using Random Decision Forests. The classes of the isolated signs are assigned as the mode of class labels of its frames.

Table 6.5. Temporal Template Size and Interval Steps optimization results. TS: Template Size, IS: Interval Steps.

	IS: 1	IS: 2	IS: 3	**IS: 5**
TS: 9	72.44 ± 17.68	82.09 ± 12.29	88.38 ± 6.72	92.74 ± 3.88
TS: 11	74.87 ± 16.54	86.15 ± 8.52	89.72 ± 5.16	94.21 ± 2.23
TS: 13	77.47 ± 16.77	88.30 ± 8.50	91.61 ± 5.81	95.55 ± 2.05
TS: 15	79.61 ± 15.68	88.55 ± 5.45	93.08 ± 2.93	95.80 ± 1.62
TS: 17	79.95 ± 14.67	89.68 ± 6.23	93.20 ± 3.92	**95.93 ± 1.57**
TS: 19	82.97 ± 10.77	90.90 ± 4.87	94.55 ± 2.52	94.97 ± 2.18
TS: 21	84.85 ± 10.17	91.82 ± 3.70	94.92 ± 1.88	95.35 ± 2.02
TS: 23	85.82 ± 10.61	92.83 ± 4.57	95.68 ± 1.96	95.51 ± 2.78

As it can be seen in Table 6.5 the recognition performance increases as the represented temporal window gets larger. However, the recognition performance converges near 95%. The highest recognition performance is obtained by using 17 frames to construct temporal templates while taking every fifth frame.

6.4. Comparing the Temporal Modeling and Classification Methods

Using the best performing feature setups, we conducted experiments in order to evaluate and compare the performance of temporal modeling and classification methods. For Dynamic Time Warping (DTW) based classification, we have optimized the k values of the k-Nearest Neighbors (k-NN) algorithm. For Temporal Template (TT) based Random Decision Forest (RDF) classifiers, we used template size and interval steps that we have found to be optimum in our previous experiments. The experiments were conducted on 662 sign samples of 33 sign classes belonging to three native TİD users. Leave-one-out cross-validation was done for both of the experiments setups. As it can be seen from Table 6.6, DTW based approach performs slightly better than the TT based approach. Further comparison of the methods can be seen in the experiments, in which we evaluate the effectiveness of the tree-based activity diagram.

Table 6.6. Comparison of the temporal modeling and classification methods.

Temporal Modeling Method	Recognition Performance
DTW + k-NN	96.72 ± 2.92
TT + RDF	95.93 ± 1.57

6.5. Effectiveness of Tree-based Activity Diagram

Using the best performing feature and normalization setups, we evaluate the effectiveness of using the tree-based activity diagram, which is explained in detail in Chapter 4. We use both of the temporal modeling and classification approaches and report our experiment results in Table 6.7.

As it can be seen from Table 6.7, using the tree-based activity diagram increases the recognition performance. Furthermore, as the recognition task is divided into subtasks, the best performing temporal modeling and classification methods can be chosen for each answer group and combined in order achieve higher recognition rates (See Combined column in Table 6.7).

Table 6.7. Evaluation of using the tree-based activity diagram. (nClasses: Number of classes)

Setup	nClasses	DTW + k-NN	TT + RDF	Combined
w/o Activity Diagram	33	96.75 ± 2.92	95.63 ± 1.57	N/A
Answer Group 1	2	100 ± 0	100 ± 0	100 ± 0
Answer Group 2	2	100 ± 0	100 ± 0	100 ± 0
Answer Group 3	9	98.77 ± 2.14	100 ± 0	100 ± 0
Answer Group 4	14	96.15 ± 3.95	94.46 ± 4.52	96.15 ± 3.95
Answer Group 5	4	98.96 ± 1.80	94.10 ± 5.14	98.96 ± 1.80
Answer Group 6	2	100 ± 0	100 ± 0	100 ± 0

7. DOMAIN ADAPTATION FOR GESTURE RECOGNITION

The performance of gesture and sign language recognition systems heavily depend on the data that have been used while training the system. Difficulties in data collection and annotation, quality of the collected data and the inconsistencies of the data collected from different users and environments make it difficult to develop systems that can recognize gestures or signs from large vocabularies in an user and environment independent manner [5]. In recent years the use of Transfer Learning (Domain Adaptation) methods is proposed as a possible solution to this problem [63].

Transfer Learning methods use information from different domains that are similar to each other to improve the performance of machine learning methods in the domain in which the recognition is being done [64]. In transfer learning, the domains are grouped with respect to the direction of the information transfer. The domain from which the information is being transferred is called the Source Domain (D^S), while the domain to which the information is being transferred is called the Target Domain (D^T).

A transfer learning method's success is evaluated by comparing recognition performances of the systems that are trained with different proportions of the available target domain data. There are three criteria that a transfer learning method has to meet in order to be successful while being compared with a system that only uses target domain data. These criteria are: Having higher accuracy while using a small or no proportion of target domain data in training (Higher Start); Increasing the system's performance faster while the proportion of target domain data that is used in training is increased (Higher Slope); Having higher performance when all the target domain data is used in training (Higher Asymptote). These three criteria are being visualized in Figure 7.1.

The effect of transfer learning methods on performance is heavily dependent on

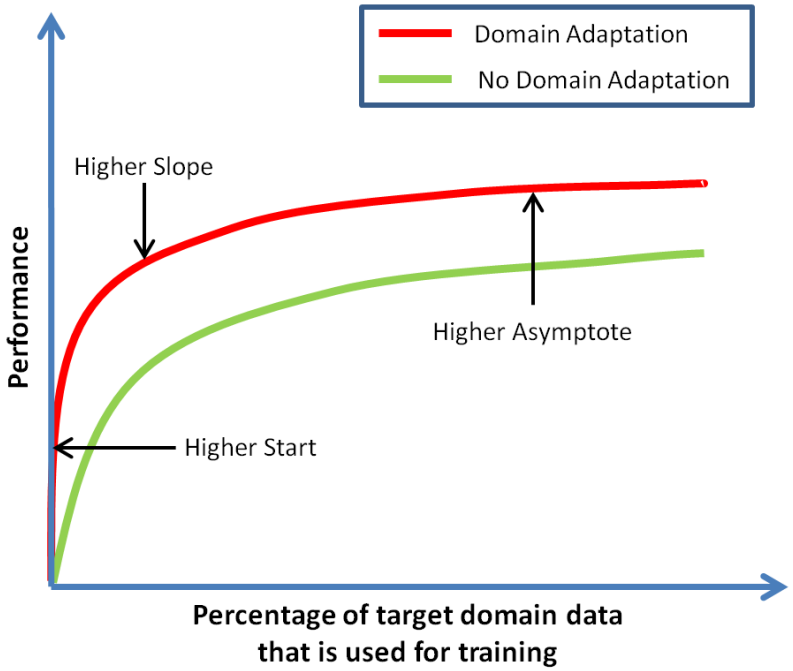

Figure 7.1. Three criteria for a successful transfer learning application.

the choice of the domains. If the transfer is done between irrelevant domains, the recognition performance may not improve or even get worse, which is called Negative Transfer. In order to avoid negative transfers, transfer must be done between similar domains.

Although transfer learning methods have proven to be successful in the related field of activity recognition [65] and suggested to be used for improving sign language and gesture recognition systems [63] there are very few studies conducted on this topic. Farhadi et al. proposed transferring word models that are learned on avatar data to new domain of human signers [66]. In their study, they use a set of shared words that are labeled in both the avatar and the human datasets and a set of target words that are only labeled in the avatar datasets. They build discriminative feature spaces from common projections obtained from shared words and use these spaces to build word classifiers for human signers by only using avatar datasets. In a more recent study, Venkatesan examined boosting based transfer learning methods for gesture recognition using accelerometer data, and report improvements [67].

In order to evaluate the applicability of transfer learning methods to the gesture and sign language recognition problems, we proposed using a domain adaptation method, namely Frustratingly Easy Domain Adaptation [3], and conducted experiments on transferring information from hand written digits to the hand drawn digit gesture recognition domain [68]. For modeling the gestures we used Hidden Markov Models [6], which is widely used in the sign language [5] and gesture recognition [69,70] fields.

7.1. Frustratingly Easy Domain Adaptation

In a transfer learning problem, data coming from similar domains are expected to have similar features. However, as samples from different domains come from different distributions, classical machine learning algorithms lack the ability to discriminate between classes that have samples from different domains and require preprocessing of the samples. In order to solve this problem, Daumé III [3] proposed an easy to implement feature augmentation method for adapting samples from different domains.

