Real – time Two Hand Gesture Recognition with Condensation and Hidden Markov Models

Tanatcha Chaikhumpha
Computer Science Department, Faculty of Informatics
Mahasarakham University,
Mahssarakham, Thailand
dksweet2013@gmail.com

Phattanaphong Chomphuwiset
Computer Science Department, Faculty of Informatics
Mahasarakham University,
Mahssarakham, Thailand
phatthanaphong.c@gmail.com

Abstract— **This paper presents a technique for two-hand gesture recognition. There are two main processes performed to recognize hands, i.e. hand tracking and gesture cognition. In the hand tracking process, a condensation density propagation (Condensation algorithm) is used to localize and track hands (centered at the center of palms) when they are moving. Hidden Markov Model is implemented in recognition part to understand the gestures performed by human. There are 8 gestures used in this study. The experiment is conducted on the collected data (of 8 hand gestures). The results show that the proposed technique provides a promising results, achieving 96.25%.**

Keywords—Real-time processing; Hand gesture recognition; Hand tracking; condensation; HMM;

I. INTRODUCTION

With the advanced technology available currently, hand gesture recognition is one of expedient computer vision applications that aimed to perform by in real-time systems [1]. To achieve the task of gesture recognition in video images, image processing and machine learning techniques are involved. The process can be decomposed from a very low-image processing procedures to a high level computation to understand hand gestures. The recognition is to assign a certain class of gesture types to an image extracted from the videos. There are three main processes that have been implemented to recognize hand gestures [1],[2] , which are as follows: (1) image acquisition and hand detection, (2) feature extraction and (3) gesture recognition, can be performed by as a classification module that is aimed to classify the gesture into different types.

II. GESTURE RECOGNITION SYSTEM

In this is work, we propose a system that recognizes both isolated and meaningful 8 gestures from real-time stereo color images. In general, images are acquired as a sequence of still images. Objects (hands) the images are detected and the trajectory of the moving objects is determined to identify a gestures obtained by the movement of hands in the images. There are a number of algorithms that have been proposed to identify and recognize gesture using, for example, HMM. Our main motivation is to improve gesture recognition of a person who is using two hands. To achieve the task, skin segmentation is performed to separate hands regions in images. In addition, false positive rejection of head area will be excluded on the detected hands using HAAR-like feature. The overall process is depicted in Fig. 1, and the details of the proposed technique are explained in the following sections.

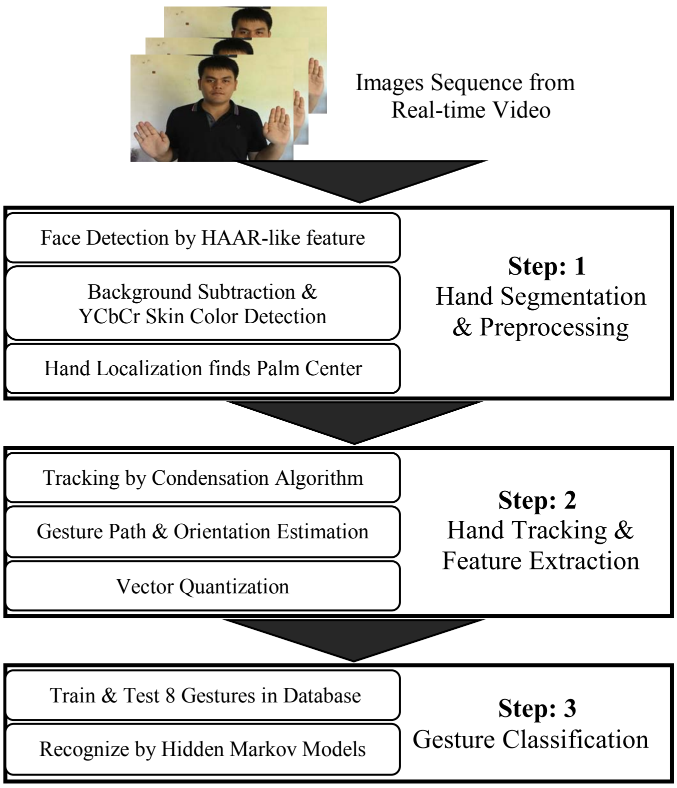


Fig.1. Gesture recognition system using Hidden Markov Models (HMM).

A. *Hand segmentation*

Detecting hands is the first primary step in gesture recognition. In this paper, an area-based technique is used to localize the regions of hands before morphological operations are carried out to smooth the hand regions. There are three steps in this work; face detection, background subtraction and skin color detection with YCbCr color space [3],[5].

