Key Video Object Plane Selection by MPEG-7 Visual Shape Descriptor for Summarization and Recognition of Hand Gestures

M.K. Bhuyan
ECE Department
IIT Guwahati
manas_kb@iitg.ernet.in

D. Ghosh
ECE Department
IIT Guwahati
ghosh@iitg.ernet.in

P.K. Bora
ECE Department
IIT Guwahati
prabin@iitg.ernet.in

Abstract

The use of human hand as a natural interface for human-computer interaction (HCI) serves as the motivation for research in hand gesture recognition. Vision-based hand gesture recognition involves visual analysis of hand shape, position and/or movement. Due to co-articulation that occurs during transition from one gesture to the next, problem is encountered in continuous hand gesture recognition. This may be tackled by identifying the key frames in the gesture sequence. Key frames are those that best represent the content of a video sequence in an abstracted manner. In this paper, we present an object-based scheme for key frame extraction using Angular Radial Transformation (ART) shape descriptor for gesture representation. We propose a finite state machine (FSM) in which gestures are represented and subsequently recognized by the sequence of key frames and the corresponding key frame duration. Experimental results obtained demonstrate the effectiveness of our proposed scheme for key frame extraction, subsequent gesture summarization and finally gesture recognition.

1. Introduction

One very interesting field of research in Pattern Recognition that has gained much attention in recent times is Gesture Recognition. Gesture may be described as the manner in which a person moves his body and limbs to express an idea or sentiment. People frequently use gestures to communicate in their day-to-day life. Therefore, gestures are a natural means of conveying information. This has motivated to use gestures for communicating with computers. Thus, gestures provide an attractive and user-friendly alternative to interface devices like keyboard, mouse and joysticks for human-computer interaction (HCI). Accordingly, the basic aim of gesture recognition research is to create a system which can identify/interprete specific human gestures automatically and use them to convey information (i.e., communicative as in sign-language communication) or for device control (i.e., manipulative as in controlling robots without any physical contact between human and computer). One type of human gesture of particular interest is hand gesture where the position and shape of the hand convey information. Although hand gestures are complicated to model since the meanings of hand gestures depend on people and cultures, a set of specific hand gesture vocabulary can be always predefined in many applications, so that the ambiguity can be limited.

To exploit the use of hand gestures in man-machine communication it is necessary that the static and/or dynamic configuration of the human hand be measurable by the machine. Initial attempts to solve this problem resulted in mechanical devices, e.g., glove-based devices, that directly measure hand/arm joint angles and its spatial position. But, glove-based gestural interface requires to wear a cumbersome glove that carries a load of cables connecting it to the computer. This hinders the naturalness with which the user can interact with the computer.

Awkwardness in using gloves is overcome by using vision-based non-contact interaction techniques. They use color-based vision segmentation, silhouettes or edges to track the hand and fingers. Unfortunately, most of the works on vision-based gestural HCI have mainly been focused on the recognition of static hand gestures or postures. But, hand gestures are in general dynamic actions where the motion of the hands conveys as much meaning as their posture does. So, appropriate interpretation of dynamic gestures on the basis of hand movement in addition to shape and position is necessary.

All approaches to hand/finger tracking require to locate hand regions in video sequences. Skin color offers an effective and efficient way to segment hand regions out. However, many of these techniques are plagued by some special difficulties such as large variation in skin tone, unknown lighting conditions and dynamic scenes. A solution to this is the 3D model-based approach, in which the hand configuration is estimated by taking advantage of 3D hand models [7]. However, they lack the simplicity and computa-

tional efficiency. An alternative approach is the appearance-based models [10]. Although it is easier for the appearance-based approach to achieve user-independence than model-based approach, there are two major difficulties associated with this approach, viz., automatic feature selection and training data collections. Another important approach for hand tracking is the prediction algorithm combined with Kalman tracking and application of probabilistic reasoning for final inference [12]. But, this approach requires fine initial model and also requires additional computations for rotation and change in shape of the model. Moreover, it is very much sensitive to the background noise.

