2nd Hellenic Conf. on AI, SETN-2002, 11-12 April 2002, Thessaloniki, Greece,
Proceedings, Companion Volume, pp. 343-354

Hidden Marcov Models for Sign Language Recognition: a Review

V.N. Pashaloudi and K.G. Margaritis

Parallel and Distributed Processing Laboratory
Department of Applied Informatics
University of Macedonia
156 Egnatia str., P.O. Box 1591, 54006, Thessaloniki, Greece
{vpasx,kmarg}@uom.gr
http://macedonia.uom.gr/~{vpasx,kmarg}

Abstract. Hidden Marcov Models (HMMs) are statistical models of sequential data that have been successfully used in many applications in artificial intelligence, pattern recognition and speech recognition. Hidden Marcov Models have intrinsic properties, which make them very attractive for sign language recognition. Explicit segmentation on the word level is not necessary for either training or recognition. Language and context models can be applied on several different levels and much related development of this technology has already been done in the field of speech recognition. Consequently, sign language recognition seems an ideal machine vision application of HMM technology, offering the benefits of problem scalability, well defined meanings, a pre-determined language model, a large base of users and immediate applications for a recognizer. In this paper we present a review of the work that has been done in the field of gesture recognition and sign language recognition.

1 Introduction

Sign language (SL) is the usual method of communication for hearing-impaired people, who often have to communicate with other people through a sign-language interpreter. However, they cannot always communicate when they wish, since the number of interpreters is limited. Therefore, there is an increasing need for an automatic SL translation system, based on pattern recognition and classification techniques. Signs can be performed by one or two hands and they are called one-handed or two-handed respectively. In one-handed signs the hand that performs the sign is always the same and is called *dominant* hand. A sign in sign language usually represents a whole word. It is a movement of the dominant hand or of both two hands. However, in most sign languages special words, such as last names, have to be spelled out, by using sequences of the static (not time-varying) signs, that represent the alphabet letters. Thus, one aspect of SL recognition is the recognition of static signs, called *postures*, and the other aspect is the recognition of hand movements, called *gestures*. Both aspects are significant, though the latter is more complicated, since it compensates with the recognition of hand morphs that change through time. Furthermore, the recognition of SL sentences can be confronted in two ways: *isolated sign recognition*, where each sign has start and endpoint, and *continuous sign recognition*, where there are no marked boundaries between signs. In the latter case the performance of each sign is affected by the preceding and the subsequent sign (*co-articulation*).

Different methods have been used for sign language recognition, involving neural networks, image processing algorithms, fuzzy systems [1], [2], [3], [4], [5], [6], [7], [8], [9] or Hidden Marcov Models. By now many systems use "datagloves" as input devices for the recognition of SL gestures. These approaches suffer from limiting the user's freedom of movement. Video-based techniques are less intrunsive ant therefore more comfortable to utilise.

Hidden Marcov Models (HMMs) are statistical models of sequential data that have been successfully used in many applications. The property of HMMs to compensate time and amplitude variances of signals has been proven useful for speech and character recognition. Due to these characteristics HMMs appear as an ideal approach to SL recognition. Like speech, sign language can be considered a non-deterministic time signal. Instead of words or phonemes we here have a sequence of signs. However, unlike speech recognition, where the smallest unit is the phoneme, linguistics have not yet agreed on subunits for signs. Thus, most commonly each sign is modelled with one HMM. In this paper we are going to discuss the use of Hidden Marcov Models in SL recognition, focusing in gesture recognition within SL. We first report HMMs' use in speech recognition, since it comprises the research area, where HMMs were primarily applied. Then some of the research work in gesture recognition is reported. Finally, works that use HMMs in SL recognition are presented. It should be noted that, what we call learning throughout this review paper, is also called parameter estimation in statistics and system identification in control and engineering.

