EVALUATION OF MOTION VELOCITY AS A FEATURE FOR SIGN LANGUAGE DETECTION

A Thesis

by

AISHWARYA

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Xiaoning Qian
Co- Chair of Committee,	Frank Shipman
Committee Members,	Ulisses Braga-Neto
	Weiping Shi
Head of Department,	Miroslav Begovic

May 2018

Major Subject: Computer Engineering

ABSTRACT

Popular video sharing websites contain a large collection of videos in various sign languages. These websites have the potential of being a significant source of knowledge sharing and communication for the members of the deaf and hard-of-hearing community. However, prior studies have shown that traditional keyword-based search does not do a good job of discovering these videos.

Dr. Frank Shipman and others have been working towards building a distributed digital library by indexing the sign language videos available online. This system employs an automatic detector, based on visual features extracted from the video, for filtering out non-sign language content. Features such as the amount and location of hand movements, symmetry of motion etc. have been experimented with for this purpose. Caio Monteiro and his team designed a classifier which uses face detection to identify the region-of-interest (ROI) in a frame, and foreground segmentation to estimate amount of hand motion within the region. It was later improved upon by Karappa et al. by dividing the ROI using polar coordinates and estimating motion in each division to form a composite feature set.

This thesis work examines another visual feature associated with the signing activity i.e. speed of hand movements. Speed based features performed better compared to the foreground-based features for a complex dataset of SL and non-SL videos. The F1 score showed a jump from 0.73 to 0.78. However, for a second dataset consisting of videos with single signers and static backgrounds, the classification scores dipped. More consistent performance improvements were observed when features from the two feature sets were used in conjunction. F1 score of 0.76 was observed for the complex dataset. For the second dataset, the F1 score changed from 0.85 to 0.86.

Another associated problem is identifying the sign language in a video. The impact of speed of motion on the problem of classifying American Sign Language versus British Sign Language was found to be minimal. We concluded that it is the location of motion which influences this problem more than either the speed or the amount of motion.

Non-speed related analyses of sign language detection were also explored. Since the American Sign Language alphabet is one-handed, it was expected that videos with left-handed signing might be falsely identified as British Sign Language, which has a two-handed alphabet. We briefly studied this issue with respect to our corpus of ASL and BSL videos and discovered that our classifier design does not suffer from this issue. Apart from this, we explored speeding up the classification process by computing symmetry of motion in the ROI on selected keyframes as a single feature for classification. The resulting feature extraction was significantly faster but the precision and recall values depreciated to 59% and 62% respectively for a F1 score of .61.

DEDICATION

To my

Mother, Father, Ankit and Kanak

for their immense love and support.

ACKNOWLEDGEMENTS

I express my sincere gratitude to Dr. Frank Shipman for the amazing experience this research work has been. His guidance, patience, and constant encouragement is what drove this thesis work all the way.

I would like to thank Dr. Qian for agreeing to be my committee chair despite my project being outside the department. His cooperation has been integral to the completion of this work. I would like to extend my gratitude to my committee members, Dr. Braga-Neta and Dr. Shi for their guidance and support throughout the course of this research.

Another huge thank you to my lab mate Caio Duarte Monteiro without whose guidance and suggestions this work would not have been possible. A special thank you to Gabriel Dzodom for introducing me to the research group.

Thanks also to Ms Katie Bryan and the rest of the staff of the ECEN Graduate Office for making this journey possible for me.

CONTRIBUTORS AND FUNDING SOURCES

This work was supervised by a dissertation committee consisting of Dr. Frank Shipman of the Department of Computer Science and Engineering and Dr. Xiaoning Qian, Dr. Ulysses Braga-Neto and Dr. Weiping Shi of the Department of Electrical and Computer Engineering.

All other work conducted for the thesis was completed by the student independently, without outside financial support.

Results of experiments by Caio Duarte Monteiro, a fellow graduate student of the Department of Computer Science and Engineering were used for comparison only.

NOMENCLATURE

SL	Sign Language
ASL	American Sign Language
BSL	British Sign Language
Non-SL	Not in Sign Language
PMP	Polar Motion Profile
PCA	Principal Component Analysis
SVM	Support Vector Machine
RBF	Radial Basis Function
GMM	Gaussian Mixture Model
ROI	Region of Interest

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Page

CHAPTER I

INTRODUCTION

Sign Language is the primary medium for sharing knowledge within the deaf and hard-of-hearing community. According to an article published in 2006 [1], a survey of various works of literature on sign language suggests that approximately 0.5 million people use the American Sign Language. With the increase in ease of recording and uploading videos on an online platform, video sharing websites now contain a large collection of videos in various sign languages, on a variety of topics.

Members of the sign language community use traditional text-based lookup techniques to locate videos online. The precision of text-based search depends on the relationship between the search string and metadata (title, description, comments etc.) associated with the video. It is found that metadata supplied by the uploader is often inaccurate or is completely missing. This makes the process of searching for these videos cumbersome for the community.

A study conducted by Shipman et. al [2] quantifies the problem of locating sign language videos. They appended terms like “ASL” and “sign language” to search strings on popular video sharing sites and analyzed the set of videos returned. Only 42% of the videos returned by the search were on the relevant topic and in sign language. About 46% of them did not contain sign language at all. This can be attributed to the fact that the terms like “ASL” and “sign language” can have different meanings based on context. They experimented with different combinations of the search terms using them separately as well as in conjunction. They discovered that attempts to improve the quality of the search strings by adding specific terms led to a decrease in the number of matched videos being returned and did not improve the absolute number of useful matches. The study thus

showed that the problem of locating sign language videos cannot be solved effectively by altering the search strings.

The experiments in [2] are restricted to the American Sign Language. The problem compounds when we consider other sign languages that a signer might use. Sign languages around the world have evolved separately and differ significantly from each other in grammar and vocabulary. For example, both American Sign Language and British Sign Language contain signs for the English alphabet. However, in BSL these signs employ both hands while in ASL they are one-handed. Search strings used to locate sign language videos online should be language specific to be useful to the community.

34% of the videos returned for experimental search strings in [2] were on topic but not in sign language. It is suggested that the search precision will improve by using a low-cost classifier to eliminate videos that do not contain sign language. Such a classifier can be trained by using (i) metadata associated with videos, (ii) features extracted from the actual video frames through image processing. This work deals with the design of classifiers to detect sign language in a video clip using video features.

In order to detect sign language in a video, we first need to understand the observable structures associated with the signing activity. Studies by linguists like Stokoe [3] describe each sign as a combination of hand-shapes, hand locations, and hand movements. Along with this, facial expressions, head-shoulder movements, and body posture are also a part of effective communication.

The feature set for training a classifier to detect sign language should take these fundamental structures into consideration. Monteiro et. al in [4][5] built a classifier that takes into

account the amount and location of hand motion in the video clip. Gebre et al.[6] used hand shapes and orientation in addition to location to build a classifier.

One of the initial approaches by Monteiro et al. resulted in the five-feature classifier of [7]. It used the total amount of motion, the spread of the motion, the amount of non-facial motion, speed and symmetry of hand motion to form a five-dimensional feature set. These features were later replaced by a Polar Motion Profile (PMP) [4][5][8] of hand movements. PMP examines the video frames using a polar coordinate system which incorporates location of motion into the feature set much more strongly as compared to [7]. These techniques are discussed in detail in the next chapter.

Speed of motion was completely dropped from the feature set after the five-feature classifier. This thesis work revisits this feature to generate speed-frames, which are the first derivative of the foreground frames of [8]. The algorithm for Polar Motion Profile generation can be used on these speed frames to preserve the effect of locality of motion. It was observed that classifiers based on location and amount of movement can misclassify videos that contain hand gestures accompanying speech as sign language videos. Speed frames are expected to have better performance statistics for such video clips and so can potentially improve overall precision and recall of the classification. Details of this technique and related experiments are outlined in Chapter 3. Chapter 4 provides the results of the experiments.

CHAPTER II

LITERATURE REVIEW

Problems related to sign language communication have been studied by researchers in the field of signal processing, human computer interaction, machine learning etc. for a long time. Video processing, gesture recognition, hand tracking etc. are problems closely related to this domain. This chapter begins with the discussion of some of the developments in this domain leading up to the motivations for this thesis work.

