Feature Extraction and Sign Recognition for Greek Sign Language

Vassilia Pashaloudi
and Konstantinos Margaritis
Parallel and Distributed Processing Laboratory
Department of Applied Informatics
University of Macedonia
156 Egnatia str., P.O. Box 1591, 54006, Thessaloniki, Greece
Email: {vpasx, kmarg} @uom.gr

***Abstract*— The work presented in this paper aims at developing a system that could recognize both isolated and continuous Greek Sign Language (GSL) sentences.**

A feature extraction method, that uses geometric properties of the morph of the hand, is used to derive descriptive features from each image, representing a GSL alphabet letter. The feature vectors produced by the method are given in sequences as input to Hidden Markov Models (HMMs). We distinguish four cases in the expreriments that have been performed: *isolated letter* recognition, recognition of *sequences of letters*, *isolated word* recognition and *connected word recognition*. In the former two classes of experiments, HMMs representing letters have been built, in order to confirm feature extraction method's eligibility; in the latter two classes whole-word HMMs have been built for each word out of a 26 sign vocabulary. Recognition rates reached 95.8% and 90.5% for the first and second class of experiments, respectively, while, in the experiments using whole-word HMMs, we achieved recognition rates of 97.4% for isolated, and 86.2% for connected word recognition.

I. Introduction

Sign Languages (SLs) are the basic means of communication between hearing impaired people. Static morphs of the hands, called *postures*, together with hand movements, called *gestures*, and facial expressions form words and sentences in SLs, corresponding to words and sentences in spoken languages. Researchers have shown an increasing interest in SL recognition and have approached the problem through different ways. Though facial expressions play an important role in SL, in the research works that have been developed, they are excluded from the area of interest, as their presence would complicate the already difficult problem. Thus, interest is concentrated on hands forming *signs*.

A *sign* usually represents a whole word in SL. It can be constituted of either a *posture*, a *gesture* or a sequence of these elements. In special cases, where an unusual word or a proper name must be represented in SL, finger spelling is used. In the latter case words are formed as sequences of postures representing alphabet letters.

Sequences of signs form SL sentences and the recognition process involves distinguishing sequential signs from each other. When there are artificial pauses between signs the recognition process is called *Isolated SL recognition*, while when there is continuous flow of signs it is called *Continuous SL recognition*.

SL recognition covers two areas of interest. The first one concerns the mean througn which the computer understands the sign and extracts useful data from it; the second regards to the method used in building a reliable recognition system that exploits the information obtained by the sign.

In the first sense, recognition systems can be divided to systems that use electromechanical devices, called *glove-based* systems and systems that exploit machine vision and image processing techniques, called *visual-based* systems. Systems of the first class use devices that measure the different gesture parameters, such as hand's position, angle and location of the fingertips. *Waldron* and *Kim* [3] used such a system in order to recognize Isolated American Sign Language (ASL) signs with Neural Networks. *Holden* and *Owens*, in [4], proposed a hand gesture recognition system, that either uses a VR glove or vision techniques to obtain data. "Datagloves" have been also used by Liang and Ouhyoung in [8], [9], [10], when working on Taiwanese Sign Language. In [10] they presented a Hidden Markov Model based system for 250 signs formed by 51 fundamental postures, 6 orientations and 8 motion primatives; they achieve recognition rates of 80.4%. However, such systems are quite expensive for wide use and, moreover, inconvenient for the user, as they require gloves and wires. *Visual-based* systems have been used by many researchers to increase naturalness in user's movement ([11]- [13], [17]).

The second area of interest in SL recognition regards the method used in building the recognition system, that exploits the information obtained by the first step. Neural Networks ([2], [5]), Image Processing Algorithms ([1]), Fuzzy Systems ([4]), Adaptive Neuro-Fuzzy Inference Systems ([20]) or Hidden Markov Models, have been used for that purpose.

