Utilizing Invariant Descriptors for Finger Spelling American Sign Language Using SVM

Omer Rashid, Ayoub Al-Hamadi, and Bernd Michaelis

Institute for Electronics, Signal Processing and Communications (IESK)
Otto-von-Guericke-University Magdeburg, Germany
{Omer.Ahmad,Ayoub.Al-Hamadi}@ovgu.de

Abstract. For an effective vision-based HCI system, inference from natural means of sources (i.e. hand) is a crucial challenge in unconstrained environment. In this paper, we have aimed to build an interaction system through hand posture recognition for static finger spelling American Sign Language (ASL) alphabets and numbers. Unlike the interaction system based on speech, the coarticulation due to hand shape, position and movement influences the different aspects of sign language recognition. Due to this, we have computed the features which are invariant to translation, rotation and scaling. Considering these aspects as the main objectives of this research, we have proposed a three-step approach: first, features vector are computed using two moment based approaches namely Hu-Moment along with geometrical features and Zernike moment. Second, the categorization of symbols according to the fingertip is performed to avoid mis-classification among the symbols. Third, the extracted set of two features vectors (i.e. Hu-Moment with geometrical features and Zernike moment) are trained by Support Vector Machines (SVM) for the classification of the symbols. Experimental results of the proposed approaches achieve recognition rate of 98.5% using Hu-Moment with geometrical features and 96.2% recognition rate using Zernike moment for ASL alphabets and numbers demonstrating the dominating performance of Hu-Moment with geometrical features over Zernike moments.

1 Introduction

Human Computer Interaction (HCI) is emerged as a new field which aims to bridge the communication gap between humans and computers. An intensive research has been done in computer vision to assist HCI particularly using gesture and posture recognition [1][2]. Many pioneering techniques have been proposed to solve the research issues however a natural mean of interaction still remains and yet to address.

Sign language recognition is an application area for HCI to communicate with computers and is categorized into three main groups namely finger spelling, word level sign and non-manual features [3]. In sign language, Hussain [4] used Adaptive Neuro-Fuzzy Inference Systems (ANFIS) model for the recognition of Arabic Sign Language. In his approach, gloves are used for detection of fingertip and

G. Bebis et al. (Eds.): ISVC 2010, Part I, LNCS 6453, pp. 253–263, 2010.

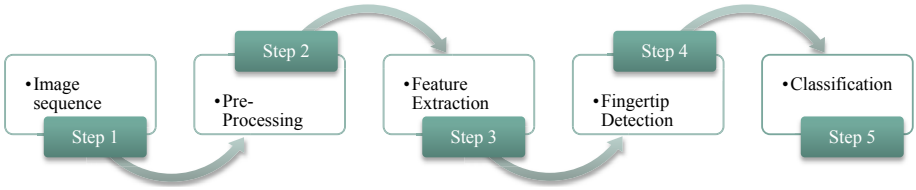

Fig. 1. Shows the framework of posture recognition system

wrist location with six different colors. Similarly, another approach is proposed by Handouyahia et al. [5] which presents a recognition system based on shape description using size functions for International Sign Language. Neural Network is used to train alphabets from the features computed for sign languages. However, the computed features in their proposed approach are not rotation invariant.

Other approach includes the Elliptic Fourier Descriptor (EFD) used by Malassiotis and Strintzis [6] for 3D hand posture recognition. In their system, orientation and silhouettes from the hand are used to recognize 3D hand postures. Similarly, Licsar and Sziranyi [7] used Fourier coefficients from modified Fourier descriptor approach to model hand shapes for the recognition of hand gestures. Freeman and Roth [8] suggested a method by employing orientation histogram to compute features for the classification of gesture symbols, but huge training data is used to resolve the rotation problem. Through out the literature, it is observed that the coarticulation such as hand shape, position and remains a fundamental research objective in vision based hand gesture and posture recognition systems.