The most dominant research topic on sign languages is the transcription problem. Even after two decades of work, translation of sign languages remains one of the challenging problems in the field of gesture recognition [9]. First research works in this field appeared in late 90s when Starner et al. [10] used wearable computers to track a signer's hands and trained a Hidden Markov Model on a vocabulary of 40 signs. They achieved 92% accuracy in word recognition. More recently, Zafrulla et al. used Kinect depth-mapping camera to achieve sentence verification rates of 76.12% [11]. Work in this field has mainly used two approaches for feature extraction, direct hand tracking using data gloves and body trackers and video-based approaches. HMM appears to be employed frequently, followed by Neural Networks and it variants[9]. Performance is limited by visual complexity of sign languages, lack of a large annotated video corpus for training and the underdeveloped linguistics of SL [9].

Detecting sign language in a video is a simpler problem. Shipman et al. designed an automatic detector based on five visual features extracted from the a video clip [5]. These features were extracted through common video analysis techniques and achieved 82% precision and 90% recall on an especially created dataset that contained many likely false positive videos. Karappa et al. used face detection and background modelling to estimate the amount and location of

movement of a signer's hands to achieve classification with 81% precision and 94% recall [8] thus improving the F1 score compared to [5]. They introduced the idea of polar motion profiles associated with each video clip. Cherniavsky et al. used the principles of sign language detection to optimize the rate of frames transmitted per second in a cell phone conversation involving signing [12].

Another related domain is sign language identification. Monteiro et al used face detection and polar motion profile of videos to achieve an F1 score of 98% for BSL versus LSF classification, and an F1 score of 70% for ASL versus BSL classification [4]. Gebre et al. used skin detection to track hands and face of the signer; hand shapes, arrangements and the motion vectors of the hands were passed to a random forest algorithm for identifying the sign language [6]. They were able to classifying British SL and Greek SL with an F1 score of about 95%.

Automatic detection techniques like the ones described above were employed by Shipman et al [7] to build a distributed library of sign language videos that improves the retrieval of sign language videos through online searches. Figure 1 shows the architecture of the Sign Language Digital Library.

The system does not provide streaming or viewing services. Instead, its purpose is to build a corpus of videos that contain sign language. The crawler locates potential videos on video sharing sites using either extended search strings [2] or by following existing sign language videos to recognize relevant video portals, uploaders etc.. Identification of these videos is done through two channels: using automatic classifiers of [5][8] and by employing community feedback. Community involvement creates a feedback loop which further refines the corpus.

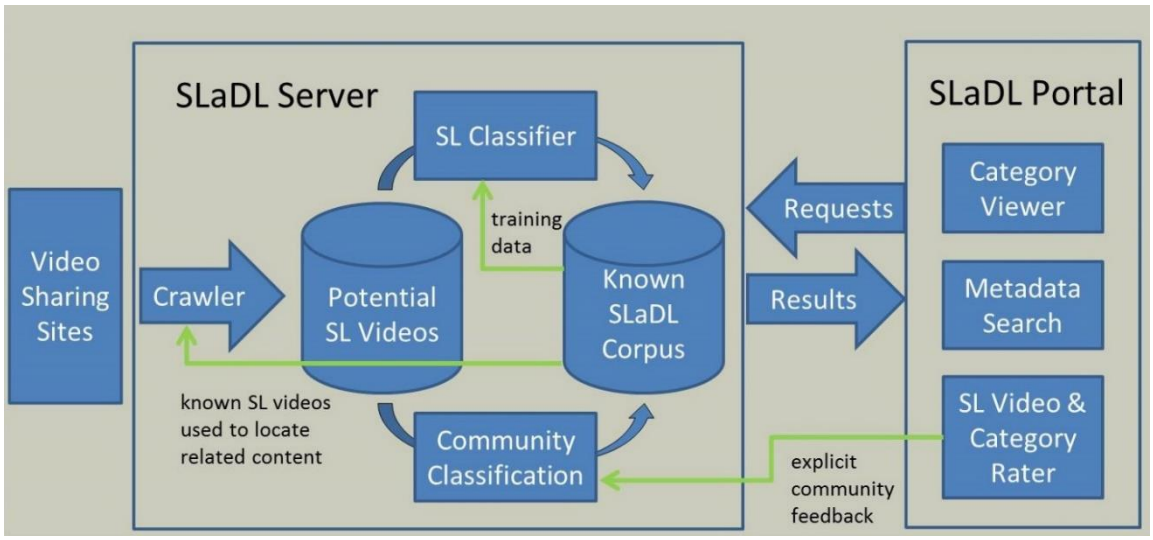


Figure 1: Architecture of Sign Language Digital Library (Adapted from [7])

Works by Montiero et al.[5], Karappa et al.[8], Duggina et al. [13] focus on the SL classifier module in the architecture of Figure 1. They attempt to build an optimized classifier for real-time application. Current thesis work too focuses on the same module along with a preliminary study of few other related issues.

CHAPTER III

APPROACH

We begin this chapter with a detailed description of the PMP based classifier of [8]. This classifier has a precision rate of 81% and recall rate of 94%. In his thesis work [14], Duggina suggested optimizations to the design of [8] to reduce the time required for feature extraction. The PMP classifier along with recommendations of [14] forms the base classifier for the current work. Later sections describe the idea behind *speed frames* and present a set of experiments assessing where the speed frames improve classification performance.

3.1 CLASSIFIER BASED ON POLAR MOTION PROFILES

Videos present in the dataset are of varying length. A one-minute clip from the middle of the video is used for the purpose of feature extraction. At the frame rate of 30 frames per second, a one-minute clip corresponds to 1800 frames.

Each of these frames is processed by a face detection algorithm and a background modeling algorithm in parallel. The face detection helps to delineate a region-of-interest within the frame which covers the face and the range of motion for the hands of a potential signer. A video that does not contain a face does not warrant an investigation for sign language content for obvious reasons. Simultaneously, a background model is generated for the entire clip using adaptive Gaussian Mixture Model and used to generate a foreground image of each frame.

A polar coordinate system centered at the face is imposed on the region-of-interest. The amount of motion in regions, defined by their polar coordinates, is estimated using the foreground image. The values are averaged over multiple ROIs and multiple frames to generate the Polar Motion Profile of a video. This PMP feature set is sent to a linear, binary classifier for detecting sign language videos. Figure 2 shows the architecture of this classifier.

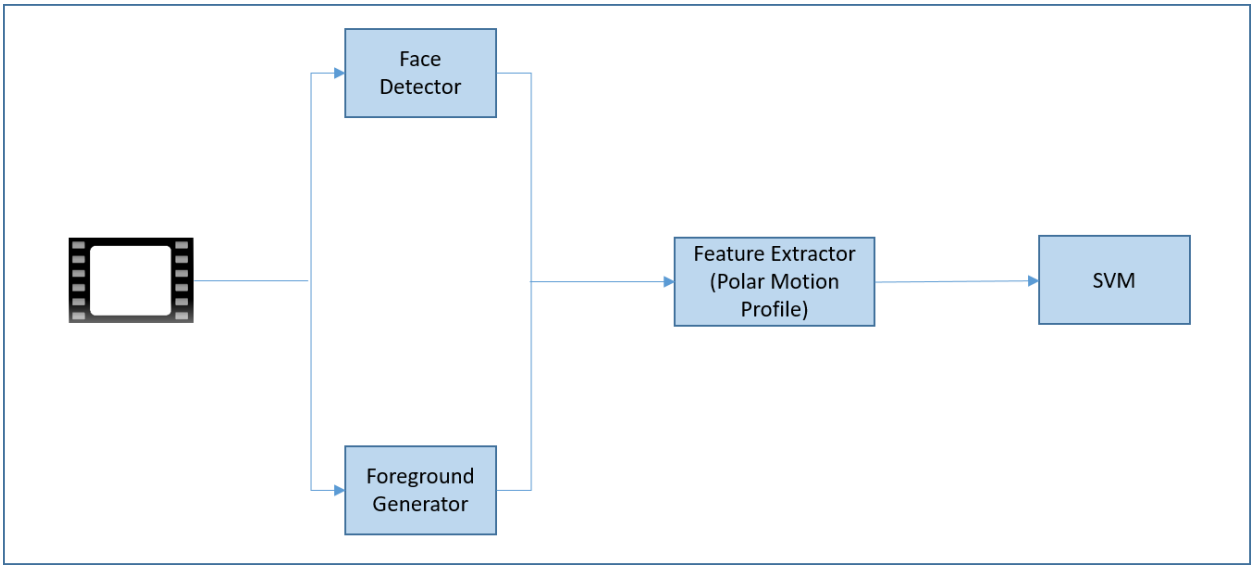


Figure 2: Architecture of the PMP Classifier

3.1.1 Face Detection

Our video-processing module is written in C++ with OpenCV integration. For face detection, we use a pre-trained, Haar-cascade classifier for frontal face available in OpenCV license. It is based on the features proposed in the Viola-Jones algorithm [15] for face detection. The window size for feature calculations in 20x20. Viola-Jones uses the AdaBoost algorithm for selecting best features for classification. The training step is time-consuming but a cascade of boosted classifiers makes the detection step relatively fast.