2 Hidden Marcov Models

Given a set of N states s_i we can describe the transitions from one state to another at each time step t as a stochastic process. The transition probability to reach state s_i in the first time step is denoted as π_i. Assuming that the transition probability α_{ij} of state s_i to state s_j only depends on the preceding states, we call this process a Marcov chain. The further assumption, that the actual transition only depends on a very preceding state leads to a first order Marcov chain.

We can now define a second stochastic process that produces, at each time step t, symbol vectors x. The emission probability of a vector x only depends on the actual state, but not on the way the state was reached. The emission probability density $b_i(x)$ for vector x at state s_i can either be discrete or continuous.

This doubly stochastic process is called a Hidden Marcov Model (HMM), if only the vectors x are observable, but not the state sequence. A HMM, λ, is defined by its parameters $\lambda = (A, B, \pi)$. The NxN matrix A represents the transition probabilities a_{ij} from state s_i to s_j, B denotes the vector of the emission densities $b_i(x)$ of each state s_i and π is the vector of the initial transition probabilities π_i.

The first problem we have to solve, when working with HMMs, is *the evaluation problem:* Given the Observation sequence O, compute the probability $P(O|\lambda)$, that a HMM $\lambda = (\Pi, A, B)$ generated O. This problem corresponds to maximum likelihood recognition of an unknown data sequence with a set of HMMs, each of which corresponds to a sign. For each HMM, the probability, that it generated the unknown sequence, is computed, and then the HMM with the highest probability is selected as the recognized sign. For computing this probability a method called **Forward-Backward** algorithm is used [10].

Another problem we have to cope with is that, given the parameters of a HMM λ and an observation sequence O of vectors O_t of the signal, we have to find the state sequence q, that emits, with a high probability, the same symbol vectors as observed from the signal. This problem can be solved with the Viterbi algorithm [10], [11]. This is called the *decoding problem.*

The last problem we have to deal with is the *estimation problem:* Adjust the parameters of an HMM λ such that they maximize $P(O|\lambda)$ for some O. This problem corresponds to training the HMMs with data, such that they are able to recognize previously unseen data correctly after the training phase. No analytical calculation method is known for maximizing $P(O|\lambda)$ for given observation sequences, but an iterative procedure, called the **Baum-Welch** procedure, maximizes $P(O|\lambda)$ locally ([10], [11]).

The most commonly used HMM *topology* is the **left-right** model or else **Bakis** model, where transitions only flow forward from lower states to the same state or higher states.

3 Speech recognition with HMMS

Since speech recognition has been the most common application of HMMs, we will discuss some application specific issues, although this discussion is relevant to many other applications. The basic speech recognition problem may be stated as follows: given a sequence of acoustic descriptors (e.g. spectral vectors), find the sequence of words intended by the speaker who pronounced those descriptors. There are two ways to approach the problem of Speech recognition: **Isolated** and **Continuous** speech recognition [11].

3.1 Isolated Speech recognition

For each word of a vocabulary, W, a separate N-state HMM is designed. A speech signal of a given word is represented as a time sequence of coded spectral vectors. It is assumed that the coding is done using a spectral codebook with M unique spectral vectors; hence, each observation is the index of the spectral vector closest to the original speech signal. Thus, for each vocabulary word, there exists a training sequence, consisting of a number of repetitions of sequences of codebook indices of the word (by one or more talkers). The first task is to build individual word models. This task is done by optimally estimate model parameters for each model (**estimation problem**). We then use the solution to the **decoding problem**, to segment each word training sequences into states and then study the properties of the spectral vectors that lead to the observations occuring in each state. The goal here would be to make refinements on the model (e.g. more states, different codebook size, etc.) so as to improve its capability of modeling the spoken word sequences. Finally, once the set of W HMMs has been designed and optimized, recognition of an unknown word is performed using the solution to the **evaluation problem**, to score each word model based upon the given test observation sequence and select the word whose model score is highest (i.e. the highest likelihood).

