Sign Language Recognition using Moment-Based Size Functions

M. Handouyahia * D. Ziou † S. Wang †

*Input Output Division
Statistics Canada, R. H. Coats
Ottawa, Ontario, Canada, K1A OT6
handmoh@statcan.ca

†Dépt. de Mathématiques et d'Informatique
Université de Sherbrooke
Sherbrooke, Québec, Canada, J1K 2R1
{ziou, wang}@dmi.usherb.ca

Abstract

This paper presents a system for the recognition of sign language based on a theory of shape representation using size functions proposed by P. Frosini [5]. Our system consists of three modules: feature extraction, sign representation and sign recognition. The first performs an edge detection operation, the second uses size functions and inertia moments to represent hand signs, and the last uses a neural network to recognize hand gestures. Sign representation is an important step which we will deal with. Unlike previous work [15, 16], a new approach to the representation of hand gestures is proposed, based on size functions. Each sign is represented by means of a feature vector computed from a new pair of moment-based size functions. The work reported here indicates that moment-based size functions can be effectively used for the recognition of sign language even in the presence of shape changes due to differences in hands, position, style of signing, and viewpoint.

1 Introduction

Gesture (or sign language) is widely used in the deaf community. In the foreseeable future, gesture inputs will be widely applied for human-computer interface. R. Watson [19] presented a review of the most recent work related to hand gesture interface techniques: the glove-based techniques, vision-based techniques and the analysis of drawing gestures. We are interested in the vision-based technique, which is a natural way of constructing a human-computer interface.

Sign language consists of static and dynamic hand gestures. Static gestures are characterized by hand posture as determined by a particular finger-thumb-palm configuration, number of fingers, finger orientation and position, and finally palm orientation; dynamic ones characterized by hand movements. Study of this subject has a number of potential applications in machine vision and multimedia technology [4]. We are interested in posture recognition, especially the sign language alphabet. Signs may be different when made by different people, or even by the same person at different times. Changes in the apparent shape of signs may be due to difference in hands, style of signing, viewpoint, and the image acquisition system. Figure 1 illustrates images of 25 signs in the International Alphabet Sign Language (ISL) given to us by C. Uras and used in the paper.

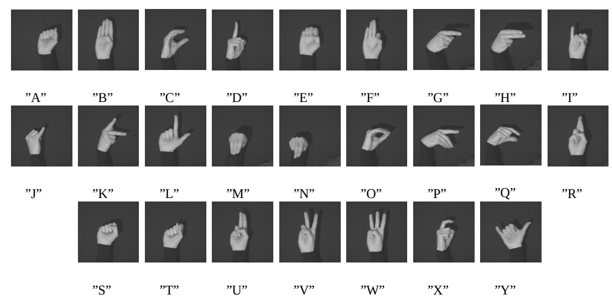


Figure 1: *Grey level images of the alphabet of the International Sign Language from "A" to "Y" given to us by C. Uras. The "Z" sign cannot be performed statically and was not considered in the current work.*

In this paper, we are interested first in studying the properties of existing methods for 2D shape representation and recognition, proposed in previous work; we concentrate on possible Pattern Recognition techniques that could be used to solve the problem of posture recognition. We show that generally, these techniques are unlikely to be appropriate for sign language. Then we propose the use of a new object representation method called *size functions* for understanding the ISL alphabet. First, size functions that capture the shape from the apparent outline of the various signs are presented. Representation of signs can be achieved simply by using a pair of size functions based on each sign's principal axis of inertia, rather than the families used in previous work done by C. Uras and A. Verri [15, 16]. Then, a training set of feature vectors computed from the proposed pair of size functions is built from real images. Finally, the recognition

Table 1: *The properties of sign representation and recognition of several existant methods.* [0]*American Sign Language.* [1]*Electronic dispotifs (PowerGlove, DataGlove et CyberGlove).* [2] *Japanese Sign Language.* [3]*International Sign Language.* [4]*The type of vocabular is pre-defined and has any configuration.*