Previous work by Monteiro et al. used 5 different pre-trained face detectors available in OpenCV. The bounding boxes returned by each of the face detectors were compared for overlap and a majority vote was taken. Duggina studied the impact of using a single detector on the overall performance of the sign language classifier [13] and concluded that using the tree-based gentle AdaBoost frontal face detector (Haar-cascade Frontal Face Alt2) alone resulted in significant

speed-up of the pipeline with a small trade-off in accuracy. All subsequent work has used a single detector.

3.1.2 Background Subtraction

We use Gaussian Mixture-based background/foreground segmentation proposed by Zivkovic [16]. A statistical model of the scene is created by modeling the variation in values at each pixel separately using a mixture of Gaussians. The probability density function at a pixel x is given as

$$\hat{p}(\vec{x}|X_T, BG) = \sum_{m=1}^{M} \hat{\pi}_m N(\vec{x}; \widehat{\vec{\mu_m}}, \widehat{\sigma_m^2} I)$$

where $X^T = \{x^t, \dots, x^{t-T}\}$ is a set of pixel values over a time period T, $\hat{\pi}_m$, $\widehat{\vec{\mu_m}}$and $\widehat{\sigma_m^2}$ are the mixing weights, estimated means and variances respectively of the m Gaussian components. These variables are adjusted for each new data sample $\vec{x}^t$. Zivkovic suggests that the number of Gaussian components should be variable to increase the adaptability of the model to changes is illumination etc. If a new data sample $\vec{x}^t$ is beyond several standard deviations of all the components, a separate Gaussian component is allowed to be generated.

The components are included in the background model if the total sum of their weights is above a certain threshold. This threshold is derived from the ratio of foreground to background pixels required for a specific segmentation task.

We use the BackgroundSubtractorMOG2 class provided by OpenCV which is an implementation of the design in [16]. It allows several tunable parameters: the background ratio, the initial variance of the components, the minimum and maximum variance for any component, a complexity reduction parameter which defines the number of samples needed to accept that the

backgroundRatio = 0.9	bShadowDetection = true	fVarInit = 50
fVarMin = 10	fVarMax = 100	fCT = 0.05

Table 1: Values Assigned to the Parameters of the BackgroundSubtractorMOG2 reference class provided by OpenCV 3.1.0

component exists and a shadow detection threshold which decides if the foreground object detected is a shadow or not. Table 1 presents the values assigned to these parameters based on OpenCV documentation and empirical studies [8]. The segmentation process is followed by morphological erosion and dilation to remove small discontinuous foreground objects.

3.1.3 Extraction of Polar Motion Profiles

Face detection returns the coordinates of a bounding box around each face detected in a frame. This box is expanded to 5 times its height and width to include the regions of possible hand movement. The resulting box is our region-of-interest for feature extraction.

Hand location is an important characteristic that ascribes meaning to signs [3]. Polar Motion Profile encodes location information into the feature set by imposing a polar coordinate system on the ROI with the face as the origin. The angular coordinate divides the ROI into 360 sectors, while the radial coordinate is used to divide it into 100 concentric regions around the face. The proportion of foreground to background pixels in each such region is estimated and concatenated to form the PMP feature. Figure 3 [8] shows the various steps involved in the generation of the polar motion profile for a video.

3.1.4 Training and Classification

Polar motion profile is calculated separately for each ROI in each frame. The overall PMP of a video is the average PMP across ROIs and frames.

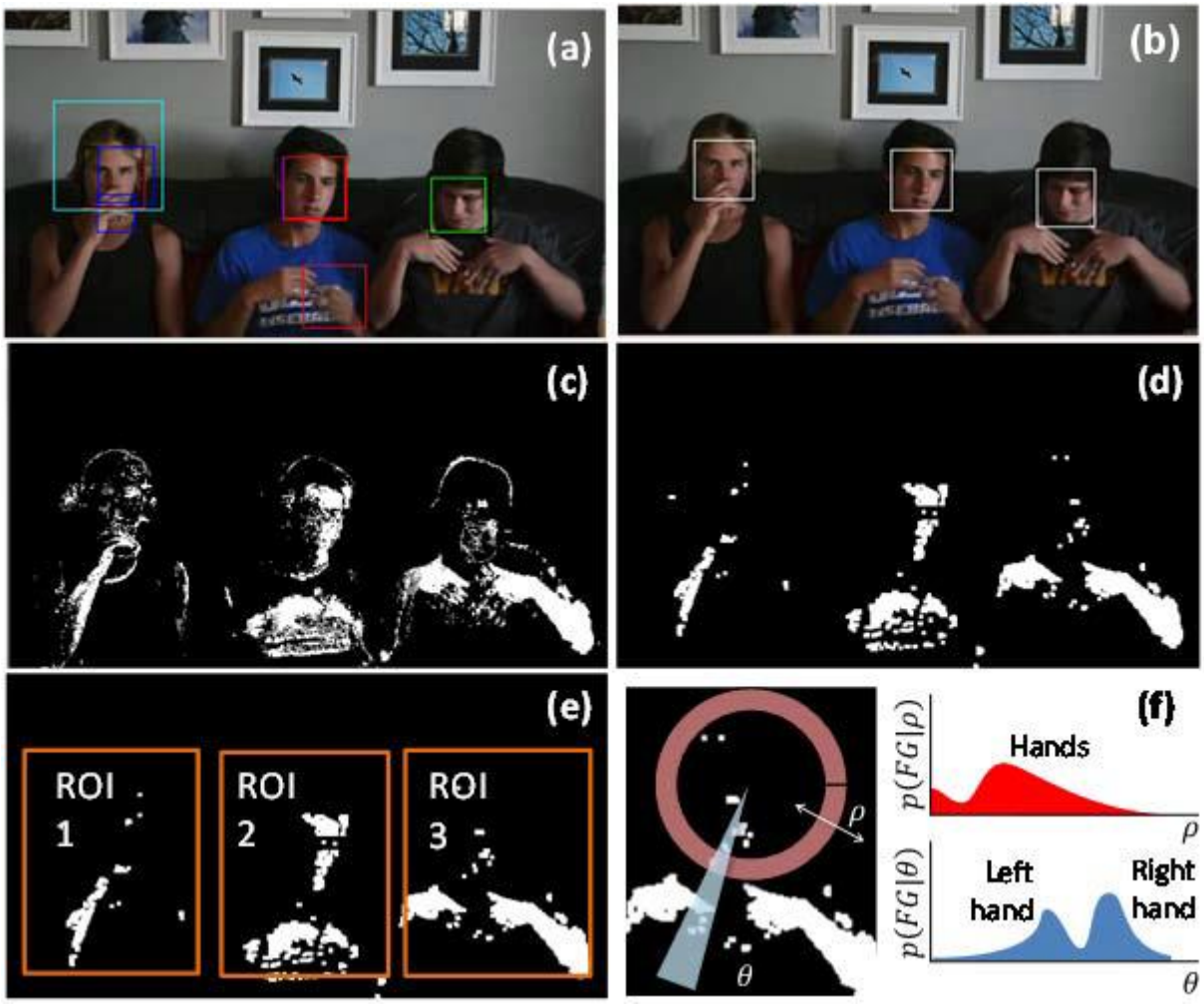


Figure 3: Stages of PMP computation. (a) Faces detected by Haar cascades. (b) Computing final face locations by taking a majority vote of (a). (c) Foreground (FG) pixels returned by the background model. (d) Refined FG after de-noising. (e) ROIs for each face detected in (b). (f) Computation of PMPs in a video frame. (Adapted from [8])

The hand-generated datasets used to evaluate alternative approaches to sign language classification contain approximately 200 videos. PMP is a 460-dimensional feature and the dataset is extremely small in comparison. Prior to using the PMP for training, its dimensionality is reduced by applying Principle Component Analysis. Current techniques reduce the number of dimensions to six.

The resulting set of six features is used to train a Support Vector Machine with Gaussian Radial Basis Function kernel. The dataset is repeatedly divided into testing and training sets using random sampling. In order to better analyze the performance of the classifiers and whether the techniques would likely benefit from larger training sets, experiments are conducted for 4 different training set sizes: 15, 30, 45 and 60.

3.2 SPEED ESTIMATION

The speed of an object is a measure of the change in its position per unit time. Consider the foreground objects identified using the GMM background model. If we take the absolute difference between the foreground images of two consecutive frames, we can estimate the amount of displacement in the position of foreground objects per frame which is proportional to the speed of their motion.

