Real-Time Capable System for Hand Gesture Recognition Using Hidden Markov Models in Stereo Color Image Sequences

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

ABSTRACT

This paper proposes a system to recognize the alphabets and numbers in real time from color image sequences by the motion trajectory of a single hand using Hidden Markov Models (HMM). Our system is based on three main stages; automatic segmentation and preprocessing of the hand regions, feature extraction and classification. In automatic segmentation and preprocessing stage, YC_bC_r color space and depth information are used to detect hands and face in connection with morphological operation where Gaussian Mixture Model (GMM) is used for computing the skin probability. After the hand is detected and the centroid point of the hand region is determined, the tracking will take place in the further steps to determine the hand motion trajectory by using a search area around the hand region. In the feature extraction stage, the orientation is determined between two consecutive points from hand motion trajectory and then it is quantized to give a discrete vector that is used as input to HMM. The final stage so-called classification, Baum-Welch algorithm (BW) is used to do a full train for HMM parameters. The gesture of alphabets and numbers is recognized by using Left-Right Banded model (LRB) in conjunction with Forward algorithm. In our experiment, 720 trained gestures are used for training and also 360 tested gestures for testing. Our system recognizes the alphabets from A to Z and numbers from 0 to 9 and achieves an average recognition rate of 94.72%.

Keywords

Gesture Recognition, Computer Vision & Image Processing, Pattern Recognition, Application.

1. INTRODUCTION

Sign language recognition from hand motion or hand posture is an active area in gesture recognition research for Human Computer Interaction (HCI). A gesture is a spatio-temporal pattern [Nia04b] which may be static or dynamic or both as in Sign Language Recognition (SLR). Over the last few years, several methods [Dey06a, Hos01a, Ngu05a, Nob02a, Seb00a, Vas03a] have been suggested for the recognition of the sign language from the hand motion but these differ from one another in their models. Some of these models are Syntactical Analysis [Hos01a], Neural Network [Dey06a], Hidden Markov Models [Seb00a] etc.

HMM are widely used in handwriting, speech recognition, part-of-speech tagging and machine translation. Another advantage of using discrete HMM is that the output distributions are automatically learned by the training process. Vassilia et. al. [Vas03a] have developed a system that could recognize both isolated and continuous Greek Sign Language (GSL) sentences where the orientation vector is extracted from images and used in sentences as input to HMM. Ho-Sub et. al. [Hos01a] introduced a hand gesture recognition method which used combined features of location, angle and velocity to determine the discrete vector that is used as input to HMM. This method runs over the alphabets A to Z, numbers 0 to 9, six edit commands and six drawing elements. Nianjun et. al. [Nia04b] proposed a method to recognize the 26 letters from A to Z by using a different HMM topologies with different states. But, these methods run off-line over a non complex background. Nguyen et. al. [Ngu05a] introduced a hand gesture recognition system to recognize real time gesture in unconstrained environments and the system was tested to a vocabulary of 36 gestures including the American Sign Language (ASL) letter spelling alphabet and digits. Nobuhiko et. al. [Nob02a] introduced a method to obtain features from image sequence where a person is performing the Japanese Sign Language (JSL) in a complex background and to recognize the JSL word. The previous two methods [Ngu05a, Nob02a] run in real time over a complex background, but they are studying the posture of the hand, not the motion trajectory of the hand as in it is our system.

We develop a system to recognize the alphabets (A - Z) and numbers (0 - 9) in real time from color image sequences by the motion trajectory of a single hand using HMM. Our system depends upon following main steps; using GMM for skin color detection, the orientation between two consecutive points is extracted as basic feature, BW algorithm for training and forward algorithm for testing in conjunction with LRB model. Moreover, each alphabet and each number is based on 30 video (20 for training and 10 for testing) where the input images are captured by a Bumblebee stereo camera that has 6mm focal length for about 2 to 5 second at 15 frames per second with 240×320 pixels image resolution on each frame. The recognition rates achieved on training and testing gestures are 99.16% and 94.72% respectively. The rest of this paper is organized as follow; Section 2 demonstrates with the suggested system in three subsections. The experimental results are described in Section 3. Finally, the summary and conclusion are presented in Section 4.