Authors/ Properties.	Size of Vocabulary.	Type of Vocabulary.	Capture.	Representation.	Recognition.	Succesv rate %
Gourley	26	ASL[0]	Elect.[1]	Templates	Perceptron Neural Network	95
Harling	5	ASL[0]	Elect.[1]	Templates	Perceptron Neural Network	96
Murkami et al.	42	JSL[2]	Elect.[1]	Templates	Perceptron Neural Network	98
Takahashi et al.	46	JSL[2]	Elect.[1]	Joint and orientation coding	Matching Templates	65
Gao	13	D. Set[4]	Camera	Convex/Concav coding	BackPropagation Neural Network	80
Uras et al.	25	ISL[3]	Camera	First size functions familly	K-Nearest Neighbor	85
Uras et al.	25	ISL[3]	Camera	Second size functions familly	K-Nearest Neighbor	86
Freeman	15	D. Set[4]	Camera	Orientation Histograms	K-Nearest Neighbor	75
Our method	25	ISL[3]	Camera	Momennt-Based Size Functions	Perceptron Neural Network	90

stage is performed using a neural network. This new interpretation allowed us to develop an improved version of systems recently proposed for ISL recognition (Fig. 5). In the next section, related work particulary on shape representation techniques, is briefly reviewed. Then, size functions are introduced and illustrated through a simple example in Section 3. Section 4 describes the representation of signs which can be obtained by a pair of size functions. Finally, the implementation and the experimental results are illustrated and summarized in Section 5.

2 Related Work

Attempts at machine sign language recognition began to appear in the literature over ten years ago. Table 1 shows several methods used for posture recognition.

The systems proposed are based om either instrument gloves or computer vision. Instrument glove techniques [14, 12, 7, 10] use a variety of sensors to provide information about hand shape, position and orientation. This captures the richness of hand gestures and make it simple enough to recognize postures efficiently. However, the user holds some hardware which can be awkward and uncomfortable. Furthermore, the cost of Datagloves is still high. We are interested in approaches based on computer vision and used for posture recognition, this removes the need for intrusive cabling and makes tracking multiple objects feasible. Gao [6] used a new chain coding-based representation and Back Propagation Neural network to achieve the recognition of thirteen pre-defined hand postures. Freeman [3] describes a vision-based system to recognize fifteen postures using orientation histograms as a representation method and the rule of k-nearest neighbor to achieve the recognition.

More recently a new framework for the analysis of visual shapes has been proposed by P. Frosini [5]. In this framework a shape is represented in terms of integer-valued functions of two real variables, called *size functions*. In his recent work, Uras [15] used a family of size functions to represent the sign language alphabet; the recognition stage was performed using the k-nearest neighbor rule.

The methods presented in Table 1 are difficult to compare for two main reasons. First, they were not all tested in the same manner using the same gesture set; and second, they do not all use the same set of postures. Sign recognition from static grey-level images requires a system that is tolerant to image shifting, scaling, rotation, degradation, and deformation. In what follows, we are interested in the study of existing features according to these criteria. Furthermore, we need features that provide enough information to distinguish between various signs and are computationally inexpensive.

Table 2: *Feature evaluation: (1) Scale invariance. (2) Translation invariance. (3) Rotation invariance. (4) Invariance to lightning changes. (5) Robustness to the number of fingers. (6) Computational complexity. (7) Completeness: We are not including this criteria in this table because, it is difficult to compare the features with regarding to the completeness.*