In this work, we define *speed frames* as the difference frame generated by comparing the foreground image of two consecutive frames. The pixels that were not a part of foreground objects in the previous image but are so in the current image, or vice versa, are assigned the value 'white' in the speed frames. The pixels that maintain their value between the two are assigned 'black'. Speed frames can be considered to be the first derivative of the foreground images. Foreground images represent the amount of motion whereas speed frames represent the speed of motion. Figure 4 shows consecutive frames from the original video, their respective foreground images and the speed frame based on them.

3.3 PROPOSED EXPERIMENTS

3.3.1 Speed as a Feature

This work explores the hypothesis that speed of hand/arm motion plays a role in detecting sign language in a video. It was similar to the study done in [7] for each one of the five features.

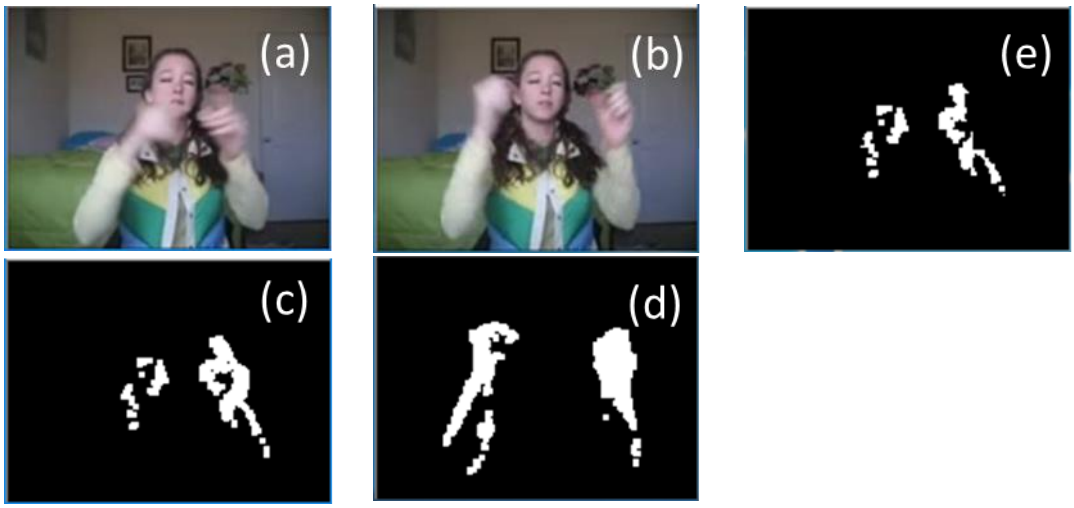


Figure 4: Generation of speed frames. (a) Previous frame (b) Current frame (c) Foreground image of (a). (d) Foreground image of the (b) (d) The speed frame generated using (c) and (d)

A single mean *speed* value is calculated for each video clip. Speed of motion in a frame is estimated as the total number of white pixels within the region of interest in a speed frame. The value is averaged over multiple ROIs and 1800 frames and passed to the SVM classifier.

3.3.2 Speed-Based PMP

The location of hand movements is important in defining the meaning of signs in any sign language and so it must be integrated into the speed-related feature set as well. This experiment applies the algorithm for extracting polar motion profile on the speed frames. Figure 5 shows the architecture of this classification system. Speed frames are generated using the foreground images and combined with the face detection results to generate the speed-based PMP.

The time required for feature extraction is expected to remain the same because generation of the speed frames does not require foreground images to be stored on the disk. We need to store the foreground image of the preceding frame only. This can be saved in a single image variable which gets reused as the algorithm proceeds.

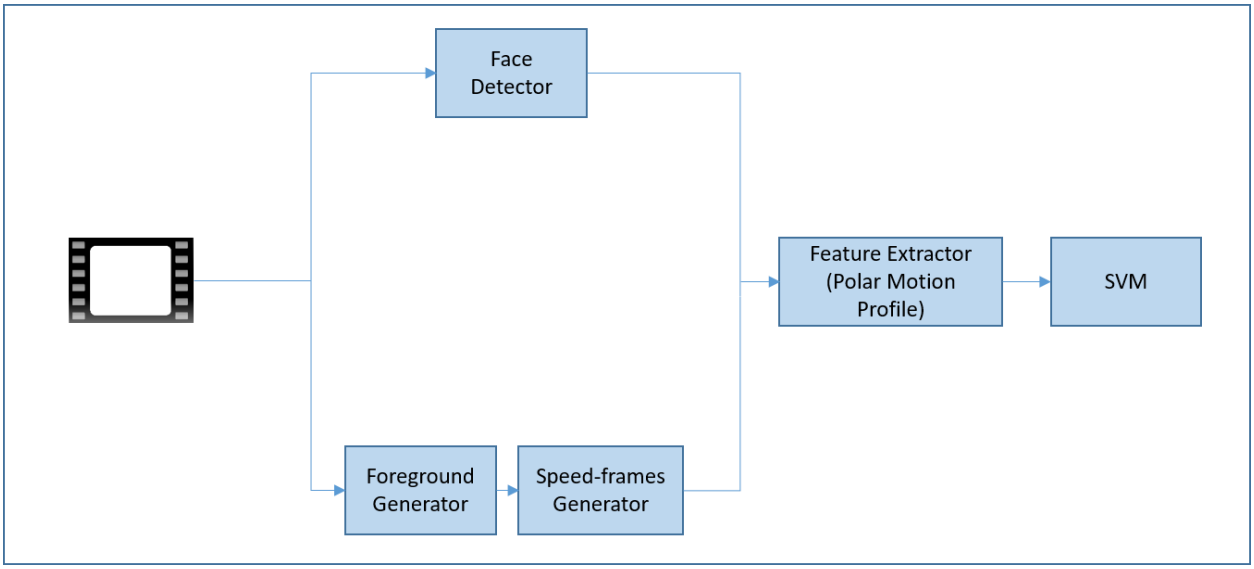


Figure 5: Architecture of the Speed Frames - Based Classifier

3.3.3 Combined Feature Set

The classifier of Figure 5 takes into account the speed and location of hand movements for sign language detection. Sign language videos available online encompass a wide range of contents. For certain types of videos, such as videos of signing music lyrics that depend on the composition of the music, there is the strong potential for misclassification with the speed based classifier. A better alternative is to combine the features generated using the amount of motion and speed of motion.

One of the ways to combine these features is to apply PMP extraction on the foreground images and speed frames in parallel and apply dimensionality reduction on the individual feature sets. The resulting features can be concatenated and sent to the classifier. Figure 6 shows the architecture of this type of classifier.

Two such classifiers will be evaluated for our experiments. *Combined Classifier 1* takes the 6 best features from both of the sets and sends a feature set of size 12 to the SVM classifier.

Since the number of videos available in the corpus remains the same, it is possible that increasing the dimensionality of the feature set can result in overfitting by the classifier. Hence, *Combined Classifier 2* takes 3 features from each of the sets and thus preserves the dimensionality of the feature set at 6.

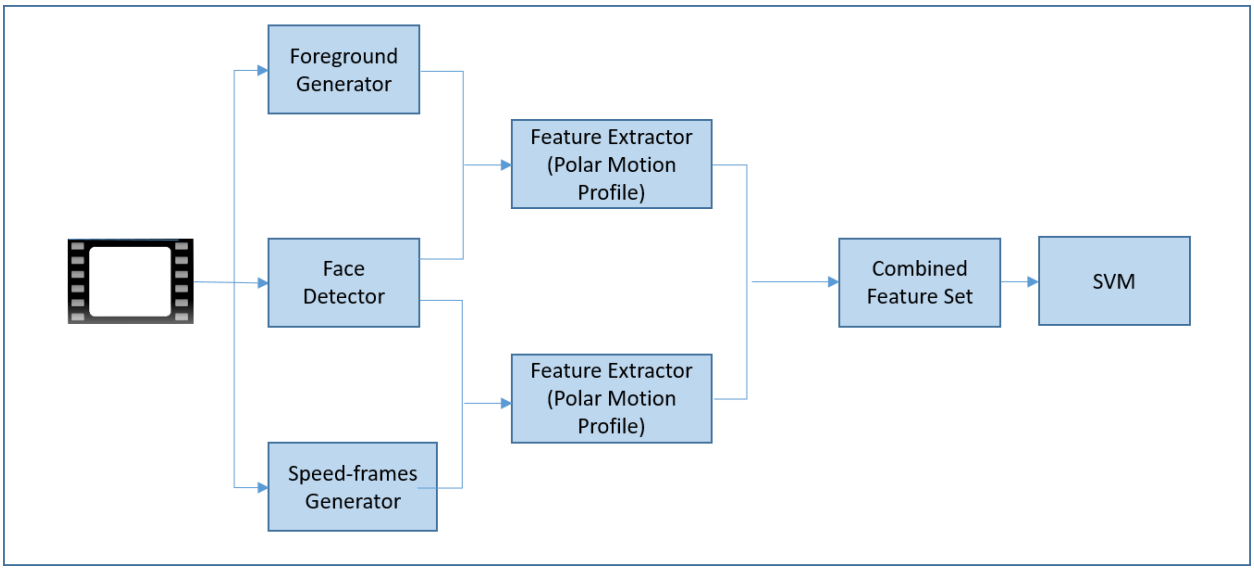


Figure 6: Architecture of the Combined Classifier

Background modeling, speed frames generation and face detection proceed in parallel. So does the PMP generation with two separate inputs. Hence the time required for feature generation should remain similar for either the foreground or speed PMP generation alone, given that the system has sufficient memory resources and multiple cores.