2. GESTURE RECOGNITION SYSTEM

Our system is designed to recognize the alphabets and numbers in real time from stereo color image sequences by the motion trajectory of a single hand using HMM. For automatic initialization, color and 3D-information are used on the basis of clustering of 3D-points in order to overcome the difficulties of overlapping regions. In particular, the gesture recognition system consists of three main stages (Fig.1 and Fig. 10):

- Automatic segmentation and preprocessing; the hand is segmented, localized and tracked to generate its motion trajectory (gesture path) by using GMM for skin color detection.
- Feature extraction; determine the discrete vector which is used as input to HMM by the orientation quantization.
- Classification; the hand motion trajectory is recognized by using discrete vector, LRB model and forward algorithm of HMM.

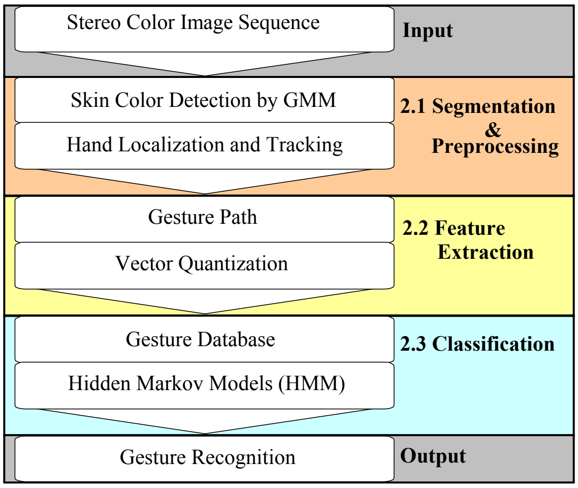


Figure 1. Gesture recognition system using HMM.

The hand graphical gesture consists of 26 alphabet characters from A to Z and 10 Arabic numbers from 0 to 9 where the gesture shapes are shown in Fig. 2.

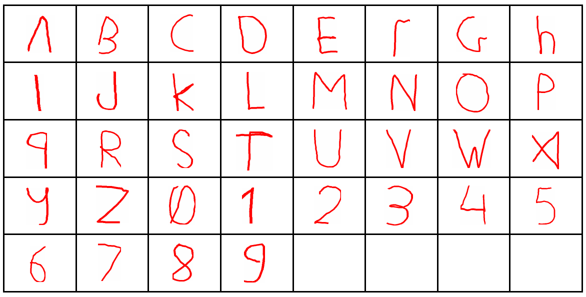


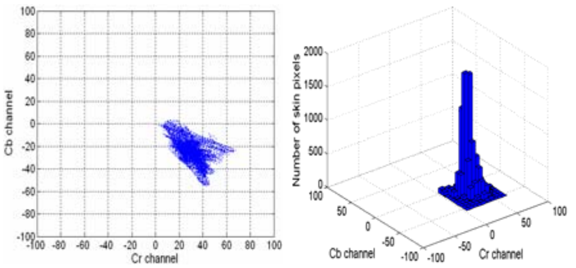
Figure 2. Alphabets and Numbers gesture shapes that are used in our system from hand graphical motion.

2.1 Segmentation and Preprocessing

Automatic segmentation and preprocessing is an important stage in our system where the segmentation of the hand takes place using 3D and color information. For the removal of remaining errors, morphological operations are used as a preprocessing. This stage contains two steps; in the first step the skin color is detected by using Gaussian Mixture Model (GMM) over YC_bC_r color space. In the second step, the hand is localized and tracked by using a blob analysis for hand region.

2.1.1 Skin Color Detection via a GMM

YC_bC_r color space is used in our system where Y channel represents brightness and (C_b,C_r) channels refer to chrominance [Ask04a, Noo06a]. We ignore Y channel in order to reduce the effect of brightness variation and then use only the chrominance channels which are fully representing the color. In a chrominance plane, human skin color is found in a small area (Fig. 3(a)), so each pixel is classified as skin or non skin by using Gaussian model. The GMM technique begins with modeling of skin and non skin by using a database of skin and non skin pixels respectively. A large database of skin pixel is used to train the Gaussian model where the mean vector and covariance matrix of the database characterize the model. In our system, we collect images that contain human skin pixels (Fig. 4) and also images for non skin pixels (Fig. 5).



(a) (b)

Figure 3. The distribution and histogram of the skin training data (a) Distribution values of skin pixels for training data (b) Histogram for skin training data over (C_b,C_r) channels.

A variant of k-means clustering algorithm [Phu02a] for Gaussian clusters performs the model training to determine the initial configuration.

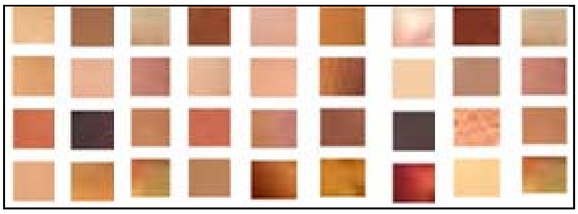


Figure 4. Database of skin pixel where these cropped images were collected from the World Wide Web for different races.