Selected feature/Criteria	(1)	(2)	(3)	(4)	(5)	(6)
Basic chain code	No	Yes	No	No	No	Yes
Convex-Concav coding	Yes	Yes	Yes	No	Yes	Yes
Fourier Desc.	No	No	No	No	No	Yes
Hu Invariant moments	Yes	Yes	Yes	No	No	No
Alt Invariant moments	Yes	Yes	No	No	No	No
Principal axes	Yes	Yes	No	No	No	Yes
Grey level histogram	No	Yes	Yes	No	No	Yes
Hist. of local orientation	Yes	Yes	No	No	No	Yes
Size Functions	Yes	Yes	No	Yes	Yes	No
Our method	Yes	Yes	No	Yes	Yes	Yes

To summarize, a visual inspection of different hand signs suggests features which satisfy the following conditions:

1. Scale invariance: The signs can be executed by different persons with different hand size.
2. Translation invariance: The location of the hand sign in the image can change.
3. Slight rotation invariance over the image plan: The orientation of the apparent viewed sign can be slightly different from one person to another. However, we suggest a representation which is sensitive to the orientation of signs. For example, if we turn the sign "D" through an angle of 90^o, we will obtain an unknown sign.
4. Invariance to lighting changes: The illumination conditions of the scene can change.
5. Robustness to the combination states of all parts (fingers) of the hand: Features must be sensitive to the number of fingers and their position.
6. Computational complexity: The representation must be fast to compute.
7. Completeness: It must be sufficient to distinguish between different signs.

Table 2 shows the properties of features which have been widely used in pattern recognition and image analysis, including size functions based representation [16] and our method: (1) Basic chain code and (2) Convex-Concav chain coding [6], (3) Fourier descriptor [9], (4) Invariant moments of Hu [11], (5) Invariant moments of Alt [2, 1], (6) Principal axes of inertia [8], (7) Grey level histogram, and (8) Histogram of local orientation [3]. As we can see from Table 2, these techniques are unlikely to be appropriate for the representation of sign language with regard to the criteria fixed at the begining of this section. However, we chose principal axes of inertia as a basic feature in building a new representation model around a mathematical tool called size functions. Furthermore, as we can see in Table 2, principal axes are sensitive to sign orientation. Before presenting this new model, in the next section we will describe size functions.

3 Size Functions

In this section we will introduce the main idea of size functions using an example. Let us assume that Fig.2(a) reproduces the graph of a function φ, named the *measuring function*, obtained as the result of some measurement on any sign. Let us evaluate the size function l_φ at a specific point (x, y) of the real plane, where $x < y$.

The shaded regions of Fig.2(b) and (c) identify the parts of the graph with $\varphi \le x$ and $\varphi \le y$ respectively. The superposition of Fig.2(b) and (c) is displayed in Fig.2(d). The darker regions of Fig.2(d) clearly correspond to the intersection between the parts of the graph with $\varphi \le x$ and $\varphi \le y$. The size function l_φ (see Fig.2(d)) can be defined by the number of distinct light regions which have a segment in common with at least one dark region. In fact, l_φ equals the number of connected components of the set $\varphi^{-1}((-\infty, y])$ which contain at least a point of the set $\varphi^{-1}((-\infty, x])$. Hence, in the specific example of Fig.2(d), $l_\varphi = 2$, since only the leftmost and the rightmost light regions contribute to l_φ.

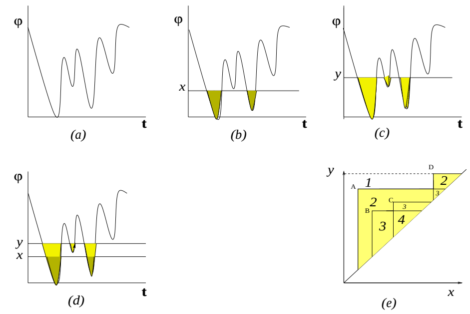


Figure 2: *Example of size function. (a) Graphe of some measuring function φ. (b) and (c) The shaded regions identify the set of points with $\varphi \le x$ and $\varphi \le y$ respectively. (d) The darker shading identify the regions with $\varphi \le x$ and $\varphi \le y$. (e) Graph of the size function l_φ for all possible values of x and y. Each label denotes the value of l_φ within the undelying region.*

Let us briefly recall the main results and properties of the size function l_φ [17].