The main contribution of the paper can be elaborated in two aspects: two set of invariant feature vectors are extracted to handle the coarticulation issues and the performance analysis of these feature vectors are demonstrated; and the symbols are categorized according to the detected fingertip by computing the curvature analysis to avoid the mis-classification among the posture signs. The remainder of the paper is organized as follows. Section 2 demonstrates the posture recognition system for ASL. Experimental results are presented in section 3 to show the performance of proposed approaches. Finally, the concluding remarks are sketched in section 4.

2 Posture Recognition System

In this section, components of proposed posture recognition systems are presented as shown in Fig. 1.

2.1 Pre-processing

The image acquisition is done by Bumblebee2 camera which gives 2D images and depth images. The depth image sequences are exploited to select region of interest for segmentation of objects (i.e. hands and face) where the depth lies in range from 30 cm to 200 cm (i.e. in our experiments) as shown in Fig. 2(a). In this region, we extract the objects (i.e. hands and face) from skin color distribution

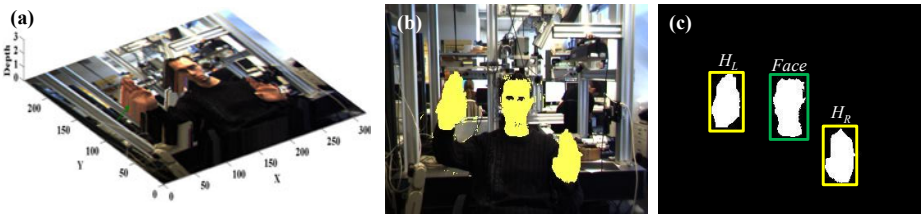

Fig. 2. (a) Original Image with selected depth region (b) Results of Normal Gaussian distribution using the depth Information (c) Detected hands and face

and are modeled by normal Gaussian distribution characterized by mean and variance as shown in Fig. 2(b). We have used YC_bC_r color space because skin color lies in a small region of chrominance components where as the effect of brightness variation is reduced by ignoring the luminance channel. After that, skin color image is binarized and the contours are extracted by computing chain code representation for detection of hands and face as shown in Fig. 2(c).

2.2 Feature Extraction

Two different approaches are employed and analyzed in the proposed approach for the extraction of posture features and are described in the following section.

Hu Moments and Geometrical Feature Vectors: In the first approach, Hu-Moment (i.e. statistical feature vector (i.e. FV)) F_{Hu} and geometrical feature vectors F_{Geo} are computed for posture recognition and is formulated as:

$$F_{Hu,Geo} = F_{Hu} \wedge F_{Geo} \tag{1}$$

Statistical FV: Hu-Moments [9] are derived from basic moments which describe the properties of objects shape statistically (i.e. area, mean, variance, covariance and skewness etc). Hu [9] derived a set of seven moments which are translation, orientation and scale invariant. Hu invariants are extended by Maitra [10] to be invariant under image contrast. Later, Flusser and Suk [11] derived the moments which are invariant under general affine transformation. The equations of Hu-Moments are defined as:

$$\phi_1 = \eta_{20} + \eta_{02} \tag{2}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \tag{3}$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \tag{4}$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \tag{5}$$

$$\begin{aligned}\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]\end{aligned} \tag{6}$$

$$\phi_6 = (\eta_{20} - \eta_{02})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \tag{7}$$

$$\phi_7 = (3\eta_{12} - \eta_{03})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] \quad (8)$$

These seven moments are derived from second and third order moments. However, zero and first order moments are not used in this process. The first six Hu-Moments are invariant to reflection [12], however the seventh moment changes the sign. Statistical FV contain the following set:

$$F_{Hu} = (\phi_1, \phi_2, \phi_3, \phi_4, \phi_5, \phi_6, \phi_7)^T \quad (9)$$

where ϕ_1 is the first Hu-Moment. Similar is the notation for all other features in this set.

