

# Hand Tracking and Hand Gesture Recognition for Human Computer Interaction

Bai yu<sup>†</sup>, Sang-Yun Park<sup>\*\*</sup>, Yun-Sik Kim<sup>\*\*\*</sup>, In-Gab Jeong<sup>\*\*\*\*</sup>,  
Soo-Yol Ok<sup>\*\*\*\*\*</sup>, Eung-Joo Lee<sup>\*\*\*\*\*</sup>

## ABSTRACT

The aim of this paper is to present the methodology for hand tracking and hand gesture recognition. The detected hand and gesture can be used to implement the non-contact mouse. We had developed a MP3 player using this technology controlling the computer instead of mouse. In this algorithm, we first do a pre-processing to every frame which including lighting compensation and background filtration to reducing the adverse impact on correctness of hand tracking and hand gesture recognition. Secondly, YCbCr skin-color likelihood algorithm is used to detecting the hand area. Then, we used Continuously Adaptive Mean Shift (CAMSHIFT) algorithm to tracking hand. As the formula-based region of interest is square, the hand is closer to rectangular. We have improved the formula of the search window to get a much suitable search window for hand. And then, Support Vector Machines (SVM) algorithm is used for hand gesture recognition. For training the system, we collected 1500 hand gesture pictures of 5 hand gestures. Finally we have performed extensive experiment on a Windows XP system to evaluate the efficiency of the proposed scheme. The hand tracking correct rate is 96% and the hand gestures average correct rate is 95%.

**Key words:** Hand Tracking, Hand Gesture Recognition, YCbCr, CAMSHIFT, SVM.

## 1. INTRODUCTION

Hand tracking—the continuous monitoring of hand position—as a perceptual form of multi-mod-

el input has many conceivable practical applications in Human-Computer Interface (HCI). Hand position information is presently being applied to navigation and control in immersive virtual reality system. Similar applications exist for other sorts of highly visual, reactive environments, such as computer games, simulations, and visualizations of large data spaces.

Hand gestures are a powerful, and probably the most natural and intuitive, human to human communication modality. So, in recent years, there has been a tremendous amount of research on hand gesture recognition in the area of HCI as an alternative to conventional interface devices [1]. And the application is so much such as interactive games, performance analysis, sign language translation, remote control of home and many others.

At the moment, the hand tracking and hand gesture cognition that has been developed is not sufficient for our needs. Tracking contours with snakes

※ Corresponding Author : Eung-Joo Lee, Address: Dept. of I.C. Eng., Tongmyong University, 535 YongDang-Dong, Nam-Gu, Busan 608-711, Korea, TEL : +82-51-629-3700, FAX : +82-51-629-3719, E-mail : ejlee@tu.ac.kr  
Receipt date : Oct. 6, 2010, Revision date : Dec. 6, 2010  
Approval date : Dec. 6, 2010

<sup>†</sup> Dept. of Information Communication Engineering, Tongmyong University(E-mail: baiyu163-com@163.com)

<sup>\*\*</sup> Dept. of Information Communication Engineering, Tongmyong University(E-mail: sypark@silla.ac.kr)

<sup>\*\*\*</sup> Dept. of Information Communication Engineering (E-mail: radio025@tu.ac.kr)

<sup>\*\*\*\*</sup> Dept. of Car & Fire Station, Kyungpook Provincial College(jeongig@hanmail.net)

<sup>\*\*\*\*\*</sup> Dept. of Game Engineering, Tongmyong University (E-mail: sooyol@tu.ac.kr)

<sup>\*\*\*\*\*</sup> Dept. of Information Communication Engineering, Tongmyong University

※ This research was supported by the SMBA (Research Program No.00043845-1).

[2-4], using Eigen space matching techniques[5], or convolving images with feature detectors are far too computationally expensive. And some systems use template to recognize the hand gesture. But this method is not exact enough. Some of the earlier gesture recognition systems attempted to identify gestures using glove-based devices that would measure the position and joint angles of the hand [6]. However, these devices are very cumbersome and usually have many cables connected to a computer. At the same time, while the development of the times, the people do not satisfy the hand tracking or hand gesture recognition under simple background with specified light condition. So that, we need much more better methods to tracking hand and hand gesture recognition. This paper is a system to develop hand tracking and hand gesture HCI which aimed at using hand instead of mouse to control computer under complex background with most light condition.

As the light condition and complex background have a greatly impact on the correctness of hand tracking and hand gesture recognition, in this paper we first do a pre-processing to every frame which including lighting compensation and background filtration. Then we use the YCbCr skin color likelihood algorithm which is based on the maximum likelihood principle to detect hand [7,8]. YCbCr color space is good for hand detection because the impact by the brightness changes is small. For using on dynamically tracking hand, we use Continuously Adaptive Mean Shift (CAMSHIFT) which is modified from the mean shift algorithm [9-11]. On the hand gesture recognition, we choose classification method. It is an important data mining task, learns a classifier from labeled training data to classify unlabeled testing data. As the support vector machine (SVM) is powerful tool with state-of-the-art performance on classification, which has both strong theoretical and excellent empirical success [12,13], we final use SVM algorithm to recognize hand gesture. Fig. 1 is the system block diagram.