Hidden Markov Models (HMMs) are statistical models that have been successfully used in speech and character recognition; thus, they form an attractive choise for sign language recognition, since SL signs appear sequentially in time axis. *Vogler* and *Metaxas* in [14] described an HMM-based system for continuous ASL recognition over a 53-sign vocabulary. Three video cameras are used interchangeably with an electromagnetic tracking system for obtaining 3D movement parameters of the signer's arm and hand. *Hienz et al.* developed a video-based continuous SL recognition system for German Sign Language, using HMMs. In the feature extracting phase they calculate parameters regarding size, shape and position of the fingers and hands, while the signer wears cotton gloves with colour markers. Recognition rate exceeds 95%, working on a 52 sign vocabulary. *Starner*, *Pentland* and *Weaver* in [13] presented a visual-based approach for the recognition of ASL sentences. A small feature vector, constituing of parameters regarding position and orientation of the hand, is constructed from each image. Signs are formed either by bare hands or hands wearing colloured cotton gloves and feature vectors are fed to HMMs for recognition. An accuracy of 97% per word on a 40 word lexicon has been reached.

In our work we concentrate on Greek Sign Language (GSL) recognition through the use of Hidden Markov Models. Signs are formed as continuous flows of GSL letters. Recognition is performed on isolated signs or on sequences of them constituing sentences, with no marked boundaries between subsequent signs. The terms "sign" and "word" are used interchangeably in our work.

This paper is organized into five sections. In section II, we give a summary of HMMs theory and briefly discuss their use in SL recognition. The proposed feature extraction method and the HMM system, that we have implemented, are presented in section III. In section IV, we specify the way data collection is implemented and give experimental results for isolated and continuous recognition over a 26-sign vocabulary from the Greek Sign Language. Finally, in section V we give a few concluding remarks and refer to our future aims.

II. HIDDEN MARKOV MODELS

A. Definition of Hidden Markov Models

Hidden Markov Models are a state-based statistical model especially suitable for modeling a signal over time. They have been successfully used in speech recognition and recently in handwriting, gesture and sign language recognition. We now give a summary of the basic theory of HMMs. Detailed theory is presented in [6]- [7].

Given a set of N states s_i we assume that a system can be in one of these states at each time interval; 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} from state s_i to state s_j only depends on the preceding states, we call this process a Markov chain. The further assumption, that the actual transition only depends on a very preceding state leads to a first order Markov 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 double stochastic process is called a Hidden Markov 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 state-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 state-transition probabilities π_i.

Given the definition of HMMs, there are three basic problems to be solved [6]:

- *The evaluation problem:* Given the observation sequence $O = O_1, O_2, \ldots, O_T$, compute the probability $P(O|\lambda)$, that a HMM $\lambda = (A, B, \pi)$ 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 and can be solved with the *Forward-Backward* algorithm.
- *The decoding problem:* given the parameters of a HMM, λ, and an observation sequence, $O = O_1, O_2, \ldots, O_T$, we have to find the state sequence, $S = s_1, s_2, \ldots, s_T$, that emits with a high probability the same symbol vectors as observed from the signal. A formal technique for finding this best state sequence is called the *Viterbi algorithm*.
- *The estimation problem:* Adjust the parameters of an HMM λ such that they maximize $P(O|\lambda)$ for one or more observation sequences 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. Viterbi training gives a solution, by iteratively adjusting the parameters A, B and π. In every iteration, the most likely path through an HMM is calculated. This path gives a new assignment of observation vectors O_t to the states s_j.

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

B. HMMs in Sign Language Recognition

HMMs can be successfully used in processing both speech and two-dimensional sign data, because their state-based nature enables them to capture variations in the duration of signs, by remaining in the same state for several time frames.

Having solved the fundamental problem of selecting good features from the images, the next problem we have to cope with is designing the HMM system, with the optimal way, in the sense of achieving the highest possible recognition results.

The recognition process can be examined from two different aspects: according to the kind of element it attempts to recognize (phoneme or word) and according to whether there are artificial pauses between signs or not.