Let the samples from source and target domain, D^S and D^T, have the features f_S and f_T that lie in the $S = \mathbb{R}^F$ space. A new augmented input space $\breve{S} = \mathbb{R}^{3F}$ is defined. Then mapping functions Ψ^S and Ψ^T are defined for each domain to map samples from the space S to the new space $\breve{S}$. These mapping functions are as following:

$$\Psi^S(f_S) =< f_S, f_S, \mathbf{0} >, \qquad \Psi^T(f_T) =< f_T, \mathbf{0}, f_T > . \tag{7.1}$$

The $\mathbf{0}$ are zero vectors in the space $S = \mathbb{R}^F$. By mapping the samples from different domains to this new augmented space using the respective mapping functions, the source and target domain samples that belong to different classes become linearly separable. Furthermore, samples from the target domain are weighted more, as these samples will have less distance to a given test sample that comes from the target domain.

7.2. Quantization of Coordinates to Trajectories

In order to apply the Frustratingly Easy Domain Adaptation to our problem of gesture recognition, we represent our samples as series of trajectories by quantizing the directions between consequent coordinate points. In both of the domains, samples are represented as time series of coordinate points. However, hand drawn digit gestures lie in 3D space while hand written digits lie in 2D space. The first thing we do is to take the projection of gestures from the 3D space to the 2D space by omitting the z coordinates, in order to have both domains' samples in the same space.

Although it is possible to use the 2D coordinate information for the feature augmentation method, the coordinates need normalization in size and space. Therefore we first interpolate each sample to have the same number of coordinate points and then using these resampled coordinates, we create trajectory representations. The trajectory representations are obtained by quantizing the movement direction between sequential coordinate points. The possible direction angles (360 degrees) are segmented into eight 45 degree segments. Instead of starting the segments from 0 degrees, we give an offset of 22.5 degrees to each segment, in order the trajectories to have more natural movement directions (i.e. Trajectory Up having the -22.5–22.5 degree segment) as it can be seen in Figure 7.2.

After each sample is converted into a series of trajectories, we apply the Frustratingly Easy Domain Adaptation and transform each sample into a series of domain adapted vectors. Finally, these vectors are used to train Hidden Markov Models [6] for each digit which will be used for gesture recognition.

7.3. Experiments

In order to evaluate the applicability of domain adaptation to the gesture recognition problem we applied the Frustratingly Easy Domain Adaptation method [3], to a gesture recognition problem. We propose using hand written digits as source domain and transfer the information to the target domain, which is hand drawn digit gestures.

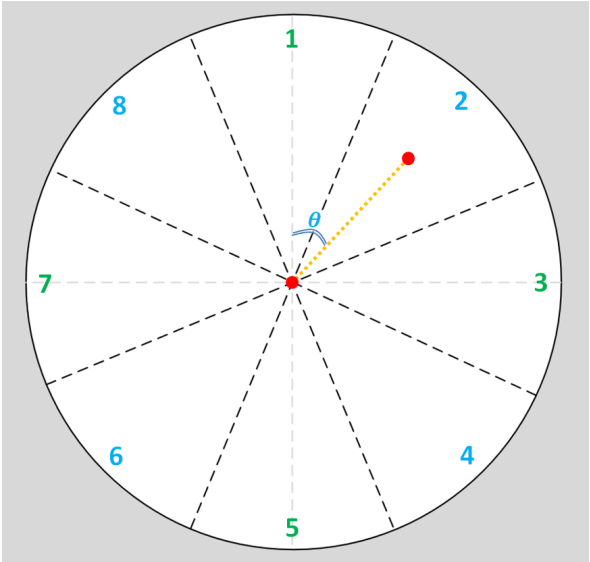

Figure 7.2. Quantized segments and a trajectory sample with the angle θ that belongs to the 2nd segment.

Both domains have coordinates as common features. Therefore, they are both represented by the trajectories that are calculated from consequent coordinates. To model these trajectories through time we used Hidden Markov Models. As the input space of our HMMs are domain adapted trajectory vectors, we used Mixture Models (more specifically Gaussian Mixture Models, assuming our samples to come from a normal distribution) to represent the observations. We used the HMM implementations of Bayesian Network Toolbox [71] in all of our experiments.

7.3.1. Dataset

In our experiments we used two datasets that represent the source and the target domains. The target domain dataset is collected by Keskin et al. [72], in which the users draw digits in mid air. The dataset was recorded by using the Microsoft Kinect [15] sensor and the 3D coordinates of user's joints are given for each frame. The dataset consist of 13 users, including both left and right handed users, who repeat each digit 10 times. Hand drawn digit gesture coordinate samples can be seen in Figure 7.3.

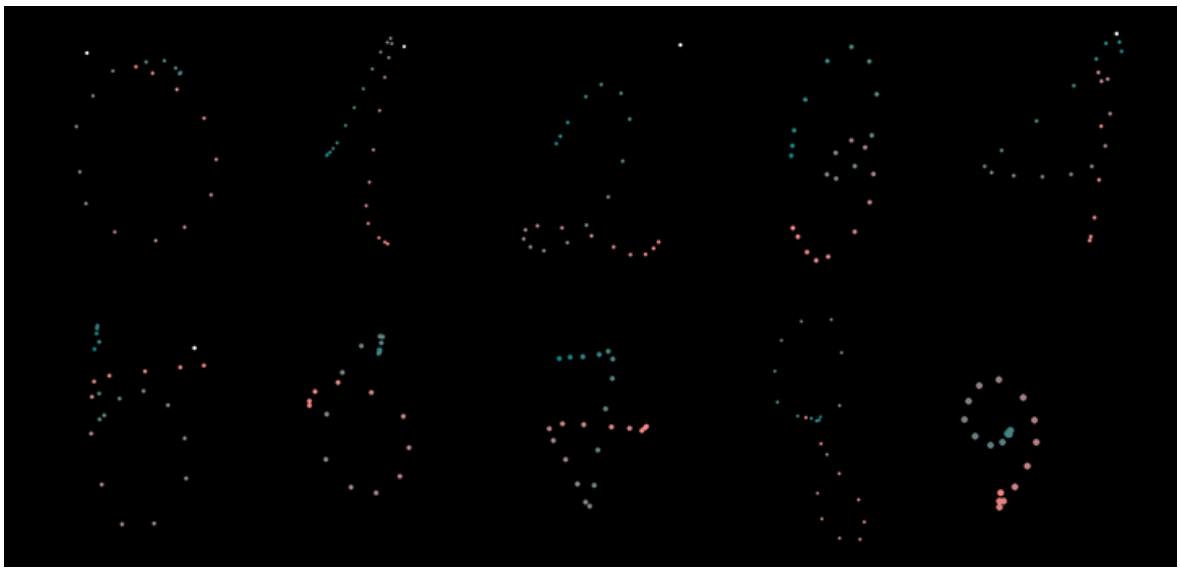

Figure 7.3. Coordinate samples from the hand drawn digit gesture dataset that is the target domain.

As the source domain dataset we used the Chars74k dataset that is collected by Campos et al. [73]. It consists of 55 samples of each hand drawn digit and provides 2D coordinate series of each sample. Hand written digit coordinate samples can be seen in Figure 7.4.

In order to have the source and the target domain samples in the same coordinate space, we took the projection of the 3D coordinates of hand gesture coordinates to the 2D coordinate space.

7.3.2. Experiment Setup

With the purpose of examining the effects of the domain adaptation to the recognition performance we have designed four experiment setups that have different training datasets.

Only the Target Domain Data (T_T): In this setup we have only used the target domain samples for training. All of the users' samples are randomly split into two

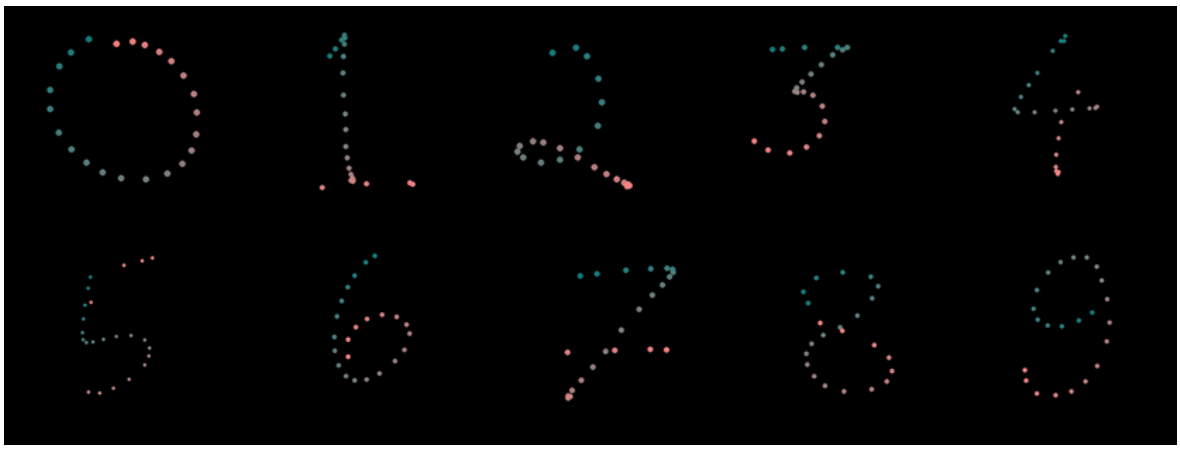

Figure 7.4. Coordinate samples from the hand written digit dataset (Chars74k) that is the source domain.

groups, 60% of the samples are used for training and the remaining 40% is used for testing.