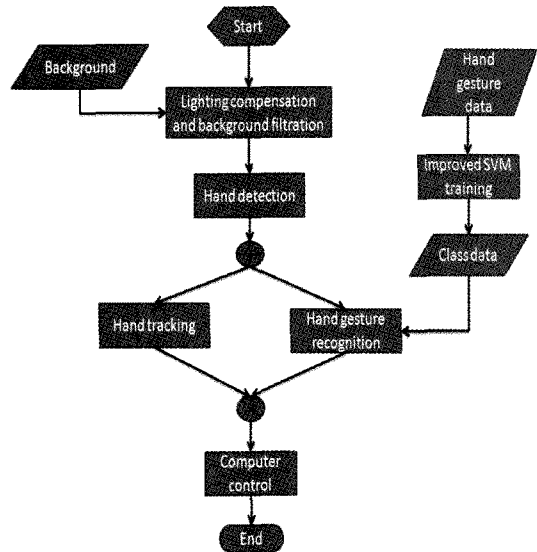


Fig. 1. The whole flow chart of proposed algorithm.

## 2. LIGHTING COMPENSATION AND BACKGROUND FILTRATION

### 2.1 Lighting compensation

At present, digital camera has been becoming more and more popular on computer vision. However, if the subjects were located between the camera and the light source, the subject will turn out to the dark or under-exposure, it is called black-light image [14]. For getting better image, we will do lighting compensation according the Just Noticeable Difference (JND). JND is that the minimal gray difference between two points in an image, which can just be resolved by the human vision; it is also called the minimal resolvable contrast (MRC).

The Human Vision Contrast Resolution (HVCR) enables the image, taken from the scotopic conditions and unresolved by human vision, to turn into a clearly visible image [15]. Let the HVCR threshold in the scotopic conditions is expressed as formula (1):

$$JND(x) = 22.98e^{-0.057x} \quad 0 \leq x \leq 47 \quad (1)$$

Where  $JND(x)$  represents the Just Noticeable

Difference of the human vision at the background gray  $x$ . It can be seen that the HVCR will be enhanced greatly if such a nonlinear limitation of HVCR can be compensated.

The formula (1) demonstrates that the HVCR threshold is very low in the scotopic conditions. Within 30 gray levels, the image can not be resolved by the human vision though there are the various gray in an image. That is to say, the portion of the image information can not be utilized. HVCR is compensated to the JND level if the original gray difference of one gray level between two adjacent points is expanded to JND. The cumulative gray distance is computed by following formulae (2):

$$T(O) = \begin{cases} O(x,y) & O(x,y) = 0 \\ k \sum_{i=0}^{O-1} JND(i) & O(x,y) > 0 \end{cases} \quad (2)$$

Where  $T(O)$  denotes the destination gray after compensation, corresponding to the original gray  $O$ .  $JND(i)$  represents the just noticeable difference at the gray  $i$ .  $O(x, y)$  denotes the original gray at the pixel point  $(x, y)$  in an image, ranged from 0 to 255.  $k$  is the compensation depth.

## 2.2 Background filtration

As the background may be have the similar chrominance with the skin color in CbCr; if we do not filtrate the background, it will give us trouble in terms of detecting hand. Filtering the background is essential [16].

We will use image subtraction to reducing the background's disturbance.

The image subtraction is the difference of two images. And the difference between two images  $f(x, y)$  and  $h(x, y)$ , expressed as follows:

$$g(x, y) = f(x, y) - h(x, y) \quad (3)$$

This is obtained by computing the difference between all pairs of corresponding pixels form  $f$

and  $h$ .

Note that most image are displayed using 8 bits (or 24-bit color images consists of three separate 8-bit channels), so we expect image values not be outside the range from 0 to 255. However, the values in a difference image can range from a minimum of -255 to a maximum of 255. We use a method that add 255 to the negative value and then divide by 2. This method is fast and simple implement.

For filtrate the background, firstly, we will adjust the camera to a suitable position, and save an image of background. Secondly, we get other image that includes hand. Thirdly, we calculate the difference of two images using image subtraction pixel by pixel. Finally, processing the negative values and saving the result.

## 3. HAND DETECTION

Skin color is important information of hand. It is not depending on the gesture of hand, and has more stability can be distinguish from most background color. Here, we use YCbCr color space for detecting skin color, because the merit of YCbCr color space is that the impact by the brightness changes is small [7]. And CbCr that independent two-dimensional distribution can limit skin color distribution range much better. Experimental proof: Although the skin color of different person may be very difference, the difference in chromaticity is much smaller than the difference in brightness. That is: different people have very similar chromaticity of skin color, just have large differences in the brightness. And the chromaticity CbCr of skin color has good clustering [17].

Skin color likelihood is based the nature of skin color clustering in the distribution of CbCr and the maximum likelihood principle [17]. The equation as follows:

$$p(CbCr) = \exp[-0.5(X-M)^T C^{-1}(X-M)] \quad (4)$$

$X(CbCr)^T$ ;  $C^{-1}$  is covariance inverse matrix

[0.0077, -0.0041, -0.0041, 0.0047];  $M$  is the average of skin color in CbCr [103.0056, 140.1309]. Firstly, we take the image from RGB to YCbCr, and save CbCr to the matrix  $X$ . Secondly, according to the algorithm calculate the probability of skin color of every pixel, and save the result to the matrix  $p$ . Thirdly, change the image to gray scale image by the matrix  $p$ . Fourthly, through median filtering to reduce noise.