3.2 Connected Speech Recognition

A more interesting task is that of recognizing connected speech. For the solution of this problem **language models** are used. A language model is a crucial element of modern speech recognition systems, because most word sequences are very unlikely in a particular language, and in a particular semantic context.

The solutions generally adopted are based on representing the language model in a graphical form and using search techniques to combine the constraints from the acoustic model ($P(O_1, \ldots, O_T | w_1, \ldots, w_L)$) with those from the language model ($P(w_1, \ldots, w_L)$).

A very common type of language model is based on restricting the context to word **bigrams** ($P(w_i | w_{i-1}$) or **trigrams** ($P(w_i | w_{i-1}, w_{i-2})$). Such language models have a simple Marcovian interpretation and can be combined with the acoustic HMMs to build a large HMM in which transition probabilities between HMMs representing a word (possibly in the context of other words) are obtained from the language model.

4 Gesture Recognition Using HMMs

Since the problem of gesture recognition in sign language is part of the general gesture recogniton problem, we thougnt it would be useful to report some works that have been implemented in this research field. Furthermore, thorough reports in this area can be found in [12] and [13].

Yamato *et al.* in [22] use HMMs to recognize image sequences of six different tennis strokes among three subjects. To apply HMMs, one set of time-sequential images is transformed into an image feature vector sequence, and the sequence is converted into a symbol sequence by vector quantization. In learning the six human action categories, the parameters of the six HMMs are optimized so as to best describe the training sequences from each category. To recognize an observed sequence, the HMM that best matches the sequence is chosen. This experiment is significant because it uses a 25x25 pixel quantized subsampled camera image as a feature vector. Even with such low-level information, the model can learn the set of motions and recognize them with up to 90% accuracy.

Nam and **Wohn** in [21] described an HMM-based method for recognizing the space-time hand movement pattern. An HMM models the spacial variance and time-scale variance in the hand movement. As for the recognition of the continuous, connected hand movement patterns, an HMM-based segmentation method is used in the specific work. Data are obtained by the use of one Data-Glove. To deal with the dimensional complexity caused by the 3D problem space, a plane fitting method is employed and the 3D data is reduced into 2D. These 2D data are then encoded as the input to HMMs. In addition to the hand movement, which is regarded as the primary attribute of the hand gesture, they also consider the hand configuration (posture) and the palm orientation. These three major attributes are processed in parallel and rather independently, followed by the inter-attribute communication for finding the proper interpretation. They use a small number of object description gestures and action indicating ones, and achieve a recognition accuracy of 80%.

Rigoll *et al.* present in [23] a real-time system for gesture recognition. The recognition is based on global motion features, extracted from each difference image of an image sequence. The HMMs, that are used as statistical classifier, are trained on a database of 24 isolated gestures, performed by 14 different people. With the use of global motion features, a recognition rate of 92.9% is achieved for a person and background independent recognition.

5 Sign Language Recognition with HMMs

HMMs are an attractive choice for processing three-dimensional sign data, because their state-based nature enables them to describe how a sign changes over time and to capture variations in the duration of signs, by remaining in a state for several time frames.

There are two ways to approach the recognition problem in Sign language. **Isolated recognition** attempts to recognize one single sign at a time. Hence, it requires clearly marked boundaries between signs. Such a boundary could simply be silence, that is, a brief resting phase after each sign, during which the signer performs no movements. **Continuous recognition**, on the other hand attempts to recognize an entire stream of signs, without any artificial pauses or any other form of marked boundaries between the individual signs.

Starner and **Pentland** in their whole research on American Sign Language (ASL) recognition use video cameras to capture the signs. Signs are formed with bare hands or with hands wearing coloured gloves [15], [16], [17]. In [14] they describe a system for the recognition of short sentences of ASL, with a vocabulary of 40 signs. Signs are modeled with four-states HMM. A single color camera is used for image recording. In the first experiment, where gloves are used, it is attained 92% word accuracy. In the second experiment a word accuracy of 99% is attained.

