A Framework for Continuous Multimodal Sign Language Recognition

Daniel Kelly, Jane Reilly Delannoy, John Mc Donald and Charles Markham
Computer Science Department
National University of Ireland Maynooth
Ireland
dankelly@cs.nuim.ie

ABSTRACT

We present a multimodal system for the recognition of manual signs and non-manual signals within continuous sign language sentences. In sign language, information is mainly conveyed through hand gestures (Manual Signs). Non-manual signals, such as facial expressions, head movements, body postures and torso movements, are used to express a large part of the grammar and some aspects of the syntax of sign language. In this paper we propose a multichannel HMM based system to recognize manual signs and non-manual signals. We choose a single non-manual signal, head movement, to evaluate our framework when recognizing non-manual signals. Manual signs and non-manual signals are processed independently using continuous multidimensional HMMs and a HMM threshold model. Experiments conducted demonstrate that our system achieved a detection ratio of 0.95 and a reliability measure of 0.93.

Categories and Subject Descriptors

I.4.8 [**Computing Methodologies**]: Image Processing and Computer Vision—*Scene Analysis*

General Terms

Algorithms

Keywords

Sign Language, Non-Manual Signals, HMM

1. INTRODUCTION

Sign Language is a form of non-verbal communication where information is mainly conveyed through hand gestures. Hand gestures can be classified into several categories such as conversational gestures, controlling gestures, manipulative gestures and communicative gestures [42].

There have been many studies on hand gestures, and on sign language in particular, in psycholinguistic research. Stokoe [30] identified the four building blocks of sign language; the hand shape, the position, the orientation and the movement. With these building blocks in mind hand gestures can be classified as either hand postures (hand shape and orientation) or spatiotemporal gestures (position and movement) [43].

A number of works [13, 15, 31, 1, 5, 8, 14, 18, 24, 29, 32, 44, 39, 9] deal only with isolated gesture recognition where the user either performs the gestures one at a time, starting and ending at a neutral position, or with exaggerated pauses, or while applying an external trigger between each word.

One of the main difficulties with recognizing a gesture within a continuous sequence of gestures is that the hand(s) must move from the end point of the previous gesture to the start point of the next gesture. These inter gesture transition periods are called movement epenthesis [21] and are not part of either of the signs. As such, an accurate recognition system must be able to distinguish between valid sign segments and movement epenthesis. Extending isolated recognition to continuous signing requires automatic detection of movement epenthesis segments so that the recognition algorithm can be applied on the segmented signs.

One proposed solution to movement epenthesis detection is an explicit segmentation model were subsets, of features from gesture data, are used as cues for valid gesture start and end point detection [27, 20]. The limitation of this explicit segmentation model arises from the difficulty in creating general rules for sign boundary detection that could be applied to all types of gestures [25].

An approach to dealing with continuous recognition without explicit segmentation is to use Hidden Markov Models (HMM) for implicit sentence segmentation. Starner et al. [28] and Bauer and Kraiss [4] model each word or subunit with a HMM and then train the HMMs with data collected from full sentences. A downside to this is that training on full sentence data may result in a loss in valid sign recognition accuracy due to the large variations in the appearance of all the possible movement epenthesis that could occur between two signs.

Wang et al. [41] also use HMMs to recognize continuous signs sequences with 92.8% accuracy, although signs were assumed to end when no hand motion occurred. Assan et al. [1] model the HMMs such that all transitions go through a single state, while Gao et al. [10] create separate HMMs that model the transitions between each unique pair of signs that occur in sequence. Vogler at al. [37] also use an explicit epenthesis modeling system where one HMM is trained for every two valid combinations of signs.

While these works have had promising results in gesture

ICMI-MLMI'09, November 2–4,2009, Cambridge, MA, USA.
Copyright 2009 ACM 978-1-60558-772-1/09/11 ...$10.00.

recognition and movement epenthesis detection, the training of such systems involves a large amount of extra data collection, model training and recognition computation due to the extra number of HMMs required to detect movement epenthesis. The techniques we implement to deal with these issues are based on the work of Kelly et al [17]. These works introduced a HMM based gesture recognition framework which accurately spots and classifies gestures, within a continuous sequence of sign language, as one of a number of pre trained gestures as well as calculating the likelihood that the given gesture sequence is or is not a movement epenthesis.

Since sign language communication is multimodal it involves not only hand gestures (i.e., manual signing) but also non-manual signals (NMS) conveyed through facial expressions, head movements, body postures and torso movements. Recognizing Sign Language communication therefore requires simultaneous observation of manual and non-manual signals and their precise synchronization and signal integration. Thus understanding sign language involves research in areas of face and facial expression recognition tracking and human motion analysis and gesture recognition.

Over the past number of years there has been a significant amount of research investigating each of these non-manual signals attempting to quantify their individual importance. Works such as [2, 33, 3] focused on the role of head pose and body movement in sign language. These researchers found evidence which strongly linked head tilts and forwards movements to questions, or affirmations. The analysis of facial expressions for the interpretation of sign language has also received a significant amount of interest [12, 11]. Computer-based approaches which model facial movement using *Active Appearance Models* (AAMs) have been proposed [38, 40, 35].