Geometrical FV: Geometrical FV consist of circularity and rectangularity, and are computed to exploit hand shapes with the standard shapes like circle and rectangle. This feature set varies from symbol to symbol and is useful to recognize ASL signs. Geometrical FV is stated as:

$$F_{Geo} = (Cir, Rect)^T \quad (10)$$

Circularity Cir and rectangularity $Rect$ are measures of shape that how much object's shape is closer to circle and rectangle respectively. These are defined as:

$$Cir = \frac{Perimeter^2}{4\pi \times Area}\ ,\ Rect = \frac{Area}{l \times w} \quad (11)$$

Length l and width w are calculated by the difference of largest and smallest orientation in the rotation. Orientation of object is calculated by computing the angle of all contour points using central moments. Statistical and geometrical FV set are combined together to form a feature vector set and is defined as:

$$F_{Hu,Geo} = (\phi_1, \phi_2, \phi_3, \phi_4, \phi_5, \phi_6, \phi_7, Cir, Rect)^T \quad (12)$$

Zernike Moments: Teague [13] examines that Cartesian moment can be replaced by orthogonal basic set (i.e. Zernike polynomial), resulting in an orthogonal moment set. The magnitudes of Zernike moments are invariant to rotation and reflection [14]. However, translation and scaling invariance can easily be achieved like central moments.

$$A_{pq} = \frac{(p+1)}{\Pi} \sum_x \sum_y I(x,y)\left[V_{pq}(x,y)\right], x^2 + y^2 \leq 1 \quad (13)$$

where $I(x,y)$ is image pixel and p and q defines the moment-order. Zernike polynomials $V_{pq}(x,y)$ are defined in polar form $V_{pq}(r,\theta)$ as:

$$V_{pq}(r,\theta) = R_{pq}(r)e^{-jq\theta} \quad (14)$$

where R_{pq} is a radial polynomial and is defined as:

$$R_{pq}(r) = \sum_{s=0}^{\frac{p-|q|}{2}} \frac{(-1)^s (p-s)! r^{p-2s}}{s!\left(\frac{p+|q|}{2} - s\right)!\left(\frac{p-|q|}{2} - s\right)!} \quad (15)$$

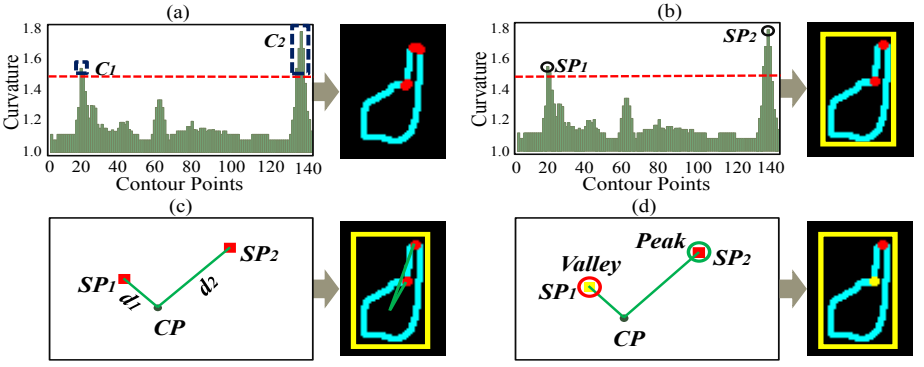

Fig. 3. (a) It shows clusters (i.e. C_1 and C_2) whose threshold is above $\sqrt{2}$. (b) Maximum local extreme selected contour point (i.e. SP_1 and SP_2) from these clusters. Red points show values above threshold $\sqrt{2}$ (i.e. candidates for fingertips).(c) Red points show the selected contour points (i.e. SP_1 and SP_2). Distance is calculated from center point (CP) and normalization is done. (d) Normalized values greater than 0.5 are detected as fingertip (i.e. peak) marked by red point. Yellow marks represent values less than 0.5.