System Overview. The hand tracking stage of the system does not attempt a fine description of hand shape. Instead, it produces only a coarse description of hand shape, orientation and trajectory. In both two experiments the hands are tracked by their color. In both cases the resultant shape, orientation and trajectory information is input to an HMM for recognition of the signed words. Explicit segmentation on the word level is not necessary for either training or recognition. Language and context models can be applied on several different levels.

In the first method, the subject wears distinctly colored cloth gloves on each hand. To find each hand initially, the image is initially scanned until a pixel of the appropriate color is found. Each pixel checked is considered part of the hand. Then the whole hand is captured.

In the second method, the hands are tracked, based on skin tone. It has been found that all human hands have approximately the same hue and saturation, and vary primarily in their brightness. By using this information, an a priori model of skin color is built and this model is used to track the hands much as it is done in the gloved case. Since the hands have the same skin tone, "left" and "right" are simply assigned to whichever hand is leftmost and rightmost.

Feature extraction. Aside from the position of the hands, some concept of the shape of the hand and the angle of the hand relative to horizontal considered necessary. Thus, an eight element feature vector, consisting of each hand's x and y position, angle of axis of least inertia, and eccentricity of bounding ellipse, is formed.

Training an HMM network. When using HMMs to recognize strings of data such as continuous speech, cursive handwriting or ASL sentences, several methods can be used to bring context to bear in training and recognition. A simple context modeling method is **embedded training**. While initial training of the models might rely on manual segmentation, or in this case, evenly dividing the evidence among the models, embedded training trains the models in situ and allows model boundaries to shift through a probabilistic entry into the initial states of each model. Generally a sign can be affected by both the sign in front of it and the sign behind it. For phonemes in speech, this is called *"co-articulation"*. While this can confuse systems trying to recognize isolated signs, the context information can be used to aid recognition. For example, if two signs are often seen together, recognizing the two signs as one group may be beneficial.

Liang and **Ouhyoung** have worked on Taiwanese Sign Language (TSL) recognition, starting from TSL Alphabet [18] and continuing with gesture recognition ([19], [20]). In all their systems they use "dataglove" as a mean of inputting data. In [19] they define a time-varying parameter (TVP) in a stream of data, as the parameter that changes its value along time axis. A discontinuity occurs when the number of TVPs of a stream of gesture input is under a threshold. A posture recognition takes place if and only if a discontinuity is detected. Then HMMs are employed for posture recognition.

System Overview. A frame of input data is send to 50 HMMs (50 is the number of TSL postures) and the posture with the maximal evaluation resulted from the corresponding HMM is the winner. Posture analysis comprises three major modules: input module, vector quantization and HMM estimation. The input module receives a stream of gesture input, and if a discontinuity happens, the input module forwards a frame of raw data to vector quantization, solving, in that way, the end point detection problem. Gesture recognition is based on looking up the vocabularies composed of several possible paths of sequences of postures. A gesture in TSL consists of one or more postures sequentially moved or posed to some direction. Each gesture can be thought as a vocabulary in a lexicon, and a sentence is a sequence of gestures. After posture analysis, the results are decoded into several candidate postures. The gesture composition composes several possible gestures according to a lexicon. Every one of these possible gestures may contain one or more postures, that is, each candidate posture may be combined with those candidate postures in the previous one or more frames. Gesture-level match evaluates these gestures according to the probabilities of associated postures, and their corresponding probabilities in this language. The above process is defined as the *gesture model*.

In [20], the same authors extend their work by implementing the position, orientation and motion model in addition to the posture model. They implement a system with a lexicon of 250 signs and use 196 collected training sentences in TSL. The system uses HMMs for 51 fundamental postures, 6 orientations and 8 motion primitives. A sentence of gestures based on these vocabularies can be continuously recognized in real-time with an average recognition rate of 80.4%.