#### 4. HAND TRACKING BASED ON CAMSHIFT ALGORITHM

Computer vision hand tracking is an active and developing field, yet the hand trackers that have been developed are not sufficient for our needs. We want a tracker that will track a given hand in the presence of noise. And it must run fast and efficiently so that objects may be tracked in real time (24 frames per second) while consuming as few system resources as possible for example that running on inexpensive consumer cameras [17].

We propose a method named mean shift algorithm, which is a simple iterative procedure that climbs the gradient of a probability distribution to find the nearest dominant mode. The mean shift algorithm operates on probability distributions. To track hand in video frame sequences, the image data has to be represented as a probability distribution. Distributions derived from video image sequences change over time, so the mean shift algorithm has to be modified to adapt dynamically to the probability distribution it is tracking. The new algorithm that meets all these requirements is called CAMSHIFT.

##### 4.1 CAMSHIFT algorithm

CAMSHIFT algorithm is a dynamic change in the distribution of the density function of the gradient estimate of non-parametric methods [10]. The course of algorithm is as follows:

1. Select a search window  $W$  size of  $s$  in skin

color probability distribution.

2. Calculate the zeroth moment.

$$Z_{00} = \sum_x \sum_y I(x, y) \quad (5)$$

Calculate the first moment of  $x$  and  $y$

$$Z_{10} = \sum_x \sum_y xI(x, y) \quad (6)$$

$$Z_{01} = \sum_x \sum_y yI(x, y) \quad (7)$$

Where  $I(x, y)$  is the pixel (probability) value at position  $(x, y)$  in the image, and  $x$  and  $y$  range over the search window.

3. Calculate the mean search window location (the centroid) is

$$x_c = Z_{10} / Z_{00} \quad y_c = Z_{01} / Z_{00} \quad (8)$$

4. Set the search window size equal to a function of the zeroth moment found in step 2.  
5. Repeat steps 2, 3 and 4 until convergence (mean location moves less than a preset threshold).

The zeroth moment reflect the area of object in the image, and the chart of skin color probability distribution is discrete gray scale image which have the max value is 255. So the relation between the size of search window  $s$  and  $Z_{00}$  as follows:

$$s = 2\sqrt{Z_{00} / 256} \quad (9)$$

Consider the symmetry;  $s$  get results close to the singular.

By calculating the second-order moment can be obtained the long axis, short axis and the direction angle of object. Second-order moment as follows:

$$Z_{20} = \sum_x \sum_y x^2 I(x, y) \quad (10)$$

$$Z_{02} = \sum_x \sum_y y^2 I(x, y) \quad (11)$$

$$Z_{11} = \sum_x \sum_y xy I(x, y) \quad (12)$$

The direction angle of object is

$$\theta = \frac{1}{2} \arctan \left[ \frac{2(Z_{11}/Z_{00} - x_c y_c)}{(Z_{20}/Z_{00} - x_c^2) - (Z_{02}/Z_{00} - y_c^2)} \right] \quad (13)$$

Suppose

$$\begin{aligned} a &= Z_{20}/Z_{00} - x_c^2 \\ b &= Z_{11}/Z_{00} - x_c y_c \\ c &= Z_{20}/Z_{00} - y_c^2 \end{aligned} \quad (14)$$

The long axis  $l$  and short axis  $w$  is

$$l = \sqrt{\frac{(a+c) + \sqrt{b^2 + (a-c)^2}}{2}} \quad (15)$$

$$w = \sqrt{\frac{(a+c) - \sqrt{b^2 + (a-c)^2}}{2}} \quad (16)$$

When CAMSHIFT algorithm track a specific color object, the images do not have to calculate each frame all the pixels of the color probability distribution, just calculate pixel color probability distribution in the area that larger than the current search window. This can save a very large computing [17].

#### 4.2 Modifications made

By the formula-based region of interest are square, but hand is closer to rectangular, when the hand rotation or hand towards the camera angle change, the aspect ration of regular of hand changes.

Assuming that there is no rotating hand, the region of interest with a width of  $b$ , height  $h$  the search window size of  $s$

$$b * h = s^2 \quad (17)$$

As the long axis  $l$  and short axis  $w$ , it has

$$b/h = w/l \quad (18)$$

So,

$$b = \sqrt{w/l} * s, \quad h = \sqrt{l/w} * s \quad (19)$$

When the target spins, we can get the rotation  $\theta$  from the formula. Taking into account the rotation of a rectangular rigid body, the region of inter-

est to determine the width and height:

$$\begin{aligned} b_n &= (b \cos \theta - h \sin \theta) \\ h_n &= (b \sin \theta - h \cos \theta) \end{aligned} \quad (20)$$

As the hand are flexible, and not a true rectangular. An amendment to the formula can be obtained following:

$$\begin{aligned} b_n &= (b \cos \theta - 0.5h \sin \theta) \\ h_n &= (0.5b \sin \theta - h \cos \theta) \end{aligned} \quad (21)$$

Experiments show that when  $\theta < 10$ , the calculation of  $b$  and  $h$  as a region of interest can be satisfied with the results; when  $\theta > 10$ , the calculation of  $b_n$  and  $h_n$  as a region of interest can get good results.

