International Journal of Multimedia and Ubiquitous Engineering
Vol.11, No.11 (2016), pp.365-374
http://dx.doi.org/10.14257/ijmue.2016.11.11.33

A Robust Method for Hand Gesture Recognition Using Support Vector Machine

Nadhir Ben Halima

College of Computer Science and Engineering, Taibah University, Yanbu branch, Saudi Arabia
nbenhalima@taibahu.edu.sa

Abstract

The aim of this paper is to explore and recognize hand gestures against complex backgrounds and lighting conditions. Mostly, previous approaches use off-line recognition for the hand motion, which make any gesture system not realistic and not suitable for on-line applications. Furthermore, 3D dynamic features are further extracted from the hand gesture path based on stereo strategy. Thus, the extracted features of dynamic affine-invariants such as location, orientation and velocity are derived from 3D spatio-temporal hand motion. After that the feature vectors are constructed by quantization and then employed to multi-class Support Vector Machine for training or texting. An application of isolated gestures from alphabet characters (A-Z) and numbers (0-9) are trained and tested with superior performance. Furthermore, the results are promising on several video samples in different situations.

Keywords*: Human-computer Interaction, Pattern Recognition, Multi-class Support Vector Machine*

1. Introduction

Nowadays there are still many problems to automatically recognize hand gesture[1] requiring real-world applications notably in the field of Human-Computer Interaction (HCI). In real-time, there is a problem to extract and recognize the hand gesture at the same time, due to the truth which says the same gesture has different trajectory in duration and shape on a par with one person. The gestures can be classified into two classes according to the inclusion of the hand motions. The first one is called posture in which static hand positions stay in the same space whereas movement of dynamic hands and fingers are referred to as gestures as depicted in Figure1.

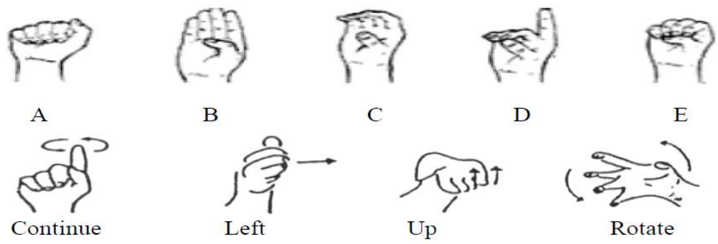

Figure 1. The Above Samples Represent the Posture for Alphabets from A to E, and the Down Samples Refer to The Gestures [1]

The motivation behind gesture interpretation is to make the performance interaction between human and computer close to the interaction between computer and computer.

[1] A hand movement, which represents spatio-temporal pattern, is called gesture, while hand posture means static morph of the hand.

ISSN: 1975-0080 IJMUE

Furthermore, one of most HCI application area is sign language recognition used to communicate with computers. Sign language is classified into three groups so-called non-manual, spelling finger and word level sign [2-4]. Here, the finger spelling is to provide the words character by character. The main interaction through non-manual features and word level sign vocabulary contains the facial expressions such as the body and the mouth position. To understand the research matter, the algorithms used in gesture and posture classification for finger spelling are conveyed and studied for sign languages.

The motivation behind this revision is to employ and promote recognition approach to strongly carry out with high classification rates. Experimentally, the hand segmentation and a good selection for features are substantial for recognition process. Various models are used to recognize alphabet characters and numbers of sign language. In [5], Neuro-Fuzzy Inference Systems (ANFIS)-based model is employed to recognize the Arabic Sign Language (ASL). In this model the colored gloves are used to prevent and alleviate the segmentation problem in addition to make the system able to find the best features. Handouyahia *et al*. [6] presented a new approach to learn and recognize alphabet characters of International Sign Language (ISL) by using Neural Network (NN). The advantage of using NN is to ease, train and test the sign languages according to the estimated features. In other side, Elliptic Fourier Descriptor (EFD), based on Strintzis and Malassiotis, is used to classify the 3D hand posture [7]. Additionally, the orientation and silhouettes features are extracted from the detection area of the hand to recognize its posture. In similar, Licsar and Sziranyi employed the coefficients of Fourier description model in order to get the hand shape that make the system deal with hand gestures [8]. Moreover, Freeman and Roth are to employ the orientation histogram to classify the gesture symbols, but they used a huge training dataset in order to convey the angle problem as well as alleviating the misclassification among character symbols [9].