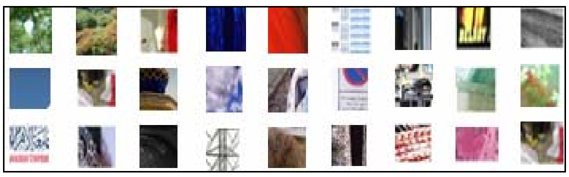


Figure 5. Database of non skin pixel where these cropped images were collected from the World Wide Web for different background.

Suppose that $x=[C_b\ C_r]^T$ represents the chrominance vector of an input pixel. The probability of skin pixel over vector x for mixture model is a linear combination of its probabilities and is calculated as follows:

$$p(x \mid skin) = \sum_{i=1}^{K} p(x \mid i).p(i) \qquad (1)$$

where K is the number of Gaussian components and is estimated by a constructive algorithm that is used the criteria of maximizing likelihood function [Raj98a]. $p(i)$ is the mixture weight and $p(x \mid i)$ is the Gaussian density model for the i^{th} component.

$$p(x \mid i) = \frac{e^{-1/2(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)}}{(2\pi)^{f/2}\sqrt{|\Sigma_i|}} \qquad (2)$$

where μ_i and Σ_i represent the mean vector and the covariance matrix of i^{th} component respectively and f is the dimension of feature space, $x \in R^f$.

$$\sum_{i=1}^{K} p(i) = 1\ ; \quad 0 \le p(i) \le 1 \qquad (3)$$

After K is decided which takes the value 4 in our experiment, the Expectation Maximization (EM) algorithm [Min99a, Ric84a] is used to estimate the maximum likelihood of parameters (mean, covariance and mixture weight) which run on the training data. The EM algorithm starts with initial parameter values where k-means algorithm is used to determine the initial configuration as in Table 1.

Cluster	Mean μ	Covariance Σ	Weight
1	$\begin{bmatrix}119.5\ ;\ 144.1\end{bmatrix}$	$\begin{pmatrix}35.81 & -13.55\\ -13.55 & 14.88\end{pmatrix}$	0.2422
2	$\begin{bmatrix}110.3\ ;\ 153.2\end{bmatrix}$	$\begin{pmatrix}13.34 & 2.12\\ 2.12 & 5.73\end{pmatrix}$	0.2612
3	$\begin{bmatrix}98.6\ ;\ 165.9\end{bmatrix}$	$\begin{pmatrix}46.09 & -21.65\\ -21.65 & 46.82\end{pmatrix}$	0.1668
4	$\begin{bmatrix}103.1\ ;\ 157.3\end{bmatrix}$	$\begin{pmatrix}16.83 & -1.26\\ -1.26 & 16.94\end{pmatrix}$	0.3298

Table 1. Gaussian Mixture Model for skin color.

Moreover, the initial parameters are now old parameters and then EM algorithm estimates the new parameter values. In the next iteration, the new parameters become old and this process is repeated until the convergence is achieved [Sob96a] (i.e. the change in log-likelihood between two iterations is less than a threshold). EM algorithm is done by the following two steps:

• Expectation step

$$p^{old}(i \mid x_n) = \frac{p^{old}(x_n \mid i).p^{old}(i)}{\sum_{j=1}^{K} p^{old}(x_n \mid j).p^{old}(j)} \quad (4)$$

• Maximization step

$$\mu_i^{new} = \frac{\sum_{n=1}^{N} p^{old}(i \mid x_n).x_n}{\sum_{n=1}^{N} p^{old}(i \mid x_n)} \quad (5)$$

$$\Sigma_i^{new} = \frac{\sum_{n=1}^{N} p^{old}(i \mid x_n).(x_n - \mu_i^{new})(x_n - \mu_i^{new})^T}{\sum_{n=1}^{N} p^{old}(i \mid x_n)} \quad (6)$$

$$p^{new}(i) = \frac{1}{N}\sum_{i=1}^{N} p^{old}(i \mid x_n) \quad (7)$$

where N is the number of data points x_n. For the probability *p(x| non-skin)*, a non skin color pixels are modeled as a unimodel Gaussian in order to reduce the computational complexity of skin probability calculation (Table 2).