Assam and **Grobel** present in [24] a video-based sign language recognition system, that aims to the recognition of 262 different signs taken from the SL of the Netherlands. The work deals with isolated and connected SL recognition by using HMMs and the system achieves recognition rates up to 94% for isolated and 73% for a reduced vocabulary of connected signs.

Feature extraction. The user wears coloured cotton gloves: one with seven colours -marking the five fingers, the palm and the back of the dominant hand- and one with an eighth colour for the non-dominant hand. Information, that reflects the manual parameters of sign language, is extracted by using a threshold algorithm. The algorithm generates in/out-code for the colours of the glove and the extracted information is used to form the feature vectors.

Training. The authors train the HMMs for isolated signs first. In order to recognize the connected signs they just update the recognition algorithm: they solve the problem of detecting the transitions between signs by modelling the transitions by a separate HMM.

Vogler and Metaxas presented in [25] a framework for recognizing isolated and continuous ASL sentences from three-dimensional data. The data are obtained by using physics-based three-dimensional tracking methods and then presented as input to HMMs for recognition. To improve recognition performance, they model context-dependent HMMs and present a method of coupling three-dimensional computer vision methods and HMMs by temporally segmenting the data stream with vision methods. Then they use the geometric properties of the segments to constrain the HMM framework for recognition. They use a 53 sign vocabulary and achieve recognition rates of 95% to 98% for different sets of features in isolated recognition experiments. In continuous recognition experiments they achieved 87.7% recognition rate in the 3D context independent experiments, 89.9% in the 3D context dependent experiments and 83.6% in the 2D context dependent experiments.

System Overview. Their approach of tracking human arms consists of two parts. In the first part, the parts of a moving articulated object are identified and their shape and motion are estimated. The authors achieve this by segmenting the apparent body contour of a moving human into the constituent parts. The second part of the algorithm consists of using the extracted three-dimensional shape of the arm to track the three-dimensional position and orientation of a subject's body parts. They use three cameras placed in a mutually orthogonal configuration. At every image frame and for each body part, they derive a subset of the cameras that provide the most informative views for tracking. This active and time varying selection is based on the visibility of a part and the observability of its predicted motion from a certain camera.

Training. Once there are clearly marked boundaries between signs, HMM recognition is performed. The recognition process extracts the signal corresponding to each sign individually. It then picks the HMM that yields the maximum likelihood for that signal as the recognized sign. Training the HMMs to maximize recognition performance is also comparatively straightforward. Initially, all signs in the training set are labeled. For each sign in the dictionary, the training procedure computes the mean and covariance matrix over the data available for that sign and assigns them uniformly as the initial output probabilities to all states in the corresponding HMM. It also assigns initial transition probabilities, uniformly to the HMM's states. Unlike the initial output probabilities, the initial transition probabilities do not influence the performance of the fully trained H-MMs greatly. The training procedure then runs the Viterbi algorithm repeatedly on the training samples, so as to align the training data along the HMM's states. The aligned data are then used to estimate better output probabilities for each state individually. This alignment yields major improvements in recognition performance, because it increases the chances of the **Baum-Welch reestimation** algorithm converging to an optimal or near-optimal maximum. After constructing the HMMs, the training procedure finishes by reestimating each HMM in turn with the Baum-Welch reestimation algorithm, which maximizes $P(O|\lambda)$ locally.

The by far most challenging problem in **isolated recognition** is extracting a feature vector that optimizes recognition performance. The number of states and the topology used for the HMMs is also important. Sign language as a time-varying process lends itself naturally to a left-right model topology. Finding the optimum number of states, which depends on the frame rate and the complexity of the signs involved, is an empirical process. The same model topology for all signs has been used and has been determined experimentally that for the task in question a model with 9 states is sufficient (left-right topology). The output probabilities are single Gaussian densities with diagonal covariance matrices.