3.3.4 ASL vs BSL

The intuition for including speed in the detection of sign language is that speed may help differentiate between hand and arm motion during signing and other uses for gestures. When considering the problem of differentiating between sign languages, language characteristics such

as the locality of common pairs of signs could result in differences in the distribution of observed hand/arm speeds across a wide range of topics.

This subsection discusses the experiments conducted for automatic identification of sign languages. Sign languages use fingerspelling to represent letters of an alphabet. Fingerspelling can be one-handed such as in American, French or Irish sign language or two-handed such as in the British Sign Language. Figure 7 taken from [4] is a good example of this contrast.

Due to this, the region of hand motion is different in one-handed and two-handed languages. Monteiro et al. have used the foreground PMP features for classifying sign language videos into ASL and BSL videos [4]. Their classifier will serve as the base classifier for comparison. We test the performance of the classifiers designed in sections 3.3.2 and 3.3.3 on ASL vs BSL classification and present the findings.

Figure 7: Letter 'E' signed in ASL(left) and BSL(right)

CHAPTER IV

EVALUATION

4.1 VIDEO CORPUS

The videos used in the experiments throughout this work have been taken from corpus created by Monteiro et al. [7] and Karappa et al. [8] for their studies. There are two datasets used to evaluate performance for discriminating between sign language vs. non-sign language videos. These are called SL Non-SL Dataset A and SL Non-SL Dataset B. A third dataset is used to assess performance on distinguishing between ASL and BSL videos.

4.1.1 SL Non-SL Dataset A

The SL Non-SL Dataset A was used in [7] was developed to determine the ability of a classifier to distinguish between an individual signing and an individual using other forms of gestures. It consists of 105 American Sign Language videos, each of which has a single signer and minimal background activity, and 105 videos which do not contain signing but were selected to be likely false positives due to their inclusion of an individual facing the camera and gesturing (e.g. a gesturing reporter.) All videos were selected from video sharing sites. This dataset was used for the evaluation of the earliest five-feature classifier by Monteiro et al. [5].

4.1.2 SL Non-SL Dataset B

Karappa et. al created a second dataset with 112 American sign language videos and 120 likely false positives that are more varied . We call this the SL Non-SL Dataset B. These videos were obtained by querying the term ‘American Sign Language’ on YouTube. The videos returned by the search were label through manual inspection. These videos have complex background as compared to dataset A and usually contain more than one person in the frame, who may or may not be signing. As a result, multiple regions-of-interest are detected within a single frame and the

PMP is a result of averaging the foreground activity over them. As a result, the precision and recall characteristics of classifiers often deteriorate on dataset B versus dataset A. However, this dataset more closely represents the type of videos that the system will be required to classify in real scenarios. The results with this dataset are more likely to better predict ultimate success in differentiating SL and Non-SL videos found on video sharing sites.

4.1.3 ASL vs BSL Dataset

The ASL vs BSL Dataset used in the sign language identification experiments consists of 90 American Sign Language videos, which are a mixture of videos from dataset A and B, and 95 British Sign Language videos which were downloaded from online video sharing sites [4]. Both the categories contain a mix of videos with static and dynamic backgrounds, single and multiple signers, etc. and closely resemble real world scenarios.

4.2 TECHNIQUE

Each of the classifiers discussed in following subsections was evaluated with training sets of four sizes: 15, 30, 45 and 60. Results presented are an average over 1000 iterations. The members of the training sets were randomly selected and the remaining videos were used for testing for each iteration. The precision, recall and F1 scores of the classifiers are used as metrics for comparison. For each classifier, we plot the values of these metrics against the size of the training set. In the resulting plots, each classifier is represented by a single curve.

As part of evaluation, the system generates a list of test videos that are incorrectly classified from both of the classes. This helps in correlation analysis of the performance on the basis of the visual features present in the video and in discussing the subclasses of videos and how their features interact with the detection and identification of sign language content.

4.3 RESULTS

4.3.1 Results of Initial Evaluations

Table 2 contains the results of using a single mean speed value as the feature for each video. This is the average number of white pixels in a speed frame ROI and thus does not include any location information beyond that the motion happened in the ROI. The results are satisfactory considering the simplicity of the feature. They indicate that incorporating speed of hand/arm motion into the feature set has the potential to improve classifier performance

Datasets	Precision	Recall	F1
Dataset A	0.745	0.6647	0.6992
Dataset B	0.5614	0.8537	0.6768

Table 2: Results of Initial Evaluation

4.3.2 Results of Speed-Frames Based Classifier

Table 3 contains the results of a PMP classifier which uses speed frames for feature extraction. Performance of the classifier on dataset B show significant improvement in both precision and recall compared to the base classifier. Precision increases from 81.2% to 84.9% while recall increases from 68.8% to 73.87%.

However, for dataset A, the performance measures do not exhibit large variations. The false set associated with dataset A contains videos with weathermen, mimes, elaborately gesturing news reporters etc., and are very likely to result in false positives. Speed of motion alone might not help to improve classification performance due to the dynamic nature of visual activity of the dataset.

4.3.3 Results of the Combined Classifiers

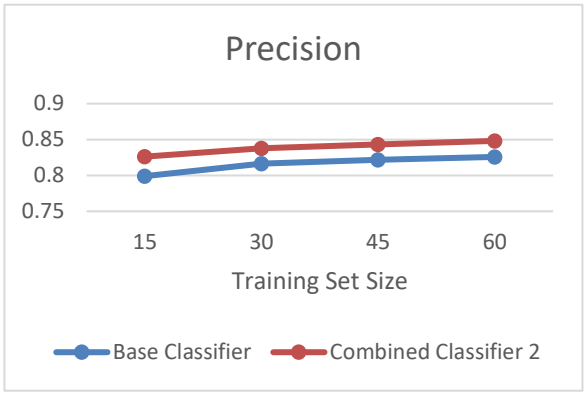
The combined classifiers incorporate both the amount and speed of motion into the feature set. The results for this type of classifier are expected to be better than the base classifier and speed-only classifier.

Dataset A												
Training Set Size	Base Classifier			Speed-Only			Combined Classifier 1			Combined Classifier 2		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
15	0.799	0.8274	0.8087	0.8057	0.8557	0.8262	0.7409	0.8853	0.8023	0.8262	0.8356	0.827
30	0.8164	0.861	0.8351	0.8208	0.8724	0.8431	0.7598	0.9059	0.824	0.838	0.8676	0.8499
45	0.8219	0.8804	0.8467	0.8183	0.8794	0.8442	0.7704	0.9119	0.8332	0.8431	0.8819	0.8594
60	0.8258	0.8889	0.8522	0.8168	0.88	0.8427	0.7721	0.9129	0.8339	0.8481	0.886	0.8629
Dataset B												
Training Set Size	Base Classifier			Speed-Only			Combined Classifier 1			Combined Classifier 2		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
15	0.7498	0.6697	0.6987	0.7755	0.7196	0.7379	0.7488	0.6561	0.679	0.7694	0.7032	0.7263
30	0.8059	0.6782	0.7317	0.8378	0.7225	0.7708	0.8336	0.6503	0.7218	0.8105	0.7045	0.7489
45	0.8158	0.6841	0.7377	0.8509	0.7321	0.7825	0.8594	0.6635	0.7442	0.8259	0.708	0.7577
60	0.812	0.6887	0.7373	0.8493	0.7387	0.7843	0.8632	0.6781	0.7533	0.8237	0.7156	0.7599

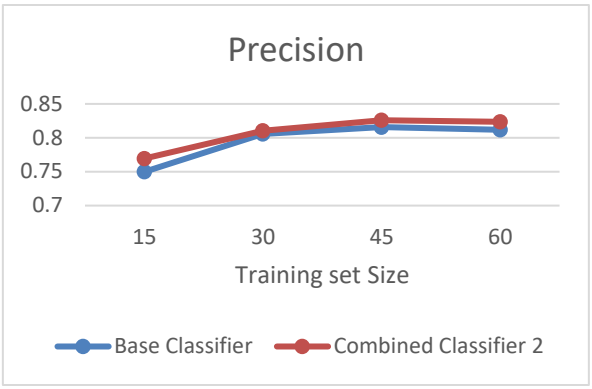
Table 3: Performance of the base classifier and 3 proposed classifiers on SL vs Non-SL classification for Dataset A and Dataset B for 4 training set sizes.