The development of a system combining manual and non-manual signals is a non-trivial task [6]. This is demonstrated by the limited amount of work dealing with the recognition of multimodal communication channels in sign language. Ma et al [23] used Hidden Markov Models (HMMs) to model multimodal information in sign language but lip motion was the only non-manual signal used. Their work was based on the assumption that the information portrayed by the lip movement directly coincided with that of the manual signs. While this is a valid assumption for mouthing, it cannot be generalized to other non-manual signals as they often span multiple manual signs and thus should be treated independently. In this paper we propose a framework for processing multimodal channels of continuous Irish Sign Language (ISL). In ISL, like most other sign languages, the key information is conveyed using manual signs while non-manual signals are used to convey grammatical structure, syntax and emotional context, as such we process these two elements independently. This paper builds on the works of Kelly et al. [17] where hand gestures are recognized from continuous manual signals. As an extension to this technique, we also analyze head movement gestures. The significance of this research lies in the integration of these techniques to create a multichannel ISL interpretation system.

2. FEATURE EXTRACTION

From the definition of a spatiotemporal gesture [30], we must track the position and movement of the hands in order to described a hand gesture sequence. We expand on the work a of hand posture recognition system proposed Kelly et al [16] to build a computer vision based feature extraction system for spatiotemporal gesture recognition. For completeness, prior to discussing our framework for continuous spotting of multimodal gestures in sign language, we briefly describe the feature tracking techniques implemented.

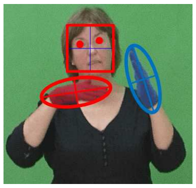


Figure 1: Extracted Features from Image

Tracking of the hands is performed by tracking colored gloves using the Mean Shift algorithm [7]. Face and eye positions are used as features for head movement recognition and also used as hand gesture cues. Face and eye detection is carried out using a cascade of boosted classifiers working with haar-like features proposed by Viola and Jones [34]. A set of public domain classifiers [22], for the face, left eye and right eye, are used in conjunction with the OpenCV implementation of the haar cascade object detection algorithm.

We define the raw features extracted from each image as follows; right hand position (RH_x, RH_y), left hand position (LH_x, LH_y), face position (FC_x, FC_y), face width (FW), left eye position (LE_x, LE_y) and right eye position (RE_x, RE_y).

3. HIDDEN MARKOV MODELS

Hidden Markov Models (HMMs) are a type of statistical model and can model spatiotemporal information in a natural way. HMMs have efficient algorithms for learning and recognition, such as the Baum-Welch algorithm and Viterbi search algorithm [26]. A HMM is a collection of states connected by transitions. Each transition (or time step) has a pair of probabilities: a transition probability (the probability of taking a particular transition to a particular state) and an output probability (the probability of emitting a particular output symbol from a given state). We use the compact notation $\lambda = \{A, B, \pi\}$ to indicate the complete parameter set of the model where A is a matrix storing transitions probabilities and a_{ij} denotes the probability of making a transition between states s_i and s_j. B is a matrix storing output probabilities for each state and π is a vector storing initial state probabilities. HMMs can use either a set of discrete observation symbols or they can be extended for continuous observations signals. Lee and Kim [19] proposed a single channel HMM threshold model using discrete observations to recognize a set of distinct gesture. We expand on their work by developing a multichannel HMM threshold model system using continuous multidimensional observation vectors. This is an important advancement as using continuous multidimensional observation vectors allows further expansion of our framework into different feature vectors without the loss of information through vector quantization which is required when using discrete observations.

To represent a gesture sequence such that it can be modeled by a HMM, the gesture sequence must be defined as a set of observations. An observation O_t, is defined as an observation vector made at time t, where $O_t = \{o_1, o_2, ..., o_M\}$ and M is the dimension of the observation vector. A particular gesture sequence is then defined as $\Theta = \{O_1, O_2, ..., O_T\}$.

To calculate the probability of a specific observation O_t, we implement probability density function of an M-dimensional multivariate gaussian (see Equation 1).

$$\aleph(O_t;\mu,\Sigma) = (2\pi)^{-\frac{N}{2}}|\Sigma|^{-\frac{1}{2}}exp\big(-\tfrac{1}{2}(O_t-\mu)^T\Sigma^{-1}(O_t-\mu)\big) \quad (1)$$

Where μ is the mean vector and Σ is the covariance matrix.

4. MANUAL SIGN RECOGNITION

We expand on the work of Lee and Kim [19] to develop a HMM threshold model system which models a parallel HMM network to recognize two hand signs and identify movement epenthesis. A specific HMM, called a threshold model, is created to model movement epenthesis by calculating the likelihood threshold of an input gesture and provide a confirmation mechanism for provisionally matched gesture patterns. For a network of HMMs $\Lambda = \{\lambda_1, \lambda_2, ..., \lambda_C\}$, where λ_c is a dedicated gesture HMM used to calculate the likelihood that the input gesture is belonging to gesture class c, a single threshold model $\overline{\lambda}$ is created to calculate the likelihood threshold for each of the dedicated gesture HMMs. It is not in the scope of this paper to describe the threshold model in detail and readers should consult the works of Lee and Kim [19] and Kelly et al [17] for a more detailed discussion on the HMM threshold model technique.

4.1 Manual Sign Feature Processing

A spatiotemporal gesture is defined by the hands' position and movement, where the position refers to the hands' location relative to the body and movement traces out a trajectory in space. Kelly et al [17] perform a number of experiments on isolated spatiotemporal gestures and movement epenthesis to find the best performing feature vector. Results showed that the best performing feature vector was a five dimensional vector describing the position of the hand relative to the eyes (RP_x, RP_y), the direction the hand was moving (V_x, V_y) and the distance between the two hands (D_H).