We have used Zernike moments upto $4th$ order moment. The feature vector set for Zernike moment is as under:

$$F_{Zernike} = (Z_1, Z_2, Z_3, Z_4, Z_5, Z_6, Z_7, Z_8, Z_9)^T \tag{16}$$

Normalization: The normalization is done for features to keep them in a particular range and is defined as:

$$c_{min} = \mu - 2\sigma \ , \ c_{max} = \mu + 2\sigma \tag{17}$$

$$nF_i = (F_i - c_{min}) \ / \ (c_{max} - c_{min}) \tag{18}$$

$nF_{Hu,Geo}$ is the normalized feature for Hu-Moment with geometrical FV . c_{max} and c_{min} are the respective maximum and minimum values used for the normalization of these features. Similar is the case of Zernike moments (i.e.$nF_{Zernike}$).

2.3 Fingertip Detection for Categorization

Given the contour of detected hand, curvature is estimated by considering the neighbor contour points to detect the fingertip [15]. Mathematically, curvature gives the ratio of length (i.e. sum of distances that a curve has) and displacement measures the distance from the first to last point if curve covers a straight line. It is computed from the following equation:

$$curv\,(k) = length/displacement \tag{19}$$

$$length = \sum_{i=(k-n/2)}^{i=(k+n/2)} \|(P_i - P_{i+1})\| \tag{20}$$

Table 1. Confusion Matrix of one Detected Fingertip

Sign	A	B	D	I	H/U
A	99.8 / 98.98	0.0 / 0.02	0.0 / 0.0	0.0 / 0.8	0.2 / 0.2
B	0.0 / 0.1	98.18 / 96.3	1.0 / 1.6	0.0 / 0.0	0.82 / 2.0
D	0.0 / 0.0	0.0 / 2.0	98.67 / 96.1	1.33 / 1.9	0.0 / 0.0
I	0.58 / 2.0	0.0 / 0.0	0.8 / 2.1	98.62 / 94.62	0.0 / 0.38
H/U	0.0 / 0.0	3.08 / 2.6	0.0 / 0.4	0.24 / 0.8	96.68 / 96.2

Table 2. Confusion Matrix of Two Detected Fingertips

Symbol	C	L	P	Q	V	Y
C	98.65 / 96.95	0.25 / 0.3	0.0 / 0.0	0.75 / 0.1	0.0 / 0.2	0.35 / 0.65
L	0.38 / 1.2	98.5 / 96.2	0.0 / 0.0	0.76 / 1.6	0.0 / 0.2	0.36 / 0.8
P	0.0 / 0.0	0.0 / 0.1	98.74 / 95.5	1.26 / 3.2	0.0 / 0.0	0.0 / 1.2
Q	0.0 / 0.0	0.0 / 0.5	3.78 / 4.2	96.22 / 94.4	0.0 / 0.0	0.0 / 0.9
V	0.2 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 1.3	99.35 / 97.4	0.45 / 1.3
Y	0.0 / 0.0	0.0 / 0.1	0.0 / 0.25	0.0 / 1.05	0.7 / 1.3	99.3 / 96.3

$$displacement = \left\| \left(P_{k-n/2} - P_{k+n/2} \right) \right\| \tag{21}$$

where k is the boundary point of object at which curvature $curv(k)$ is estimated, n is total number of pixels used for curvature estimation, and P_i and $P_{(i+1)}$ are the objects boundary points.

The principle objective is to find high curvature values from contour points which results in detection of peaks from hands contour and tends to represent the fingertip. We have adaptively determine number of contour points (i.e. after conducting empirical experiments) by exploiting the depth information which provides information about the distance of object from the camera. In this way, a candidate for the fingertip is selected when curvature value is greater than $\sqrt{2}$.