The first classifier we designed is referred to as *combined classifier 1*. The PMP extraction step is followed by principal component analysis. The top six features are picked up from the feature sets for the base and the speed-only classifier to form a 12-dimensional feature set.

As seen in Table 3, the performance of this classifier is weaker than the other three classifiers for dataset A. For dataset B, it is weak for small training set sizes and catches up with the base as the training size increases. One of the important points about the *combined classifier 1* is that the dimensionality of the feature set provided to the SVM increases from 6 to 12. More dimensions require more data points for training to avoid overfitting. We have evaluated all the classifiers on the same datasets. Thus, it is likely that the drop in performance metrics was caused due to overfitting.

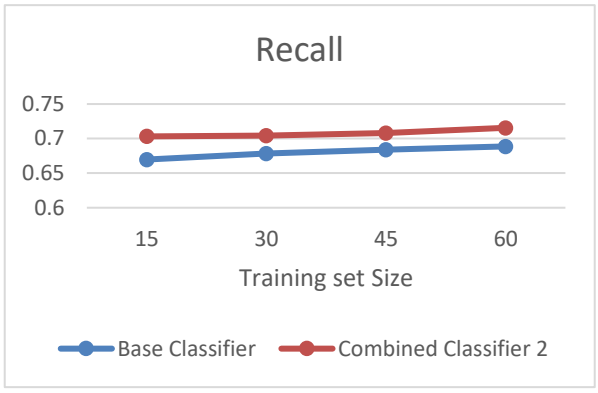


6 (a)

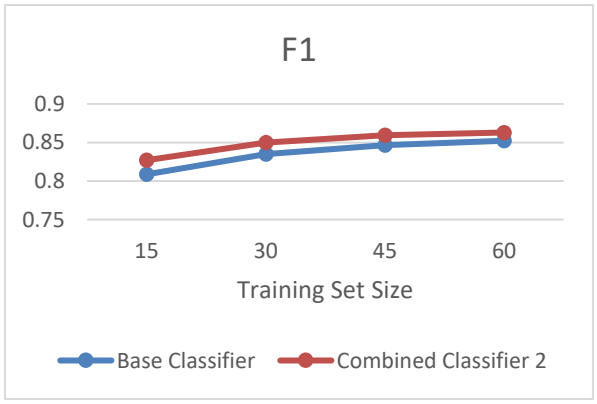


6 (d)

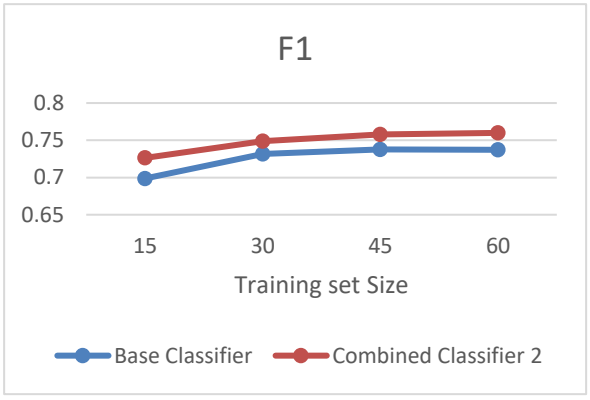
6 (b)



6 (e)



6 (c)



6 (e)

Figure 8: Comparison between the Base Classifier and Combined Classifier 2 (6 Features). (a) (b) (c) Precision, Recall and F1 Results for Dataset A, (d) (e) (f) Precision, Recall and F1 Results for Dataset B.

ASL versus BSL												
Training Set Size	Base Classifier			Speed-Only			Combined Classifier 1			Combined Classifier 2		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
15	0.6666	0.6369	0.6411	0.6685	0.6378	0.6424	0.6567	0.6015	0.6074	0.6109	0.5907	0.5908
30	0.7053	0.6861	0.6891	0.7037	0.6925	0.6916	0.6952	0.6553	0.6596	0.612	0.6139	0.6069
45	0.7132	0.7067	0.7051	0.7108	0.7252	0.7142	0.6941	0.7098	0.6917	0.6129	0.6255	0.6143
60	0.7065	0.7298	0.7135	0.7082	0.7393	0.72	0.692	0.7582	0.7172	0.6045	0.63	0.6118

Table 4: Performance of the three proposed classifiers on ASL vs BSL classification for four training set sizes

Combined classifier 2 aims to address this problem by only taking three features from each of the two feature sets after PCA. Thus, the dimensionality of the combined feature set remains same as the base classifier.

The results show improvement for both the datasets. Figure 8 shows the comparison between the performance metrics of the base classifier and *combined classifier 2*.

4.3.4 Effect on ASL vs BSL Classification

The three classifiers proposed in the previous sections were used to classify videos into those including American Sign Language and those including British Sign Language. Since ASL has a one-handed alphabet while BSL's alphabet is two-handed, the region of significant hand movements is different for them. All the classifiers evaluated in this work employ PMP features which utilize location information effectively to achieve the classification. The aim of this experiment was to evaluate how using speed frames affects the ASL vs BSL problem.

The results of the experiment are tabulated in Table 4. For the speed-frames based classifier of section 3.3.2 and *combined classifier 1* the precision and recall values exhibit very little deviation from those of the base classifier, as shown in Figure 9. This can be attributed to the fact that location of hand motion is more useful for this type of classification than either the amount of motion or the speed of motion. We can infer that using speed frames in ASL versus BSL classification has no significant gain over using the foreground images.

Classification using *combined classifier 2* shows a more interesting trend. The performance metrics show a drastic fall as compared to all the other classifiers. While generating the polar motion profile of a video, the region-of-interest is divided into 360 angular sectors and 100 concentric regions at different radial distances from the center. It results in feature set with 460 columns. This combined classifier 2 uses PCA to pick 3 features from the PMP of foreground