- The size function l_φ is computed within the triangular region $T_\varphi = \{(x, y) : \varphi^{min} \le x < y \le \varphi^{max}\}$, where φ^{min} and φ^{max} are the minimum and the maximum of the measuring function φ respectively. Morever, the normalisation $\bar{\varphi} = \dfrac{\varphi - \varphi^{min}}{\varphi^{max} - \varphi^{min}}$ ensures that the size function $l_{\bar{\varphi}}$, computed within the triangular region $T_{\bar{\varphi}}$, where $T_{\bar{\varphi}} = \{(x, y) : 0 \le x < y \le 1\}$, is invariant to scale transformation.
- $l_\varphi(x, y)$ is non-decreasing in x and non-increasing in y.
- $l_\varphi(x, y)$ is finite for $x < y$.
- $l_\varphi(x, y) = 0$ for every $x < \varphi^{min}$.
- $l_\varphi(x, y) = \infty$ for every point (x, y) such that there exists a non-isolated point Q for which $x > \varphi(Q)$ and $y < \varphi(Q)$.
- The size function inherits the invariance properties of the measuring function. Therefore, it is sufficient to find any measuring function which satisfies the invariance requirements proposed in the previous section.

According to the proposed framework, the representation of signs by means of size functions is based on the definition of some measuring function. Since for a giving measuring function two different signs can produce the same size function, the choice of an adequate measuring function is a fundamental problem and the measuring function must satisfy the requirements described in Section 2.

In the next section, we will present our approach for the sign representation, based on the new size functions generated by a new pair of measuring functions.

4 Sign Representation

Unfortunately, the theory of size functions does not provide a formal tool to determine an adequate measuring function. Therefore, the search for measuring functions must be carried out heuristically. In previous work [18, 16, 15], an implementation of the theory of size functions based on the edge of viewed postures was proposed. In [16] the sign representation is obtained by means of seventy size functions and thus requires more than 70 feature vectors, each of them corresponding to the size function of each sign computed in a particular orientation about the image plane.

Let us illustrate the empirical principles behind the search for measuring functions for the representation of the the sign language alphabet. As we noted in Section 2 an adequate measuring function must be sensitive to the orientation of the signs relative to the image plane. Here, we assume the elongated hand shape has an axis of elongation which can be used to define the orientation of the hand sign. In other words, it is necessary to find the orientation of each sign and to use it as a parameter of the measuring function. For elongated hand shapes, we choose the axis of the least second moment (the minor axis of inertia) as the orientation. In order to have more information about the hand shapes, we also consider the second axis (the major axis of inertia).

Table 3: *Orientation of the Major Principal Axis*

μ_{11}	$\mu_{20}-\mu_{02}$	ϕ	θ
0	-	0	$+\frac{\pi}{2}$
+	-	$0 \succ \phi \succ -\frac{\pi}{4}$	$+\frac{\pi}{2} \succ \theta \succ +\frac{\pi}{4}$
+	0	0	$+\frac{\pi}{4}$
+	+	$\frac{\pi}{4} \succ \phi \succ 0$	$+\frac{\pi}{4} \succ \theta \succ 0$
0	0	0	0
-	+	$0 \succ \phi \succ -\frac{\pi}{4}$	$0 \succ \theta \succ -\frac{\pi}{4}$
-	0	0	$-\frac{\pi}{4}$
-	-	$+\frac{\pi}{4} \succ \phi \succ 0$	$-\frac{\pi}{4} \succ \theta \succ -\frac{\pi}{2}$

In terms of moments, the orientation ϕ of the principal axis near the x axis is given by the following equation [13]:

$$\phi = \frac{1}{2}\tan^{-1}\left(\frac{2\mu_{11}}{\mu_{20}-\mu_{02}}\right) \qquad (1)$$

where $\phi \in [-\frac{\pi}{4}, \frac{\pi}{4}]$, μ_{11}, μ_{20} and μ_{02} are the central moments. The specific orientation of either principal axis may be determined from the specific values of μ_{11} and $\mu_{20}-\mu_{02}$. Table 3 illustrates how the orientation of the major principal axis, θ, may be determined using ϕ and the second moments.