Experimental results show contour of left hand with threshold greater than $\sqrt{2}$. In Fig. 3(a), there are two clusters named C_1 and C_2 and maximum value from these clusters is selected using maximum local extreme value. The resulted points are marked as a fingertip (i.e. SP_1 and SP_2) as shown in Fig. 3(b). It is observed that both the peaks (i.e. $\cap$) and valleys (i.e. $\cup$) can be inferred as a fingertip. Therefore, the next step is to remove valleys from being detected as a fingertip. For this purpose, selected contour points (i.e. SP_1 and SP_2) are taken and their distances are computed from the center point CP of object as shown in Fig. 3(c). The normalization is done and these points are scaled ranging from 0 to 1. We pick the points whose values are greater than 0.5 for fingertip detection. In Fig. 3(d), red mark represents the fingertip whereas the yellow mark points to a valley (not a fingertip). In this way, fingertips are successfully detected for the categorization of symbols using above defined criteria.

Four groups are formed for ASL alphabets according to number of detected fingertips (i.e.Group I with no fingertip,Group II with one fingertip,Group III with two fingertips and Group IV with three fingertips). Numbers are not grouped with alphabets because some numbers are similar to alphabets, so it is hard to classify them together (i.e. 'D' and '1' are same with small change of thumb).

Table 3. Confusion Matrix of Fingertips (i.e. all)

FT/ Gr.	1	2	3	4
0	99.8	0.2	0.0	0.0
1	1.2	96.8	2.0	0.0
2	0.0	0.0	95.1	4.9
3	0.0	0.0	0.9	99.1

Table 4. Confusion Matrix of ASL Numbers

Nr.	0	1	2	3	4	5	6	7	8	9
0	99.8 / 99.1	0.2 / 0.9	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0
1	0.3 / 1.1	99.4 / 97.7	0.3 / 1.2	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0
2	0.0 / 0.0	0.0 / 0.98	98.3 / 96.52	0.4 / 0.6	0.0 / 0.0	0.0 / 0.0	1.3 / 1.9	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0
3	0.0 / 0.0	0.0 / 0.0	0.4 / 1.0	98.2 / 96.6	0.9 / 1.3	0.0 / 0.0	0.5 / 0.6	0.0 / 0.0	0.0 / 0.5	0.0 / 0.0
4	0.0 / 0.0	0.0 / 0.0	0.0 / 0.8	0.2 / 1.2	98.2 / 94.6	1.6 / 3.4	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0
5	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 1.0	2.4 / 3.8	97.6 / 95.2	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0
6	0.0 / 0.0	0.0 / 0.0	0.8 / 1.6	0.6 / 1.2	0.0 / 0.0	0.0 / 0.0	98.6 / 97.2	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0
7	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.7 / 1.3	98.3 / 96.5	0.6 / 1.6	0.4 / 0.6
8	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.4 / 0.3	0.0 / 0.0	0.0 / 0.0	0.2 / 0.8	0.4 / 0.8	98.4 / 97.6	0.6 / 0.8
9	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.5 / 0.8	0.0 / 0.0	0.0 / 0.0	0.0 / 0.0	0.4 / 1.75	0.5 / 1.35	98.6 / 96.1

2.4 Classification

Classification is the last step in posture recognition where a posture symbol is assigned to one of the predefined classes. In proposed approach, a set of thirteen ASL alphabets and ten ASL numbers are classified using Support Vector Machines (SVM). SVM [16][17] is a supervised learning technique for optimal modeling of data. It learns decision function and separates data class to maximum width. SVM learner defines the hyper-planes for data and maximum margin is found between these hyper-planes. Because of the maximum separation of hyper-planes, it is also considered as a margin classifier. We have used Radial Basis Function (RBF) Gaussian kernel which has performed robustly with given number of features and provided optimum results as compared to other kernels.

3 Experimental Results

A database is built to train posture signs which contain 3600 samples taken from eight persons on a set of thirteen ASL alphabets and ten numbers. Classification results are based on 2400 test samples from five persons and test data used is entirely different from training data. As Zernike moments $nF_{Zernike}$ and Hu-moments with geometrical FV $nF_{Hu,Geo}$ are invariant to translation, rotation and scaling, therefore posture signs are tested for these properties.