For manual signs, we define O_t as the observation vector made at time t, where $O_t = \{RP_x, RP_y, V_x, V_y, D_H\}$. A particular hand sign sequence is then defined as $\Theta = \{O_1, O_2, ..., O_T\}$.

4.2 Manual Sign HMM Training

Our system initializes and trains a dedicated parallel HMM [36] denoted as λ_c, where $0 < c < C$ and C is the total number of manual signs to be recognized. Each parallel HMM consists of two separate HMMs, λ_{Lc} and λ_{Rc}, that model the right and left hand sign respectively.

HMM model training is carried out by an automated HMM initialization and training technique, utilizing an iterative clustering, Baum Welch and Viterbi realignment process, proposed by Kelly et al [17].

It is desirable to weight λ_{Lc} and λ_{Rc}, the left hand HMM and right hand HMM respectively, due to variations in information held in each of the hands for a particular sign. The weighting applied in our system is based on a variance measure of the observation sequences. Using data from all observation sequences Θ^k_{Lc} and Θ^k_{Rc}, where $1 \leq k \leq K$, K is the total number of training examples and Θ_{Lc} and Θ_{Rc} are the left and right hand observations respectively. The variance of the left and right hand observations are calculated by calculating the variance of each observation dimension $\sigma^2_{Lc}[i]$ and $\sigma^2_{Rc}[i]$, where $0 \leq i \leq D$ and D is the dimension of the observation vectors. The left HMM weight, ω_{Lc}, and right HMM weight, ω_{Rc}, are then calculated as using Equation 2.

$$\omega_{Lc}=\Sigma_{i=0}^{D}\frac{\sigma^2_{Lc}[i]}{(\sigma^2_{Lc}[i]+\sigma^2_{Rc}[i])\times D} \quad \omega_{Rc}=\Sigma_{i=0}^{D}\frac{\sigma^2_{Rc}[i]}{(\sigma^2_{Lc}[i]+\sigma^2_{Rc}[i])\times D} \quad (2)$$

A parallel HMM threshold model, $\overline{\lambda} = \{\overline{\lambda_L}, \overline{\lambda_R}\}$ is then created using the network of trained parallel HMMs λ_c $(0 < c < C)$. The set of parallel HMMs, to recognize the C pre-trained signs, is then denoted as $\Lambda_L = \{\lambda_{L1}, \lambda_{L2}, ..., \lambda_{LC}, \overline{\lambda_L}\}$ and $\Lambda_R = \{\lambda_{R1}, \lambda_{R2}, ..., \lambda_{RC}, \overline{\lambda_R}\}$.

4.3 Manual Sign Classification

To classify the observations, the Viterbi algorithm is run on each model given the unknown observation sequences Θ_L and Θ_R, calculating the most likely state paths through each model c. The likelihoods of each state path, which we denote as $P(\Theta|\lambda_{Lc})$ and $P(\Theta|\lambda_{Rc})$, are also calculated. We calculate the overall likelihoods of a dedicated gesture and a movement epenthesis with the equations defined in Equations 3 and 6.

$$P(\Theta|\lambda_c) = P(\Theta_L|\lambda_{Lc})\omega_{Lc} + P(\Theta_R|\lambda_{Rc})\omega_{Rc} \quad (3)$$

$$\Psi_c = \frac{P(\Theta_L|\overline{\lambda_L})\Gamma_{Lc} + P(\Theta_R|\overline{\lambda_R})\Gamma_{Rc}}{2} \quad (4)$$

Where Γ_{Lc} and Γ_{Rc} are constant scalar values used to tune the sensitivity of the system to movement epenthesis. The sequence of observations can then be classified as c if $P(\Theta|\lambda_c) \geq \Psi_c$ evaluates to be true.

5. NON-MANUAL SIGNAL RECOGNITION

While hand gestures do play central grammatical roles, movements of the head, torso and face are used to express certain aspects of ISL. In this work we will focus on a single non-manual signal, the head movement, to evaluate our techniques when recognizing non-manual features.

5.1 Head Movement HMM Training

Our system initializes and trains a dedicated HMM for each head movement gesture to be recognized. In this work we evaluate our techniques using three different head movement gestures; a left head movement, a right head movement and a left-forward movement.

To train the head movement HMMs, we recorded 18 different videos of a fluent ISL signer performing the head movements naturally within full sign language sentences. Six videos where recorded for each head movement gesture. Each head movement HMM λ^H_i (where $0 < i < I$ and I is the total number of head gestures) was then trained on the observation sequences extracted from the corresponding videos. The start and end point of each of the head movement gestures were labeled, the observation sequences Θ_i were extracted and each HMM was then trained using the iterative HMM training model proposed by Kelly at al [17]. A HMM threshold model, $\overline{\lambda^H}$ is then created using the network of trained HMMs λ^H_i (where $0 < i < I$). The set of HMMs, to recognize the I pre-trained head movement gestures, is then denoted as $\Lambda^H = \{\lambda^H_1, \lambda^H_2, ..., \lambda^H_I, \overline{\lambda^H}\}$.