Mean μ	Covariance Σ
$\begin{bmatrix} 58.65; 48.24 \end{bmatrix}$	$\begin{pmatrix} 13.22 & -8.37 \\ -8.37 & 19.01 \end{pmatrix}$

Table 2. Unimodel Gaussian for non skin color.

For the skin segmentation of hands and face in stereo image sequences an algorithm is used, which calculates the depth information in addition to skin color information [Nie06a]. The depth information can be gathered by passive stereo measuring based on cross correlation and the known calibration data of the cameras. Several clusters are composed of the resulting 3D-points [Nie06a]. The clustering algorithm can be considered as kind of region growing in 3D which used two criteria; skin color and Euclidean distance. Furthermore, this method is more robust to the disadvantageous lighting and partial occlusion which occur in real time environment (for instance, in case of gesture recognition). For more details, the reader can refer to [Nie07b]. By the given depth information from the camera set-up system (Fig. 6 (c)), the overlapping problem between hands and face is solved since the hand regions are closer to the camera rather than the face region. For removing the outliers (noise, spurious components) from the skin probability image, we use morphological operation (median filter, erosion and dilation) since there are small regions that are close to skin but does not belong to the human skin. Furthermore, the holes pixels are filled on the outer edge of an image that is not connected to the background. Thereby, the skin color is detected (hands and face). Fig. 6 (a) shows the first frame of the image sequence.

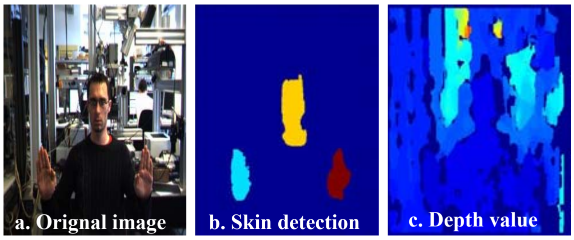


Figure 6. Skin segmentation. (a) First frame of image sequence (b) Labeled skin color detection after using morphological operation (c) Depth information of the original image from a Bumblebee stereo camera.

2.1.2 Hand Detection and Tracking

After the labeled skin image is determined (Fig. 6 (b)), the localization of two hands is found by selecting the two small areas (Fig. 7 (a)) where the face represents the big area and the furthest away from the camera. In addition, we use a blob analysis to determine the boundary area, bounding box and the centroid point of each hand region. Our attention concentrates to the motion of a single hand to detect the hand graphical trajectory for a specific alphabet or number. Consequently, we select a search area in the next frame (Fig. 7 (b)) around the bounding box that is determined from the last frame in order to track the hand and reduce the Area of Interest (AOI). If there are multiple extracted skin regions in a search area of the hand, a big region is selected since this region represents a hand at most.

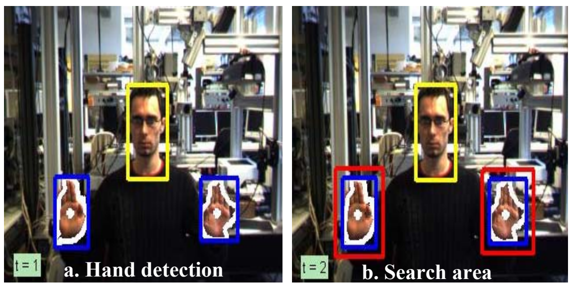


Figure 7. Hand localization and search area (a) Hand localization with a boundary area, bounding box and centroid point (b) Search area around the hands in the next frame.

Thereby, the new bounding box is calculated and the centroid point is determined. By iteration of this process, the motion trajectory of the hand so-called gesture path is generated from connecting centroid points (Fig. 8).

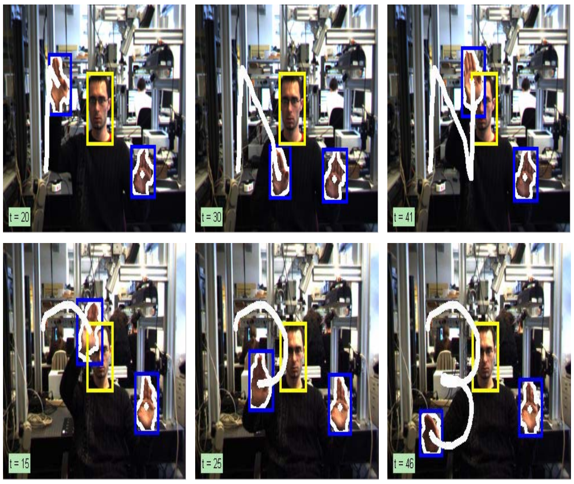


Figure 8. Gesture path for alphabet N and number 3.