Only the Source Domain Data (T_S): In this setup we have only used source domain samples for training HMMs. All of the target domain samples are used for testing.

Combined Target and Source Domain Data (T_{CTS}): In this setup the target domain samples are split in the same manner as in the T_T and same proportions of samples are used for training and testing. In addition to the target data all of the source domain samples are used for training the HMMs.

Domain Adapted Target and Source Domain Data (T_{DATS}): In this setup we use the same sample setup as in T_{CTS} for traning HMMs. However, before combining target and source domain samples we apply feature augmentation to the samples.

For all of the experiment setups we have optimized HMM parameters using grid search. The search spaces for each parameter is as following: Number of States $N = \{1, 2, 3, 4, ..., 25\}$; Resampled Observation Length $O_L = \{20, 30, 50\}$; Number of Mixtures $M = \{1, 2, 3, 5, 7, 9, 11, 13\}$.

Since the target samples are split randomly, we repeat each experiment that use target data for training five times and report the mean and standard deviation of the recognition performance. Furthermore, in order to eliminate the effects of user specific gesture patterns we do leave-one-user-out cross-validation in the experiment setups that use target domain data for training.

Finally, to examine the transfer learning method's success we incrementally add the target data into training and look for the three criteria that are visualized in Figure 7.1.

7.3.3. Results and Discussion

Using the experiment setups that are defined in Section 7.3.2, we report the mean and standard deviation results of hand drawn digit gesture recognition. In the experiments, the percentage of target domain data is incremented iteratively in order to examine its effect to the recognition performance. The results of our experiments can be seen in Table 7.1.

Table 7.1. Performance evaluation of domain adaptation.

%Target	T_{DATS}	T_{CTS}	T_T	T_S
16,6 %	40,23 ± 7,7	47,19 ± 8,0	42,19 ± 6,6	37,17 ± 5,1
33,3 %	49,54 ± 6,3	53,92 ± 6,3	50,77 ± 6,0	37,17 ± 5,1
50,0 %	56,92 ± 5,9	57,81 ± 7,4	54,65 ± 5,7	37,17 ± 5,1
66,6 %	60,58 ± 7,4	62,58 ± 5,4	61,73 ± 7,4	37,17 ± 5,1
83,3 %	65,38 ± 4,8	63,58 ± 5,7	64,65 ± 5,4	37,17 ± 5,1
100 %	68,62 ± 5,2	67,18 ± 4,8	66,96 ± 4,7	37,17 ± 5,1

It can be seen from our experiment results that the performance of domain adaptation (T_{DATS}) increases as larger proportions of the target domain data used in the training. Furthermore, domain adapted setup has the highest recognition performance when all of the target domain data is used for training. However, the domain adapta-

tion's performance is suppressed by two other setups (T_{CTS} and T_T) while low proportions of target domain data is used for training. This loss of improvement in recognition is an indication of negative transfer, meaning that there are inconsistencies between class samples from different domains.

Considering the fact that users were allowed to lift their hands while performing hand written digits, we omit digits that are affected by the domain specific features. Therefore, we choose the digits $\{2, 3, 4, 6\}$ and repeated our experiments. The results of the experiments with this subset of digits can be seen in Table 7.2.

Table 7.2. Performance evaluation of domain adaptation after removing the classes that cause negative transfer.

%Target	T_{DATS}	T_{CTS}	T_T	T_S
16,6 %	44,23	48,65	47,02	26,00
33,3 %	62,31	58,17	58,94	26,00
50,0 %	67,60	63,75	63,65	26,00
66,6 %	74,71	73,65	71,44	26,00
83,3 %	78,08	75,29	71,06	26,00
100 %	80,87	78,85	74,04	26,00

As expected, using the domain adaptation (T_{DATS}) improved the recognition performance over using the combined samples (T_{CTS}) and only using the target domain samples (T_T). As it can be seen in Table 7.2, the application of domain adaptation meets at least two of the criteria of a successful transfer learning application, that are Higher Slope and Higher Asypmtote.

8. FACIAL LANDMARK LOCALIZATION IN DEPTH IMAGES

Landmark localization is a crucial initial step for face processing applications. Such applications include but are not limited to biometrics [74], facial expression analysis [75], age estimation [76] and sign language recognition [77]. In biometrics applications, the localized landmarks are used to align faces before matching or to extract local features. On the other hand, in facial expression analysis and sign language recognition, the landmarks are tracked through time to extract features in the spatio-temporal domain. For all these different applications a better landmark localization results in a better performance of the overall system. Most of the systems use 2D images since 2D images are easy to acquire using commonly available video cameras. However, 2D face images are vulnerable to illumination and pose changes. The availability of inexpensive depth cameras has led to the widespread use of 3D face images, which overcome these difficulties. Therefore, the development of a reliable 3D facial landmark localization method has become essential.

Facial landmark localization methods generally utilize heuristic approaches as well as statistical methods. Heuristics rely on unique properties of the facial landmarks on the face: For example, the nose tip resides on the symmetry axis of the face and can be localized using the shape properties. Similarly, the corners of the eye and mouth can easily and successfully be localized by heuristics using shape properties. Such an example to these methods is [74] in which Alyüz et al. propose a heuristic method which uses curvature information, symmetry axis and shape index to locate the nose tip, the nose and the eye corners in 3D faces.

Statistical 3D landmark localization methods also exploit the features of facial landmarks such as local texture and shape. Unlike heuristic-based approaches which require a unique rule for each landmark, statistical methods utilize feature statistics in a uniform approach for all landmarks. Most recent statistical methods also use the shape

information that is extracted by using facial landmarks relative positions. Creusot et al. [78] propose a statistical facial landmark localization method utilizing shape information in addition to the local features of landmarks. Several candidates are identified on a local 3D mesh and the most probable candidate is identified through shape analysis. Another statistically motivated method using shape information proposed by Sukno et al. [79] localizes facial landmarks under occlusion and expression changes. In [79], the shape context of facial landmarks is used together with local feature analysis. Different subsets of candidate points are evaluated, resulting in robustness against missing landmarks due to occlusions. A similar concept for estimating occluded 3D landmarks is also proposed in [80], where partial Gappy Principal Component Analysis is used to restore missing landmark coordinates. In another study, Farrelli et al. [81] proposed a Random Decision Forest based framework in which patches extracted from depth images cast votes to localize facial landmarks.

Supervised Descent Method (SDM) [1] was proposed to solve nonlinear optimization problems by turning the problem into least squares form and applying regression. In 2D domain, SDM has been proven to be successful for facial landmark localization. Recently Camgoz et al. [82] achieved state-of-the-art performance on facial landmark localization in 3D depth images using SDM. They conducted experiments using Scale-Invariant Feature Transform (SIFT) [83] and Histogram of Oriented Gradients (HOG) [59] to represent local features of facial landmarks and showed that both of the features yield accurate localization results. Taking [82] as a baseline, we propose to use ridge regression for Supervised Descent Method (which we call *Supervised Ridge Descent*) instead of least squares regression for facial landmark localization in depth images. Additionally, we propose to change feature sizes in each iteration in a coarse to fine fashion. In this way, we aim to capture more details in later iterations by focusing on smaller regions.

8.1. Supervised Descent Method (SDM)

Supervised Descent Method has achieved state-of-the-art performances in several computer vision applications which previously relied heavily on nonlinear optimization

methods [1, 84]. Xiong et al. [1] proposed to approach the non-linear optimization by learning the descent directions from the training samples and then use these previously learned descent directions on new unseen test samples. SDM's best known application is facial landmark localization, also known as the IntraFace [85]. It has been used to achieve state-of-the-art performances in face tracking and alignment.