As mentioned earlier, hand gestures may be either discrete (static) or continuous (dynamic). Continuous gestures are composed of a series of gestures that as a whole bears some meaning. As a first step towards recognition, a continuous gesture sequence needs to be segmented into its component gestures. However, the process is complicated due to the phenomenon of 'co-articulation' in which one gesture influences the next in a temporal sequence [8]. This happens due to hand movement during transition from one gesture to the next. The problem is very significant in case of fluent sign language. Recognition of co-articulated gestures is one challenge in gesture recognition research.

Toward this goal, an effort has been made to solve the problem of co-articulation by selecting key frames in a sequence of gestures. In our proposed technique, we use the concept of object-based video abstraction for segmenting the frames into video object planes (VOPs), as used in MPEG-4, with each VOP corresponding to one semantically meaningful hand position. The hand position is then tracked in the binarized frame sequence using the change detection algorithm [5]. Next, we select the key VOPs on the basis of shape dissimilarity measure using Angular Radial Transformation (ART) based shape descriptor as used for shape description in MPEG-7 multimedia content description interface [6]. The key frames transform an entire video clip to a small number of representative images that are sufficient to represent a particular gesture sequence. We believe that by doing so we can cope with the co-articulation problem. Hence, our proposed scheme is capable of detecting several gestures connected sequentially without the curse of co-articulation.

Since a gesture can be defined as an ordered sequence of states in the spatial-temporal space, we represent a particular gesture as a sequence of key frames and the corresponding key frame duration, which constitute a finite state machine (FSM). For recognition, the shape similarity between the shapes of the incoming data sequence and the states of the FSM is measured by ART shape descriptor. Our proposed scheme is described in more details in the section to follow.

2. Proposed Scheme for Hand Gesture Recognition

Figure 1 shows the basic block diagram for the proposed hand gesture recognition system. From the input gesture video sequence VOPs for different hand positions are obtained. By measuring shape similarity by ART shape descriptor, key VOPs are extracted. These key VOPs are the input to the gesture classification system that uses state based approach [2] for representation and recognition of gestures. The steps involved in our proposed scheme are described below.

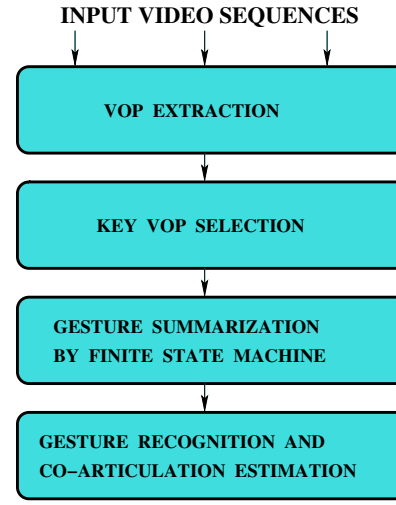

Figure 1. Block diagram for the proposed scheme

2.1. Segmentation of hand image

In our scheme, we use the inter-frame change detection algorithm [5] for segmenting out hand regions in the input video sequences. The advantage of using this algorithm is that it enables automatic detection of objects and allows larger non-rigid motion compared to object tracking methods. The small holes and falsely detected parts can be subsequently removed by morphological filtering.

2.2. Extraction of moving edge (ME) map

The edge map of images are calculated by using Canny edge detector. Edge extraction from the difference image in successive frames results in a noise-robust difference edge map DE_n because Gaussian convolution included in the Canny operator suppresses the noise in the luminance difference. The moving edge ME_n of the current frame I_n is extracted based on the edge map DE_n of the difference image $\mid I_{(n-1)} - I_n \mid$, the current frame's edge map $E_n = \phi(I_n)$, and the background edge map E_b. If DE_n denotes the set

of all pixels belonging to the edge map of the difference image, then the moving edge model generated by edge change is given by selecting all edge pixels within a small distance T_{change} of DE_n, i.e.,

$$ME_n^{change} = \{e \in E_n \mid \min_{x \in DE_n} \parallel e - x \parallel \leq T_{change}\} \quad (1)$$