The contribution of this work is to explore hand gestures via multi-class SVM. A robust method is investigated to recognize the American Sign Language (ASL) alphabets (A-Z) and Arabic numbers (0-9). This model also considers the complex backgrounds and lighting conditions to deal with the hand gesture path. Additionally, the proposed method has the ability to deal with the hand gesture against invariants of scaling, rotation and translation (*i.e.*, spatio-temporal problem). YC_bC_r color space and 3D depth map that depicted from the stereo camera are employed to adaptively localize hands based on stereo matching mechanism. The model of Support Vector Machine (SVM) is used to learn and test our experiments according the extracted features vector. The performance of our proposed technique is promising for recognizing hand gestures with low computational complexity and high performance when applied on many videos, which include confusing situations like hand overlapping and partial occlusion. The outline of this paper is treated as follows: A proposed system for recognizing hand gestures is introduced in Section 2. Section 3 provides our experimental results. Last, Section 4 concludes our work in addition to the future of our work.

2. Suggested Approach

We propose an automatic method to recognize the gesture path from the motion of hand in complex environment based on multi-class SVM. The system of recognizing hand gesture relies on three main stages (Figure 2).

(1) Preprocessing stage: here the area of hand is localized and tracked to provide the motion trajectory, called hand gesture path.

(2) Feature Extraction stage: extracts and quantizes features of orientation, location and velocity, which are employed to multi-class SVM.

(3) Classification stage: building gesture datasets, train and test various hand gestures of alphabet characters and number using multi-class SVM.

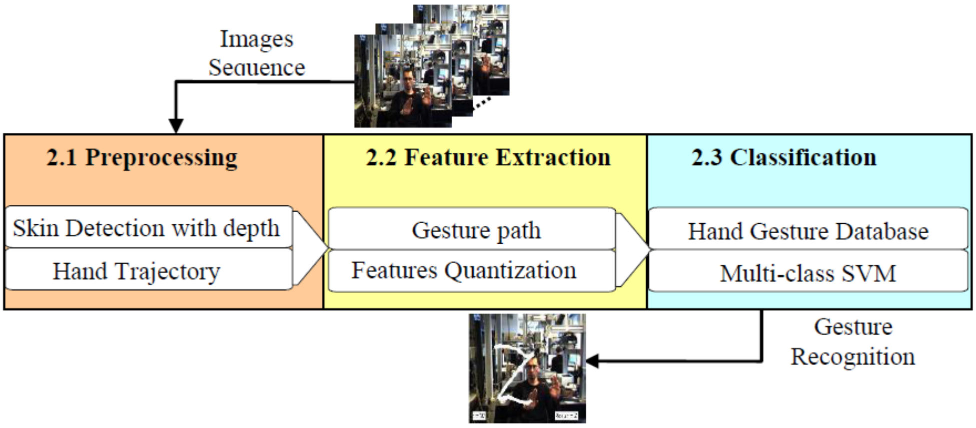

Figure 2. Hand Gesture Recognition System with Three Major Stages

2.1. Preprocessing

Preprocessing stage includes two processes. In the first process, hand skin color is segmented. Hand localization and tracking are processed in the second process. In the fist process, we rely on the YC_bC_r color space for segmentation where C_b and C_r channels refer to the chrominance but Y channel refers to brightness [10]. The motivation behind the use of color space is to eliminate Y channel and to reduce the brightness effect as well as use the channels (C_b, C_r), which completely represents the color. Since the human skin belongs to small region in chrominance plane, every pixel is classified as skin or not according to the lockup table. We note that if the pixel is classified with value 1, that means it belongs to skin region otherwise it belongs to non-skin region. Based on the depth information, which is provided via stereo camera and the threshold value, the image of skin color is obtained. Here the mean depth information value will adaptively vary for each skin region in the input of image sequences. The purpose of using depth information value is to alleviate the problem of hand–face overlapping and hand-hand overlapping. After that, the outliers in the resultant binary image like noise and spurious components are removed. Furthermore, a median filter as morphological operation is employed to carry out the erosion and dilation processes since the little regions belong to human skin (*i.e.*, classified as non-skin). In our experiment, the person wears T-shirts with long sleeves against complex environment. Additionally, we fill the outer edge of the current image frame to completely detect the skin color of the two hands and face.