### 5. HAND GESTURE RECOGNITION BASED ON SUPPORT VECTOR MACHINE

On the hand gesture recognition, we choose classification method. It is an important data mining task, learns a classifier from labeled training data to classify unlabeled testing data. There are a numbers of standard classification techniques in literature, such as simple rule based on nearest neighbor classifiers, Bayesian classifiers, artificial neural networks, decision tree, support vector machine, ensemble methods, etc. Among these techniques, support vector machine (SVM) is one of the best-known techniques for its optimization solution [18]. So that we employ SVM algorithm to recognizing hand gesture.

#### 5.1 Support vector machine

Support vector machine (SVM) is a newly developed machine learning method for classification and function approximation.

In SVM classification, we are given training data  $\{x_1 \dots x_n\}$  that are vectors in some space  $x \in R^d$ . We are also given the labels  $\{y_1 \dots y_n\}$ , where  $y \in \{-1, +1\}$ . In the simplest form, SVM is hyperplane

that separate the training data by a maximal margin. All vectors lying on one side of the hyper-plane are labeled as -1, and all the other vectors lying on the other side are labeled as +1. The training instances that lie closest to the hyperplane are called support vectors [19]. More generally, SVM allow one to project the original training data in space  $X$  to a higher dimensional feature space  $F$  via a Mercer kernel operator  $K$  [20]. In other words, we consider another set of classifier of the form:

$$f(x) = \sum_{i=1}^n \alpha_i K(x_i, x) \quad (22)$$

When  $f(x) \geq 0$ , we classify  $x$  as +1, otherwise we classify  $x$  as -1.

When  $K$  satisfies Mercer's condition [15], we can write  $K(u, v) = \phi(u) \cdot \phi(v)$ , where  $\phi: X \rightarrow F$  and " $\cdot$ " denotes an inner product. We can write  $f$  as:

$$f(x) = w \cdot \phi(x), \text{ where } w = \sum_{i=1}^n \alpha_i \phi(x_i) \quad (23)$$

Thus, by using  $K$  we are implicitly projecting the training data into a different (often higher dimensional) feature space  $F$ . But there are many hyperplanes that might classify the data. One reasonable choice as the best hyperplane is the one that represents the largest separation, or margin, between the two classes. So we choose the hyper-

plane so that the distance from it to the nearest data point on each side is maximized. Such a hyperplane is called maximal margin hyperplane. Fig. 2 (a) shows that a training data, the black solid circles represent positive labels data, and the white solid circles represent negative labels data. And there is three separable hyperplanes  $H_1$ ,  $H_2$  and  $H_3$ . From Fig. 2 (a) we can see that:  $H_3$  doesn't separate the 2 classes;  $H_1$  does it with a small margin;  $H_2$  not only separate the 2 classes, but also have the maximum margin. From Fig. 2 (b), we can see that the maximum-margin hyperplane and two margins which are both important for SVM training. Note that the data on the margins are support vectors.

By choosing different kernel functions we can implicitly project the training data from  $X$  into spaces  $F$  for which hyperplanes in  $F$  corresponds to very complex decision boundaries in the original space  $X$ . Two commonly used kernels are the polynomial function kernel  $K(u, v) = (u \cdot v + 1)^p$  and radial basis function kernel  $K(u, v) = e^{-r(u-v) - (u-v)}$  [15]. The former function kernel introduces polynomial boundaries of degree  $p$  in the original space  $X$ . The second function kernel introduces boundaries by placing weighted Gaussians upon key training instances. In our SVM

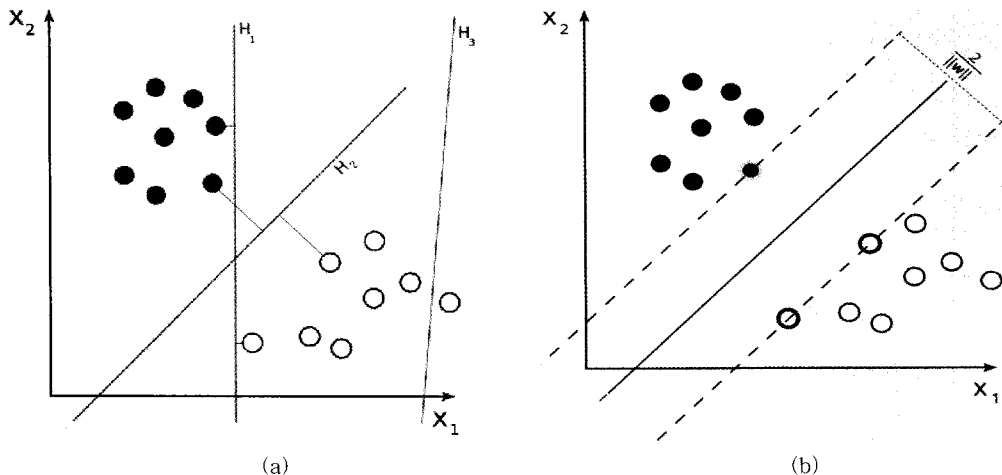


Fig. 2. (a) A training data and its three separable hyperplanes, (b) Maximal margin hyperplane and margins.

classifier, we choose the radial basis kernel function for training and classification.