Facial landmark localization using SDM starts with creating an average face shape which provides the initial landmark locations for the face images. At the beginning of the training, landmarks are placed in these initial locations (x_0). Then the shape increment (Δx) required to displace the landmarks from their current location (x_k) to its ground truth location (x_*) is calculated. This is written as a function of the features extracted from the current shape estimate (ϕ_k) as:

$$\Delta x_k = x_* - x_k = R_k \phi_k + b_k \tag{8.1}$$

To estimate the parameters of this function, R_k and b_k, the problem is written in least squares format as in Equation 8.2, where i and k represent the sample and iteration indices, respectively.

$$\underset{R_k, b_k}{\operatorname{argmin}} \sum_{x_k^i} \left\| \Delta x_k^i - R_k \phi_k^i - b_k \right\|^2 \tag{8.2}$$

By using the closed form solution of least squares regression, both R_k and b_k parameters are estimated. Then R_k and b_k are used to update the location of the landmark as:

$$x_{k+1} = x_k + R_k \phi_k + b_k \tag{8.3}$$

The training procedure continues until the landmarks converge to the actual positions. When a test sample comes, landmarks are placed in their initial positions (x_0) and their positions are updated using Equation 8.3.

8.2. Supervised Ridge Descent (SRD)

SDM was originally designed to use least square regression (LSR) to estimate its predictor parameters. While using LSR, one needs to take the inverse of the X^TX matrix, X being the observations of predictors. However, the X^TX matrix becomes singular when the observation size is large and/or the predictors are strongly correlated. To overcome the singularity issue Xiong et al. [1] proposed to use PCA to regularize their matrix before taking the inverse of it.

In this study we propose to use *ridge regression* (RR) instead of LSR, in which the matrix singularity issue is dealt by adding a $\Gamma^T\Gamma$ matrix to the X^TX matrix, Γ being the regularization term which is proportional to the identity matrix. Although we lose precision by taking the inverse of $\Gamma^T\Gamma + X^TX$ instead of X^TX, we avoid over-fitting and large variances in the estimators.

Our formalization of ridge regression can be seen in Equation 8.4, in which β_k, λ_k, b_k represent the estimator, regularization term and offset parameter of the k^{th} iteration, respectively. The rest of the parameters Δx_k^i and ϕ_k^i represent the landmarks' distance from the ground truth and their features in these positions of the i^{th} sample, respectively. As in [82] and [84] we used HOG features as facial landmark descriptors. However, in each iteration, the size of the HOG features and the regularization term's value has been decreased to be able to descend more precisely to the ground truth.

$$\underset{\beta_k, b_k}{\operatorname{argmin}} \sum_{x_k^i} \left\| \Delta x_k^i - (\phi_k^i)'\beta_k - b_k \right\|^2 + \left\| \lambda_k \beta_k \right\|^2 \tag{8.4}$$

To calculate the ridge regression estimator, β_k, for each iteration, we use Equation 8.5 in which I and λ_k represent the identity matrix with the same size as the observation matrix and the regularization term of the k^{th} iteration. Φ_k and ΔX_k are constructed by concatenating each training samples' HOG features and distances from the ground truth into two matrices, respectively. Note that both the feature matrix Φ_k and shape

increment ΔX_k are normalized to zero mean before regression.

$$\beta_k = ((\Phi_k)^T \Phi_k + \lambda_k I)^{-1} (\Phi_k)^T \Delta X_k \tag{8.5}$$

After learning the ridge regression estimator, β_k, and calculating the offset b_k for each iteration, we use Equation 8.6 to localize facial landmarks starting from the initial points which are defined by the average landmark positions of the training samples.

$$x^i_{k+1} = x^i_k + (\phi^i_k)' \beta_k + b_k \tag{8.6}$$

In Equation 8.6, ϕ^i_k, x^i_{k+1} and x^i_k represent the i^{th} sample's HOG features of the k^{th} iteration and the same sample's facial landmarks' locations of the the $k+1^{th}$ and k^{th} iterations, respectively.

8.3. Experiments

To evaluate the proposed method, we conducted experiments on the commonly used Bosphorus 3D Face Database [86]. The Bosphorus database contains 4666 face samples belonging to 105 users. Each sample's 2D color image, 3D point cloud and manually annotated 24 facial landmark positions are provided by the database. The Bosphorus database contains a variety of pose and facial expression variations as well as occluded faces, making it a challenging database.

In our experiments, we worked on samples with frontal poses which had no occluding objects covering the face. 22 of the 24 facial landmarks were selected to be localized since the other two are ear dimples and are not visible in frontal images. Selected landmarks are eye, mouth, nose and eyebrow corners, middle points of lips, eyebrows, nose, chin, and the nose saddles, all of which can be seen in Figure 8.1.

We compared our method with the state-of-the-art 3D facial landmark localization methods for depth images. A summary of these methods are given in Table 8.1. To be able to compare our method with the most successful methods, namely Sukno

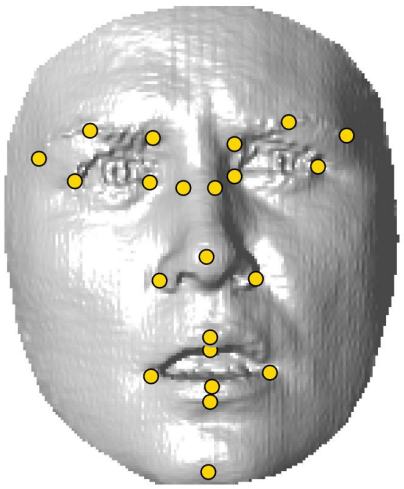

Figure 8.1. 22 landmarks used in our experiments.

et al. [79] and Camgoz et al. [82], who are both using statistical facial landmark localization methods, we used the same experimental setup as theirs and reported our results on 10 landmarks, which are common to all of the methods. We selected the frontal non-occluded face samples and divided them into two folds in which the users were exclusive to their folds. All the experiments have been done using two-fold cross-validation and we iterated Supervised Ridge Descent (SRD) six times as it usually converges after the fourth iteration.

Table 8.1. Summary of the proposed method and the state-of-the-art methods. (#LM = Number of Landmarks)

	#LM	Training Size	Test Size	Features	Method Used
Alyüz et al. [74]	5	–	2902	Shape Index	Heuristics
Creusot et al. [78]	14	99	2803	Surface Descriptors	LDA and Adaboost
Sukno et al. [79]	14	1402 x 2	1402 x 2	ASPC [87]	Statistical Shape Models
Camgoz et al. [82]	10	1446 x 2	1446 x 2	SIFT [83] - HOG [59]	SDM [1]
SRD (Our method)	22	1420 x 2	1420 x 2	HOG [59]	SDM [1] - Ridge Regression [88]

In our first experiments, our aim is to find the optimum λ_k values and HOG feature sizes. Our experiments yield optimum λ_k values to be [300.00, 110.40, 40.60,

14.94, 5.49, 2.02, 0.74] and HOG feature sizes to be [0.20, 0.17, 0.14, 0.12, 0.09, 0.06] $\times ImageSize$ for the iterations from one to six, respectively.

The SRD method has two main novelties when compared to the SDM: 1) the use of ridge regression and 2) the use of adaptive feature sizes from coarse to fine resolution. In order to evaluate the independent contributions of these novelties, we performed several experiments by incrementally adding ridge regression and adaptive features to the classical SDM. As it can be seen from Table 8.2, using ridge regression instead of least squares regression improves the performance drastically (See *SDM* and *SRD with Fixed Feature Size* columns). Similarly, using adaptive features instead of fixed features increases the performance for both SDM and SRD approaches (See *SDM* vs. *SDM with Adaptive Feature Size* columns and *SRD* vs. *SRD with Fixed Feature Size* columns). By incorporating both ridge regression and adaptive features, our SRD approach attains the best overall results (See *SRD* column).

As observed from Table 8.2, our best performing landmarks are eye and mouth corners, which have strong geometric characteristics. However, our method struggled to localize chins and nose saddles which are difficult to locate accurately even by manual annotation. These findings were also backed up as we visualized the best and worst performing facial samples which can be seen in Figure 8.3. It can be seen from Figure 8.3 that ground truth locations of nose saddles differ for each subject which is probably due to the subjective preferences of the manual annotators.

To see if these results are consistent with all the samples, we calculated the cumulative error distribution, which can be seen in Figure 8.2. By analyzing the curves of chin and nose saddles, we can confirm that both of these landmarks are problematic landmarks and their error is distributed over the whole database. This may be caused by false annotation of the data as these landmarks are more ambiguous than the others.

To compare our method with the the state-of-the-art methods, we used a subset of 10 points that most methods reported results on. As it can be seen in Table 8.3 the proposed method achieves the state-of-the-art performance on all of the landmarks

Table 8.2. Landmarks' mean and standard deviation of errors. SDM = Supervised Descent Method, SRD = Supervised Ridge Descent, FFS = Fixed Feature Size, AFS = Adaptive Feature Size.