5.2 Head Movement Recognition

Given an unknown sequence of head movement observations Θ^H, the goal is to accurately classify the head movement gesture as a non head gesture or as one of the I trained gestures. To classify the observations, the Viterbi algorithm is run on each model given the unknown observation sequences Θ^H, calculating the most likely state paths through each model i. The likelihoods of each state path, which we denote as $P(\Theta^H|\lambda_i^H)$, are also calculated. The sequence of observations can then be classified as i if Equation 5 evaluates to be true.

$$P(\Theta^H|\lambda_i^H) \geq \Psi_i^H \tag{5}$$

$$\Psi_i^H = P(\Theta^H|\overline{\lambda^H})\Gamma_i^H \tag{6}$$

Where Γ_i^H is a constant scalar value used to tune the sensitivity of the system non head movement gestures.

5.3 Head Movement Feature Processing

The goal of the head movement gesture recognition system is to spot and classify head movement gestures from within a continuous sign language sentence. An accurate head movement spotter must first be able to discriminate between positive a negative head movement gesture samples, therefore, we perform a set of experiments to find the best performing feature set when discriminating between isolated positive and negative head gestures.

To test the discriminative performance of different feature vectors, we recorded an additional 7 videos for each head gesture (21 in total), where a fluent ISL signer performed the head movement gestures within different sign language sentences. The start and end points of the head gestures were then labeled and isolated observation sequences Θ_i^{τ} were extracted. An additional set of 15 other head gesture sequence, outside of the training set, were also labeled in the video sequences to test the performance of the system when identifying negative gestures.

The classification of a gesture is based on a comparison of a weighted threshold model likelihood with the weight denoted as Γ_i^H. In our ROC analysis of the system, we vary the weight, Γ_i^H, over the range $0 \leq \Gamma_i^H \leq 1$ and then create a confusion matrix for each of the weights.

To evaluate the performance of different features, we performed a ROC analysis on the models generated from the different feature combinations and calculated the area under the curve (AUC) for each feature vector model. Table 1 shows the AUC measurement of four different features which were evaluated during our experiments. To calculate the directional vector of the head, (V_x^H, V_y^H), we used the mid point between the eyes and calculated the direction the midpoint moved from frame to frame. We used a sliding window to average the directional vector and in our experiments we evaluated the best performing window size for each feature vector. Although we evaluated each feature vector with a range of different window sizes, we report the best performing window sizes for each feature vector in Table 1.

6. CONTINUOUS RECOGNITION

Thus far we have described a framework for classifying manual signs and head movement gestures. Kelly et al [17] perform experiments to show the robustness of this framework for recognizing isolated hand gestures with a ROC area under the curve measurement of 0.949. In Section 5.3, we expanded on the work of Kelly et al [17] to recognize head movement gestures with a ROC area under the curve measurement of 0.936.

Table 1: AUC Measurements for Different Feature Combinations

Features	**Window Size**	**ROC AUC**
F_1 - Unit Direction Vector $(\hat{V}_x^H, \hat{V}_y^H)$	6	0.821
F_2 - Direction Vector (V_x^H, V_y^H)	**12**	**0.936**
F_3 - Unit Direction Vector $(\hat{V}_x^H, \hat{V}_y^H)$ + Angle Eyes (θ_{eyes})	6	0.863
F_4 - Direction Vector (V_x^H, V_y^H) + Angle Eyes (θ_{eyes})	6	0.868

In order to spot and classify manual signs and head movement gestures, we must extract three observation channels from the video streams. The three observation channels correspond to the left hand observations Θ_L, the right hand observations Θ_R and the head movement observations Θ^H. The observations Θ_L and Θ_R are combined into a parallel observation sequence Θ which will be processed by the set of parallel HMMs. Since manual and non-manual signals are independent, the recognition of Θ and Θ^H will be processed independently and will be combined after the independent spotting and recognition of gestures within each of the two independent channels.

6.1 Continuous Manual Sign Recognition

We will now describe our system for spotting and classifying manual signs within a continuous sequence, Θ, extracted from natural sign language sentences.

The first step in our spotting algorithm is gesture end point detection. To detect a gesture end point in a continuous stream of gesture observations $\Theta = \{O_1, O_2, ..., O_T\}$, we calculate the model likelihoods of observation sequence $\theta = \{O_{T-F}, O_{T-F-1}, ..., O_T\}$ where θ is a subset of Θ and F defines the length of the observation (no. of frames) subset used. In this paper we set F to the average length of the observation sequences used to train the system.

A candidate hand gesture, κ, with end point, $\kappa_e = T$, is flagged when $\exists c : P(\Theta|\lambda_c) \geq \Psi_c$. Figure 2 illustrates the likelihood time evolution of the hand gesture model "Lost" when given an observation sequence where the signer performs the "Lost" sign. It can be seen from Figure 2 that a number of candidate end points occur between $T = 16$ and $T = 21$.

$$\Phi_c(\Theta) = \tfrac{P(\Theta|\lambda_c)}{P(\Theta|\lambda_c)+\Psi_c} \tag{7}$$

For each candidate end point we calculate a corresponding start point κ_s. Different candidate start points are evaluated using the measurement shown in Equation 7 where $\Phi_c(\Theta)$ is normalized metric (between 0 and 1) which measures the strength of gesture c given observations Θ. To find a candidate start point, the metric $\Phi_c(\Theta_{s\kappa_e})$ is calculated over different values of s, where $\Theta_{s\kappa_e} = \{O_s, O_{s+1}, ..., O_{\kappa_e}\}$ and $(\kappa_e - F^2) \leq s < \kappa_e$. The candidate gesture start point κ_s, is then found using Equation 8.