## 5.2 Training and recognition

Support Vector Machine (SVM) becomes a popular classification method recently due to its ability to learn with small size of samples and to classify high-dimensional data. However, SVM is originally designed for binary classification [11]. To deal with multi-class classification, several schemes have been proposed: (1) one-against-rest, (2) one-against-one and (3) decision directed acyclic graph SVM (DAGSVM) [19].

- One-Against-Rest: In One-Against-Rest scheme,  $n$  SVM classifiers,  $SVM_i$  ( $i=\{1..n\}$ ), was trained for a  $n$ -class classification problem. During training, each SVM use the samples of its corresponding class as the positive examples and the samples from all other classes as the negative samples. During testing, the test samples are evaluated against all  $n$   $SVM_i$  ( $i=\{1..n\}$ ) and  $SVM_k$  which gives highest decision value as the predicted class is chosen.
- One-Against-One: In this scheme, one SVM classifier,  $SVM_{k,j}$ , is constructed for every pair of classes ( $i, j$ ). There are  $N(N+1)/2$  SVM classifiers in total. During testing, the samples are evaluated against all the pair wise classifiers  $SVM_{i,j}$ . The final decision of the class for the test samples are determined by a voting scheme. Every SVM classifier gives one vote for its predicted class. The class that receives most of the votes is chosen as the final class for the test sample.
- DAGSVM: In this scheme, same as the one-against-one,  $N(N+1)/2$  pair wise classifiers are constructed. During test, a list with all class candidates is created. For each test with the SVM, the class candidate which is given negative label by the SVM is removed from the list. The last class that is left on the list is chosen

to be the class for the test sample.

By experiment, the One-Against-Rest has the least run time and gets good recognition results (specific information can be seen on Part 6: experimental results). Therefore, we collect 1500 pictures of five kinds of hand gestures to training the system. Specially, the hands pictures must be only have hand gesture without much background. Firstly, we transform the pictures to gray-level and resize the picture to  $20*30$ . Secondly, we input the pictures to training the system by SVM One-Against-Rest scheme to get the characteristic vectors of each kind of hand gesture and save the characteristic vectors to hand-gesture.xml file. Thirdly, we resize the hand areas which are hoped to recognition to  $20*30$ . And compare the hand areas with the characteristic vectors of hand gesture (reading hand-gesture.xml file) by SVM.

Fig. 3 (a) shows the training data of five hand gestures respectively in five colors (red, green, blue, pine green and black red) which had mapped to the 2-dimensional space. Fig. 3 (b) shows the support vectors of five hand gestures in white circle, and the corresponding areas are assigned into the corresponding color (red, green, blue, pine green and black red).

## 6. EXPERIMENTAL RESULTS

The hand tracking and hand gesture recognition approach is implemented in Visual C++6.0 on 2.33 GHZ Core 2 Quad running Windows XP with a  $320*240$  USB Logitech Quickcam webcam.

Fig. 4 (a) illustrates the initial status of hand tracking including the hand position and the playing music position. Fig. 4 (b) shows that the final status of hand tracking. Note that, the hand position had moved to left compare with Fig. 4 (a)'s and the playing music also move to 4 from 0. Through experiment, we also can see that the improved CAMSHIFT has a suitable searching win-

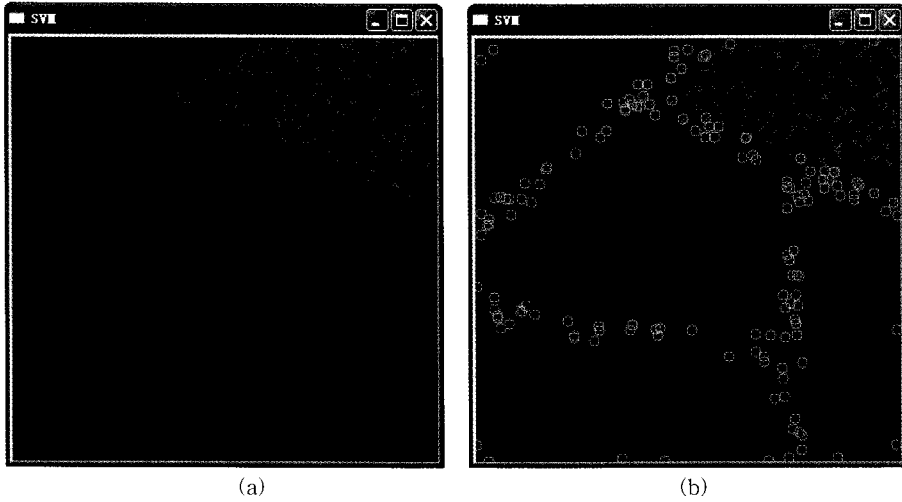


Fig. 3. (a) Training data of five hand gestures (b) Characteristic vectors of five hand gestures.