In addition to this, moving edges in the previous frames can be referenced to detect temporarily still moving edges, i.e.,

$$ME_n^{still} = \{e \in E_n \mid e \notin E_b, \min_{x \in ME_{n-1}} \parallel e - x \parallel \leq T_{still}\} \quad (2)$$

The final moving edge map for current frame I_n is expressed by combining the two maps.

$$ME_n = ME_n^{change} \cup ME_n^{still} \quad (3)$$

2.3. Extraction of VOPs and key VOP selection

The region inside the first and the last edge points in a row is a horizontal candidate for the object in a frame while that in each column is the vertical candidate. After finding all the horizontal and the vertical candidates in a frame, the VOP is generated by logical AND operation and further processing by alternative use of morphological operations like closing and filling.

After the VOPs are extracted,binary alpha planes are generated. A binary alpha plane indicates whether or not a pixel belongs to a VOP. After getting the binary alpha planes, the key VOPs are selected using ART based shape dissimilarity measure.

2.4 Region-based shape descriptor

The region-based shape descriptor expresses pixel distribution within a 2-D object region; it can describe complex objects consisting of multiple disconnected regions as well as simple objects with or without holes. Consequently, an ART based descriptor was recently adopted by MPEG-7 [4]. Conceptually, the descriptor works by decomposing the shape into a number of orthogonal 2-D basis functions (complex-valued), defined by the Angular Radial Transform (ART) [1]. The normalized and quantized magnitudes of the ART coefficients are used to describe the shape.

Angular Radial Transformation (ART)

From each shape, a set of ART coefficients F_{nm} is extracted, using the following formula:

$$F_{nm} = \ < V_{nm}(\rho, \theta), f(\rho, \theta) >$$

i.e.,

$$F_{nm} = \int_0^{2\pi} \int_0^1 V_{nm}^*(\rho, \theta), f(\rho, \theta) \rho d\rho d\theta \quad (4)$$

where $f(\rho, \theta)$ is an image function in polar coordinates and V_{nm} is the ART basis function that are separable along the angular and radial directions, that is,

$$V_{nm}(\rho, \theta) = \frac{1}{2\pi} \exp(jm\theta) R_n(\rho) \quad (5)$$

$$R_n(\rho) = \begin{cases} 1 & \text{if } n = 0 \\ 2\cos(\pi n \rho) & \text{if } n \neq 0 \end{cases} \quad (6)$$

Descriptor Representation

The ART descriptor is defined as a set of normalized magnitudes of complex ART coefficients. Twelve angular and three radial functions are used $(n < 3, m < 12)$. For scale normalization, ART coefficients are divided by the magnitude of ART coefficient of order $n = 0, m = 0$. Therefore, discarding the normalized ART co-efficient of order $n = 0, m = 0$, which is unity, we have 35 coefficients in all. To keep the descriptor size to a minimum, quantization is applied to each coefficient using four bits per coefficient. Hence, the default region-based shape descriptor has total 140 bits.

Shape Similarity Measure

The distance (or dissimilarity) between two shapes described by the ART descriptor is calculated using an $L - 1$ norm, for example, by summing up the absolute differences between ART coefficients of equivalent order $(L = 2)$.

$$Dissimilarity = \sum_i \parallel M_d[i] - M_q[i] \parallel \quad (7)$$

Here, the subscript d and q represent image in the database and query image, respectively and M is the array of ART descriptor values.

For key VOP selection, the first VOP of a video sequence is declared as a key VOP, and whenever the shape dissimilarity measure in terms of ART coefficients between the mass center aligned contours of a key VOP candidate and its temporally closest key VOP is larger than a predefined threshold, the key VOP candidate is selected as a new key VOP [3].