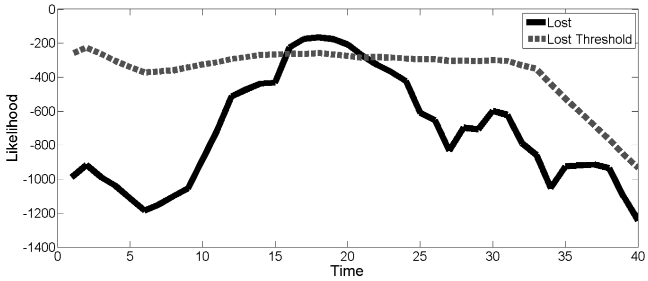


Figure 2: Likelihood evolution of "Lost" gesture model and associated threshold model

$$\kappa_s = \underset{s}{\operatorname{argmax}} \Phi_c(\Theta_{s\kappa_e}) \qquad (8)$$

The start and end point detection algorithm may flag candidate gestures which overlap and for this reason we expand on our continuous sign recognition algorithm with a candidate selection algorithm. The purpose of the candidate selection algorithm is to remove overlapping candidate gestures such that the single most likely gesture is the remaining gesture for a particular time frame.

We will use a sample sign language sentence "I Lost Book" to illustrate our candidate selection algorithm in the context of our gesture and threshold likelihood evaluation, where the system was trained on the following 8 signs; "Paper", "Alot", "Bike", "Clean", "Paint", "Plate", "Lost" and "Gone". Figure 3 illustrates the difference between the gesture model likelihood $P(\Theta|\lambda_c)$ and its corresponding threshold Ψ_c, where positive values indicates $P(\Theta|\lambda_c) \geq \Psi_c$. We illustrate 4 gesture model likelihoods as all other gesture model likelihoods never exceed their corresponding threshold.

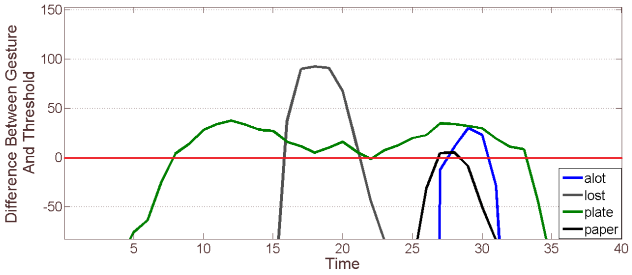


Figure 3: Gesture And Corresponding Threshold Model Likelihood Difference

The first step in the candidate selection algorithm is to cluster overlapping gestures, with the same gesture classification, together. Each of these candidate gestures, within the cluster, have an associated metric $\kappa_p = \Phi_c(\Theta_{\kappa_s\kappa_e})$. We remove all but one candidate gesture from this cluster leaving the candidate gesture, κ^B, with the highest κ_p value. We repeat this step for each cluster to produce a set of candidate gestures $\Upsilon = \{\kappa^{B1}, \kappa^{B2}, ..., \kappa^{BK}\}$, where K is the total number of clusters created from grouping overlapping gestures, with the same gesture classification, together. Figure 4 shows the time segments and Φ metrics of each candidate gesture after the first candidate selection step.

The second step in the candidate selection algorithm is an iterative selection step to remove the least probable candidate gestures as shown in Algorithm 1.

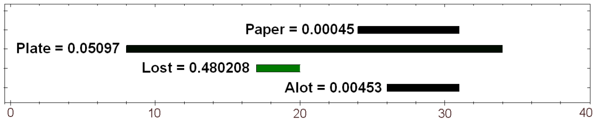


Figure 4: Candidate Gestures, Υ, after first candidate selection step

Algorithm 1 Second Step of Candidate Selection Algorithm

```
Sort($\Upsilon$) by In Order of Increasing $\kappa_P^B$
for $i \leq K$ do
  if $\exists j \in J = \{i+1, i+2, ..., K\}$, such that $\Upsilon[j]$ overlaps with $\Upsilon[i]$ then
    Remove $\Upsilon[i]$ from $\Upsilon$
  end if
end for
```

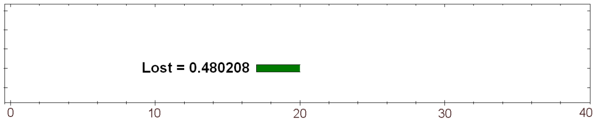


Figure 5: Recognized Gestures, Υ, after final candidate selection step

Figure 5 shows the time segments and Φ metrics of the recognized gestures after the second candidate selection step where the sign "Lost" is correctly recognized.

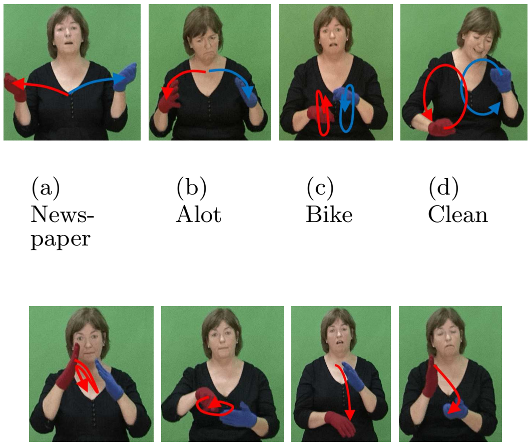
6.2 Continuous Non Manual Signal Recognition

The spotting and classifying of the non-manual channel Θ^H is then conducted using the methods described in Section 6.1 above, however to keep the notation consistent with the techniques described in Section 5, the notation Θ, c, λ_c and Ψ_c should be substituted with Θ^H, i, λ_i^H and Ψ_i^H respectively.