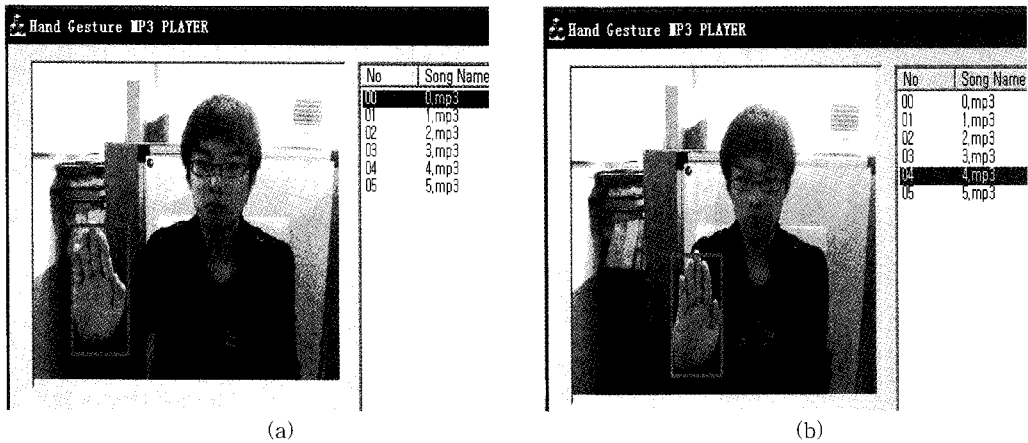


Fig. 4. (a) Hand tracking initial status (b) Hand tracking final status.

dow with hand tracking.

To evaluate the performance of one-against-rest schemes for multi-class classification, we compare the average classification rate over the 10 subjects with the other 2 SVM classifiers (one-against-one and DAGSVM). Fig. 5 shows the classification results. The x-axis in the figure shows the percentage of training data. The y-axis shows the classification rate. One-against-one SVM classifier achieves the best classification rate among all the classifiers at the cost of significantly high computation. One-against-rest and DAGSVM performs classifier show similar classification rate

with one-against-rest performs slightly better than the DAGSVM classifier. Fig. 6 shows the training time for 3 different classifiers. The one-against-one classifier need the longest training time. The training time of DAGSVM is less than the one-against-one's. One-against-rest need the least training time.

The five hand gestures recognition is respectively showed on Fig. 7 (a), (b), (c), (d) and (e). While the program recognizing the hand gesture, the program playing the corresponding music.

We have experiment the hand tracking and each of hand gesture recognition respectively 200 times



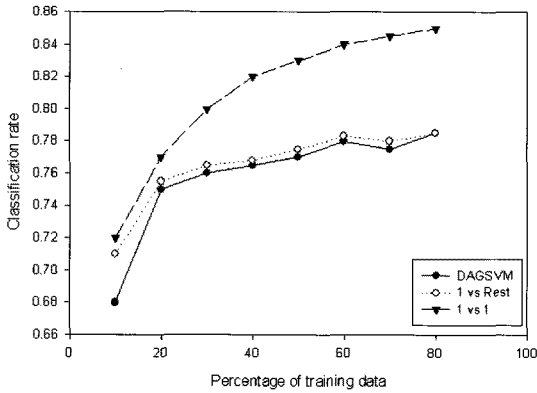


Fig. 5. Comparison of classification rate for 3 different SVM multiclass classifier: 1 vs. 1, 1 vs. Rest and DAGSVM.

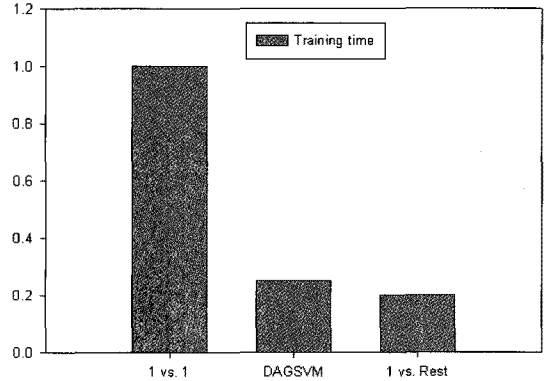


Fig. 6. Comparison of training times for 3 different SVM multiclass classifier: 1 vs. 1, 1 vs. Rest and DAGSVM.

of 10 different persons. The results are satisfactory as shown in Table 1. The hand tracking correct rate is 96% and the hand gesture average correct rate is 95%. In Table 1, COR means correct rate. ERR is that error rate. FAR is false acceptance rate and FRR is false rejection rate.

### 7. CONCLUSIONS

In this paper, a methodology of hand tracking and hand gesture recognition was proposed, it can have application in scotopic conditions and complex background, in which lighting compensation