Landmarks	SDM	SDM with AFS	SRD with FFS	**SRD (Our Method)**
Outer left eyebrow	5.01 ± 2.97	4.16 ± 2.41	4.39 ± 2.58	$\mathbf{4.13 \pm 2.36}$
Middle left eyebrow	5.17 ± 3.07	4.69 ± 2.67	4.68 ± 2.81	$\mathbf{4.37 \pm 2.56}$
Inner left eyebrow	4.02 ± 2.45	3.52 ± 1.92	3.48 ± 2.08	$\mathbf{3.13 \pm 1.74}$
Inner right eyebrow	3.86 ± 2.23	3.28 ± 1.75	3.34 ± 1.97	$\mathbf{2.99 \pm 1.66}$
Middle right eyebrow	4.68 ± 2.86	4.19 ± 2.39	4.11 ± 2.49	$\mathbf{3.88 \pm 2.25}$
Outer right eyebrow	5.02 ± 4.10	4.19 ± 3.43	4.23 ± 3.53	$\mathbf{4.02 \pm 3.33}$
Outer left eye corner	3.16 ± 2.00	2.81 ± 1.57	2.63 ± 1.68	$\mathbf{2.56 \pm 1.45}$
Inner left eye corner	2.28 ± 1.55	2.12 ± 1.23	1.93 ± 1.39	$\mathbf{1.90 \pm 1.14}$
Inner right eye corner	2.10 ± 1.46	2.03 ± 1.21	1.84 ± 1.34	$\mathbf{1.84 \pm 1.15}$
Outer right eye corner	3.04 ± 2.00	2.89 ± 1.81	2.57 ± 1.84	$\mathbf{2.51 \pm 1.63}$
Nose saddle left	7.61 ± 3.96	7.08 ± 3.77	7.16 ± 3.73	$\mathbf{6.78 \pm 3.59}$
Nose saddle right	7.77 ± 4.03	7.29 ± 3.81	7.32 ± 3.82	$\mathbf{6.92 \pm 3.66}$
Left nose peak	2.51 ± 1.99	2.21 ± 1.31	2.18 ± 1.81	$\mathbf{1.96 \pm 1.20}$
Nose tip	3.34 ± 2.41	2.96 ± 1.90	3.01 ± 2.27	$\mathbf{2.65 \pm 1.76}$
Right nose peak	2.56 ± 2.04	2.18 ± 1.23	2.18 ± 1.96	$\mathbf{1.99 \pm 1.26}$
Left mouth corner	4.37 ± 3.82	3.09 ± 1.97	3.41 ± 3.39	$\mathbf{2.92 \pm 2.13}$
Upper lip outer middle	3.66 ± 3.52	2.71 ± 1.95	2.99 ± 3.25	$\mathbf{2.46 \pm 2.04}$
Right mouth corner	4.50 ± 3.85	3.05 ± 1.92	3.54 ± 3.33	$\mathbf{2.91 \pm 2.07}$
Upper lip inner middle	3.62 ± 3.47	2.64 ± 1.90	2.84 ± 3.25	$\mathbf{2.39 \pm 1.96}$
Lower lip inner middle	4.65 ± 5.01	2.60 ± 2.09	3.56 ± 4.44	$\mathbf{2.39 \pm 2.28}$
Lower lip outer middle	5.49 ± 5.59	3.14 ± 2.35	4.30 ± 5.07	$\mathbf{2.90 \pm 2.65}$
Chin middle	6.45 ± 5.60	5.32 ± 3.60	5.65 ± 4.87	$\mathbf{5.08 \pm 3.45}$
Mean Error	4.31 ± 3.18	3.55 ± 2.19	3.70 ± 2.86	$\mathbf{3.30 \pm 2.15}$

except the nose tip. Considering the manual annotation error for the nose tip (2.96mm, see Table 8.3), our average automatic localization error (2.65mm) can still be considered as not too high.

Table 8.3. Mean and standard deviation of 10 common facial landmark localization errors on Bosphorus 3D face database.

	Inner Eye Corners	Outer Eye Corners	Nose Tip	Nose Corners	Mouth Corners	Chin
Manual Annotation [74]	2.51	–	2.96	1.75	–	–
Alyüz et al. [74]	3.70	–	3.05	3.10	–	–
Creusot et al. [78]	4.14 ± 2.63	6.27 ± 3.98	4.33 ± 2.62	4.16 ± 2.35	7.95 ± 5.44	15.38 ± 10.49
Sukno et al. [79]	2.85 ± 2.02	5.06 ± 3.67	$\mathbf{2.33 \pm 1.78}$	3.02 ± 1.91	6.08 ± 5.13	7.58 ± 6.72
Camgoz et al. [82] (SIFT)	2.26 ± 1.79	4.23 ± 2.94	2.72 ± 2.19	4.57 ± 3.62	3.14 ± 2.71	5.72 ± 4.31
Camgoz et al. [82] (HOG)	2.33 ± 1.92	4.11 ± 3.01	2.69 ± 2.20	4.49 ± 3.62	3.16 ± 2.70	5.87 ± 4.19
SRD (Our method)	$\mathbf{1.87 \pm 1.14}$	$\mathbf{2.54 \pm 1.54}$	2.65 ± 1.76	$\mathbf{1.97 \pm 1.23}$	$\mathbf{2.92 \pm 2.10}$	$\mathbf{5.08 \pm 3.45}$

8.4. Discussion

Many applications rely on the analysis of facial data to analyze, recognize and understand humans and their behaviors. Many of these applications start with facial landmark localization to be able to either align faces or track these landmarks. Thus a successful facial landmark localization is essential to the success of various facial processing tasks.

In this chapter, we presented Supervised Ridge Descent, in which we proposed using ridge regression instead of least squares regression while training Supervised Descent Method. We also use decreasing feature sizes in each iteration, which become smaller as the system iterates, turning the localization into a coarse to fine approach. Our experiments show that both improvements increase the performance significantly.

SRD was trained using HOG features in a similar manner to SDM. We experimented on the Bosphorus 3D Face Database and compared our method with the state-of-the-art methods, which work on 10 common facial landmarks of the Bosphorus 3D