1) Word-based and Phoneme-based recognition in SL: According to the first aspect, there are **whole-word** or *word-based* systems, where separate HMMs are trained for each word, and **phoneme-based** systems, where separate HMMs are trained for each phoneme. In either case HMMs are trained to yield the maximum probability for the signal representing their respective word or phoneme. The **phoneme** in SL could be the **"morph"** of the hand, that is a posture among a specific predefined set of postures, such as those appearing in Liang-Ouhyoung model [9]; or it could be considered as a **movement** or a **hold** as Vogler and Metaxas did in [15] and [16], in order to break words-signs of American Sign Language into phonemes.

The main advantage in the phoneme-based recognition is that the number of phonemes is limited in any language, including SL, wherever they have been itemized, as opposed to the unlimited number of words that can be built from phonemes. Thus, for large-scale applications the most effective and commonly used method is the phoneme-based recognition, whilst for small-scale applications **whole-word** training is more appropriate.

2) Isolated and Continuous recognition in SL: Isolated Recognition assumes that there are clearly marked boundaries between signs. Such a boundary could be "silence", that is, a brief resting phase between signs. For each unknown observation sequence representing a sign calculation of model likelihoods for all possible models and selection of the model, whose likelihood is the highest, is performed [6].

In Continuous Recognition the recognition problem is to find the optimum sequence of word models that best matches, in a maximum likelihood sense, an unknown connected word string. Since there are no artificial pauses between signs, the straightforward method of using pauses to distinguish signs fails. Furthermore, between two subsequent signs there is usually a connective sign, that must be ignored. This forms the "co-articulation problem" in SL recognition. However, HMMs offer the compelling advantage of being able to segment the streams of signs automatically with the Viterbi algorithm.

III. The Recognition System

In GSL the only morphs, that have been encountered and itemized without dispute, are the morphs that represent the 24 alphabet letters. Those morphs can be used in sequences to form proper names or specialized words that are not common in GSL. For the rest of the words in GSL each word is formed by moving one or both hands and can be considered as an image sequence through time.

In our approach we use sequences of GSL image-letters to form a word. This could straightforward lead to the recognition of proper names. The sequences are constituted of the letters used, when "spelling" the word. Thus, sequences corresponding to different words might have different lengths. Sentences are formed as continuous flows of GSL image-letters, without pauses between signs.

All the GSL alphabet letters are presented in figure 1.

Fig. 1. Morphs of the Greek Sign Language alphabet letters

From each letter we extract a feature vector by using the method described in the next chapter. In order to be sure that, the feature vector we selected to use, is appropriate for our recognition purposes, we performed experiments over isolated letters.

In this class of experiments, called *letter-case*, we built an HMM for each one of the 24 letter-morphs, trained and tested it. In Isolated recognition experiments for this case, tests have been performed over a number of letters never presented in the training phase. In Continuous recognition, tests have been performed over sequences of letters representing spelled words. The results were encouraging, so we proceeded to the second stage of building whole-word HMMs.

In this second class of experiments, the *word-case*, an HMM for each one of the words is constructed. Each HMM is trained over a number of examples of the word it represents, so that in the recognition phase it could give the maximum probability among all the other HMMs for the specific word. For these new HMMs the training examples are sequences of feature vectors, extracted from the letters participating in a word, by using the specific feature extracting method. When speaking of sentences the sequences of feature vectors are extracted from the letters participating in each word of the sentence. Unseen feature vector sequences are used in the testing phase, in order to check system's performance.

System's recognition ability can be improved through the use of a *grammar*. A *grammar* is a collection of rules that specifies the allowed sequences of words. The words in a vocabulary can be divided to a number of classes (i.e. pronoun, noun, verb) and the grammar rules restrain the flow of the words in a sentence.

The structure of our system is presented in figure 2.