In **Continuous recognition** on the other hand, there is no silence between the signs, so the straightforward method of using silence to distinguish the signs fails. HMMs offer the compelling advantage of being able to segment the streams of signs automatically with the Viterbi algorithm.

The above described system continues **Vogler's** and **Metaxas's** work of [26], where they describe an HMM-based system for continuous ASL recognition with a vocabulary of 53 signs. Three video cameras are again used interchangeably with an electromagnetic tracking system for obtaining 3D movement parameters of the signer's arm and hand. The sentence structure is unconstrained and the number of signs within a sentence is variable. They have performed two experiments, both with 97 test sentences: one without grammar and another with incorporated bigram probabilities. Recognition accuracy ranges from 92.1% up to 95.8% depending on the grammar used.

Vogler and **Metaxas** have also presented a completely different approach to continuous, whole-sentence ASL recognition, that uses phonemes instead of whole signs as the basic units [27]. According to this model the ASL signs can be broken into movements and holds, which are both considered phonemes. This model does away with the distinction between whole signs and epenthesis movements (movements between two successive signs), that the authors have made in [26]. They train HMMs to recognize the phonemes and the epenthesis movements instead of whole signs. As the number of phonemes is limited, HMM-based training and recognition of the ASL signal becomes computationally more tractable. They experiment with a 22 word vocabulary and they achieve recognition rates of 91%.

The same authors have extended their work by using **parallel HMMs** in order to model the phonemes of ASL [28]. Modelling the phonemes properly is more difficult than in speech recognition, because speech is sequential, whereas many phonemes in ASL occur in parallel during the course of a sign. For example, in many signs both the left and right hands move in a large number of possible combinations. Attempting to capture all the possible different combinations of phonemes statistically for example, by training a HMM for each combination would be untractable for large vocabularies. The set of phonemes in ASL is limited to approximately 100 total (30 handshpes, 8 hand orientations, 20 major body locations and 40 movements). Parallel HMMs are essentially regular H-MMs that are used in parallel. Thus, they are used to model independently the processes for the left and the right hand.

Hienz, Bauer and **Kraiss** in [29] have developed a video-based continuous sign language recognition system using HMMs. The system aims to automatic recognition of sign language sentences, based on a lexicon of 52 signs of German Sign Language. A single colour video camera is used for image recording. The recognition is based on HMMs, concentrating on manual sign parameters. As an additional component, a stochastic language model is utilized, which considers uni- and bigram probabilities of single and successive signs. The system achieves an accuracy of 95% using a bigram language model.

System overview. After recording, the sequence of input images is digitized and segmented. In the next processing step features regarding size, shape and position of the fingers, hands and body of the signer are calculated. Using this information a feature vector is built that reflects the manual sign parameters. Classification is performed by using HMMs. For both, training and recognition, feature vectors must be extracted from each video frame and input into the HMM. Incorporated grammars provide additional constraints on the data and simplify the recognition process.

Feature Extraction. In this approach the person who performs the sign wears simple coloured cotton gloves. The authors have chosen different gloves for the two hands: one with seven colours - marking each finger, the palm and the back of the dominant hand - and a second glove in an eighth colour for the non-dominant hand. A threshold algorithm generates Input/Output-code for the colours of the gloves, skin, body and background. In the next processing step the size and the centres of gravity (COG) of the coloured areas are calculated and a rule-based classifier estimates the position of the shoulders and the central vertical axis of the body silhouette. Using this information a feature vector is built. This vector reflects the manual parameters of sign language, without explicitly modelling them.

Hiden Marcov Modelling. Each sign is modeled with one HMM. Signs may consist of two, three or four images, which are recorded by the video camera in sequence. For each image a feature vector is calculated. The sequence of feature vectors represents the observation sequence O. The order of visited states forms the state sequence. With Bakis topology for each HMM, the system is able to compensate different speed of signing. An initial state of a sign can only be reached from the last state of a previous model.