2.5. Finite states representation of gestures

The proposed FSM for gesture recognition consists of a finite number of key frames and the corresponding key frame

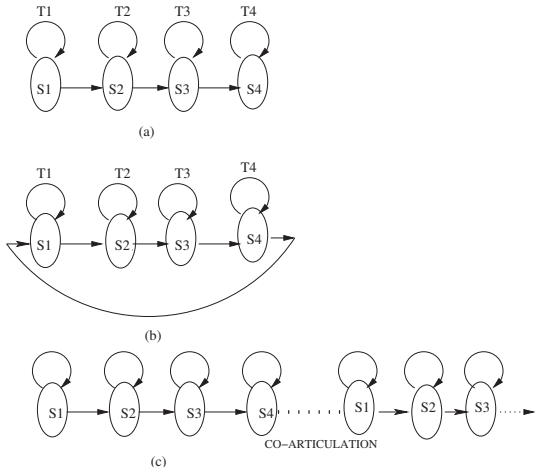

Figure 2. (a) Finite state representation and summarization of a gesture. (b) FSM for the same gesture repeated again and again. (c) FSM for the gestures connected sequentially with co-articulation.

durations as shown in Figure 2. ART based shape descriptor is used to represent the shape of the video object in the key frames. The state transition occurs only when the shape similarity of key VOPs (KVOP) and duration criteria are met. A threshold T_s is pre-defined to allow a certain degree of spatial variance in each state and a counter C_{dur} is used to judge the key frame duration criterion of FSM.

During training, the input data sequence is segmented into state sequence (i.e., finite number of key frames). This gives the representation of the particular gesture in a summarized format. Thus, an FSM for gesture summarization is constructed. The state representation algorithm is given below.

Algorithm : Training of FSM for Gesture Recognition

1. ***begin initialize*** $\mathbf{C_{max}}, \mathbf{C_{min}}, \mathbf{T_s}, \mathbf{n_{max}}$
2. $m \leftarrow 0$
3. ***do*** $m \leftarrow m + 1$
4. *read the training data sequence* d_m
5. *generate VOPs* $(VOP_1, VOP_2 .... VOP_{n_{max}})$
6. *assign* VOP_1 *as key VOP*
7. $n \leftarrow 1$
8. ***do*** $n \leftarrow n + 1$
9. ***if*** *shape of* VOP_{n-1} *and* VOP_n *are not similar*
10. *assign* VOP_n *as key VOP*
11. *count and store* C_{dur_n} *for* $KVOP_n$
12. ***until*** $n = n_{max}$
13. ***until*** $C_{min} \leq C_{dur} \leq C_{max}$ *for each KVOP*
14. *and shape convergence criterion met for each KVOP*
15. ***return*** *KVOP,* $\mathbf{C_{dur}}$
16. ***end***

The counter value $\mathbf{C_{max}}$ and $\mathbf{C_{min}}$ assign the allowable ranges of duration for each KVOP. The values of $\mathbf{C_{max}}$, $\mathbf{C_{min}}$ and $\mathbf{T_s}$ for a particular gesture are obtained by running the algorithm for several times.

2.6. Recognition and co-articulation estimation

As mentioned earlier, when different gestures are occurring sequentially co-articulation problem arises as shown in Figure 2(c). For recognition purpose the incoming gesture states are matched with the states of finite state machine. Recognition is nothing but a string matching between a data sequence and the state sequence of an FSM. For each new data sequence, the gesture recognizer decides whether to stay at the current state or to jump to the next state based on the shape similarity and duration criteria. If all the states of the FSM are passed successfully then a gesture is recognized. Else, co-articulation is detected.