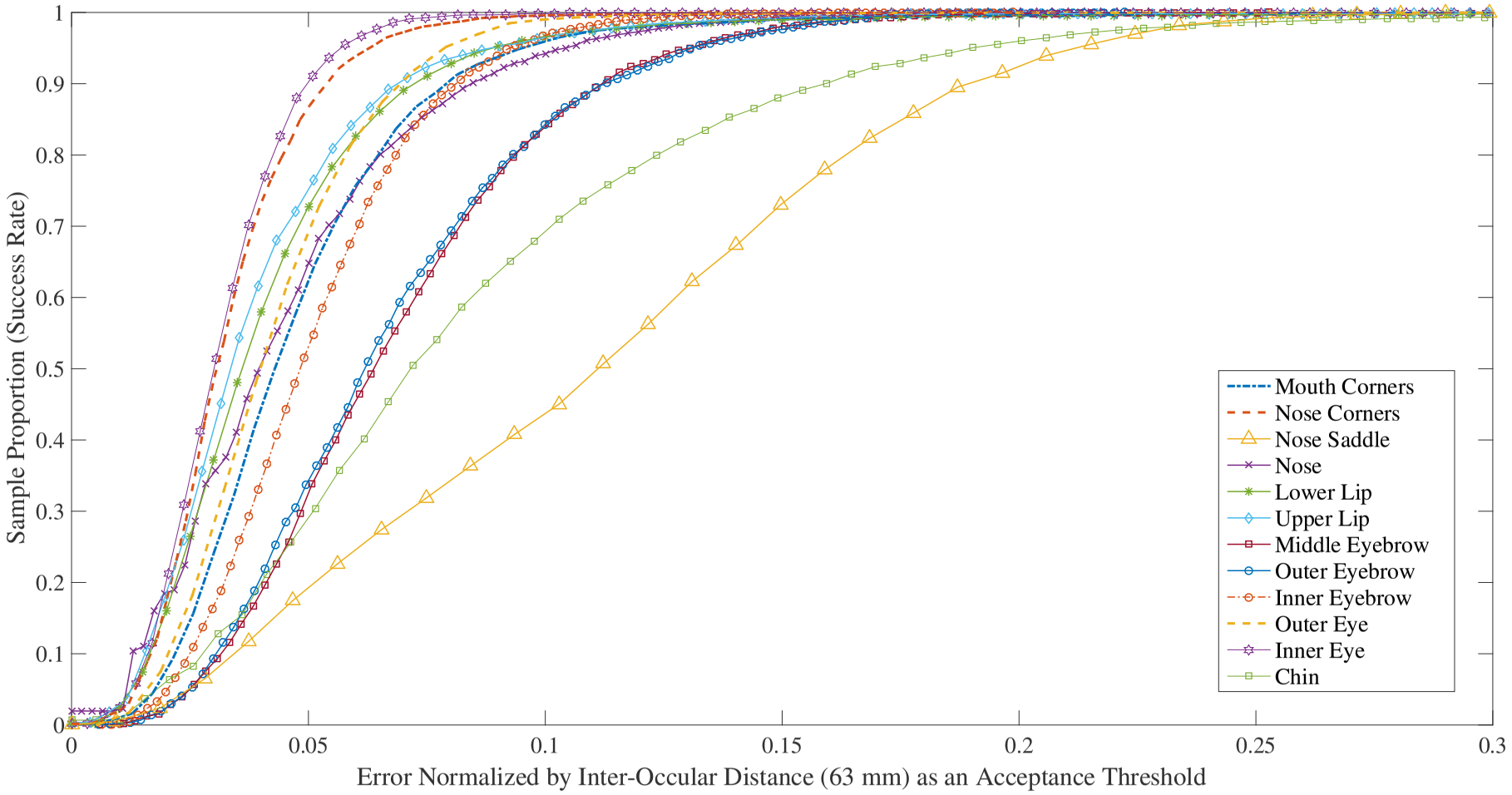

Figure 8.2. Cumulative error distribution of different landmarks.

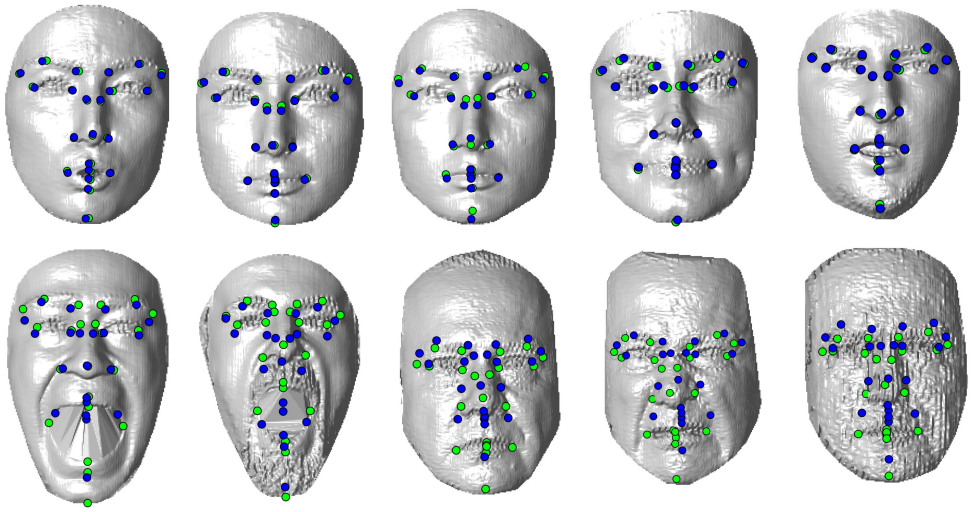

Figure 8.3. The first row shows the faces with the best landmark localization performance, while the second row shows samples with the worst performance. Green (Light) Dots = Ground Truth, Blue (Dark) Dots = Prediction. (Best seen in color)

Face database, namely, eye corners, nose tip, nose corners, mouth corners and chin. Except for the nose tip, our approach achieved state-of-the-art performance on all of the landmarks landmarks. However, our nose tip error is close to human annotation done by [74], which may indicate that the annotation variance may be the reason of this behaviour.

To improve our system, we plan to use 3D descriptors instead of 2D descriptors. To generalize our system, cross database experiments should also be conducted. Furthermore, feature learning methods can be used to learn features instead of using descriptors such as HOG.

9. CONCLUSION

In this thesis, we present an interactive communication interface for the hearing impaired, which we called HospiSign, that is designed to assist the Deaf in their hospital visits. The HospiSign platform guides its users through a tree-based activity diagram by asking specific questions and requiring the users to answer from the given options. This approach alleviates the problem of recognizing the sign displayed by the user among all the signs, since the system only looks for the possible answers to recognize the sign in each step. Our experiments show that the activity diagram not only increases the recognition performance of the system, but also makes our system more user-friendly and accessible. The system consists of a personal computer, a touch display to visualize the questions and answers, and a Microsoft Kinect v2 sensor, which provides body pose information, to capture the responses of the users. The developed software for the interface was designed in order to be easily adaptable to other applications, such as banking applications.

In order to develop the proposed system, we first collected BosphorusSign, a Turkish Sign Language (Türk İşaret Dili, TİD) corpus in health and finance domains. We have collected 859 sample signs from three categories: 487 samples belonging to the health domain, 177 samples belonging to the finance domain and 195 samples comprising commonly used signs and phrases in everyday life. The corpus is aimed to contain six repetitions from 10 native signers, making the corpus the largest available database for sign language recognition.

The database is collected using the Microsoft Kinect v2 sensor, and all the modalities that are provided by the sensor are recorded. Recording sessions have been conducted using the recording software we have developed, which guides the subject by displaying the sign samples and asking for their repetitions. The software also enables the recording person to annotate sign borders online. The script for each recording session is generated randomly so that each session would be unique.

BosphorusSign has two main target users. The first target user group is sign language recognition researchers. Sign language recognition community will be provided with recording sessions and their sign border annotations. The second target user group is the sign language linguists, who will be able to study our publicly available samples and their annotations. The corpus will also serve as a lexicon to people who would like to learn Turkish Sign Language, similar to [89].

To realized the sign language recognition module of HosipSign, we propose using several features, normalization techniques, temporal modeling approaches and classification methods. We conducted experiments in order to find the best combination of features and came to the conclusion that combining Hand Joint Distance and Hand Movement Distance features achieved the highest recognition performance. Moreover, our experiments showed us that both the Dynamic Time Warping (DTW) and Temporal Template (TT) based temporal modeling approaches and their respective classification methods, k-Nearest Neighbors and Random Decision Forests, yielded competitive results. Evaluated on a subset of BosphorusSign consisting of 662 samples belonging to 33 sign classes that are collected from three native TİD users, DTW and TT based approaches achieved 96.75% and 95.63% mean recognition performance respectively.

Moreover, we have investigated the applicability of domain adaptation techniques to the gesture recognition problem, which is closely related to the sign language recognition field, and reported improvements in the performance. As future work, we are planing to apply domain adaptation methods to improve the user independence.

Last but not least, we have studied facial landmark localization techniques in color images, which are widely used in sign language recognition, in order to extract facial gestures. We have used Supervised Descent Method to locate facial landmarks on color videos and used these locations to extract the baseline features more accurately. Furthermore, we have proposed an extension to Supervised Descent Method, which we called Supervised Ridge Descent, that uses Ridge Regression instead of Least Squares Regression. The proposed method achieved state-of-the-art facial landmark localization performance in frontal depth images.

REFERENCES

1. Xiong, X. and F. De la Torre, "Supervised Descent Method and its Applications to Face Alignment", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

2. Kadir, T., R. Bowden, E. J. Ong and A. Zisserman, "Minimal Training, Large Lexicon, Unconstrained Sign Language Recognition", *British Machine Vision Conference*, 2004.

3. Daumé III, H., "Frustratingly Easy Domain Adaptation", *arXiv preprint arXiv:0907.1815*, 2009.

4. Camgoz, N., V. Struc, B. Gokberk, L. Akarun and A. Kindiroglu, "Facial Landmark Localization in Depth Images Using Supervised Ridge Descent", *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 136–141, 2015.

5. Cooper, H., B. Holt and R. Bowden, "Sign Language Recognition", *Visual Analysis of Humans*, pp. 539–562, Springer, 2011.

6. Rabiner, L. R., "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", *Proceedings of the IEEE*, Vol. 77, pp. 257–286, 1989.

7. Berndt, D. J. and J. Clifford, "Using Dynamic Time Warping to Find Patterns in Time Series", *KDD Workshop*, Vol. 10, pp. 359–370, 1994.

8. Starner, T., J. Weaver and A. Pentland, "Real-time American Sign Language Recognition using Desk and Wearable Computer based Video", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, pp. 1371–1375, 1998.

9. Grobel, K. and M. Assan, "Isolated Sign Language Recognition using Hidden

Markov Models", *IEEE International Conference on Computational Cybernetics and Simulation, Systems, Man, and Cybernetics*, pp. 162–167, 1997.

10. Vogler, C. and D. Metaxas, "Parallel Hidden Markov Models for American Sign Language Recognition", *The Seventh IEEE International Conference on Computer Vision*, Vol. 1, pp. 116–122, 1999.