Figure 3 illustrates the hand skin segmentation and localization in which the first image of the input images sequence is depicted in Figure 3(a). In Figure 3(c), the correct regions of hands and face are correctly detected in a yellow color. So why, the two small regions in area are classified as a hand and not a face that furthest away from the camera and represent a big region area. In our work, the motion of a single hand to generate its trajectory for a specific alphabet character or number is called a gesture path. In Matlab, we call a blob analysis function to determine the specification of centroid point boundary area and bounding box for each skin region.

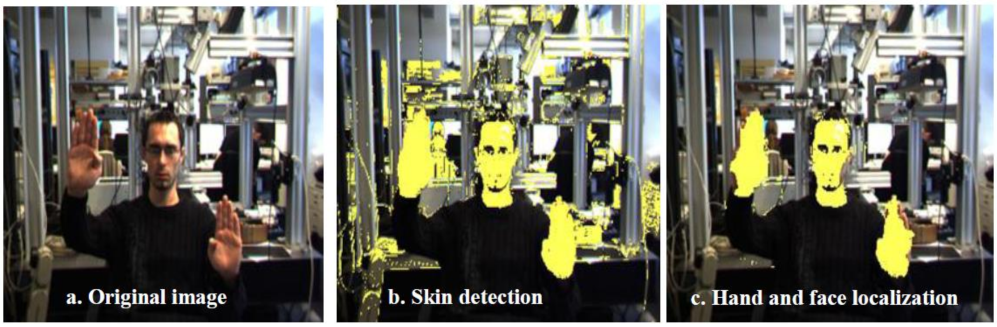

Figure 3. Hand Detection (a) Source Image (b) Skin Segmentation (c) Two Hand and Face Localization

Consequently, the bounding box in further frames depends on a surrounding search area with the purpose of reducing the cost calculation region considering the speed of hand motions. During the search area the big region is considered as a hand if we find multiple skin regions. As a result, the new bounding box is estimated as well as the new centroid point. The processes are iterated until the gesture path of alphabet character or numbers are generated via connecting the hand centroid point of all detecting frames.

2.2. Feature Extraction

In our system, there are three main features: location, velocity and orientation. As a result, the three features are quantized and then employed to the classifier either for training or testing. The features are extracted from their spatio-temporal coordinates of hand gesture in Cartesian coordinate system. Formally speaking, the hands gesture consists of connecting their centroid points (x_{hand}, y_{hand}).

According to the location features, there are two different features. First, L_c represents the distance between the current hand centroid point and the centroid point of a gesture path. The main reason behind using this feature is to locate the starting point of a gesture path. Furthermore, the location feature for the same gesture may be varied Eq.1). Second, L_{sc} measures the distance between the start point and the current point of gesture path (Eq. 2).

$$L_{C_t} = \sqrt{(x_{t+1} - C_t)^2 + (y_{t+1} - C_t)^2}, \quad t = 1, 2, ..., T-1 \tag{1}$$

$$(C_x, C_y) = \frac{1}{n}\left(\sum_{t=1}^{n} x_t, \sum_{t=1}^{n} y_t \right) \tag{2}$$

$$L_{SC_t} = \sqrt{(x_{t+1} - x_1)^2 + (y_{t+1} - y_1)^2} \tag{3}$$

Here, T represents gesture path's length and (C_x, C_y) represents the gravity center of extracted hand gesture path at point n. The gravity center is calculated in order to achieve real-time implementation. An orientation represents the second basic feature in our system, where it provides the direction along the hand motion which constructs the gesture. The orientation is a major feature in recognition processes because they are based on displacement vector. There are three orientation features in our system as θ_{1t} center gravity orientation, θ_{2t} orientation between two sequential points and θ_{3t} orientation between current point and the start point of hand gesture path (Eq. 4).