3. Experimental Results

3.1. Test video sequences and extracted key frames

In our experiment, we have used ten different gesture sequences (labelled 0-9) taken from Sebastien Marcel's gesture database and Thomas Moeslund's gesture recognition database. The results for KVOP selection for four sequences, viz.,"Rotate" sequence, "No" sequence, "Clic" sequence and "Stop Grasp OK" sequence, are shown in Figures 3-6. First row of each figure shows the original video sequences, second and third rows show the difference edge DE_n and moving edge ME_n respectively, fourth and fifth rows show horizontal and vertical candidates of a particular VOP respectively. The combination of vertical and horizontal can be found by logical AND operation which is shown in the sixth row. After morphological filtering we get the binary alpha plane corresponding to each VOP, which is shown in the seventh row in each of the figures. The last row of each figure shows the key binary alpha planes, which are sufficient to represent a particular gesture. Thus, the last row of each of the figures illustrates the summarization of long gesture video sequence, where redundant frames of the video sequences are discarded.

The classification results corresponding to different gesture sequence is shown in Table 1. It is seen that gesture recognition and classification accuracy rate of our proposed method is somewhat comparable with the other methods of gesture recognition, specifically HMM methods for gesture recognition, both the approaches give recognition rate in the range 80-95% [9], [11]. However, in our proposed approach, unlike HMM, a gesture model is available immediately. The statistical nature of an HMM precludes a rapid training phase. To train a HMM, well-aligned data segments are required, whereas in the FSM representation the training

Table 1. Experimental Results: Gesture Recognition Rate

No. of training samples per class : 50
No. of test samples per class : 25

Actual class label	No. of test pattern assigned to predefined class											Acc rate	Err rate	Rej rate
	0	1	2	3	4	5	6	7	8	9	Reject			
0	23	0	1	0	1	0	0	0	0	0	0	92.0	8.0	0.0
1	1	20	0	1	0	2	0	0	0	1	0	80.0	20.0	0.0
2	0	1	19	0	1	0	1	0	0	0	3	76.0	12.0	12.0
3	0	0	1	22	0	1	0	0	0	0	1	88.0	8.0	4.0
4	0	0	0	0	23	0	0	0	0	0	2	92.0	0.0	8.0
5	0	0	0	0	1	24	0	0	0	0	0	96.0	4.0	0.0
6	1	1	1	0	0	0	20	1	1	0	0	80.0	20.0	0.0
7	1	2	0	0	0	0	0	20	1	0	1	80.0	16.0	4.0
8	0	0	0	2	0	0	0	0	21	1	1	84.0	12.0	4.0
9	2	0	1	0	0	0	1	0	0	20	1	80.0	16.0	4.0
Average												**84.8**	**11.6**	**3.6**

data is segmented and aligned simultaneously to produce a gesture model.

4. Conclusions

From the test results it is seen that by using key frames, a particular gesture can be uniquely determined and can be represented in terms of a finite state machine with key frames and corresponding frame duration as states. One notable advantage of finite state representation of gesture is that it handles different gestures consisting of different number of states. The key frame based gesture representation is nothing but the summarization of the gesture with finite number of unique states. The advantage of key frame based state representation is that only the shape similarity measurement for key frames are required instead of all frames of the video sequence. Moreover key frame based gesture classification can solve the co-articulation problem to a greater extent. The key frame based gesture representation is equally useful for both gesture recognition and coding of video frames in compressed domain.

References

[1] Miroslaw Bober. MPEG-7 Visual Shape Descriptors. In *IEEE Trans. Circuits and Systems for Video Technology,*, vol. 11, no. 6, pp. 716-719, 2001.

[2] A.F. Bobick and A.D. Wilson. A state-based approach to the representation and recognition of gesture. In *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 12, pp. 1325-1337, 1997.

[3] B. Erol and F. Kossentini. Automatic key video object plane selection using the shape information in the MPEG-4 compressed domain. In *IEEE Trans. Multimedia*, vol. 2, no. 2, pp. 129-138, 2000.

[4] B. Erol and F. Kossentini. Local motion descriptors. In *Proc. IEEE 4th Workshop on Multimedia Signal Processing*, pp.467-472, 2001

[5] C. Kim and J.-N. Hwang. Object-based video abstraction for video surveillance systems. In*IEEE Trans. Circuits and Systems for Video Technology*, vol. 12, no. 12, pp. 1128-1138, 2002.