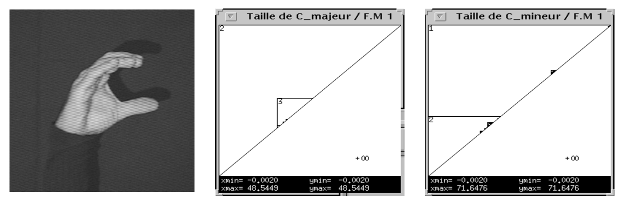
We can now define a new pair of measuring functions as follows:

$$\varphi_1(p) = \begin{cases} d(p) & \text{if } p \quad \text{lies above the major principal axis} \\ 0 & \text{else} \end{cases}$$

and

$$\varphi_2(p) = \begin{cases} d(p) & \text{if } p \quad \text{lies above the minor principal axis} \\ 0 & \text{else} \end{cases}$$

where $d(p)$ is the distance from an edge point of the sign to the center of mass. Thus we have two size functions $l_1(\varphi_1)$ and $l_2(\varphi_2)$ generated by φ_1 and φ_2 (see Fig.3(b) and Fig.3(c)).



(a) (b) (c)

Figure 3: *A paire of Size Functions. (a) The grey-level image of the sign "C". (b) The size function computed using the major axis. (c) The size function computed using the minor axis.*

Intuitively, it appears plausible that a complete representation can be obtained by looking at the four possible regions generated by the two axis. However, after much empirical experimentation, we found that the pair of measuring functions (φ_1 and φ_2) are sufficient to capture the shape of all alphabet hand signs. Note that in our case φ is defined as $\varphi : \gamma \rightarrow \Re$ and $l_\varphi : \Re^2 \longmapsto \aleph$ where γ represents the edge map of the sign. The size functions $l_1(\varphi_1)$ and $l_2(\varphi_2)$ generated by the measuring functions φ_1 and φ_2, are defined on the real pair (x, y) according to the following procedure.

Let us consider the graph G as a discrete approximation of one of the edge maps obtained using the major principal axis (the same procedure is used for the second graph generated by the other, minor, principal axis) (see Fig.3(b)).

1. Find the subgraph $G_{\varphi \le y}$ of G determined by the points p with $\varphi(p) \le y$;

2. Identify the connected components of $G_{\varphi \le y}$;

3. The size function l_φ at point (x, y) equals the number of connected components of $G_{\varphi \leq y}$ which contain at least one vertex p, with $\varphi(p) \leq x$.

The size function obtained according to the previous procedure is represented as a triangular region. In order to obtain the size function in the form of a squared matrix which can be used as an input vector to the neural network, we need to normalize the size function by following these two steps:

1. **Step 1: Normalization of the measuring functions φ_1 and φ_2**

   Let us consider a set of points χ representing the sign edge map and $\varphi_\chi(P)$ one of the measuring functions, where P is a point in χ. We define a new measuring function $\bar{\varphi}_\chi(P) = \dfrac{\varphi_\chi(P) - m}{M - m}$, where $M = \max\limits_{P \in \chi}(\varphi_\chi(P))$ and $m = \min\limits_{P \in \chi}(\varphi_\chi(P))$ are the maximum and the minimum of the measuring function φ_χ, respectively (we assume that the edge map of each sign has at least two points and $M > m$). It should be noted that we always have $\min\limits_{P \in \chi}(\bar{\varphi}_\chi(P)) = 0$ and $\max\limits_{P \in \chi}(\bar{\varphi}_\chi(P)) = 1$. We obtain $0 \leq \bar{\varphi}_\chi(P) \leq 1$. We say that $\bar{\varphi}_\chi$ is the normalization of φ. The invariance of φ implies that if two hand signs have exactly the same shape with respect to φ, then they have exactly the same size function with respect to $\bar{\varphi}$.