2.2 Feature Extraction

The feature extraction is the second stage in our system and there is no doubt that, selecting good features to recognize the hand gesture path play significant role in system performance. There are three basic features as location, orientation and velocity. The previous researches [Hos01a, Nia04b] showed that the orientation feature is the best in terms of results. Therefore, we will rely upon it as a main feature in our system. A gesture path is spatio-temporal pattern which consists of centroid points (x_{hand}, y_{hand}). So, the orientation is determined between two consecutive points from hand gesture path (Eq. 8).

$$\theta_t = \arctan\left(\frac{y_{t+1} - y_t}{x_{t+1} - x_t}\right) \; ; \; t = 1, 2, ..., T-1 \qquad (8)$$

where T represents the length of gesture path. The orientation θ_t is divided by 20° in order to quantize the value from 1 to 18 (Fig. 9). Thereby, the discrete vector is determined and then is used as input to HMM.

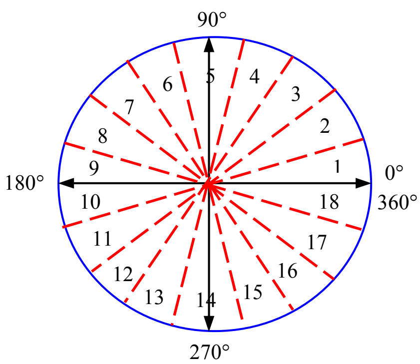


Figure 9. Discrete vector quantization range.

2.3 Classification

Classification is the final stage in our system. Throughout this stage, Baum-Welch algorithm (BW) [Law89a] is used to do a full train for the initialized parameters of HMM by a discrete vector. Moreover, the gesture path of hand motion is recognized by using Left-Right Banded model with 9 states, discrete vector in conjunction with Forward algorithm [Law89a] and building gesture database.

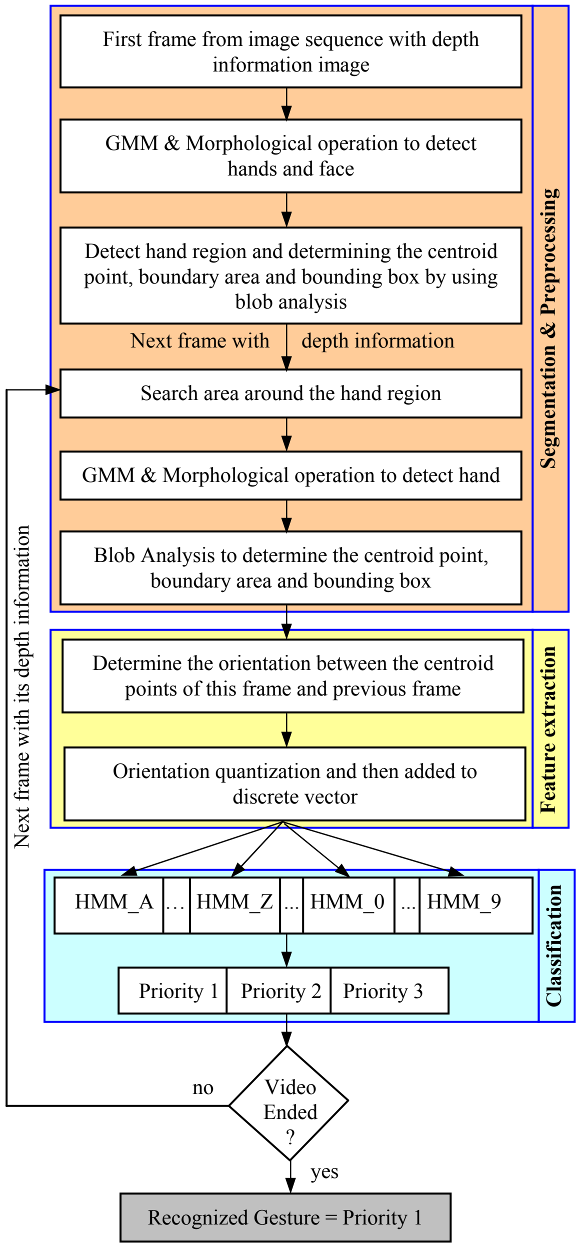


Figure 10. Real time system for alphabets and numbers gesture recognition.

The gesture database contains 30 video for each alphabet from A to Z and also for each number from 0 to 9 where HMM parameters are trained on 20 video and the other 10 video are used for testing by forward algorithm. The following subsections describe this stage in details.