7. CONTINUOUS RECOGNITION EXPERIMENTS

To evaluate the performance of our recognition framework, a set of eight different manual signs and a set of three different head movement gestures, as performed by a fluent signer, were recorded and labeled. The set of gestures were not selected to be visually distinct but to represent a suitable cross section of the manual signs and head movement gestures that can occur in sign language. Figure 6 illustrates an example of a signer performing each of the eight manual signs, and Figure 7 illustrates an example of a signer performing each of the three different head movement gesture.

The left direction head movement and right direction head movement are commonly used in sign language sentences to denote a "wh" question while the left forward direction head movement is often used to convey a yes/no question.

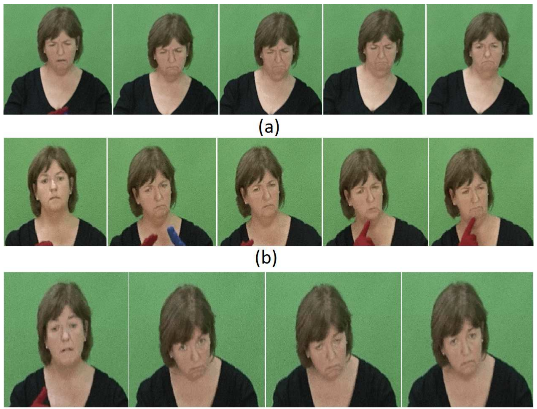


(a) Newspaper (b) Alot (c) Bike (d) Clean

(e) Paint (f) Plate (g) Lost (h) Gone

Figure 6: Example of the eight different signs the system was tested on

A total of 160 additional video clips of full unsegmented sign language sentences being performed by a fluent signer were recorded to test the performance of our continuous recognition framework. Each video clip contained at least one of the eight chosen manual signs and the three head movement gestures occurred 30 times within the 160 videos. Videos were recorded at 25 frames per second with an average length of 5 seconds. Observation sequences Θ_L, Θ_R and Θ^H were extracted from each video clip and our continuous recognition framework, described in Section 6, was used to process the observation sequences to spot and classify manual signs and head movement gestures from within the videos.



(a)

(b)

(c)

Figure 7: Example of the three different head movement gestures the system was tested on (a) Right Movement (b) Left Movement (c) Left Forward Movement

In the gesture spotting and classification task, there are three types of errors: *an insertion error* occurs when the spotter reports a nonexistent gesture, *a deletion error* occurs when the spotter fails to detect a gesture, and *a substitution error* occurs when the spotter falsely classifies a gesture. From these error measures we define two performance metrics shown in Equations 9 and 10.

$$DetectionRatio = \frac{\#CorrectlyRecognizedGestures}{\#InputGestures} \quad (9)$$

$$Reliability = \frac{\#CorrectlyRecognizedGestures}{\#InputGestures + \#InsertionErrors} \quad (10)$$

Table 2 shows the performance of our system when spotting and classifying signs within continuous sequences of video. The experiment shows an overall detection rate of 95.7% and an overall reliability of 93.8% when independently spotting and classifying manual and non-manual gestures in continuous sign language sentences.

Table 2: Continuous Spotter and Classifier Performance

Gesture	#Correct	*#Del*†	*#Ins*‡	*#Sub*††	Detection	Reliability
Gone	20	0	0	0	1.0	1.0
Alot	20	0	0	0	1.0	1.0
Lost	20	0	0	0	1.0	1.0
Plate	19	0	1	0	0.95	0.90
Bike	20	0	0	0	1.0	1.0
Paint	20	0	0	0	1.0	1.0
Paper	16	0	1	3	0.8	0.76
Clean	18	0	1	1	0.9	0.85
Head Left	11	0	1	0	0.91	0.84
Head Right	10	0	0	0	1.0	1.0
Head Left Forward	8	0	0	1	0.88	0.88
Total	**182**	**0**	**4**	**5**	**0.957**	**0.938**

† *Number of Deletion Errors*
‡*Number of Insertion Errors*
††*Number of Substitution Errors*

We also evaluate the performance of the start and end point detection relative to ground truth data labeled by a human sign language translator. Table 3 shows the average absolute difference between the spotters start and end points and the human interpreters start and end points for signs that were correctly spotted and classified. The average start point error was 8.1 frames and the average end point error was 7.6 frames. From this experiment we can conclude that our spotter is capable of detecting start points, within an average of 324 milliseconds of a human interpreter, and end points, within an average of 304 milliseconds of a human interpreter.

Table 3: Continuous Spotter and Classifier Performance

Gesture	Start Error (Frames)	End Error (Frames)
Gone	±2.5	±8.4
Alot	±1.5	±1.6
Lost	±1.5	±3.5
Plate	±8.1	±12.2
Bike	±12.1	±12.0
Paint	±26.1	±20.7
Paper	±5.9	±1.6
Clean	±4.8	±5.2
Head Left	±10.1	±7.7
Head Right	±4.0	±4.3
Head Left Forward	±12.9	±6.5
Total	**±8.1**	**±7.6**