11. Lee, H.-K. and J. H. Kim, "An HMM-based Threshold Model Approach for Gesture Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21, No. 10, pp. 961–973, 1999.

12. Chai, X., G. Li, X. Chen, M. Zhou, G. Wu and H. Li, "VisualComm: A Tool to Support Communication Between Deaf and Hearing Persons with the Kinect", *15th International ACM SIGACCESS Conference on Computers and Accessibility*, p. 76, 2013.

13. Theodorakis, S., V. Pitsikalis and P. Maragos, "Dynamic–Static Unsupervised Sequentiality, Statistical Subunits and Lexicon for Sign Language Recognition", *Image and Vision Computing*, Vol. 32, pp. 533–549, 2014.

14. Pitsikalis, V., S. Theodorakis, C. Vogler and P. Maragos, "Advances in Phonetics-based Sub-Unit Modeling for Transcription Alignment and Sign Language Recognition", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1–6, 2011.

15. Zhang, Z., "Microsoft Kinect Sensor and Its Effect", *IEEE MultiMedia*, Vol. 19, pp. 4–10, 2012.

16. Parton, B. S., "Sign Language Recognition and Translation: A Multidisciplined Approach from the Field of Artificial Intelligence", *Journal of Deaf Studies and Deaf Education*, Vol. 11, pp. 94–101, 2006.

17. Shotton, J., T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook

and R. Moore, "Real-time Human Pose Recognition in Parts from Single Depth Images", *Communications of the ACM*, Vol. 56, pp. 116–124, 2013.

18. Cox, S., "Speech and Language Processing for a Constrained Speech Translation System", *INTERSPEECH*, 2002.

19. Cox, S., M. Lincoln, J. Tryggvason, M. Nakisa, M. Wells, M. Tutt and S. Abbott, "TESSA, a System to Aid Communication with Deaf People", *Fifth International ACM Conference on Assistive Technologies*, pp. 205–212, 2002.

20. Aran, O., I. Ari, L. Akarun, B. Sankur, A. Benoit, A. Caplier, P. Campr, A. H. Carrillo and F.-X. Fanard, "Signtutor: An Interactive System for Sign Language Tutoring", *IEEE MultiMedia*, Vol. 16, No. 1, pp. 81–93, 2009.

21. Zafrulla, Z., H. Brashear, P. Yin, P. Presti, T. Starner and H. Hamilton, "American Sign Language Phrase Verification in an Educational Game for Deaf Children", *International Conference on Pattern Recognition (ICPR)*, pp. 3846–3849, 2010.

22. Weaver, K. A. and T. Starner, "We need to Communicate!: Helping Hearing Parents of Deaf Children Learn American Sign Language", *13th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 91–98, 2011.

23. Hruz, M., P. Campr, E. Dikici, A. A. Kindiroglu, Z. Krnoul, A. Ronzhin, H. Sak, D. Schorno, H. Yalcin, L. Akarun, O. Aran, A. Karpov, M. Saraclar and M. Zelezny, "Automatic Fingersign-to-Speech Translation System", *Journal on Multimodal User Interfaces*, Vol. 4, pp. 61–79, 2011.

24. Zafrulla, Z., H. Brashear, T. Starner, H. Hamilton and P. Presti, "American Sign Language Recognition with the Kinect", *International Conference on Multimodal Interfaces (ICMI)*, pp. 279–286, 2011.

25. Efthimiou, E., S.-E. Fotinea, T. Hanke, J. Glauert, R. Bowden, A. Braffort, C. Collet, P. Maragos and F. Lefebvre-Albaret, *The Dicta-Sign wiki: Enabling Web Com-*

munication for the Deaf, Springer, 2012.

26. Karpov, A., Z. Krnoul, M. Zelezny and A. Ronzhin, "Multimodal Synthesizer for Russian and Czech Sign Languages and Audio-Visual Speech", *Universal Access in Human-Computer Interaction. Design Methods, Tools, and Interaction Techniques for eInclusion*, pp. 520–529, Springer, 2013.

27. Chai, X., G. Li, Y. Lin, Z. Xu, Y. Tang, X. Chen and M. Zhou, "Sign Language Recognition and Translation with Kinect", *IEEE Conference on Automatic Face and Gesture Recognition*, 2013.

28. Gameiro, J., T. Cardoso and Y. Rybarczyk, "Kinect-Sign: Teaching Sign Language to "Listeners" through a Game", *Innovative and Creative Developments in Multimodal Interaction Systems*, pp. 141–159, Springer, 2014.

29. Zafrulla, Z., H. Brashear, H. Hamilton and T. Starner, "Towards an American Sign Langauge Verifier for Educational Game for Deaf Children", *International Conference on Pattern Recognition (ICPR)*, 2010.

30. López-Ludeña, V., C. González-Morcillo, J. López, R. Barra-Chicote, R. Cordoba and R. San-Segundo, "Translating Bus Information into Sign Language for Deaf People", *Engineering Applications of Artificial Intelligence*, Vol. 32, pp. 258–269, 2014.

31. Fenlon, J., A. Schembri, T. Johnston and K. A. Cormier, "Documentary and Corpus Approaches to Sign Language Research", *Research Methods in Sign Language Studies: A Practical Guide*, 2015.

32. Fang, G. and W. Gao, "A SRN/HMM System for Signer-Independent Continuous Sign Language Recognition", *The Fifth IEEE International Conference on Automatic Face Gesture Recognition*, pp. 312–317, 2002.

33. Dreuw, P., C. Neidle, V. Athitsos, S. Sclaroff and H. Ney, "Benchmark Databases

for Video-Based Automatic Sign Language Recognition", *Language Resources and Evaluation*, pp. 1–6, 2008.

34. von Agris, U. and K.-F. Kraiss, "SIGNUM Database: Video Corpus for Signer-Independent Continuous Sign Language Recognition", *The 4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies, Language Resources and Evaluation Conference (LREC)*, pp. 243–246, 2010.

35. Forster, J., C. Schmidt, T. Hoyoux, O. Koller, U. Zelle, J. Piater and H. Ney, "RWTH-PHOENIX-Weather: A Large Vocabulary Sign Language Recognition and Translation Corpus", *LREC*, 2012.

36. Chai, X., H. Wanga, M. Zhoub, G. Wub, H. Lic and X. Chena, *DEVISIGN: Dataset and Evaluation for 3D Sign Language Recognition*, Tech. rep., Key Lab of Intelligence Information Processing of Chinese Academy of Sciences, 2015.

37. Escalera, S., X. Baró, J. Gonzàlez, M. Bautista, M. Madadi, M. Reyes, V. Ponce-López, H. Escalante, J. Shotton and I. Guyon, "ChaLearn Looking at People Challenge 2014: Dataset and Results", *Computer Vision - ECCV 2014 Workshops*, Vol. 8925 of *Lecture Notes in Computer Science*, pp. 459–473, Springer International Publishing, 2014.

38. Stefanov, K. and J. Beskow, "A Kinect Corpus of Swedish Sign Language Signs", *Workshop on Multimodal Corpora: Beyond Audio and Video*, 2013.

39. Johnston, T., "From Archive to Corpus: Transcription and Annotation in the Creation of Signed Language Corpora", *International Journal of Corpus Linguistics*, Vol. 15, pp. 106–131, 2010.

40. Schembri, A., J. Fenlon, R. Rentelis, S. Reynolds and K. Cormier, "Building the British Sign Language Corpus", *Language Documentation and Conservation*, Vol. 7, pp. 136–154, University of Hawaii Press, 2013.

41. Prillwitz, S., T. Hanke, S. König, R. Konrad, G. Langer and A. Schwarz, "DGS Corpus Project: Development of a Corpus Based Electronic Dictionary German Sign Language/German", *3rd Workshop on the Representation and Processing of Sign Languages: Construction and Exploitation of Sign Language Corpora*, p. 159, 2008.

42. Crasborn, O. A. and I. Zwitserlood, "The Corpus NGT: An Online Corpus for Professionals and Laymen", *3rd Workshop on the Representation and Processing of Sign Languages (LREC)*, pp. 44–49, 2008.

43. Bungeroth, J., D. Stein, P. Dreuw, H. Ney, S. Morrissey, A. Way and L. van Zijl, "The ATIS Sign Language Corpus", *International Conference on Language Resources and Evaluation (LREC)*, 2008.

44. Athitsos, V., C. Neidle, S. Sclaroff, J. Nash, A. Stefan, Q. Yuan and A. Thangali, "The American Sign Language Lexicon Video Dataset", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1–8, 2008.

45. Zafrulla, Z., H. Brashear, H. Hamilton and T. Starner, "A Novel Approach to American Sign Language (ASL) Phrase Verification using Reversed Signing", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 48–55, 2010.