2. **Step 2: Normalization of the size function**

   (a) We choose a positive integer r;

   (b) For every $(i, j) \in \{1, 2, ..., r\}$ we compute the value $u_{ij} = l_{(\chi, \bar{\varphi}_\chi)}(\frac{i}{r}, \frac{j}{r})$.

The normalized size function u_{ij} is a squared matrix. It is invariant to scale changes, translation and orientation. However, as we mentioned in Section 2, the representation of signs must be sensitive to sign orientation. Consequently, our representation model is defined by both the two size functions and the principal axes. The normalisation algorithm described above may be sensitive to noise. As a consequence, it can happen that two edge maps χ_1 and χ_2 have similar shapes but the maximum and the minimum M and m of the measuring functions φ_1 and φ_2 are very different. It follows that the size functions $l_1(\varphi_1)$ and $l_2(\varphi_2)$ of χ_1 and χ_2 may be different (see Fig.4).

In order to reduce the effect of noise during normalization of the measuring functions, we can proceed as follows:

1. Assume that the set $\chi = \{P_1, P_2, ..., P_n\}$ is a finite approximation of the sign edge map.

2. Let us set $\mu = \dfrac{1}{n} \sum\limits_{i=1}^{n} \varphi_\chi(P_i)$.

3. Then we set $d_k^+ = (\sum\limits_{\varphi_\chi(P_i) > \mu} (\varphi_\chi(P_i) - \mu)^k)^{\frac{1}{k}}$, where k is a positive real number.

4. Similarly, we set $d_k^- = (\sum\limits_{\varphi_\chi(P_i) < \mu} (\mu - \varphi_\chi(P_i))^k)^{\frac{1}{k}}$.

5. Finally, we set $M_k = \mu + d_k^+$ and $m_k = \mu - d_k^-$.

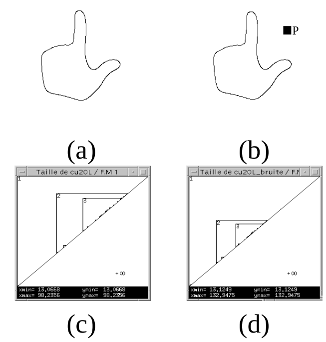


Figure 4: *Normalization of the measuring functions of noisy edge. (a). The edge map of the sign "L". (b) The noisy edge map of the sign "L". (c) and (d) are the size functions of edge map (a) and (b) respectively.*

Unlike M and m, the values M_k and m_k are stable enough when we apply some noise to the set χ . We can show that $\lim\limits_{k \to +\infty} M_k = M$, $\lim\limits_{k \to +\infty} m_k = m$, $\lim\limits_{k \to 0^+} M_k = +\infty$, and $\lim\limits_{k \to 0^+} m_k = -\infty$. So, on the one hand we cannot choose too large a value for k, otherwise M_k and m_k would be approximately M and m, which can change abruptly because of noise. On the other hand, if we choose too small a value for k, M_k and m_k are too large, so that the "window" $[0, 1] \times [0, 1]$ in which we compute the normalized size function corresponds to too large a subset $[m_k, M_k] \times [m_k, M_k]$ of the real plane before normalization, and many details are lost. For our purpose we chose $k = 4$.

In order to check whether or not this representation was correct, a simple recognition scheme has been developed and tested. Let us now discuss the construction of this recognition scheme and the experimental results obtained.