$$\theta_{1t} = \arctan\left(\frac{y_{t+1} - C_y}{x_{t+1} - C_x} \right), \ \theta_{2t} = \arctan\left(\frac{y_{t+1} - y_t}{x_{t+1} - x_t} \right), \ \theta_{3t} = \arctan\left(\frac{y_{t+1} - y_1}{x_{t+1} - x_1} \right) \tag{4}$$

Finally, the third feature is velocity, which provides an important role in recognition. The velocity V depends on the fact that every gesture is performed with various speeds. We note that the hand velocity wanes at the corner points of gesture path. Formally speaking, the velocity is computed using the Euclidean distance between two sequential points (Eq. 5).

$$V_t = \sqrt{(x_{t+1} - x_t)^2 + (y_{t+1} - y_t)^2} \tag{5}$$

As a result, the feature vector of (L_{ct}, L_{sct}, θ_{1t}, θ_{2t}, θ_{3t}, V_t) is extracted at time t. In general, we can say that any gesture is made of a consecutive sequence of feature vectors in space dimension. Once the feature vector of a gesture path is obtained, it is then clustered, projected and employed as input to the SVM classifier. In the feature space, k-means clustering algorithm [11] carries out this process and classifies the patterns of gesture into K clusters. This algorithm used the criteria of minimum distance between the feature point and the centroid of each cluster. Our experiment is based on a set of clusters where each cluster is assigned to extracted feature vector at time t. Thus, the resultant cluster indexes of gesture observation symbols are submitted to the multi-class SVM. To correctly determine the number of clusters K using k-means algorithm, we empirically considered all segmented parts that exist in all alphabet characters (A-Z) and number (0-9) such that every segment straight-line belongs to a specific cluster. Here, K is empirically assigned with one value of K = 28, 29, ..., 37.

2.3. Classification

Classification is the third task in our work where the hand gesture path that was extracted from one video sequence is assigned to one class/label (*i.e.*, class is either alphabet character A to Z or number 0 to 9). There are many problems to learn multiclass, which requires a good selection for the classifier to correctly perform the recognition task. So, our work is based on the Support Vector Machines (SVMs) classifier given its capability for outstanding generalization as well as its highly accurate paradigm. The main motivation behind using SVMs is their structure principle of risk minimization in addition to their ability to alleviate the problem of data over fitting that exist in neural network [12]. The structure of SVMs is to carry out dichotomic classes notably at a higher-dimensional space and it has the ability to construct a maximal separating hyper-plane. Here the strategy work is based on determining the separating hyper-plane by maximization of the distance between two parallel hyper-planes (Figure 4).

By using the hyper-plane, we can further perform a better separation based on the largest distance that could margin a low classifier generalization error.

Formally speaking, suppose that the training dataset is represented as $D = \{(x_i, y_i) \mid x_i \in R^d, y_i \in \{-1, +1\}\}$. Furthermore, Vapnik *et al.* [13] deals with the training problem via permitting some examples to be violated in their margin constraints. Thereby, slack variables are used to formulate the potential violations in addition to a penalty parameter for blocking the margin violations.

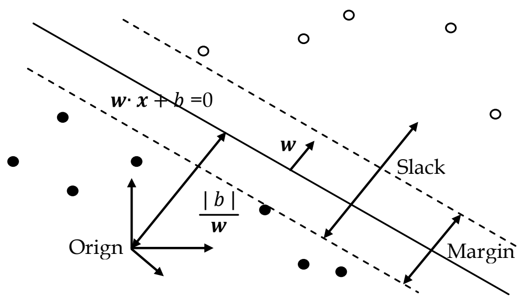

Figure 4. Margin of the Hyper-Plane

The function used to linearly train SVMs is expressed as:

$$f(x) = sign(w.x + b) \tag{6}$$

where w represents a weight vector, x represents the input sample and b refers to a used threshold. The maximum margin of data hyper-planes is maximally separated by the trained SVMs learner. To represent the minimum distance between support vectors and hyper-plane, we maximized the margin of hyper-plane. The SVMs can be formulated as following:

$$\gamma = \frac{2}{\|w\|} \tag{7}$$

where γ is the hyper-plane margin. In Figure 4, the maximization margin of hyper plane is shown and the input data is mapped and separated linearly via SVMs to high dimension domain as depicted in Figure 5.