The procedure described above can be followed in recognizing image sequences, captured at sequential time intervals, during the formation of a sign with the classic SL way. The

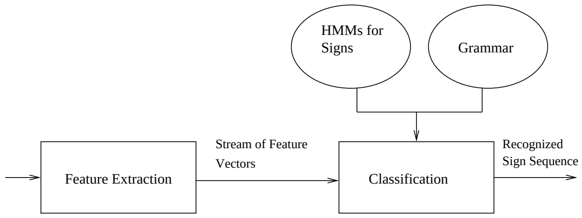

Fig. 2. Components of the continuous sign language recognition system

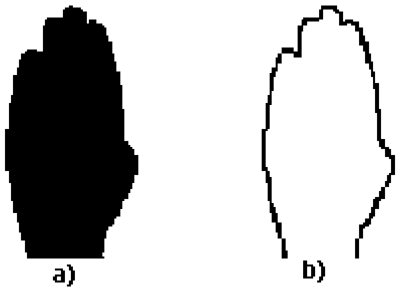

Fig. 3. a) GSL letter B (BETA) in compact form b) Border points of the letter B

only difference in this case is that an image sequence consists of images that do not represent letters.

A. Feature Extraction

Selecting good object features in any object recognition system plays significant role in system's performance. It is obvious that the whole image can not be used as a feature or features, because this would demand high calculation complexity for whatever recognition system it would fed. Furthermore, an image carries much useless information, which we desire to exclude from the feature vector. Being able to extract the usefull and semantic portion of information from an image, means reducing system's complexity and simultaneously achieving good recognition results.

In our approach we use monochrome bitmap images of the GSL alphabet letters, with size 100x100 pixels, as the initial data. Each bitmap image is transformed in a 2-dimensional array of 1's and 0's, placed in the corresponding positions of "black" and "white" pixels, respectively.

In our previous work [18], in order to extract a feature vector from each image, we were counting the number of "black" pixels in each scan line of the bitmap image; these numbers formed the feature vector. We now use a different approach, which comes close to what Al-Jarrah and Halawani did for posture recognition in Arabic Sign Language [20].

The method involves finding the centre of mass and the orientation of the morph representing a GSL letter, according to *Jain* [19]; we also retrieve the boundary points of the morph.

Our aim is to calculate the lengths of vectors that originate from the center of mass and end up to the border. Obviously, it is not necessary to use the whole number of border points, but only those that bare important information, These lie in the fingertips area.

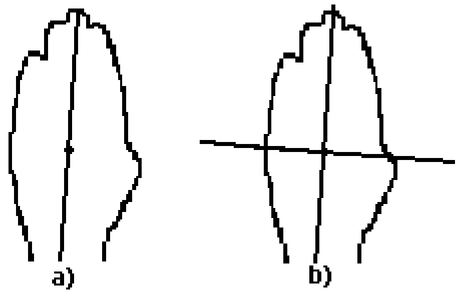

Fig. 4. a)The center of mass, denoted by a point, and the vector direction for the letter B. b) The axes formed by the vector direction and the vertical axis, passing through the center of mass.

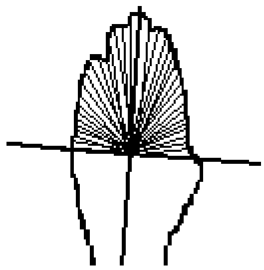

Fig. 5. The vectors whose lengths form the feature vector for the letter B.

Let the center of mass be specified by $(\overline{m}, \overline{n})$ and the morph orientation by ϑ. If we think of the center of mass as the origin in a two-dimensional axis system and the morph direction as the y-axis, then, without losing much information, we can consider as the most important parts of the morph the ones that are found in the 1st and 2nd quadrants. Thus, we are dealing with vectors, whose direction, θ, lies in the interval:

$$\vartheta - 90 \leq \theta \leq \vartheta + 90 \tag{1}$$

where ϑ is the morph orientation.

Moreover, we do not use the total number of border points lying in this area, but only those, whose vector's direction coincides to $\pm 5, \pm 10, \pm 15, ... \pm 90$ degrees relatively to the morph direction. Thus, we use only 37 points and a feature vector becomes of the form

$$f = [l_1, l_2, ..., l_{37}] \tag{2}$$

For scale-invariability we normalize the lengths into the range 0 to 100, by dividing all the lengths by the maximum vector length and then multiplying by 100. If the image is scaled, all vector lengths will be scaled by a certain factor.

The selected features also satisfy the translation and rotation-invariant property. When the position of a morph changes the lengths of the vectors do not change. Furthermore, if a gesture is rotated, its direction will change accordingly.