5 Experimental Results

Our recognition system differs from previous work on the integration of two size functions, principal axis, and neural network. The proposed system architecture is shown in Figure 5. It includes four major modules : edge detection, principal moment axis computation, size function computation, and recognition. The system applies the edge detection algorithm to find the border of the hand shape. Then we compute the principal axes. The size functions are computed and normalized. Finally, we use a three-layer feedforward neural network for the recognition. The input to the network is the feature vector obtained in the previous section, which consists of 146 features including both size functions and principal axis orientation. The output of the network is a 25-dimensional vector.

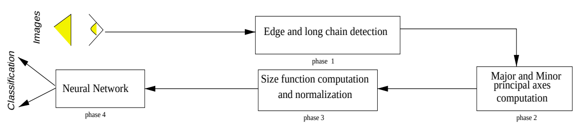


Figure 5: Recognition System Architecture

Table 4: *Sign recognition rate from training set T1, T2 and T3.*

Tests	T1	T2	T3	
Signs	S2	S1	S1	S2
“A”	2(E)	2(E)	——	——
“B”	1(F)	1(F)	——	1(F)
“C”	——	——	——	——
“D”	——	1(I)	——	——
“E”	——	1(A)	——	——
“F”	——	——	——	——
“G”	——	——	——	——
“H”	1(G)	——	1(G)	——
“I”	——	——	1(O)	——
“J”	1(M)	——	——	——
“K”	——	——	——	——
“L”	——	1(D)	——	——
“M”	2(N)	——	——	——
“N”	1(M)	——	1(M)	——
“O”	——	1(H)	——	1(H)
“P”	1(G)	——	——	——
“Q”	——	1(C)	——	1(C)
“R”	——	——	——	——
“S”	1(A)	1(T)	1(A)	1(T)
“T”	2(S)	1(S)	1(S)	——
“U”	——	2(V)	1(R)	——
“V”	——	——	——	——
“W”	1(Q)	——	——	——
“X”	1(U)	1(G)	1(O)	——
“Y”	——	——	——	——
Rates	85%	89%	93%	96%

Three series of experiments were performed based on three different tests T1, T2, T3. We explored 20 sequences of signs performed by two different subjects S1 and S2 (10 sequences each). In the first test (T1), the neural network is trained on 10 sequences performed by subject S1. Recognition was tested on 10 sequences performed by the second subject, S2. For test (T2), the neural network is trained on 10 sequences performed by subject S2. Recognition was tested on 10 sequences performed by the other subject, S1. Finally, in the last test, (T3), the neural network is trained on 5 sequences from S1 and 5 sequences from S2. Recognition was tested on 10 sequences provided by S1 (5 sequences) and S2 (5 sequences). The recognition rates are shown in Table 4.

A few comments are in order. First, it is interesting to note that if we compare our system with the system of C. Uras and A. Verri [15], we find that the recognition success is not different, however our system is efficient because we used only two size functions. It should be recalled that C. Uras and A. Verri used seventy size functions. Second, looking at Table 4, we discover that the recognition rate is higher in the last test (T3), in which the training and test sets were performed by the same subjects (S1 and S2). Notice the overall graceful degradation of the recognition rates when the training sets and the testing sets are performed by different subjects (T1 and T2). Lower error rates are expected if the training and testing sets are performed by the same subject (about 96%). Third, the confusions involve signs for which internal edges appear to be important as for ”S” and ”T”,”M” and ”N”, “A” and “E”, due to the similarity of the signs (see Fig.1). To eliminate this problem, we can use internal edges and another network which can be trained using the misclassified feature vectors of the confused signs, to improve the performance of the network. The use of 3D informations can also increase the recognition rates of the overall system.

6 Conclusion

In this paper we have shown an unencumbered, vision-based system for recognizing International Signing Language (ISL) without invoking complex hand models. The scheme is based on the use of a pair of moment-based size functions as input to a neural network classifier. Unlike previous work proposed by C. Uras [15], our scheme invloves the use of a single pair of size functions, rather than families of seventy size functions. The results obtained indicate that the system is able to recognize viewed signs efficiently with a good percentage of success. It is concluded that size functions can be useful in analyzing sign language alphabet recognition. The experience gained from studying ISL will also contribute to the study of other sign languages, such as ASL(American Sign Language) and may be useful for the interpretation of a large set of virtual reality commands. Some improvements are needed, such as the use of internal edge information and a second network trained on refused and misclassified signs.