Training HMMs on Continuous Sign Language. Training HMMs on continuous Sign Language is very similar to training isolated signs. One of the advantages of HM modelling is that it can absorb a range of boundary information of models automatically for continuous sign language recognition. Given a specific number of observation (training) sequences, O, the result of the training are the model parameters $\lambda = (\Pi, A, B)$. These parameters are later used for the recognition procedure. Since the entire sentence HMM is trained on the entire observation sequence for the corresponding sentence, sign boundaries are considered automatically. With this kind of training, variations caused by preceding and subsequent signs are incorporated into the model parameters. The model parameters of the single sign must be reconstructed from this data afterwards. The overall training is partitioned into the following components: the estimation of the model parameters for the complete sentence, the detection of the sign boundaries and the estimation of the model parameters for single signs.

Language modeling technology has already been developed by the speech recognition community. The task of language modeling is to capture the inherent linguistic constraints in sign language in order to improve recognition accuracy. In the specific approach a stochastic grammar, known as unigram and bigram probabilities has been implemented. In a bigram model, the probability of a sign depends on the preceding sign. Such a modeling technique contains both syntactic and semantic information.

6 Conclusions

From the above presentation of different research works we can gather some conclusions, that may be proven useful for the researchers in the area.

In almost all presented works the HMMs model signs that represent whole words. However, there are few works, in which HMMs model phonemes such as in [18], [27] and [28] . In that way the latter works have common aspects with speech recognition.

Works on SL recognition have employed either "datagloves" ([14], [15], [16], [17], [23], [24], [25], [26], [27], [28], [29]) or video cameras ([18], [19], [20], [21]) as a means of data inputting. Both techniques have presented significant good results both on isolated and continuous sign language recognition. However, "datagloves" are expensive devices, whilst video cameras are not. Considering the case that a specific research work is widely used, the case of video cameras seams a more approachable solution.

References

1. Charayaphan, C., Marble, A.: Image Processing System for Interpreting Motion in American Sign Language. Journal of Biomedical Engineering, Vol 14 (1992) 419-425
2. Fels, S., Hinton, G.: Glove-Talk: A Neural Network Interface Between a Data-Glove and a speech Synthesizer. IEEE transactions on Neural Networks, Vol. 4 No. 1 (1993)
3. Waldron, M., Kim, S.: Isolated ASL Sign Recognition System for Deaf Persons. IEEE Transactions on Rehabilitation Engineering, Vol. 3, No 3 (1995)
4. Dorner, B., Hagen, V: Towards an American Sign Language Interface. Artificial Intelligence Review, Vol. 8 Iss. 2-3 (1994) 235-253
5. Holden, E., Owens, R., Roy, G.: Hand Movement Classification using an adaptive fuzzy Expert System. International Journal of Expert Systems Research and Applications, Vol. 9 Iss. 4 (1996) 465-480
6. Sagawa, H., Takeuchi, M., Ohki, M.: Methods to Describe and Recognize Sign Language Based on Gesture components Represented by Symbols and Numerical Values. Knowledge-Based Systems, Vol. 10 Iss. 5 (1998) 287-294
7. Huang, Ch., Huang, W.: Sign Language Recognition Using Model- Based Tracking and a 3D Hopfield Neural Network. Machine Vision and Applications, Vol. 10, (1998) 292-307
8. Su, M.: A fuzzy Rule-Based Approach to Spatio-Temporal Hand Gesture Recognition. IEEE Transactions on Systems, Man, and Cybernetics-Part C:Applications and Reviews, Vol. 30, No. 2, (2000) 276-281
9. Pashaloudi, V., Margaritis, K.: On Greek Sign Language Alphabet Character Recognition: using Back-Propagation Neural Networks. Proccedings of the 5th Hellenic European Research on Computer Mathematics and its Applications (HERCMA) Conference. Athens (2001)
10. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of IEEE 77(2) (1989) 257-286
11. Bengio, Y.: Marcovian Models for Sequential Data. Dept. Informatique et Researche Opérationelle. Université de Montreal Canada (1999)
12. Watson, R.: A survey of gesture recognition techniques. Technical Report TCD-CS-93-11, Department of Computer Science, Trinity