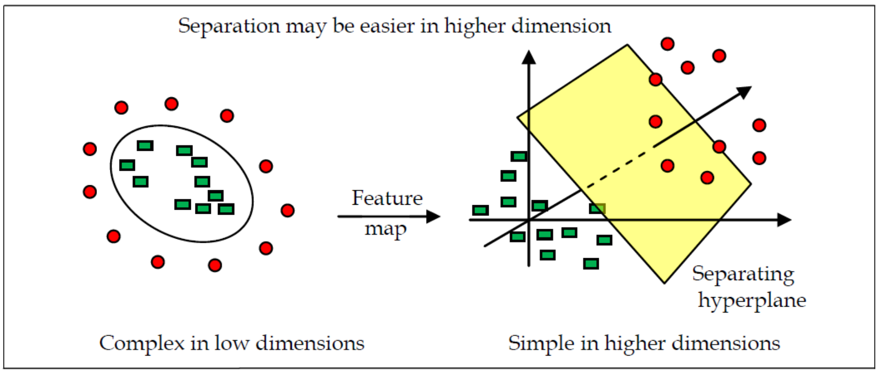

Figure 5. Mapping Process for Input Data from Complex Low Dimension to Simple High Dimension

As a result, the mapping does not influence the time of training process because it has kernel trick and dot product. Since the number of extracted features is high, the SVMs classifier is a better classifier and robust to the dimensionality curse. There are many researches applied on SVMs which achieved promising results against large domains. One advantage of SVM is to achieve the regression optimization during the training and the testing processes. Additionally, the SVM architecture can be updated due to the domain problem by kernel type, margin and duality characteristics. There are many problems such as non-linear and local minima which are supported by SVMs. Above all, SVMs have the ability to discriminately distinguish classes as well as separate them correctly.

Algorithm 1: MultiClass SVM (Training & Testing)

Input: Training set (v_1, l_1), …, (v_N, l_N)
Output: Multiclass Classifier

Training: Binary SVMs and graded relevance scores

for *j = 1 to (k - 1) do*
- For all samples from C_1 to C_j classes, set labels to (+1) and all samples from C_{j+1} to C_k, set labels to (-1)
- Train j^{th} binary SVM
- Classify the training samples
- **if** ($j > 1$), compute fuzzy scores σ_p for all training samples v_p classified as (+1) and define (j - 1) thresholds by splitting the curve of sorted relevance scores into equally spaced intervals.
- **if** ($j < k$), compute fuzzy scores σ_n for all training samples v_n classified as (-1) and define (k-j-1) thresholds by splitting the curve of sorted relevance scores into equally spaced intervals.

end for

Testing: Classification of a new sample z_l

for *j = 1 to (k - 1) do*
- Classify z_l by j^{th} model
- **if** (z_l is classified as (+1))
- **if** ($j = 1$) class$_j$(z_l) C_1 **else** use the defined thresholds to decide class$_j$(z_l)

  else

  if ($j = k - 1$) class$_j$(z_l) C_k **else** use the defined thresholds to decide class$_j$(z_l)

  end if

end for

The decision of SVMs to determine the highest score among all classes is based on the relevance scores which can be estimated in case of two equal-value classes.

So, our algorithm proceeds as follows: suppose that the training data set of N pairs take the form (v1, l1),...,(vN, lN), in which vi ∈Rd represents the vector of reduced feature for a given input image i. In addition to that, the label of crowd density level is expressed as li ∈{C1,…, Ck}. The main idea here is to estimate every binary classifier by relevance scores. By using fuzzy membership, we can automatically grade the crowd using this score, which is employed as a measure for constructing the graded ground truth [13]. It is noted that the ground truth is built in binary label with no manual effort. In feature space, the binary classifier of SVM determines the hyper-plane to optimally separate the two classes. In addition to that, it measures the distance according to the hyper-plane to assign the sample in one class. The fuzzy score (i.e., either positive/negative class posterior probability) is relied on the function of fitting sigmoid and then identified after computing the decision value of f(xs) for every xs training sample (Eq. 8).