Feature vectors are finally transformed to Hidden Markov Model Toolkit (HTK) format, which has been used for all modeling and training tasks.

B. System Design

1) Building HMMs: In applications of the kind the number of states and the topology used for the HMMs is important.

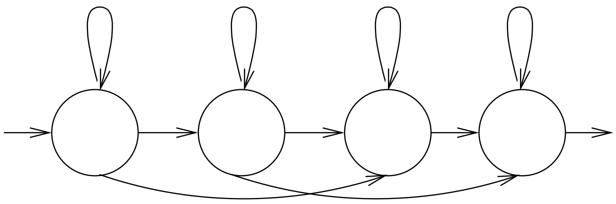

Fig. 6. Left-Right model with 4 states

Sign language as a time-varying process lends itself naturally to a left-right model topology, which we also used here. The initial topology for an HMM can be determined by estimating how many different states are involved in specifying a sign. Finding the optimum number of states, which depends on the frame rate and the complexity of the signs involved, is an empirical process. The idea used in our task is to let the number of states correspond roughly to the average number of observations in a sign. In this manner each state corresponds to an observation interval. Attentive experiments showed that we can achieve better results by tailoring different topologies for "small" and "large" signs. Thus we used 6 different topologies of 3, 4, 5, 6, 7 and 8 states and issue the signs among them, according to the number of letters they are constituted of. An example of a four-state HMM with skip transitions is presented in figure 6. The output probabilities are single Gaussian densities with diagonal covariance matrices.

2) Syntax Used: For recognition, HTK's Viterbi recognizer is used with a **grammar**, according to GSL syntax:

[pronoun] noun {adjective} verb

where square brackets [ ] denote optional selection of an item and braces { } denote zero or more repetitions of an item. We do not use a **strong** grammar, though the choice would reduce the error rates. We chose this option in order to have a syntax closer to the natural way of "speaking" with GSL.

IV. Experiments

A. Tools used

We performed isolated and continuous GSL recognition experiments by using Entropic's Hidden Markov Model Toolkit (HTK) Version 3.1 for all HMM modelling and training tasks. For the result analysis the HTK result analysis tool, **HResults**, has been used.

B. Definitions

In the tables that we are going to present below, the first line gives the sentence-level accuracy based on the total number of files which are identical to the **transcription** files. A **transcription** file is a file that HTK produces as the result of the recognition process for a specific test file. The second line is the word accuracy based on Dynamic Programming matches between the label files and the transcriptions [21]. *H* is the number of correct labels, *D* is the number of deletions, *S* is the number of substitutions, *I* is the number of insertions and *N* is the total number of labels in the defining transcription files. The percentage number of labels correctly recognized is given by

$$\%Correct = \frac{H}{N}x100 \quad (3)$$

The accuracy measure is calculated by subtracting the number of insertion errors from the number of correct labels and dividing by the total number of signs.

$$\%Correct = \frac{H-I}{N}x100 \quad (4)$$

TABLE I

Single Letter Recognition Results

	% Correct	% Accuracy	Numerical Results
SENT	95.83		H=23, S=1, N=24
WORD	95.83	95.83	H=23, D=0, S=1, I=0, N=24

TABLE II

Recognition Results for Sequences of Letters

	% Correct	% Accuracy	Numerical Results
SENT	90.59		H=77, S=8, N=85
WORD	97.49	97.49	H=428, D=0, S=11 I=0, N=439

C. Single Letter Recognition

1) Data Collection for HMMs: For each one of the 24 letters we built an one-state HMM, with non-emitting entry and exit states attached to it. Images of letters, that differ from each other in morph, as if they are formed by different users, are used as training examples. Each HMM is trained over 11 images and tested over examples that have never been presented to the system during the training phase.

2) Results: The recognition results for single letter recognition case are presented in table I.

D. Recognition of Sequences of Letters

1) Data Collection for HMMs: The training examples are formed of feature vectors representing sequences of letters in spelled words. A total of 392 examples have been collected. 307 examples have been used for training purposes and 85 examples for testing.

2) Results: The recognition results for sequences of letters are presented in table II.