3.1 ASL Alphabets

Thirteen alphabets are categorized into four groups according to fingertips and are presented as follows:

Group I: There is no misclassification between 'A' and 'B' because these two signs are very different. Precisely, statistical FV show correlation in Hu-Moments for

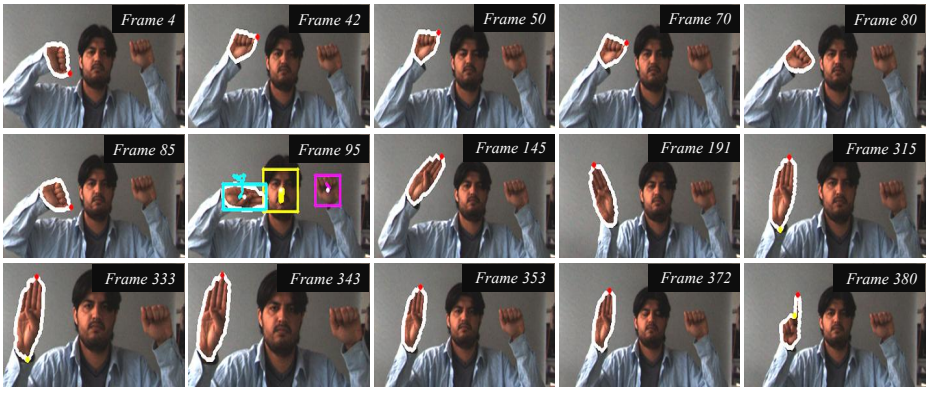

Fig. 4. Sequence starts with recognition of posture signs 'A' till frame 90 with different orientations and back and forth movement. Similarly, posture sign 'B' is recognized from frame 91 to frame 372. 'D' is the last recognized sign in sequence.

features (i.e. ϕ_3,ϕ_4,ϕ_5,ϕ_6,ϕ_7) whereas in Zernike moments, features (i.e. Z_4,Z_8,Z_9) are very different for these posture symbols. Discrimination power of SVM can be seen from this behavior and it classifies posture signs 'A' and 'B' correctly.

Group II: Table 1 shows confusion matrix of classes with one fingertip detected for Hu-Moment with geometrical FV (i.e. above in fraction $\frac{\mathbf{99.8}}{98.98}$) and Zernike moments (i.e.below in fraction $\frac{99.8}{\mathbf{98.98}}$). It can be seen that from first approach, alphabet 'A' results in least misclassification whereas 'H'/'U' has the maximum misclassification with other posture alphabets. Misclassifications between 'B' and 'H/U' are observed in Hu-Moment features during back and forth movement.

Group III: Confusion matrix of group with two detected fingertips is shown in Table 2. During experiments, it is seen that highest misclassification exists between 'P' and 'Q' signs because of their shape and geometry. Besides, statistical FV in this group lies in similar range and thus possesses strong correlation which leads to misclassification between them. Similar is the case of Zernike moments.

Group IV: The posture sign 'W' is always classified because it is the only element in this group.

3.2 ASL Numbers

The classes for ASL numbers are tested for translation, orientation and scaling, and are shown in Table 4. It is found that the least misclassification of number '0' with other classes because its geometrical features are entirely different from others. Highest misclassification exists between numbers '4' and '5' as the similarity between these signs (i.e. thumb in number '5' is open) is very high. Zernike moments also give similar results as the first approach. Average recognition rate for ASL numbers are 98.6% and 96.4% from Hu-Moment with geometrical FV and Zernike moments respectively.