$$\sigma_s = \frac{1}{1+\exp(af(x_s)+b)} \tag{8}$$

Here, the two parameters, *a* and *b*, are adapted based on the training dataset. Additionally, the training sample, positive or negative, is sorted according to the fuzzy relevance scores. An adaptive threshold is determined to re-categorize the data sample of each set into various graded crowd level. The multi-SVM classifier is employed to correctly deal with the multiclass problem in which the classes are connected to their increasing relevance score. The multi-SVM algorithm has two advantages. The first is to decrease the time complexity because of constructing only (*k*-*1*) binary SVMs. Finally, every binary classification can easily be transferred to multi-class via relevance scores σ_S.

3. Experimental Results

In our work, the hand is segmented and localized against complex background in stereo color images. The hand was tracked to generate the gesture path by using color information, depth map, Gaussian mixture model and mean-shift algorithm as well as

Kalman filters [15]. When the chrominance channel is employed only for the analysis, the segmentation of skin color turns into more powerful. So, YC_bC_r color space is considered with no Y channel to diminish the brightness variation effect. The Gaussian model is learned based on a collection of skin and non-skin datasets from the internet. Here, the skin dataset includes 20272 skin pixels of 28 various races subjects, but the non-skin dataset contains 128675 non-skin pixels of 96 various image scenes. The initial parameters configuration of GMM is assigned and they are computed using k-mean clustering technique based on the skin and non-skin datasets.

To track the hand, the histogram of hand color in conjunction with Epanechnikov kernel increases the density estimation via assigning small weight to next pixels from the center [15]. The similarity based on Bhattacharyya coefficient [16] is measured by maximizing Bayes error to emerge the likeness between hand candidate and hand target. So, Bhattacharyya coefficient provides the preferable match for hand's target sequentially. The use of mean depth based on the previous candidate hand helps solve the problem of overlapping between hands and face (Figure 6).

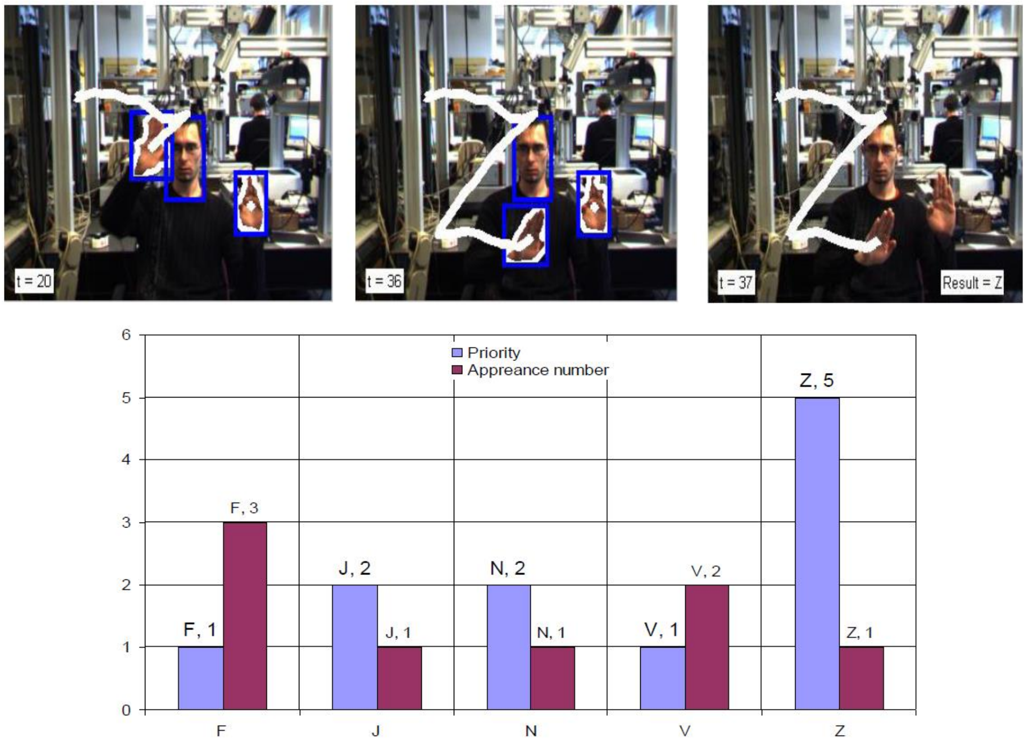

Figure 6. Result of Gesture Path 'Z' where the Bottom Graph Displays the Highest Five Priorities. The Appearance Number of Z is to 1 While its Priority to 5