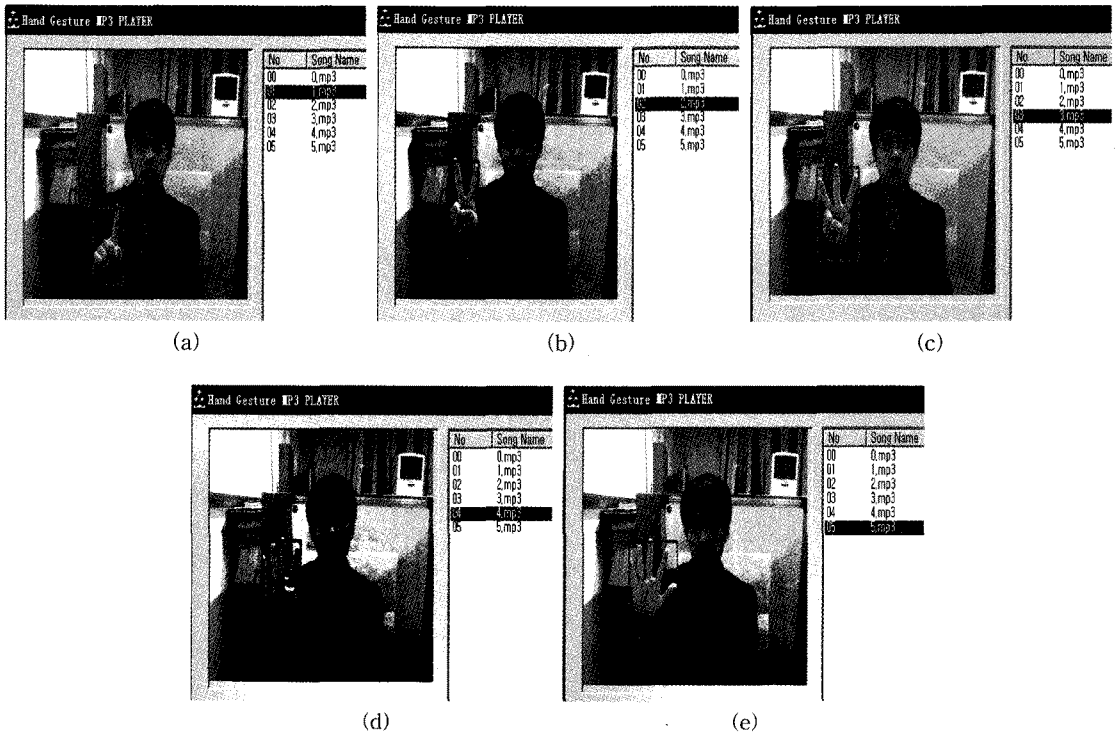


Fig. 7. (a) Hand gesture 1 (b) Hand gesture 2 (c) Hand gesture 3 (d) Hand gesture 4 and (e) Hand gesture 5.

Table 1. Hand tracking and hand gesture recognition results

|               | Numbers | Percent |
|---------------|---------|---------|
| Total test    | 200     | 100%    |
| Tracking COR  | 192     | 96%     |
| Tracking ERR  | 8       | 4%      |
| Gesture 1 COR | 188     | 94%     |
| Gesture 1 FAR | 2       | 1%      |
| Gesture 1 FRR | 10      | 5%      |
| Gesture 2 COR | 194     | 97%     |
| Gesture 2 FAR | 1       | 0.5%    |
| Gesture 2 FRR | 5       | 2.5%    |
| Gesture 3 COR | 190     | 95%     |
| Gesture 3 FAR | 4       | 2%      |
| Gesture 3 FRR | 6       | 3%      |
| Gesture 4 COR | 195     | 97.5%   |
| Gesture 4 FAR | 2       | 1%      |
| Gesture 4 FRR | 3       | 1.5%    |
| Gesture 5 COR | 193     | 96.5%   |
| Gesture 5 FAR | 3       | 1.5%    |
| Gesture 5 FRR | 4       | 2%      |

and background filtration is first used to every frame to reducing the adverse impact of light condition and complex background. And then YCbCr skin-color likelihood is used to detect the areas of hand. Then we track the hand movement by CAMSHIFT algorithm which had been improved searching window to suit hand shape. After that we abstract the characteristic vectors of five hand gestures which be saved on hand-gesture.xml file by using SVM algorithm. The detected hand is applied to SVM algorithm through reading characteristic vectors (hand-gesture.xml file) for hand gesture recognition. Finally, we use the tracking result and gesture recognition result to control a MP3 player. The experimental results had proved that our method have good tracking and gesture recognition performance while have the least training time and finer classification rate.

The further work is focus on improving the propose algorithm for the purpose of dealing with hand

images under multi-illumination conditions while have higher accuracy rate.

## REFERENCES

- [1] Li wei and Eung-Joo Lee, "An IP-TV Remote Controller Based on Hand Posture Recognition Using Fourier Descriptor," Conference on Korea Multimedia Society 2009, Vol.12, No.2, pp.510-511, 2009.
- [2] K. Sobottka and I. Pitas, "Segmentation and tracking of faces in color images," IEEE Conference on *AFGG 1996*, pp.236-241, 14-16, 1996.
- [3] K. Takaya, "Tracking a video object with the active contour (snake) predicted by the optical flow," IEEE Conference on *CCECE 2008*, pp. 369-372, 4-7, 2008.
- [4] A. Aksel and S.T. Acton, "Target tracking using the snake particle filter," IEEE Conference on *SSIAI 2010*, pp.33-36, 23-25, 2010.
- [5] M.K. Agarwal, "Eigen space based method for detecting faulty nodes in large scale enterprise systems," IEEE Conference on *NOMS 2008*, pp. 224-231, 7-11, 2008.
- [6] V.I. Pavlovic, R. Sharma and T.S. Huang, "Visual interpretation of hand gestures for human-computer interaction: a review," IEEE *Transactions on Pattern Analysis and Machine Intelligence*, Vol.19, No.7, pp.677-695, 1997.
- [7] D. Chai and A. Bouzerdoum, "A Bayesian approach to skin color classification in YCbCr color space," IEEE Conference on *TENCON 2000*, pp. 421-424, 24-27, 2000.
- [8] N. Habili, Cheng-Chew Lim and A. Moini, "Automatic human skin segmentation based on color information in the YCbCr color space," IEEE Conference on *Information, Decision and Control 2002*, pp.377-382, 11-12, 2002.
- [9] Yizong Chen, "Mean shift, mode seeking, and