46. Matthes, S., T. Hanke, A. Regen, J. Storz, S. Worseck, E. Efthimiou, A.-L. Dimou, A. Braffort, J. Glauert and E. Safar, "Dicta-Sign: Building a Multilingual Sign Language Corpus", *The 5th Workshop on the Representation and Processing of Sign Languages: Interactions Between Corpus and Lexicon (LREC), European Language Resources Association*, 2012.

47. Gutierrez-Sigut, E., B. Costello, C. Baus and M. Carreiras, "LSE-Sign: A lexical database for Spanish Sign Language", *Behavior Research Methods*, pp. 1–15, 2015.

48. Özsoy, A. S., E. Arık, A. Goksel, M. Kelepir and D. Nuhbalaoglu, "Documenting Turkish Sign Language (TID)", *Current Directions in Turkish Sign Language Research*, pp. 55–70, 2013.

49. Stokoe, W. C., "Sign Language Structure", *Annual Review of Anthroplogy*, pp. 365–390, 1980.

50. Hanke, T., "HamNoSys: Representing Sign Language Data in Language Resources and Language Processing Contexts", *LREC*, Vol. 4, 2004.

51. Sutton, V., *Sign Writing*, Deaf Action Committee (DAC), 2000.

52. Sloetjes, H. and P. Wittenburg, "Annotation by Category: ELAN and ISO DCR.", *LREC*, 2008.

53. Hanke, T., "iLex-A tool for Sign Language Lexicography and Corpus Analysis", *LREC*, 2002.

54. Süzgün, M. M., H. Ozdemir, N. C. Camgoz, A. A. Kindiroglu, D. Basaran, C. Togay and L. Akarun, "HospiSign: An Interactive Sign Language Platform for Hearing Impaired", *International Conference on Computer Graphics, Animation and Gaming Technologie (Eurasia Graphics)*, 2015.

55. Johnston, T. and L. De Beuzeville, "Auslan Corpus Annotation Guidelines", *Centre for Language Sciences, Department of Linguistics, Macquarie University, Sydney (Australia)*, 2011.

56. Sloetjes, H. and P. Wittenburg, "Annotation by Category: ELAN and ISO DCR.", *LREC*, 2008.

57. Kubuş, O., *An Analysis of Turkish Sign Language (TİD) Phonology and Morphology*, Master's Thesis, Middle East Technical University, 2008.

58. Liddell, S. K., *Grammar, Gesture, and Meaning in American Sign Language*, Cam-

bridge University Press, 2003.

59. Dalal, N. and B. Triggs, "Histograms of Oriented Gradients for Human Detection", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 886–893, 2005.

60. Jolliffe, I., *Principal component analysis*, Wiley Online Library, 2002.

61. Kreyszig, E., *Advanced Engineering Mathematics (Fourth ed.)*, Wiley, 1979.

62. Breiman, L., "Random Forests", *Machine Learning*, Vol. 45, No. 1, pp. 5–32, 2001.

63. Guyon, I., V. Athitsos, P. Jangyodsuk, B. Hamner and H. J. Escalante, "ChaLearn Gesture Challenge: Design and First Results", *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1–6, June 2012.

64. Pan, S. J. and Q. Yang, "A Survey on Transfer Learning", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 22, pp. 1345–1359, 2010.

65. Cook, D., K. D. Feuz and N. C. Krishnan, "Transfer Learning for Activity Recognition: A Survey", *Knowledge and Information Systems*, Vol. 36, No. 3, pp. 537–556, 2013.

66. Farhadi, A., D. Forsyth and R. White, "Transfer Learning in Sign Language", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2007.

67. Venkatesan, A., *A Study of Boosting based Transfer Learning for Activity and Gesture Recognition*, Master's Thesis, Arizona State University, 2011.

68. Camgoz, N. C., A. A. Kindiroglu, L. Akarun and O. Aran, "Domain Adaptation for Gesture Recognition using Hidden Markov Models", *IEEE 22nd Signal Processing and Communications Applications Conference (SIU)*, pp. 2050–2053, 2014.

69. Mitra, S. and T. Acharya, "Gesture Recognition: A Survey", *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, Vol. 37, No. 3, pp. 311–324, 2007.

70. Suarez, J. and R. R. Murphy, "Hand Gesture Recognition with Depth Images: A Review", *The 21st IEEE International Symposium on Robot and Human Interactive Communication*, pp. 411–417, 2012.

71. Murphy, K., "The Bayes Net Toolbox for MATLAB", *Computing Science and Statistics*, Vol. 33, pp. 1024–1034, 2001.

72. Keskin, C., A. T. Cemgil and L. Akarun, "DTW Based Clustering to Improve Hand Gesture Recognition", *Human Behavior Understanding*, Vol. 7065 of *Lecture Notes in Computer Science*, pp. 72–81, 2011.

73. de Campos, T. E., B. R. Babu and M. Varma, "Character Recognition in Natural Images", *International Conference on Computer Vision Theory and Applications*, 2009.

74. Alyuz, N., B. Gokberk and L. Akarun, "Regional Registration for Expression Resistant 3-D Face Recognition", *IEEE Transactions on Information Forensics and Security*, Vol. 5, pp. 425–440, 2010.

75. Valstar, M., J. Girard, T. Almaev, G. McKeown, M. Mehu, L. Yin, M. Pantic and J. Cohn, "FERA 2015-Second Facial Expression Recognition and Analysis Challenge", *11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, 2015.

76. Dibeklioglu, H., F. Alnajar, A. Ali Salah and T. Gevers, "Combining Facial Dynamics With Appearance for Age Estimation", *IEEE Transactions on Image Processing*, Vol. 24, pp. 1928–1943, 2015.

77. Arı, s., A. Uyar and L. Akarun, "Facial feature tracking and expression recogni-

tion for sign language", *23rd IEEE International Symposium on Computer and Information Sciences (ISCIS)*, pp. 1–6, 2008.

78. Creusot, C., N. Pears and J. Austin, "A Machine-Learning Approach to Keypoint Detection and Landmarking on 3D Meshes", *International Journal of Computer Vision*, Vol. 102, pp. 146–179, 2013.

79. Sukno, F. M., J. L. Waddington and P. F. Whelan, "3-D Facial Landmark Localization With Asymmetry Patterns and Shape Regression from Incomplete Local Features", *IEEE Transactions on Cybernetics*, 2014.

80. Alyuz, N., B. Gokberk, L. Spreeuwers, R. Veldhuis and L. Akarun, "Robust 3D Face Recognition in the Presence of Realistic Occlusions", *5th IAPR International Conference on Biometrics (ICB)*, pp. 111–118, 2012.

81. Fanelli, G., M. Dantone, J. Gall, A. Fossati and L. Van Gool, "Random Forests for Real Time 3D Face Analysis", *International Journal of Computer Vision*, Vol. 101, pp. 437–458, 2013.

82. Camgoz, N. C., B. Gokberk and L. Akarun, "Facial Landmark Localization in Depth Images Using Supervised Descent Method", *23th IEEE Signal Processing and Communications Applications Conference (SIU)*, pp. 1997–2000, 2015.

83. Lowe, D. G., "Distinctive Image Features from Scale-Invariant Keypoints", *International Journal of Computer Vision*, Vol. 60, pp. 91–110, 2004.

84. Xiong, X. and F. De la Torre, "Supervised Descent Method for Solving Nonlinear Least Squares Problems in Computer Vision", *arXiv preprint arXiv:1405.0601*, 2014.

85. De la Torre, F., W.-S. Chu, X. Xiong, F. Vicente, X. Ding and J. Cohn, "Intraface", *11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pp. 1–8, 2015.

86. Savran, A., N. Alyuz, H. Dibeklioglu, O. Celiktutan, B. Gokberk, B. Sankur and L. Akarun, "Bosphorus Database for 3D Face Analysis", *Biometrics and Identity Management*, Vol. 5372 of *Lecture Notes in Computer Science*, pp. 47–56, Springer Berlin Heidelberg, 2008.

87. Sukno, F. M., J. L. Waddington and P. F. Whelan, "Rotationally Invariant 3D Shape Contexts using Asymmetry Patterns", *GRAPP-International Confernece on Computer Graphics Theory and Applications*, 2013.

88. Hoerl, A. E. and R. W. Kennard, "Ridge Regression: Biased Estimation for Nonorthogonal Problems", *Technometrics*, Vol. 12, pp. 55–67, 1970.

89. İsmail Arı and P. Santemiz, "Türk İşaret Dili Kaynak Sitesi", `www.cmpe.boun.edu.tr/tid/`, accessed at January, 2016.