Sequentially, the mean shift vector is estimated and then optimized recursively using the mean-shift function. We notice that the hand target location with its waif estimation is computed at each step of mean-shift optimization. After that, the Kalman iteration predicates and drives the hand target position. Among successive image frames, the gesture path is constructed by joining the centroid points of target hand regions. We used Kinect camera system to capture image sequences at 30FPS with 240 × 320 image resolution. The proposed system was carried out by Matlab language and C++ language.

An application of isolated hand gesture path for alphabet characters (A-Z) and numbers (0-9) was conducted to address the effects of recognition in real-time. We considered multi-class SVM classifier for a gesture path of a single hand in color image sequences. Our experiment demonstrated that each gesture (*i.e*, from A to Z and number from 0 to 9, equal to 36 symbols) was based on 30 videos where ten were for testing and twenty were for learning. We can say that the dataset contains 360 image sequences for testing and 720 image sequences for learning gestures. In Figure 6 (bottom), the result is related to gesture path 'Z' because of highest priority which coupled appearance number of alphabet character Z.

To evaluate our proposed method, we base on the following criteria;

We considered *t* refers to number of testing data with value 10 for every gesture path such that the valid gesture recognition is denoted by u_j and invalid gesture path by $\bar{u}_j$ as follows;

$$t = u_j + \bar{u}_j \ , \qquad j = 1,2,\ldots,36 \tag{9}$$

Here, the index *j* refers to the hand gesture path of alphabets characters or number. Eq. 10 estimates the valid recognition percentage for each gesture path while the total percentage of recognizing all testing gesture paths are provided by Eq. 11.

$$h_j = \frac{u_j}{t}.100 \tag{10}$$

$$\Omega = \frac{1}{36}\sum_{j=1}^{36}\Gamma_j \tag{11}$$

where Γ_j represents the result of each number or alphabet character and Ω refers to total recognition value of our testing dataset. In similar, we estimate the recognition on training dataset for each alphabet character and number. It is noted that the higher priority is estimated using multi-class SVM in addition to the number of appearance for each gesture path as in Figure 6. Our experiment introduced and yielded promising results and achieved 98.61% and 93.89% as recognition rates for learning and testing dataset respectively.

4. Conclusion and Future Work

In this work, we proposed a robust method for recognizing Arabic numbers (0 - 9) and the Latin alphabet characters (A - Z) using multi-class SVM in color videos. The gesture path of a single hand will take place using multi-class SVM, which is adequate for real-time implementations. The proposed method is described in three main stages.

First, the region of target hand is detected, localized and tracked using g mean-shift procedure with Kalman filter as a preprocessing stage. Second, the extracted affine features of orientations, locations and velocity are quantized using K-mean clustering, and then employed to the classifier either for learning or testing. Finally, multi-class SVM algorithm is to performing the learning and testing processes. The hand gesture dataset of alphabet characters A to Z and number 0 to 9 includes on 360 and 720 videos for testing and learning respectively. Our experiment introduced and yield promising results and achieved 98.61% and 93.89% recognition rates for learning and testing dataset respectively. Hereafter, our researches will investigate the hand gesture path by fingertip detection which makes the system more realistic.

Acknowledgement

I would like to deeply thank Dr.-Ing. Mahmoud Elmezain, faculty of Science, Tanta University, Egypt on his ample efforts by supplying me the hand gesture dataset that helped me to apply my research.