1) Face Detection using HAAR-like feature: Face detection is one of the steps of this work. The objective is to separate head or face regions from hands. Haar-like feature-based

978-1-5386-2615-3/18/$31.00 ©2018 IEEE

technique is used [1],[2]. The feature is generated and a classification is performed using AdaBoost technique.

a) AdaBoost Learning Algorithm: AdaBoost is learning algorithm that try to improve a classifier by tweaking a week classifier to a strong one. In this work, sub-windows are classified to into positive (face) and negative (non-face) objects [5],[9]. The classification is performed by adjusting a weak classifier of the features $j(h_j)$ in Eq. (1).

$$h_j(x) = \begin{cases} -1 & if \quad f_j(x) \le 0 \\ 1 & otherwise \end{cases} \tag{1}$$

The value $f_j(x)$ is the sum of the difference in black and white pixels of Haar-like feature is the Eq. 2.

$$f(x) = Sum_{(Black)} - Sum_{(White)} \tag{2}$$

b) Cascade Classifiers: Cascade Classifiers is a way of classifying the desired shape. The classification mentioned above is repeated several times (stage), which in each round will cut off the negative area in every cycle found. At the end of the process, the number of sub-windows that are negative will be reduced to the desired shape. The identification of a desired shape to do this with the picture-in integral as mentioned in the second to make it easier to process.

2) Background Subtraction: moving objects in image scenses are identufied by a background substraction approach. The technique can produce excellent outcomes for detecting fast moving objects in real-time video with background images unchanged or a static background image. Therefore, we apply the technique to idenifyt moving objects in images, as set of hand candidates, before non-hand object are rejected [3],[5].

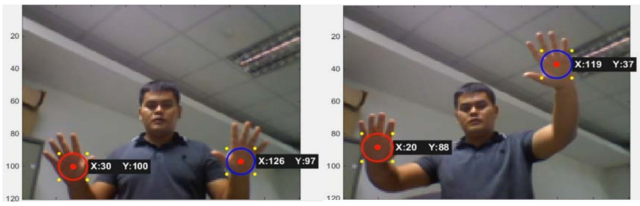
3) Skin Color Detection with YCbCr Color Model

Skin Color Detection with YCbCr Color Model: Moving objects in images comprises with hands and other non-related objects. To detect hands, skin segmentation is applied to of skin is applied. This is to identify hands and reject other object such as face and hed areas. Y CbCr color space is used in our system where Y channel represents brightness and (Cb,Cr) channels refer to chrominance [8]. The difference of color, it is divided into two color Cb and Cr Cb is demonstrated by nuances of blue and cited in the blue division Cr shows the difference of red. And referred to in the red by the YCbCr color fidelity and precision. In terms of illumination [1],[2],[6].

(a) (b) (c)

Fig, 2. Show hand detection (a) original image, (b,c) image after remove background and face

4) Hand Localization finds Palm Center: After the labeled skin (Fig. 2(c)), the localization of two hands is performed by selecting the palm area (Fig. 3.).



(a) (b)

Fig. 3. Hand localization is (a) Hand localization with a circle and centroid point (b) Search area around the detected hands.

B. Hand Tracking and Feature Extraction

1) Hand Tracking: After finding the midpoint of both hands, the next step is to track the movement of the hands -- based on tracking data from the midpoint of both hands [7],[8]. To track the movement of the hands, a Condensation Algorithms (Conditional Density Propagation Algorithm) is implemented. The condensation function algorithm is capable of tracking objects with more than one location and is not prone to computational burden. The algorithm iterativly determines model parameters. Given model parameters, $\{s_t^{(n)}, \pi_t^{(n)}, c_t^{(n)}\}, n = 1,\ldots,N$ At time t from the sample data set and $\{s_{t-1}^{(n)}, \pi_{t-1}^{(n)}, c_{t-1}^{(n)}, n = 1,\ldots,N\}$ At time t -1, the calculation is as follows:

a) Select the following examples.