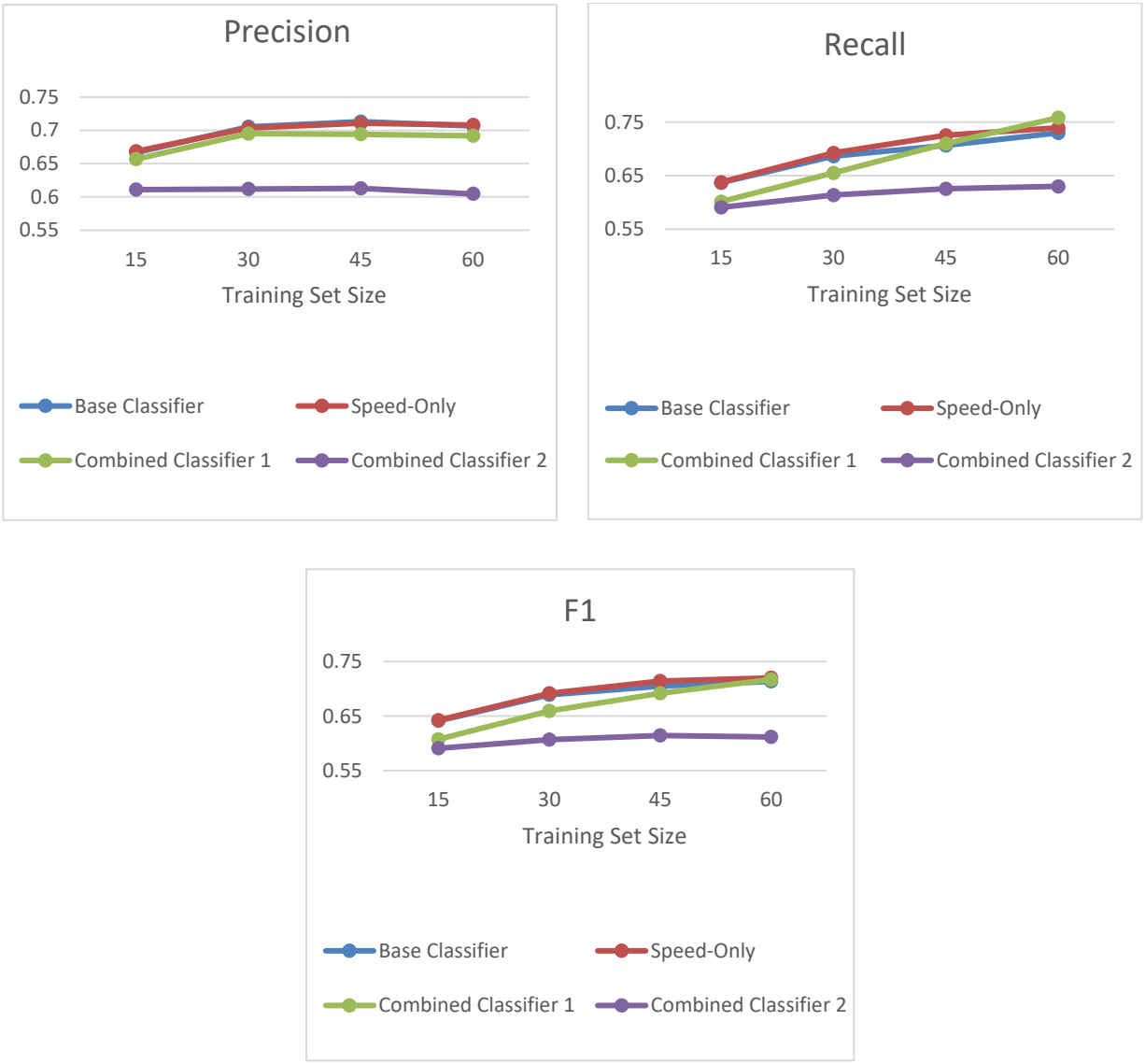


Figure 9: Comparison of the Precision, recall and F1 Scores of the 4 Classifiers for ASL vs BSL Classification

images and another 3 from the PMP of speed-frames. Based on the results, we can conclude that the 3 dimensions do not encode sufficient location information to provide good classification performance.

4.4 ANALYSIS OF MISCLASSIFIED VIDEOS

To gain a clearer perspective of the results, we compared the performance of the speed-frames based classifier versus the performance of the base classifier on individual videos in the datasets. We calculated a misclassification rate for each video as the ratio of the number of times it was misclassified to the total number of times it was a part of the test set. It is observed that while the misclassification rate for most of the videos does not show much change for the two approaches, there are a few videos for which it changes drastically. We group the videos that show more than 50% change in misclassification rate into 2 categories: those with positive and negative change. We calculate the mean misclassification rates for each of these groups and compare them for all the 4 classifiers. The results are tabulated in Table 5.

Dataset A				
Number of videos	Base Classifier	Speed-Only	Combined Classifier 1	Combined Classifier 2
7	9.32%	81.88%	68.25%	40.26%
5	81.55%	23.08%	72.25%	35.71%
others	14.77%	15.21%	15.87%	13.26%
Dataset B				
Number of videos	Base Classifier	Speed-Only	Combined Classifier 1	Combined Classifier 2
4	7.31%	81.61%	32.43%	36.62%
10	85.62%	12.90%	53.61%	55.07%
others	20.72%	18.56%	19.33%	19.63%

Table 5: Comparison of mean misclassification rates observed for the 3 groups of videos when applying 4 different classifiers for ASL vs BSL classification. The groups are divided by comparing absolute misclassification rates of individual videos for base vs speed-only classifier.

In dataset A, there are 5 videos whose classification shows huge improvement when speed-frames are used instead of foreground image. For another 7 videos classification is much better with the latter approach. Similarly in dataset B, using speed-frames improves the classification for 10 videos and deteriorates for 4 others. If these videos are removed from the datasets, the average classification rate does not show large fluctuations.

These special videos are responsible for the overall performance metrics observed in Table 3. Since dataset A contain more number of videos in the first set the overall performance of speed-frames based classifier on dataset A is shows deterioration. Better performance of speed-frames on dataset B can be similarly explained.

We manually observed these anomalous video in an attempt to spot similarities between them. In some videos, a person signs to the lyrics of a song while the music plays in the audio. The classification performance of *speed-frames* was found to deteriorate for these videos. On the other hand, in a few videos, the signing activity is frequently punctuated. For example, one person is giving signing lessons to another in a video. Each sign is followed by a few seconds of inactivity while the learner masters the sign. *Speed-frames* performed better for such videos.

This leads us to the conclusion that there are certain types of sign language videos for which speed-frames can prove to be a useful feature. Meta data features might help to identify these videos prior to applying video feature extraction.

As we observe in Table 5, the misclassification rates for the combined classifiers is less extreme for all the groups. With proper empirical tuning combined classifiers can provide better performance than the first two classifiers.

4.5 ADDITIONAL DIRECTIONS

While the speed frames are the focus of this work, certain other dimensions of the classification problem were also studied as part of the research process. This subsection briefly covers some of those studies explaining the motivation behind them and their results.

4.4.1 Using Symmetry of Motion with Key-Frames Based Approach

There have been attempts to optimize the feature extraction pipeline to reduce the time taken for classifying a new video. Face detection, background subtraction and PMP generation form the major bottlenecks in the pipeline. In [13] attempts have been made to move towards real time video classification by establishing a speed-accuracy tradeoff. One of the suggested techniques is the *key-frame based approach* [13]. Instead of using 1800 continuous frames for feature extraction, we work with a limited number of inter-leaved frames. Face detection is called only for these frames. Instead of GMM based background modelling, the previous keyframe selected is considered to be the background model for the current frame. The pixels whose values change between the two frames are considered foreground pixels. PMP features extracted from these *keyframes* are used for classification. With 10 keyframes, Shipman et al. achieved 69% precision and 74% recall for SL vs non-SL classification.

Another prior result comes from when the individual features of the *five-feature* classifier were compared. Shipman et al. discovered that the symmetry of motion outperformed the other four features combined [7].

In this work we attempt to combine the findings of [7] and [13] to design a fast classifier based on the symmetry feature. 10 keyframes are extracted from the video. The region-of-interest identified through face detection is divided along the vertical center. Regions on the left and right

side of the division are compared to count the total number of white pixels symmetric to each other in the foreground image. The number is averaged over 10 frames and passed to the SVM classifier.

The feature extraction process proved to be extremely fast but the classifier incurred considerable loss of accuracy. A precision rate of 60% and recall rate of 62% were achieved with 20 keyframes on dataset A. Results are shown in Table 6. Given that the keyframe results in [13] were significantly better than these results, it appears symmetry of motion is valuable primarily when considering continuous motion and not across gaps in the video.

Dataset A						
Training Set Size	Base Classifier			20 keyframes		
	Precision	Recall	F1	Precision	Recall	F1
15	0.799	0.8274	0.8087	0.6045	0.5076	0.5239
30	0.8164	0.861	0.8351	0.602	0.5567	0.5611
45	0.8219	0.8804	0.8467	0.6004	0.5832	0.5771
60	0.8258	0.8889	0.8522	0.5975	0.622	0.6022

Table 6: Results of Symmetry of Motion Feature with Key-Frames Based Approach for Dataset A

4.4.2 Left Handed Signers Problem

The excellent performance of the PMP feature has been attributed to the detailed information it encodes about the location of hand motion. Unfortunately, since the American Sign Language is one-handed, severe dependence on location of motion can lead to videos with left handed signers being constantly misclassified. Figure 10 shows the difference between left-handed and right-handed signing.

As a part of this thesis work, attempts were made to understand how this problem affects the ASL vs BSL classification results. The 90 ASL videos in the ASL vs BSL dataset were

manually inspected to determine if the signer was left handed or right handed. Out of the 90, 18 videos contained left handed signing. The decision was ambivalent for another 16 videos.