References

[1] Y. Abu-Mustapha and A. Psaltis. *Recognitive Aspects of the Moment Invariants. IEEE Trans. Patt. Anal. and Machine Intell.*, 6:5698–5706, 1984.

[2] Franz L. Alt. *Digital Pattern Recognition by Moments*. *Journal of the ACM*, 11:240–258, 1962.

[3] William T. Freeman and Michal Roth. *Orientation Histograms for Hand Gesture Recognition*. *Int. Workshop. on Automatic Face and Gesture Recognition*, Switzerland, 1995.

[4] Weissman C Freeman W. *Television Control by Hand Gesture*. *Int. Workshop on Automatic Face and Gesture Recognition*, 26-28:179–183, 1995, Switzerland.

[5] P. Frosini. *Measuring Shapes by Size Function*. In *Proc. of SPIE on Intelligent Robotics and Computer Vision, Boston, Mass.*, volume 1607, pages 3–26, 1991.

[6] Wen. Gao. *Enhanced User by Hand Gesture Recognition*. *CHI'95 Wokshop on User Interface by Hand Gesture, Denver*, pages 45–53, 1995.

[7] C. Gourley. *Neural Networks Utilizing Posture Input for Sign Language Recognition*. *Technical report Computer Vision and Robotics Research Laboratory, University of Tenessee Knoxville*, novembre, 1994.

[8] M. Handouyahia. *Sign Language Recognition using Moment-Based Size Functions*. *M.Sc. en Informatique, Département de Mathématique et d'informatique, Université de Sherbrooke, Sherbrooke*, 1998.

[9] Robert M. Haralick and Linda G. Shapiro. *Computer and Robot Vision*, volume 2. *Addison-Wesley Publishing Company*, 1993.

[10] Philip A. Harling. *Gesture Input Using Neural Networks*. Technical report, *University of York, UK.*, 1993.

[11] M. Hu. *Visual Pattern Recognition by Moment Invariants*. *IRE Trans. Inf. Theory*, 8:179–187, 1962.

[12] K. Murakami and H. Taguchi. *Gesture Recognition using Recurrent Neural Networks*. *Actes de CHI'91 Workshop on User Interface by Hand Gesture, ACM*, pages 237–242, 1991.

[13] Richard J. Prokop and Anthony P. Reeves. *A Survey of Moment-Based Techniques for Unoccluded Object Representation and Recognition*. *CVGIP: Graphical Models and Image Processing*, 54, no.5:438–460, september 1992.

[14] T. Takahashi and F. Kishino. *Hand Gesture Coding Based on Experiments Using a Hand Gesture Interface Device*. *SIGCHI Bulletin*, vol.23:67–74, 1991.

[15] C. Uras and A. Verri. *Sign Language Recognition: An Application of the Theory of Size Functions*. *6th British Machine Vision Conference*, 2:711–720, 1995.

[16] A. Verri and C. Uras. *On the Recognition of the Alphabet of the Sign Language Through Size Functions*. *Proc, XII Int. Conf. Patt. Recog., Jerusalem*, II:334–338, 1994.

[17] A. Verri and C. Uras. *A Metric-Topological Approach to Shape Representation and Recognition*. *Image and Vision Computing*, 14:189–207, 1996.

[18] A. Verri, C. Uras, P. Frosini, and M. Ferri. *On the Use of Size Functions for Shape Analysis*. *Biol. Cybern*, 70:99–107, 1993.

[19] R. Watson. *A Survey of Gesture Recognition Techniques*. *Dep. of Comp. Sci., Trinity College, Dublin2. Technical Report*, pages TCD–CS–93–11, 1993.