- Random number generator r is uniform distribution $r \in [0,1]$.
- Find the j at least to makes $c_{t-1}^{(j)} \ge r$
- Determined $s_t'^{(n)} = s_{t-1}^{(j)}$

b) Predicted new position by sampling $p(X_t \mid X_{t-1} = s_{t-1}'^{n})$

c) Find weight $\pi_t^{(n)} = p(z_t \mid X_t = s_t^{(n)})$ use of images z_t

d) Update data by adding examples $\{s_t^{(n)}, \pi_t^{(n)}, c_t^{(n)}\}$ new to the list (t) for $p(x_t)$

With $\pi_t^{(n)} = p(z_t \mid X_t = s_t^{(n)})$

$c_t^{(0)} = 0,$

$c_t^{(n)} = c_t^{(n-1)} + \pi_t^{(n)} \qquad (n = 1 \ldots N)$

e) State Estimation of the state by calculating the Eq. (3)

$$\varepsilon[f(x_t)] = \sum_{n=1}^{N} \pi_t^{(n)} f(s_t^{(n)}) \tag{3}$$

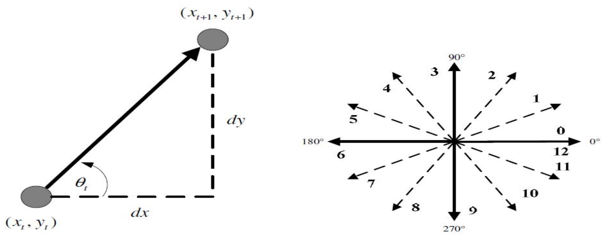
After calculating State Estimation will have trajectory show in (Fig. 4.)

Fig. 4. the trajectory of hand tracking with Condensation Algorithms

2) Feature Extraction: Selecting good features to recognize the hand gesture path plays significant role in system performance. There are three basic features; location, orientation and velocity. 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}) the orientation is determined between two consecutive points from hand gesture path by Eq. (4).

$$\theta_t = \arctan\left(\frac{y_{t+1} - y_t}{x_{t+1} - x_t}\right) \qquad t = 1, 2, \ldots, T-1 \tag{4}$$

where T is represents the length of gesture path. The orientation θ_t is quantized by dividing it by 30° in order to generate the codewords from 1 to 12 (Fig. 5). Also the codewords contain zero codeword notably in case of continuous gesture. Thereby, the discrete vector is determined and then is used as input to HMM.



(a) (b)

Fig. 5. The orientation and its codewords (a) orientation between two consecutive points (b) directional codewords from 1 to 12 including also zero codeword.

C. *Hand Gesture Recognition*

The final step in our system is performing a classification. Throughout this stage, the isolated and continuous gestures paths are recognized by HMM forward-backward algorithm in conjunction with Viterbi path over its discrete vector. Moreover, Baum- Welch algorithm is used to do a full training for the initialized HMM parameters by the discrete vector to construct gestures database. The gestures database contains 50 video sequences for each isolated gesture from gesture control 8 gesture by 1 gesture has 50 video. So, 8 gestures will have a total of 400 videos for training with Baum-Welch algorithm. The following subsections describe this stage in some details.

1) Hidden Markov Models: The markov model is a mathematical model of stochastic process where these processesgenerate a random sequence of outcomes according to certain probabilities [4],[10]. An HMM is a triple $\lambda = (A, B, \pi)$ as follows:

f) The set of states $S = \{S_1, ..., S_N\}$ where N is a number of states.

g) Initial probability for each state $\pi_i, i = 1, 2, \ldots, N$ such that $\pi_i = P(s_i)$ at the initial step.

h) 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.

i) The set of possible emission (an observation) $O = O_1, O_2, \ldots, O_T$ where T is the length of gesture path.

j) The set of discrete symbols $V = v_1, v_2, \ldots, v_M$ where M represents the number of discrete symbols.

k) 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.

2) HMM states: The number of states is an important parameter because the excessive number of states can generate the over-fitting problem if the number of training samples is insufficient compared to the model parameters. The number of states in our gesture recognition system is based on the complexity of each gesture number (TABLE I) and is determined by mapping each straight-line segment into a single HMM state. We used Left-Right Banded (LRB) model that also each state can go back to itself or the following state only.

3) Initializing parameters for LRB model: Matrix A is the first parameter and is determined by Eq. (7). It depends on the duration time d of states for each number such that d is defined as;

$$d = \frac{T}{N} \tag{6}$$

where T is the length of gesture path and N represents the number of states that has a value 5 in our system.

$$A = \begin{pmatrix} a_{ij} & 1-a_{ij} & 0 & 0 & 0 \\ 0 & a_{ij} & 1-a_{ij} & 0 & 0 \\ 0 & 0 & a_{ij} & 1-a_{ij} & 0 \\ 0 & 0 & 0 & a_{ij} & 1-a_{ij} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{7}$$

Such that:

$$a_{ij} = \frac{1}{d} \tag{8}$$

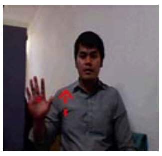
The second important parameter is matrix B that is determined by Eq. 9. Since HMM states are discrete, all elements of matrix B can be initialized with the same value for all different states.