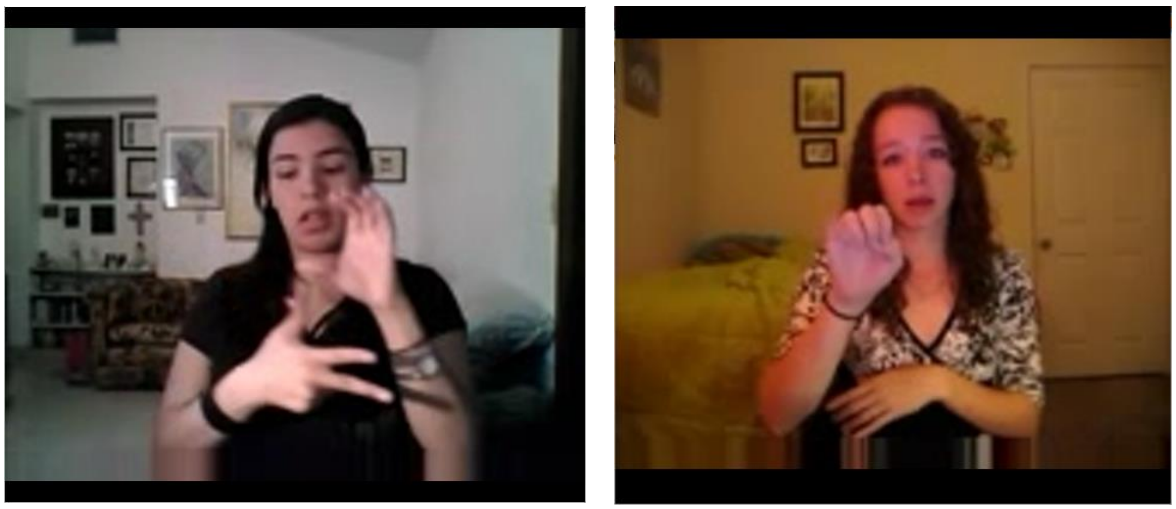


Figure 10: Left-Handed Signing (left) and Right Handed Signing (right) in ASL

ASL vs BSL classification results were generated for the same dataset using the base classifier. For each video, we calculated its misclassification rate with the left handed/right handed labels.

Surprisingly, left-handedness of the signer showed no discernable correlation with the misclassification rate. Misclassification rate of these videos ranged from 0% to 87% with a mean of 23% as shown in Figure 11. Mean misclassification rate for the entire ASL dataset is 25%. Due to the small number of left handed signing videos, we were not able to pursue further work in this direction.

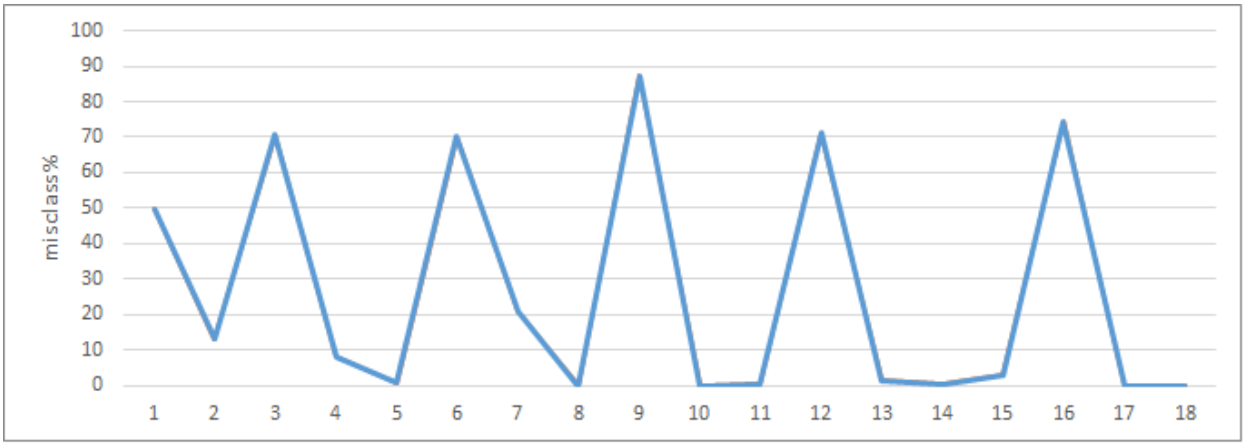


Figure 11: Variation of misclassification rate for videos with left-handed signing in the ASL vs BSL classification using base classifier.

CHAPTER V

DISCUSSIONS AND FUTURE WORK

This work provides a detailed study of the importance of speed of hand motion in detecting sign language videos. While we have attempted to explore the idea from multiple angles, through 3 classifier designs and by employing varied datasets, it is far from comprehensive.

Study of the classification results of speed-frames classifier for SL and nonSL dataset showed that including speed into the feature set can bring huge improvements for certain categories of videos. If there was a mechanism to identify such videos in a pre-processing step, we would observe good classification gains for overall classification. The dataset needs to be expanded so that we find more such videos. Only then will we be able to characterize them effectively. Combining meta data features with visual features can prove to be helpful in this regard.

We have described a classifier that combines speed and foreground features. It will be interesting to explore the effect cascading these features instead of combining.

This work discusses the key frames based approach for designing a fast classifier. It is a preliminary study and has scope for improvement. While our focus here has been towards facilitating the discovery of sign language videos on the internet, fast classifiers can have diverse applications. For example, it can be used in a real-time video conferencing system to focus the camera on the signer. The design presented in this work uses a single symmetry value calculated over the entire region of interest. It can be extended by dividing the ROI into a small number of regions.

Left handed signers is an interesting problem for one-handed sign languages. While it does not have an observable impact on the sign language detection problem, it can derail transcription

and recognition algorithms based on tracking hand shapes and orientations. Currently the dataset of left-handed signers is small. With an extended dataset, classifiers can be trained to differentiate left-handed and right-handed signers. This can be used as a preprocessing step for transcription/recognition.

Research in the field of sign languages focuses largely on the transcription problem. Shipman et al. must be credited for driving the work on detecting sign language video. This thesis takes forward the research in the field of sign language detection and identification through several experiments and observations, while leaving scope for more.

REFERENCES

[1] R. Mitchell, T. Young, B. Bachleda, and M. A Karchmer, "How Many People Use ASL in the United States? Why Estimates Need Updating," *Sign Lang. Stud.*, vol. 6, 2006.

[2] F. M. Shipman, R. Gutierrez-Osuna, and C. D. D. Monteiro, "Identifying Sign Language Videos in Video Sharing Sites," *ACM Trans. Access. Comput.*, vol. 5, no. 4, pp. 1–14, Mar. 2014.

[3] W. C Stokoe, "Sign Language Structure: An Outline of the Visual Communication Systems of the American Deaf," *J. Deaf Stud. Deaf Educ.*, vol. 10, pp. 3–37, 2005.

[4] C. D. D. Monteiro, C. M. Mathew, R. Gutierrez-Osuna, and F. Shipman, "Detecting and Identifying Sign Languages through Visual Features," in *2016 IEEE International Symposium on Multimedia (ISM)*, 2016, pp. 287–290.

[5] C. D. D. Monteiro, R. Gutierrez-Osuna, and F. M. Shipman, "Design and Evaluation of Classifier for Identifying Sign Language Videos in Video Sharing Sites," *Proc. 14th Int. ACM SIGACCESS Conf. Comput. Access.*, pp. 191–198, 2012.

[6] B. G. Gebre, P. Wittenburg, and T. Heskes, "Automatic sign language identification," in *2013 IEEE International Conference on Image Processing*, 2013, pp. 2626–2630.

[7] F. Shipman, R. Gutierrez-Osuna, T. Shipman, C. Monteiro, and V. Karappa, "Towards a Distributed Digital Library for Sign Language Content," in *Proceedings of the 15th ACM/IEEE-CE on Joint Conference on Digital Libraries*, 2015, pp. 187–190.

[8] V. Karappa, C. D. D. Monteiro, F. M. Shipman, and R. Gutierrez-Osuna, "Detection of sign-language content in video through polar motion profiles," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1290–1294.

[9] A. Er-Rady, R. Faizi, R. O. H. Thami, and H. Housni, "Automatic sign language

recognition: A survey," in *2017 International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, 2017, pp. 1–7.

[10] J. W. A. P. T.Starner, "Real-time american sign language recognition using desk and wearable computer based video," *IEEE trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 12, pp. 1371–1375, 1998.

[11] Z. Zafrulla, H. Brashear, T. Starner, H. Hamilton, and P. Presti, "American Sign Language Recognition with the Kinect," in *Proceedings of the 13th International Conference on Multimodal Interfaces*, 2011, pp. 279–286.

[12] N. Cherniavsky, R. E. Ladner, E. A. Riskin, and Others, "Activity detection in conversational sign language video for mobile telecommunication.," in *FG*, 2008, pp. 1–6.

[13] F. M. Shipman, S. Duggina, C. D. D. Monteiro, and R. Gutierrez-Osuna, "Speed-Accuracy Tradeoffs for Detecting Sign Language Content in Video Sharing Sites," in *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility - ASSETS '17*, 2017, pp. 185–189.

[14] S. Duggina, "Evaluation of Alternative Face Detection Techniques and Video Segment Lengths on Sign Language Detection," Texas A&M University, 2015.

[15] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, 2001, vol. 1, p. I----511.

[16] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, 2004, vol. 2, pp. 28–31.