2.3.1 Hidden Markov Models

Markov model is a triple $\lambda = (A, B, \Pi)$ and is described as follows [Hos01a, Nia03a, Nia04b, Sus07a]:

- The set of states S= $\{s_1, s_2, \ldots, s_N\}$ where N is the number of states.
- An initial probability for each state Π_i , i=1, 2, ..., N such that $\Pi_i = P(s_i)$ at the initial step.
- An N-by-N transition matrix $A = \{a_{ij}\}$ where a_{ij} is the probability of a transition from state s_i to s_j; $1 \le i, j \le N$ and the sum of the entries in each row of matrix A must be 1 because this is the sum of the probabilities of making a transition from a given state to each of the other states.
- The set of possible emission (an observation) O= $\{o_1, o_2, \ldots, o_T\}$ where T is the length of gesture path.
- The set of discrete symbols $V = \{v_1, v_2, \ldots, v_M\}$ where M represents the number of discrete symbols.
- An N-by-M observation matrix $B = \{b_{im}\}$ where b_{im} gives the probability of emitting symbol v_m from state s_i and the sum of the entries in each row of matrix B must be 1 for the same pervious reason.

Evaluation, Decoding and Training are the main problems of HMM and they can solved by using Forward-Backward algorithm, Viterbi algorithm [Law89a] and Baum-Welch algorithm respectively. Also, HMM have three topologies: Fully Connected (Ergodic model) where any state in it can be reached from any other state, Left-Right model such that each state can go back to itself or to the following states and Left-Right Banded (LRB) model that also each state can go back to itself or the following state only (Fig. 11).

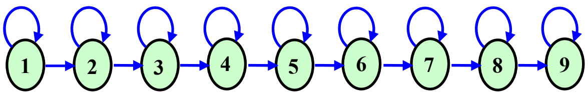


Figure 11. Left-Right Banded model with 9 states.

2.3.2 Initializing HMM parameters

Before describing the initialization of HMM parameters, it will be more convenient to explain, why we use left-Right Banded model with 9 states. LRB model is restricted and simple for training data that will be able to match the data to the model. Since fully connected model has many transitions rather than LRB model, its structure data can lose easily. This also applies to Left-Right model. Moreover, previous researches [Hos01a, Nia04b] showed that the using of 9 states for LRB model are the best in terms of results. So, we used LRB model with 9 states in our system. At most, selecting good initial parameters for HMM, achieves better recognition results. The initial vector Π takes the following value;

$$\Pi = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^T \tag{9}$$

That is because we use 9 states and in order to ensure that it begins from the first state.
The matrix A depends on the duration time d of states for each alphabet or each number and is determined by Eq. 11 such that d is defined as:

$$d = \frac{T}{N} \tag{10}$$

where T is the length of gesture path and N represents the number of states.

$$A = \begin{pmatrix} a_{ii} & l & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{ii} & l & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a_{ii} & l & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{ii} & l & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{ii} & l & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{ii} & l & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a_{ii} & l & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{ii} & l \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{11}$$

Such that,

$$a_{ii} = 1 - \frac{1}{d} \ ; \qquad l = 1 - a_{ii} \tag{12}$$

Matrix B is the important parameter in our system and is determined by Eq. 13. Since HMM states are discrete, all elements of matrix B can be initialized with the same value for all different states.

$$B = \{b_{im}\} \ ; \quad b_{im} = \frac{1}{M} \tag{13}$$

where i, m represent the number of states and number of discrete symbols respectively.

2.3.3 Baum-Welch and Forward Algorithm

Baum-Welch algorithm plays a very important role in our system where it is used to do a full train for the initialized HMM parameters. This algorithm estimates the new matrix A, new matrix B and vector Π where the inputs of it are discrete vector and the initialized parameters. Since our system is trained on 20 video for each alphabet and number, the value of matrix A and matrix B for them is averaged. According to the Forward algorithm, the other 10 video for each alphabet and number are tested where this algorithm is built on discrete vector, matrix A, matrix B and vector Π as inputs for it. The forward algorithm is used to solve the evaluation problem by computing the $P(O|\lambda)$ which give us the probability of the observation.

Define the forward (alpha) values as follows;

$$\alpha_t(i) = p(O_1 = o_1, ..., O_t = o_t, S_t = s_i \mid \lambda) \quad (14)$$

$$Step1: \quad \alpha_1(i) = \Pi(i).b_{im} \quad (15)$$

$$Step2: \quad \alpha_{t+1}(j) = \sum_{i=1}^{9} \alpha_t(i).a_{ij}.b_{jt+1} \quad (16)$$

Then, the forward value is computed by two steps according to the Eq. 15 and Eq.16.