E. Isolated Word Recognition

1) Data Collection for HMMs: The sign vocabulary that we used is listed in table III. It comprises 26 words.

The goal in choosing the vocabulary was to be able to express sentences that could have occured in a natural conversation.

Since the rotation and scale-invariability are satisfied by the feature extraction method we have described above, we only need to satisfy the signer independency. To achieve this aim we use different morphs of the GSL letters, as if they are

TABLE III
THE COMPLETE 26 SIGN VOCABULARY

pronouns	I, you, he, we, you (plural), they
nouns	car, train, airplain, dog, cat, house, horse, bicycle
verbs	want, don'twant, have, don'thave, love, hate
adjectives	black, white, red, yellow, small, big

TABLE IV
ISOLATED WORD RECOGNITION RESULTS

	% Correct	% Accuracy	Numerical Results
SENT	97.44		H=76, S=2, N=78
WORD	97.44	97.44	H=76, D=0, S=2, I=0, N=78

formed by different signers. We have built a large database of all the GSL letters, where each letter is represented by multiple nonidentical images. We form the training examples for each word by retrieving randomly letter-morphs, from the above described database. In that way, we enhance systems' tolerance to different signers.

For the isolated recognition case, we use training files, each one of which holds a single example of a word, with no leading or trailing silence. Each sign has 6-8 examples available for the training set, and 3 examples available for the test set. Thus a total of more than 240 examples has been collected over a range of 26 signs. The length of the words ranges between 3-8 image-letters.

2) Results: The analysis of the experiments for the isolated word recognition case is presented in table IV. We can see that for this case the recognition rate is notably high.

F. Connected Word Recognition.

1) Data Collection for HMMS: The training examples in this case are sequences of feature vectors that stand for the sequences of the words constituing a sentence. In continuous recognition we do not have pauses between words in the collected sentences, neither do we have to deal with the "co-articulation problem", since we use only GSL letters in sequences, in order to form sentences.

We have collected up to now 314 continuous GSL sentences, each between 2 and 5 signs long, with a total of approximately 1250 signs. Each sign was between 3-8 frames long, as in the isolated recognition case. The only constraints on the order and occurence of signs were those dictated by the syntax of GSL, which has been previously described. We divided the set of the collected sentences in two subsets: the training set, consisting of 234 sentences and the test set, consisting of the rest 80 sentences. The 80 test sentences were never used in the training process. The task is to correctly recognize the words in the given sentence in order and without inserting any additional words. Error and accuracy will be measured as in the continuous speech recognition literature, incorporating substitution, insertion and deletion errors.

TABLE V
CONNECTED WORD RECOGNITION RESULTS

	% Correct	% Accuracy	Numerical Results
SENT	86.25		H=69, S=11, N=80
WORD	96.04	96.04	H=267, D=1, S=10, I=0, N=278

2) Results: In continuous recognition we used the method of **embedded training**, in order to achieve better recognition results, since we had also performed experiments without using the method and had achieved lower recognition rates. The results of the experiments performed for the connected word recognition case with the use of embedded training are presented in table V.

The 86.25% recognition rate can be considered quite good, since the number of training sentences can be augmented and, thus, the recognition rate can get even higher.

V. CONCLUSION

In this paper we present a system designed for the purpose of recognition of GSL words and sentences. Having test the appropriateness of the chosen feature extraction method, we proceeded to system implementation. For HMM building, we chose the whole-word approach. Thus, we trained 26 HMMs, each one representing a GSL word.

We used monochrome bitmap images of the GSL alphabet letters in order to form GSL words and extracted a feature vector from each image, by using geometric features of the hand morph. The feature vector produced from each image comprises 37 elements, a quite small number, that enhances systems' performance. The feature vector sequences were used to train the 26 HMMs.

We have worked on isolated and continuous GSL recognition. Through the use of Hidden Markov Models, low error rates were achieved.

The sentences that we collected up to now and used as training sentences gave high recognition rates. In the continuous recognition case we used sentences, where there is not any kind of artificial pauses between the words, but only sequences of GSL letters form the sentences.