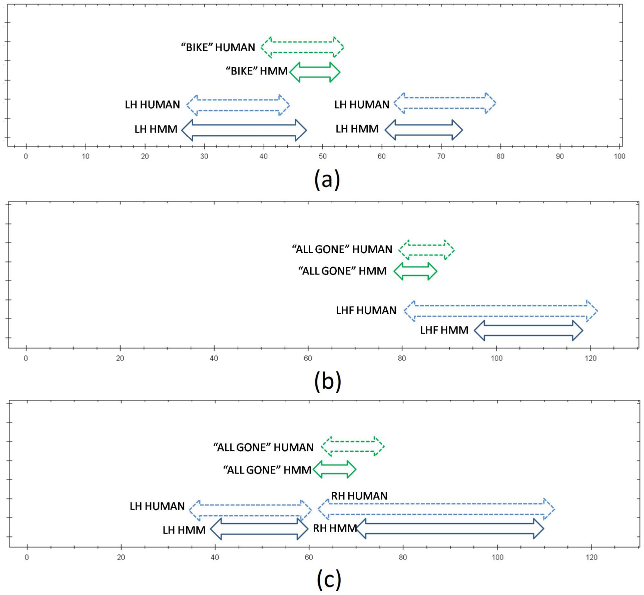


Figure 8: Multimodal gesture labeling comparison of a human interpreter vs. our recognition system

8. CONCLUSION

In this paper we have discussed current methods of continuous sign recognition. The downside of these methods is that unnatural constraints are put on the signer, such as pauses between words, or the explicit training of models to handle movement epenthesis must be carried out. The method we have proposed in this paper can recognize gestures from within natural unconstrained sign language sentences which requires that a set of dedicated gesture models be trained, and as a result of this training a single threshold model can be created to identify negative samples.

We have also discussed the importance of non-manual signals in sign language. We have highlighted there are currently a limited number of works which incorporate both manual and non-manual signals into a single framework for continuous automatic sign language recognition. The principal contribution of our work is that we have developed a multimodal framework for spotting and classifying manual and non-manual gestures from continuous sign language sentences. The system which we have proposed is unique as, unlike current works, each manual and non-manual signal is processed independently within our multimodal framework.

Experiments conducted demonstrate that our system achieved a detection ratio of 0.957 and a reliability measure of 0.938. Experiments also showed that our gesture spotting system successfully flagged gesture start points and end points within ±324 milliseconds and ±304 milliseconds respectively when compared to a human interpreter. Through these experiments we have proved the robustness of our system when recognizing a number of different manual and non-manual signals.

Another contribution of this paper is that we have expanded on the work of Lee and Kim [19] where they proposed single channel HMM threshold model using discrete observations to recognize a set of distinct gestures. Although our system is based on their HMM threshold model, it differs in that we have developed a multichannel HMM threshold model using continuous multidimensional observation vectors. This is an important advancement as using continuous multidimensional observation vectors allows further expansion of our framework into different feature vectors without the loss of information through vector quantization which is required when using discrete observations. As a result of this, we hypothesize that our framework is extendable to model many of the modes of communication in sign language. Future work will involve testing this hypothesis by incorporating different modes of non-manual communication, such as facial expressions, to our framework.

Acknowledgment

The Authors would like to acknowledge the financial support of the Irish Research Council for Science, Engineering and Technology (IRCSET).

9. REFERENCES

[1] M. Assan and K. Grobel. Video-based sign language recognition using hidden markov models. In *Proceedings of the International Gesture Workshop on Gesture and Sign Language in Human-Computer Interaction*, pages 97–109, London, UK, 1998. Springer-Verlag.

[2] B. Bahan. *Nonmanual Realisation of Agreement in American sign language*. PhD thesis, University of California, Berkely, 1996.

[3] C. Baker-Shenk. Factors affecting the form of question signals in asl. *Diversity and Diachrony*, 1986.

[4] B. Bauer and K.-F. Kraiss. Towards an automatic sign language recognition system using subunits. In *GW '01: Revised Papers from the International Gesture Workshop on Gesture and Sign Languages in Human-Computer Interaction*, pages 64–75, London, UK, 2002. Springer-Verlag.

[5] B. Bauer and K.-F. Kraiss. Video-based sign recognition using self-organizing subunits. *Pattern Recognition, 2002*, 2:434–437 vol.2, 2002.

[6] S. C., W. Ong, and S. Ranganath. Automatic sign language analysis: A survey and the future beyond lexical meaning. *IEEE Trans. PAMI*, 27(6):873–891, 2005.

[7] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, 2:142–149 vol.2, 2000.

[8] Y. Cui and J. Weng. Appearance-based hand sign recognition from intensity image sequences. *CVIU*, 78(2):157–176, 2000.

[9] L. Ding and A. Martinez. Modelling and recognition of the linguistic components in american sign language. *Journal of Image and Vision Computing*, In Press, 2009.

[10] W. Gao, G. Fang, D. Zhao, and Y. Chen. Transition movement models for large vocabulary continuous sign language recognition. *IEEE FG 2004*, pages 553–558, May 2004.

[11] R. Grossman and J. Kegl. Moving faces: Categorization of dynamic facial expressions in american sign language by deaf and hearing participants. *Journal of Nonverbal Behavior*, 31(1):23–38, 2007.

[12] R. B. Grossman and J. Kegl. To capture a face: A novel technique for the analysis and quantification of facial expressions in american sign language, 2006.

[13] E.-J. Holden and O. Robyn. Visual sign language recognition. *Mutli-Image Analysis*, 2001.

[14] C.-L. Huang and W.-Y. Huang. Sign language recognition using model-based tracking and a 3d hopfield neural network. *Mach. Vision Appl.*, 10(5-6):292–307, 1998.