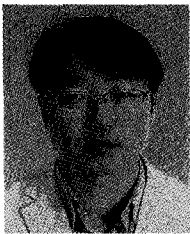
- clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.17, No.8, pp. 790-799, 1995.
- [10] Xia Liu, Hongxia Chu and Pingjun Li, "Research of the Improved Camshift Tracking Algorithm," *IEEE Conference on ICMA 2007*, pp. 968-972, 5-8 Aug. 2007.
- [11] Qiang Liu, Canhui Cai, K.N. Ngan and Hongliang Li, "Camshift based Real-Time Multiple Faces Match Tracking," *IEEE Conference on ISPACS 2007*, pp. 726-729, 2007.
- [12] Cervantes. J, Xiaoou Li and Wen Yu, "SVM Classification for Large Data Sets by Considering Models of Classes Distribution," *IEEE Conference on MICAI 2007*, pp. 51-60, 4-10, 2007.
- [13] Dong Seong Kim, Ha-Nam Nguyen and Jong Sou Park, "Genetic Algorithm to Improve SVM Based Network Intrusion Detection System," *IEEE Conference on AINA 2005*, pp. 155-158, 28-30, 2005.
- [14] Daw-Tung Lin and Chia-Ching Huang, "Image Back-Light Compensation with Fuzzy C-Means Learning Algorithm and Fuzzy Inferencing," *IEEE Conference on ISSPA 2003*, pp. 433-436, 1-4, 2003.
- [15] Shen Yuhao, Meng Chen and Fu Zhenhua, "A Synthesized SVM and Its Application in Fault Diagnosis for Circuit Board Based on Virtual Instrument," *IEEE Conference on ICEMI 2009*, pp. 907-911, 16-19, 2009.
- [16] Sang-Yun Park and Eung-Joo Lee, "Hand Gesture Recognition Algorithm Robust to Complex Image," *Journal of Korea Multimedia Society 2010*, Vol.13, No.2, pp. 1000-1015, 2010.
- [17] Bai yu and Eung-Joo Lee, "The hand Mouse: Hand Detection and Hand Tracking," *International Conference on Multimedia, Information Technology and its Application 2009 (MITA 2009)*, pp. 244-245, 19-21, 2009.
- [18] M. Niranjan, "Support Vector Machine: a Tutorial Overview and Critical Appraisal," *IEEE Colloquium on ASPR 1999*, pp. 2/1, 1999.
- [19] Chh-Wei Hsu and Chih-Jen Lin, "A Comparison of Methods for Multiclass Support Vector Machines," *IEEE Transactions on Neural Networks 2002*, Vol.13, No.2, pp. 415-425, 2002.
- [20] Bai yu and Eung-Joo Lee, "Hand Posture Recognition Based on SVM Using on Mobile Phone," *Conference on Korea Multimedia Society 2009*, Vol.12, No.2, pp. 188-189, 2009.



**Bai Yu**

received his B. S. at Dalian Polytechnic University in China (2004-2008). From 2008 to now he is a M. S student of Tongmyong University, Korea. His main research tops are image processing, computer vision, biometrics and hand

recognition.



**Sang-Yun Park**

received his B. S. in Physics, M. S. in Multimedia Application from KyungSung University, Korea, in 1992, 1998, respectively. And now(2006-2011) he is a Ph.D student of Tong-Myoung University in Korea.

His main research interests include in computer vision and biometrics.



**Yun-Sik Kim**

received his B. S. in Information Communication Engineering from Tongmyong University, Korea, in 2009. From 2009 to now he is a M. S student of Tongmyong University, Korea.

His main research tops are image processing, computer vision, biometrics and hand recognition.



**Soo-Yol Ok**

received his M. S. and Ph. D. in Master's Program in sciences and Engineering, Doctoral Program in Engineering from Tsukuba University, Japan, in 1998, and July. 2001, respectively. In March 2004, he joined

the Department of Game Engineering of Tongmyong University, Busan, Korea, as a professor. His main research interests are in Human Interface Control, Game AI, and Computer Graphics.



**In-Gab Jung**

received his B. S. , M. S. and Ph. D. in Electronic Engineering from Kyungpook National University, Korea, in 1990, 1992, and Feb. 1997, respectively. Since 1997, he has worked in Gyeongbuk Provincial College,

Korea. His main research are image processing and computer vision.



**Eung-Joo Lee**

received his B. S. , M. S. and Ph. D. in Electronic Engineering from Kyungpook National University, Korea, in 1990, 1992, and Aug. 1996, respectively. Since 1997 he has been with the Department of Information &

Communications Engineering, Tongmyong University, Korea, where he is currently a professor. From 2000 to July 2002, he was a president of DigitalNetBank Inc.. From 2005 to July 2006, he was a visiting professor in the Department of Computer and Information Engineering, Dalian Polytechnic University, China. His main research interests includes biometrics, image processing, and computer vision.