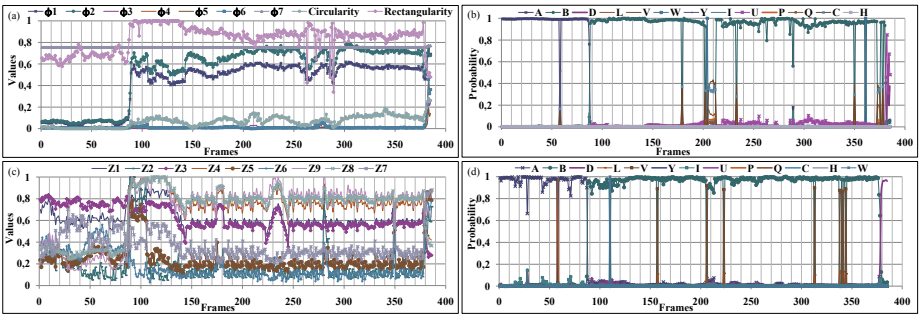

Fig. 5. (a) and (b) Shows the feature vectors values and classification probability for Hu-Moment with circularity and rectangularity. (c) and (d) shows the feature vectors and classification results for Zernike moment.

3.3 Misclassification of Fingertip Groups

The fingertip probability classification of a group with other groups is shown in Table 3. It is observed that the misclassification exists in neighboring groups and is due to the reason that only +/- 1 detected fingertip is wrong or wrongly detected for posture signs. A strong misclassification between the detected fingertips of group III and group IV can be seen in Table 3. Wrong fingertip detection from group III leads to the detection of three fingertips and thus 'W' is recognized from posture sign. Similarly, if two fingertips are detected in group IV, it leads to the classification from group with two fingertips detected class signs.

3.4 Test Sequences

Fig. 4 presents the sequence in which detected posture signs are 'A', 'B' and 'D'. In this sequence, first posture sign changes from 'A' to 'B' is occurred in frame 90 followed by another symbol change at frame 377 from 'B' to 'D'. The symbols in this sequence are tested for rotation and scaling, and it can be seen that Hu-Moment with geometrical FV doesn't affect much under these conditions. However, in our case, Zernike moments are affected under rotation which leads to misclassification. Moreover, it is observed that Zernike moments are most robust to noise than Hu-Moment with geometrical FV. Fig. 5(a) and (b) presents feature values of Hu-Moment with geometrical FV and classification probabilities of this approach for test sequence samples. Moreover, Zernike moments features and classification probabilities are presented in Fig. 5(c) and (d).

Fig. 6(a) presents the sequence in which classified posture symbols are 'W', 'A', 'Y', 'D' and 'V'. Classification probabilities of Hu-Moment with geometrical features and Zernike moments are shown in Fig. 6(b) and (c) respectively. It can be seen from the graphs that classification errors are occured due to wrong fingertip detection and thus results in classification of symbol in a different group.

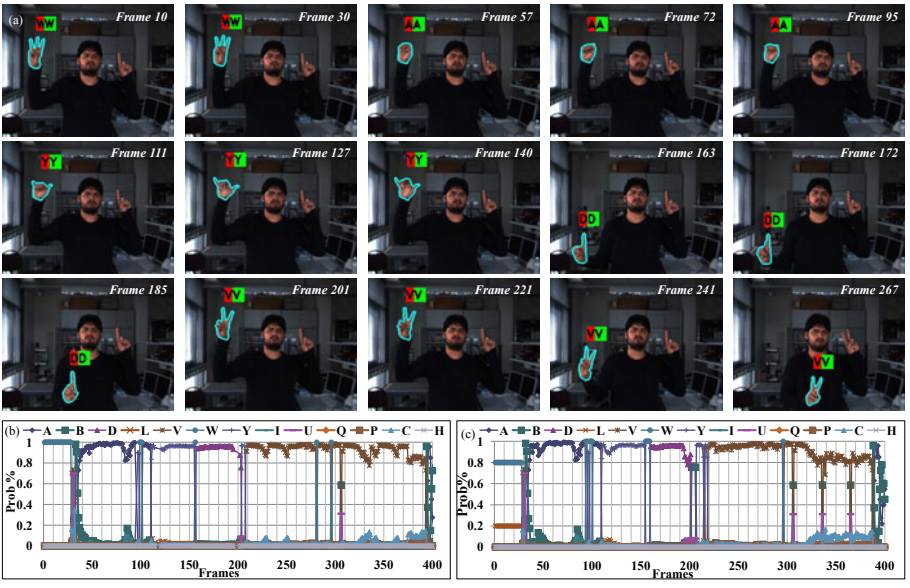

Fig. 6. (a) Presents the sequence in which recognized posture symbols are 'W', 'A','Y','D' and 'V'. (b) shows classification probabilities of Hu-Moment with geometrical features and (c) shows classification probabilities of Zernike moments.