[15] K. Imagawa, H. Matsuo, R. Taniguchi, D. Arita, S. Lu, and S. Igi. Recognition of local features for camera-based sign language recognition system. In *ICPR '00*, page 4849, Washington, DC, USA, 2000. IEEE Computer Society.

[16] D. Kelly, J. McDonald, T. Lysaght, and C. Markham. Analysis of sign language gestures using size functions and principal component analysis. In *IMVIP 2008*, 2008.

[17] D. Kelly, J. McDonald, and C. Markham. Recognizing spatiotemporal gestures and movement epenthesis in sign language. In *IMVIP 2009*, 2009.

[18] T. Kobayashi and S. Haruyama. Partly-hidden markov model and its application to gesture recognition. In *ICASSP '97*, page 3081, Washington, DC, USA, 1997. IEEE Computer Society.

[19] H. K. Lee and J. H. Kim. An hmm-based threshold model approach for gesture recognition. *IEEE PAMI*, 21(10):961–973, 1999.

[20] R. H. Liang and M. Ouhyoung. A real-time continuous gesture recognition system for sign language. In *IEEE FG 1998*, page 558, Washington, DC, USA, 1998. IEEE Computer Society.

[21] J. R. Liddell, S.K. American sign language: The phonological base. *Sign Langauge Studies*, 64.

[22] L. A.-C. M. Castrillt'on-Santana, O. Dt'eniz-Sut'arez and J. Lorenzo-Navarro. Performance evaluation of public domain haar detectors for face and facial feature detection. *VISAPP 2008*, 2008.

[23] J. Ma, W. Gao, and R. Wang. A parallel multistream model for integration of sign language recognition and lip motion. In *ICMI '00: Proc of the 3rd Intl Conf on Adv in Multimodal Interfaces*, pages 582–589, 2000.

[24] H. Matsuo, S. Igi, S. Lu, Y. Nagashima, Y. Takata, and T. Teshima. The recognition algorithm with non-contact for japanese sign language using morphological analysis. In *Proceedings of the International Gesture Workshop on Gesture and Sign Language in Human-Computer Interaction*, pages 273–284, London, UK, 1998. Springer-Verlag.

[25] S. C. W. Ong and S. Ranganath. Automatic sign language analysis: A survey and the future beyond lexical meaning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(6):873–891, 2005.

[26] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, Feb 1989.

[27] H. Sagawa and M. Takeuchi. A method for recognizing a sequence of sign language words represented in a japanese sign language sentence. In *IEEE FG 2000*, page 434, Washington, DC, USA, 2000. IEEE Computer Society.

[28] T. Starner, A. Pentland, and J. Weaver. Real-time american sign language recognition using desk and wearable computer based video. *IEEE PAMI*, 20(12):1371–1375, 1998.

[29] T. Starner, J. Weaver, and A. Pentl. Real-time american sign language recognition from video using hidden markov models. *IEEE PAMI*, 20:1371–1375, 1998.

[30] J. Stokoe, William C. Sign language structure: An outline of the visual communication systems of the american deaf. *Journal of Deaf Studies and Deaf Education, v10 n1 p3-37 Win 2005*, 2005.

[31] S. Tamura and S. Kawasaki. Recognition of sign language motion images. *Pattern Recogn.*, 21(4):343–353, 1988.

[32] N. Tanibata, N. Shimada, and Y. Shirai. Extraction of hand features for recognition of sign language words. In *In International Conference on Vision Interface*, pages 391–398, 2002.

[33] E. van der Kooij, O. Crasborn, and W. Emmerik. Explaining prosodic body leans in sign language of the netherlands: Pragmatics required. *Journal of Pragmatics*, 38, 2006. Prosody and Pragmatics.

[34] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *CVPR, IEEE*, 1:511, 2001.

[35] C. Vogler and S. Goldenstein. Facial movement analysis in asl. *Universal Access in the Information Society*, 6(4):363–374, 2008.

[36] C. Vogler and D. Metaxas. Parallel hidden markov models for american sign language recognition. In *In ICCV*, pages 116–122, 1999.

[37] C. Vogler and D. Metaxas. A framework for recognizing the simultaneous aspects of american sign language. *Computer Vision and Image Understanding*, 81:358–384, 2001.

[38] U. von Agris, M. Knorr, and K.-F. Kraiss. The significance of facial features for automatic sign language recognition. pages 1–6, 2008.

[39] U. von Agris, D. Schneider, J. Zieren, and K.-F. Kraiss. Rapid signer adaptation for isolated sign language recognition. In *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, page 159, Washington, DC, USA, 2006. IEEE Computer Society.

[40] U. von Agris, J. Zieren, U. Canzler, B. Bauer, and K.-F. Kraiss. Recent developments in visual sign language recognition. *Universal Access in the Information Society*, 6(4):323–362, 2008.

[41] C. Wang, S. Shan, and W. Gao. An approach based on phonemes to large vocabulary chinese sign language recognition. In *IEEE FG 2002*, page 411, Washington, DC, USA, 2002. IEEE Computer Society.

[42] Y. Wu and T. Huang. Human hand modeling, analysis and animation in the context of hci, 1999.

[43] Y. Wu, T. S. Huang, and N. Mathews. Vision-based gesture recognition: A review. In *Lecture Notes in Computer Science*, pages 103–115. Springer, 1999.

[44] M. H. Yang, N. Ahuja, and M. Tabb. Extraction of 2d motion trajectories and its application to hand gesture recognition. *IEEE PAMI.*, 24(8):1061–1074, 2002.