13. Cédras, C., Shah, M.: Motion-Based Recognition: A Survey. Image and Vision Computing, Vol. 13 (2) (1995) 129-155
14. Starner, T., Pentland, A.: Real-time American Sign Language recognition from video using hidden Markov models. Perceptual Computing Section Technical Report No. 375 MIT Media Lab Cambridge MA (1996)
15. Starner, T., Pentland, A.: Visual Recognition of American Sign Language Using Hidden Marcov Models. In International Workshop on Automatic Face and Gesture Recognition. Zurich Switzerland (1995)
16. Starner, T., Weaver, J., Pentland, A.: A Wearable Computer Based American Sign Language Recognizer. The First International Symposium on Wearable Computers (1997) 130-137
17. Starner, T., Weaver, J., Pentland, A.: Real-Time American Sign Language Recognition Using Desk and Wearable Computer Based Video. Pattern Analysis and Machine Intelligence (1998)
18. Liang, R.-H., Ouhyoung, M.: A Real-time Continuous Alphabetic Sign Language to Speech Conversion VR System. Computer Graphics Forum. Vol 14, No 3 (1995) C67-C77
19. Liang, R.-H., Ouhyoung, M.:A Sign Language Recognition System Using Hidden Markov Model and Context Sensitive Search. Proceedings of the ACM Symposium on Virtual Reality Software and Technology. HongKong (1996) 59-66
20. Liang, R.-H., Ouhyoung, M.: A Real-time Continuous Gesture Recognition System for Sign Language. Proceedings of the Third International Conference on Automatic Face and Gesture Recognition. Nara Japan (1998) 558-565
21. Nam, Y., Wohn, K.Y.: Recognition of Space-Time Hand-Gestures using HMM. ACM Symbosium on Virtual Reality Software and Technology. Hong Kong (1996) 51-58
22. Yamato, J., Ohya, J., Ishii, K.: Recognizing Human Action in Time-Sequential Images using Hidden Marcov Models. In Proceedings of the International Conference on Computer Vision. IEEE Press (1992) 379-385
23. Rigoll, G., Kosmala, A., Eickeler, S.: High Performance Real-Time Gesture Recognition Using Hidden Marcov Models. In Proceedings of the International Gesture Workshop. Bielefeld Germany (1997) 69-80
24. Assan, M., Grobel K.: Video-Based Sign Language Recognition using Hidden Marcov Models. In Proceedings of the International Gesture Workshop. Bielefeld Germany (1997) 97-109
25. Vogler, C., Metaxas, D.: ASL Recognition based on a Coupling Between HMMs and 3D Motion Analysis. In Proceedings of the International Conference on Computer Vision. Mumbai India (1998) 363-369
26. Vogler, C., Metaxas, D.: Adapting HMMs for ASL Recognition by using Three-Dimensional Computer Vision Methods. In Proceedings of the IEEE International Conference on Systems Man and Cybernetics. Orlando FL (1997) 156-161
27. Vogler, C., Metaxas, D.: Toward Scalability in ASL Recognition: Breaking Down Signs into Phonemes. Gesture Workshop. Gif sur Yvette, France (1999)
28. Vogler, C., Metaxas, D.: Parallel Hidden Marcov Models for American Sign Language Recognition. Gesture Workshop. Corfu, Greece (1999)
29. Hienz, H., Bauer, B., Kraiss, K.-F.: HMM-Based Continuous Sign Language Recognition Using Stochastic Grammars. Gesture Workshop. Gif sur Yvette, France (1999)