Results show that recognition rate of feature vector computed from Hu-Moments with the geometrical features (i.e. 98.5%) dominate the performance over recognition outcome of feature vectors measured from Zernike moments (i.e. 96.2%).

4 Conclusion

In this paper, we have proposed a three-step approach for recognition of ASL posture signs. Two moment based approaches are analyzed and their features are extracted which are invariant to translation, orientation and scaling. Besides, the fingertip is detected for ASL alphabets and used as a measure to categorize thus avoids the misclassifications of posture signs. SVM is applied for recognition of ASL signs and recognition results of both approaches are computed and analysed. The recognition results based on Hu-Moment with geometrical FV outperforms the other approach using feature vectors computed with Zernike moments.

Acknowledgement

This work is supported by Transregional Collaborative Research Centre SFB/TRR 62 Companion-Technology for Cognitive Technical Systems funded by DFG and Forschungspraemie (BMBF-Froederung, FKZ: 03FPB00213).

References

1. Jaimes, A., Sebe, N.: Multimodal human-computer interaction: A survey. Computer Vision and Image Understanding, 116–134 (2007)
2. Karray, F., Alemzadeh, M., Saleh, J., Arab, M.: Human-computer interaction: Overview on state of the art. International Journal on Smart Sensing and Intelligent Systems 1, 137–159 (2008)
3. Bowden, R., Zisserman, A., Kadir, T., Brady, M.: Vision based interpretation of natural sign languages. In: Proc. 3rd ICCVS, pp. 391–401 (2003)
4. Hussain, M.: Automatic recognition of sign language gestures. Master thesis, Jordan University of Science and Technology (1999)
5. Handouyahia, M., Ziou, D., Wang, S.: Sign language recognition using moment-based size functions. In: Int. Conf. of Vision Interface, pp. 210–216 (1999)
6. Malassiotis, S., Strintzis, M.: Real-time hand posture recognition using range data. Image and Vision Computing 26, 1027–1037 (2008)
7. Licsar, A., Sziranyi, T.: Supervised training based hand gesture recognition system. In: ICPR, pp. 999–1002 (2002)
8. Freeman, W., Roth, M.: Orientation histograms for hand gesture recognition. In: Int. Workshop on Automatic FG Recognition, pp. 296–301 (1994)
9. Hu, M.: Visual pattern recognition by moment invariants. IRE Trans. on Information Theory 8, 179–187 (1962)
10. Maitra, S.: Moment invariants. In: IEEE Conf. on CVPR., vol. 67, pp. 697–699 (1979)
11. Flusser, J., Suk, T.: Pattern recognition by affine moment invariants. Journal of Pattern Recognition 26, 164–174 (1993)
12. Davis, J., Bradski, G.: Real-time motion template gradients using intel cvlib. In: IEEE ICCV Workshop on Framerate Vision (1999)
13. Teague, M.: Image analysis via the general theory of moments. Journal of the Optical Society of America, 920–930 (1980)
14. Bailey, R., Srinath, M.: Orthogonal moment features for use with parametric and non-parametric classifiers. IEEE Trans. on PAMI 18, 389–399 (1996)
15. Kim, D.H., Kim, M.J.: A curvature estimation for pen input segmentation in sketch-based modeling. Computer-Aided Design 38, 238–248 (2006)
16. Cristianini, N., Taylor, J.: An Introduction to Support Vector Machines and other kernel based learning methods. Cambridge University Press, Cambridge (2001)
17. Lin, C., Weng, R.: Simple probabilistic predictions for support vector regression. Technical report, National Taiwan University (2004)