3. EXPERIMENTAL RESULTS

Our system proposed good results to recognize the alphabets and numbers in real time from color image sequences by the motion trajectory of a single hand using HMM. In our experimental results, each alphabet from A to Z and each number from 0 to 9 was based on 30 video which 20 for training and 10 for testing. In other words, our database contains 720 video for training gestures and 360 video for testing gestures. The system was implemented in Matlab language and the input images were captured by a Bumblebee stereo camera system that has 6 mm focal length for about 2 to 5 second at 15 frames per second with 240×320 pixels image resolution on each frame.

Figure 12. System output for alphabet R, where at t=28 the high priority is alphabet F, at t=45 the high priority is alphabet P and at t=70 the result is R.

Fig. 12 shows the output of our system for alphabet R. The following criteria evaluated our result as follows:

The testing data is considered as, $\tau = 10$, for each alphabet or each number where these test data include valid gesture v and also invalid gesture $\bar{v}$ such that;

$$\tau = v_j + \bar{v}_j \ ; \qquad j = 1,2,...,36 \quad (17)$$

where j represents the index of alphabets from A to Z and numbers from 0 to 9. The valid percentage for each alphabet and each number is calculated by Eq.18 and the total percentage for all testing data is determined by Eq. 19.

$$\eta_j = \frac{v_j}{\tau}.100 \quad (18)$$

$$\Re = \frac{1}{36}\sum_{j=1}^{36} \eta_j \quad (19)$$

where η_j is the result of each alphabet or a number and $\Re$ represents the value of all testing data. Similarly, training data was calculated with 20 video for each alphabet and number. For each hand graphical video, the higher priority was computed by forward algorithm to recognize the alphabet or number in our real system frame by frame (Fig. 13).

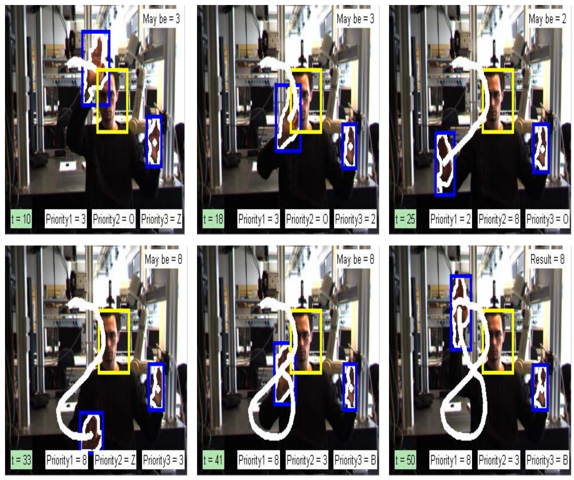


Figure 13. System output for number 8, where at t=18 the high priority is number 3, at t=25 the high priority is number 2 and at t=50 the result is 8.

The recognition was achieved on training and testing gesture with 99.16% and 94.72% respectively where the yield of training data higher than testing data in our system.

4. SUMMARY AND CONCLUSION

This paper proposes a system to recognize the alphabets (A - Z) and numbers (0 - 9) from color image sequences by the motion trajectory of a single hand using HMM which is suitable for real time application. The system consists of three main stages. The first stage is the automatic segmentation and preprocessing, where the hand is localized and tracked to generate its gesture path by using GMM for skin color detection. In the second stage so-called feature extraction, the discrete vector is obtained by quantization of the orientation where this vector is used as input to HMM. The final stage is the classification which can be able to recognize the hand graphical gesture by using LRB model, BW algorithm and Forward algorithm. Our database contains 720 video for training and 360 video for testing. Our results show that; an average recognition rate is 94.72% and 99.16% for testing and training video respectively. In future, our research focuses on: the motion trajectory will be determined by a fingertip instead of the centroid point for the hand region. Also, our system will be developed to recognize the American Sign Language words and sentences.

5. ACKNOWLEDGMENTS

This work was supported by Bernstein-Group (BMBF: 01GQ0702) and NIMITEK grants (LSA: XN3621E/1005M).

REFERENCES

[Ask04a] Askar,S., Kondratyuk,Y., Elazouzi,K., Kauff,P., Schreer,O.: Vision-Based Skin-Colour Segmentation of Moving Hands for Real-Time Application, 1st European CVMP, pp. 79-85, 2004.