$$b_{im} = \frac{1}{M} \qquad (9)$$

where i, m run over the number of states and the number of discrete symbols respectively. The third HMM parameter is the initial vector π which takes value;

$$\pi = (1 \quad 0 \quad 0 \quad 0 \quad 0)^T \qquad (10)$$

TABLE I. TABLE SHOW ORIENTATION SEQUENCE 8 GESTURE

Gesture	Gesture	Left hand feature	Right hand feature
Move Up		Orientation sequence = NULL	Orientation sequence = [3, 3, 3, 3, 3]
Move Down		Orientation sequence = NULL	Orientation sequence = [9, 9, 9, 9, 9]
Spread		Orientation sequence = [6, 6, 6, 6, 6]	Orientation sequence = [12, 12, 12, 12, 12]
Reduce		Orientation sequence = [12, 12, 12, 12, 12]	Orientation sequence = [6, 6, 6, 6, 6]
Shuffle		Orientation sequence = [3, 3, 3, 3, 3]	Orientation sequence = [9, 9, 9, 9, 9]
Move Right		Orientation sequence = NULL	Orientation sequence = [6, 6, 6, 6, 6]
Move Left		Orientation sequence = NULL	Orientation sequence = [12, 12, 12, 12, 12]
Rotate		Orientation sequence = [9, 10, 10, 10, 10]	Orientation sequence = [9, 4, 4, 4, 4]

III. EXPERIMENTAL RESULTS

Our proposed system provides good results to recognize gesture in real-time from color image sequences via the motion trajectory of a single hand and multiple using HMM. In our experimental, the results gesture from 8 gesture, which 50 video samples for training and testing by 4 persons each person test 8 gesture and 1 gesture testing 50 time.

TABLE II. HMM GESTURE RECOGNITION RESULTS FOR 8 GESTURE

Gesture	Data Test	Correct	Error	Accuracy (%)
Rotate	200	186	14	93
Shuffle	200	195	5	97.5
Reduce	200	192	8	96
Spread	200	191	9	95.5
Move Up	200	193	7	96
Move Down	200	194	6	97
Move Left	200	195	5	97.5
Move Right	200	195	5	97.5
Average	200	192.625	7.375	96.25

IV. SUMMARY AND CONCLUSION

This paper proposes a system to recognize meaningful gestures for control gesture from color image sequences by the motion trajectory of a single hand and multiple hands using condensation tracking and HMM which is suitable for real-time application. Our results show that; an average recognition rate is 96.25%.

REFERENCES

[1] S. Mitra and T. Acharya, "Gesture Recognition: A Survey," IEEE Transactions on systems, Man and Cybernetics, Part C (Applications and reviews), Vol. 37(3), April 2007, pp.311-324.

[2] Z. K. Rafiqul and A. I. Noor, "Hand Gesture Recognition: A Literature Review," International Journal of Artificial Intelligence & Applications (IJAIA), Vol. 3(4), 2012, pp. 161-174.

[3] X. Wenkai and J. L. Eung, "A New NUI Method for Hand Tracking and Gesture recognition Based on User Experience," International Journal of Security and Its Applications, Vol. 7(2), 2013, pp. 149-158.

[4] S. C. Feng, M. F. Chih and L. H. Chung, "Hand gesture recognition using a real-time tracking method and hidden Markov models," Image and Vision Computing, Vol. 21, 2003, pp. 745-758.

[5] S. P. Han and H. J. Kang, "Real-Time Hand Gesture Recognition for Augmented Screen using Average Background and CAMshift," The 19th Korea -Japan Joint Workshop on Frontiers of Computer Vision, 2013, pp. 18-21.

[6] F. M. Fayed, E. A. Youness, M. Elmezain and F. S. Dewdar, "Hand Gesture Spotting and Recognition in Stereo Color Image Sequences Based on Generative Models," International Journal of Engineering Science and Innovative Technology (IJESIT), Vol, 3(1), 2014, pp. 78-88.

[7] M. Z. Hong and M. P. Chi, "Movement Tracking in Real-time Hand Gesture Recognition," 9th IEEE/ACIS International Conference on Computer and Information Science, 2010, pp. 240-245.

[8] R. Aditya, V. Namrata, C. Santanu and B. Subhashis, "Recognition of dynamic hand gestures," Pattern Recognition, Vol. 36, 2003, pp. 2069-2081.

[9] Y. L. Hsiang and J. L. Han, "Real-Time Dynamic Hand Gesture Recognition," International Symposium on Computer, Consumer and Control, 2014, pp. 658-661.

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