References

[1] N. P. Vassilia and G. M. Konstantinos, "On Feature Extraction and Sign Recognition for Greek Sign Language", International Conference on Artificial Intelligence and Soft Computer, **(2003)**, pp. 93-98.
[2] F. M. Ghaled, E. A. Youness, M. Elmezain and F. Sh. Dewdar, "Vision-Based Hand Gesture Spotting and Recognition Using CRF and SVM", Journal of Software Engineering and Applications, vol. 8, no. 7, **(2015)**, pp. 313-323.
[3] F. M. Ghaled, E. A. Youness, M. Elmezain and F. Sh. Dewdar, "SVM and HMM-Based Hand Gesture Spotting and Recognition in Stereo Color Image Sequences", Information: An International Interdisciplinary Journal, vol. 18, no. 18, **(2015)**, pp. 3641-3654.
[4] R. Bowden, A. Zisserman, T. Kadir, and M. Brady, "Vision Based Interpretation of Natural Sign Languages", In International Conference on Computer Vision Systems, **(2003)**.
[5] M. Hussain, "Automatic Recognition of Sign Language Gestures", Master Thesis, Jordan University of Science and Technology, **(1999)**.
[6] M. Handouyahia, D. Ziou, and S. Wang, "Sign Language Recognition Using Moment-Based Size Functions", In International Conference of Vision Interface, **(1999)**, pp. 210-216.
[7] S. Malassiotis and M. Strintzis, "Real-time Hand Posture Recognition using Range Data", In Image and Vision Computing, ISSN 0262-8856, vol. 26, no. 7, **(2008)**, pp. 1027-1037.
[8] A. Licsar, and T. Sziranyi, "Supervised Training Based Hand Gesture Recognition System", In International Conference on Pattern Recognition, **(2002)**, pp. 999-1002.
[9] W. Freeman and M. Roth, "Orientation histograms for hand gesture recognition", In International Workshop on Automatic Face and Gesture Recognition, **(1994)**, pp. 296-301.
[10] S. Askar, Y. Kondratyuk, K. Elazouzi, P. Kauff and O. Schreer, "Vision-Based Skin-Colour Segmentation of Moving Hands for Real-Time Application", 1st European CVMP, **(2004)**, pp. 79-85.
[11] C. Ding and X. He, "K-means Clustering via Principal Component Analysis", In International Conference on Machine Learning (ICML), **(2004)**, pp. 225-232.
[12] V. Vapnik, "The Nature of Statistical Learning Theory", Springer, New York, NY, USA, **(1995)**.
[13] M. Redi and B. M´erialdo, "A multimedia retrieval framework based on automatic graded relevance judgments", In 18th International Conference on Multimedia Modeling (MMM), **(2012)**.
[14] M. Elmezain, A. Al-Hamadi, J. Appenrodt and B. Michaelis, "A Hidden Markov Model- Based Continuous Gesture Recognition System for Hand Motion Trajectory", In International Conference on Pattern Recognition (ICPR), **(2008)**, pp. 519-522.
[15] D. Comaniciu, S. Ramesh and P. Meer, "Kernel-Based Object Tracking", In IEEE Transaction on PAM I, ISSN 0162-8828, vol. 25, no. 5, **(2003)**, pp. 564-577.
[16] S. Khalid, U. Ilyas, S. Sarfaraz and A. Ajaz, "Bhattacharyya Coefficient in Correlation of Gary-Scale Objects", In Journal Multimedia, ISSN 1796-2048, vol. 1, no. 1, **(2006)**, pp. 56-61.

Author

Nadhir Ben Halima, received his BSc in Computer Engineering from the National School of Computer Sciences (ENSI), Mannouba, Tunisia, the M.S degree in Communication Networks Engineering from SantAnna School of Advanced Studies, Pisa, Italy in 2006 and the PhD in Information and Communication Technology in 2009 from the University of Trento, Italy. In 2009 he was a visiting researcher at the Department of Electrical and Computer Engineering at North Carolina State University, Raleigh, USA. Since September 2011, he is Assistant Professor and Vice Dean of the College of Computer Science and Engineering at Taibah University, Yanbu Branch, Saudi Arabia. His research interest includes Wireless and Sensor Networks, Cognitive Networks, Image Processing, Image Watermarking and Multi-Robot Systems.