[Dey06a] Deyou,X.: A Network Approach for Hand Gesture Recognition in Virtual Reality Driving Training System of SPG, ICPR, 18th International Conference on Pattern Recognition (ICPR 06), pp. 519-522, 2006.

[Hos01a] Ho-Sub,Y., Jung,S., Young,J.B., Hyun,S.Y.: Hand Gesture Recognition using Combined Features of Location, Angle and Velocity, Pattern Recognition 34(7), pp. 1491-1501, March 2001.

[Law89a] Lawrence,R.R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proceeding of the IEEE, Vol. 77, No. 2, pp. 257-286, February 1989.

[Ngu05a] Nguyen,D.B., Enokida,S., Toshiaki,E.: Real-Time Hand Tracking and Gesture Recognition System, GVIP 05 Conference, CICC, pp. 362-368, December 2005.

[Nia03a] Nianjun,L., Brian,C.L., Peter,J.K.: Evaluation of HMM Training Algorithms for Letter Hand Gesture Recognition, IEEE International Symposium on Signal Processing and Information Technology, pp. 648-651, December 2003.

[Nia04b] Nianjun,L., Brian,C.L., Peter,J.K., Richard,A.D.: Model Structure Selection & Training Algorithms for a HMM Gesture Recognition System, In International Workshop in Frontiers of Handwriting Recognition, pp. 100-106, October 2004.

[Nie06a] Niese,R., Al-Hamadi,A., Michaelis,B.: A Stereo and Color-based Method for Face Pose Estimation and Facial Feature Extraction, ICPR 2006 (IEEE), The 18th International Conference on Pattern Recognition, Vol. 1, pp. 299- 302, 2006.

[Nie07b] Niese,R., Al-Hamadi,A., Michaelis,B.: A Novel Method for 3D Face Detection and Normalization, Journal of Multimedia, ISSN,1796-2048, 2007.

[Nob02a] Nobuhiko,T., Nobutaka,S., Yoshiaki,S.: Extraction of Hand Features for Recognition of Sign Language Words, In International Conference on Vision Interface, pp. 391-398, 2002.

[Noo06a] Noor,S.M.S., Jamilin,J.L.M., Roslan,I., Selman, Y., Azhana,A., Adzly,A., Dzulkifli,M.: Sign Language to Voice Recognition: Hand Detection Techniques for Vision-Based Approach, In Technology-Assisted Education, pp. 967-972, 2006.

[Min99a] Ming-Hsuan,Y., Narendra,A.: Gaussian Mixture Modeling of Human Skin Color and Its Applications In Image and Video Databases, In the 1999 SPIE/EI&T Storage and Retrieval for Image and Video Databases, pp. 458-466, January 1999.

[Phu02a] Phung,S.L., Bouzerdoum,A., Chai,D.: A Novel Skin Color Model in YC_bC_r Color Space and its Application to Human Face Detection, In IEEE International Conference on Image Processing (ICIP), pp. 289-292, 2002.

[Raj98a] Raja,Y., Mckenna,S.J., Gong,S.: Colour Model Selection and Adaptation in Dynamic Scenes, In Proceedings European Conference on Computer Vision, pp. 460-474, 1998.

[Ric84a] Richard,A.R., Homer,F.W.: Mixture Densities, Maximum Likelihood and the EM Algorithm, SIAM Review, Vol. 26, No. 2, pp. 195-239, April 1984.

[Seb00a] Sebastien,M., Olivier,B., Jean-Emmanuel,V., Daniel,C.: Hand Gesture Recognition using Input-Output Hidden Markov Model, Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 456-460, 2000.

[Sob96a] Sobottka, K., Pitas, I.: Face Localization and Feature Extraction Based on Shape and Color Information, Proceedings of IEEE International Conference on Image Processing, pp.483-486, 1996.

[Soo05a] Soontranon,N., Aramvith,S., Chalidabhongse,T.: Improved Face and Hand Tracking for Sign Language Recognition, ITCC 2005, Vol. 2, pp.141-146, April 2005.

[Sus07a] Sushmita,M., Tinku,A.: Gesture Recognition: A Survey, IEEE Transactions on Systems, MAN, and Cybernetics, pp. 311-324, May 2007.

[Vas03a] Vassilia,N.P., Konstantinos,G.M.: On Feature Extraction and Sign Recognition for Greek Sign Language, Proceedings of the 7th IASTED International Conference Artificial Intelligence and Soft Computer, pp. 93-98, July 2003.

[Xia00a] Xiaojin,Z., Jie,Y., Alex,W.: Segmenting Hands of Arbitrary Color, Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 446-453, March 2000.