Our future aim is to perform experiments over sequences of unclassified morphs of the hand, which will be used to form GSL words and sentences. The same procedure can be followed in these experiments, too, and we believe that we can achieve equally high recognition results.

REFERENCES

[1] C. Charayaphan and A. Marble, "Image Processing System for Interpreting Motion in American Sign Language", *Journal of Biomedical Engineering*, Vol. 14, pp. 419-425, 1992.

[2] S. Fels and G. Hinton, "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] M. Waldron and S. Kim, "Isolated ASL Sign Recognition System for Deaf Persons", *IEEE Transactions on Rehabilitation Engineering*, Vol. 3, No. 3, 1995.

[4] E. Holden, R. Owens and G. Roy, “Hand Movement Classification using an adaptive Fuzzy Expert System”, *International Journal of Expert Systems Research and Applications*, Vol. 9, Iss. 4, pp. 465-480, 1996.

[5] V. Pashaloudi and K. Margaritis, “On Greek Sign Language Alphabet Character Recognition: using Back-Propagation Neural Networks”, *Procedings of the 5th Hellenic European Research on Computer Mathematics and its Applications (HERCMA) Conference*, Athens, 2001.

[6] L.R. Rabiner, “A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition”, *Proceedings of IEEE 77(2)*, pp. 257-286, 1989.

[7] Y. Bengio, “Markovian Models for Sequential Data”, Dept. Informatique et Researche Opérationelle, Université de Montreal Canada, 1999.

[8] R.-H. Liang and M. Ouhyoung, “A Real-time Continuous Alphabetic Sign Language to Speech Conversion VR System”, *Computer Graphics Forum*, Vol. 14, No. 3, pp. C67-C77, 1995.

[9] R.-H. Liang and M. Ouhyoung, “A Sign Language Recognition System Using Hidden Markov Model and Context Sensitive Search”, *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pp. 59-66, HongKong, 1996.

[10] R.-H. Liang and M. Ouhyoung, “A Real-time Continuous Gesture Recognition System for Sign Language”, *Proceedings of the Third International Conference on Automatic Face and Gesture Recognition*, pp. 558-565, Nara, Japan, 1998.

[11] T. Starner and A. Pentland, “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.

[12] T. Starner, J. Weaver and A. Pentland, “A Wearable Computer Based American Sign Language Recognizer”, *The First International Symposium on Wearable Computers*, pp. 130-137, 1997.

[13] T. Starner, J. Weaver and A. Pentland, “Real-Time American Sign Language Recognition Using Desk and Wearable Computer Based Video”, *Pattern Analysis and Machine Intelligence*, 1998.

[14] C. Vogler and D. Metaxas, “ASL Recognition based on a Coupling Between HMMs and 3D Motion Analysis”, *Proceedings of the International Conference on Computer Vision*, pp. 363-369, Mumbai, India, 1998.

[15] C. Vogler and D. Metaxas, “Towards Scalability in ASL Recognition: Breaking Down Signs into Phonemes”, *Gesture Workshop*, Gif sur Yvette, France, 1999.

[16] C. Vogler and D. Metaxas, “Parallel Hidden Markov Models for American Sign Language Recognition”, *Gesture Workshop*, Corfu, Greece, 1999.

[17] H. Hienz, B. Bauer and K.-F Kraiss, “HMM-Based Continuous Sign Language Recognition Using Stochastic Grammars”, *Gesture Workshop*, Gif sur Yvette, France, 1999.

[18] V. Pashaloudi and K. Margaritis, “Hidden Markov Models for Greek Sign Language Recognition”, *Proceedings of 2nd WSEAS International Conference on Speech, Signal and Image Processing (ICOSSIP)*, Skiathos, Greece, 2002.

[19] A.K. Jain, *Fundamendals of Digital Image Processing*, Englewood Cliffs : Prentice Hall, NJ, 1989.

[20] O. Al-Jarrah and A. Halawani, “Recognition of gestures in Arabic Sign Language using neuro-fuzzy systems”, *Artificial Intelligence 133*, pp. 117-138, 2001.

[21] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev and Ph. Woodland, *The HTK Book*.