[6] B.S.Manjunath, Philippe Salembier and Thomas Sikora, ed., *Intoduction to MPEG-7, Multimedia Content Description Interface*. John Wiley and Sons, Ltd.

[7] V.I. Pavlovic, R. Sharma and T.S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. In *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 677-695, 1997.

[8] A. Shamaie, W. Hai and A. Sutherland. Hand-Gesture Recognition for HCI. In *ERCIM News (on line edition), http://www.ercim.org/publication/Ercim_News*, no. 46, 2001,

[9] Christian Vogler and Dimitris Metaxas. ASL Recognition Based on a Coupling Between HMM and 3D Motion Analysis. In *Proceedings of the International Conference on Computer Vision*, pp. 363-369, 1998.

[10] Y. Wu and T.S. Huang. Self-supervised learning for visual tracking and recognition of human hand. In *Proc. 17th Natl. Conf. Artificial Intelligence (AAAI'2000)*, pp. 243-248, 2000.

[11] J. Yamato, J. Ohya, K. Ishii. Recognizing Human Action in Time-Sequential Images using Hidden Markov Model. In *Proc. of IEEE CVPR*, pp. 379-385, 1992

[12] J. Zieren, N. Unger and S. Akyol. Hands tracking from frontal view for vision-based gesture recognition. In *Proc. DAGM Symp.*, pp. 531-539, 2002.

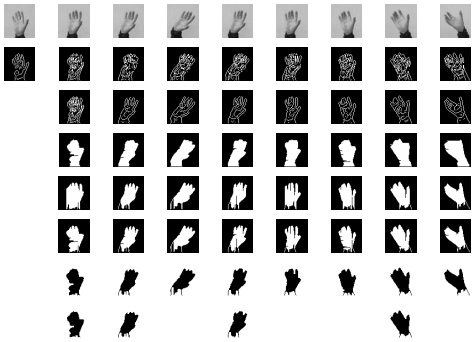

Figure 3. Test results for “Rotate" sequence

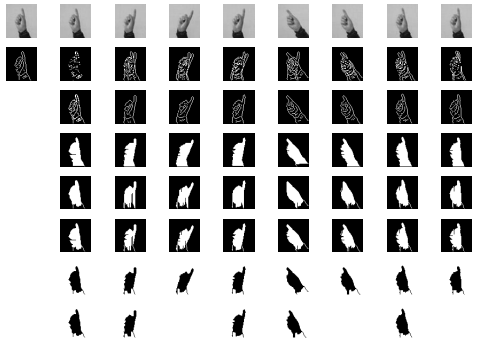

Figure 4. Test results for “No" sequence

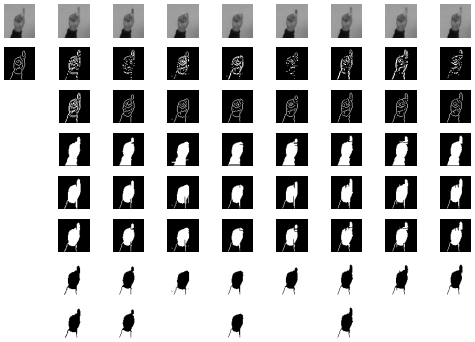

Figure 5. Test results for “Clic" sequence

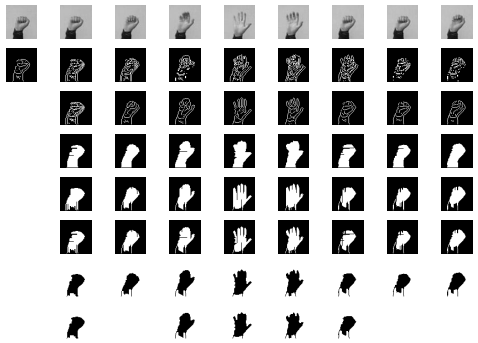

Figure 6. Test results for “Stop Grasp OK" sequence