

A Real Time Hand Tracking System for Interactive Applications

Siddharth Swarup Rautaray
Indian Institute of Information Technology
Allahabad

Anupam Agrawal
Indian Institute of Information Technology
Allahabad

ABSTRACT

In vision based hand tracking systems color plays an important role for detection. Skin color detection is widely used in different interactive applications, e.g. face and hand tracking, detecting people in video databases. This paper implements an effective hand tracking technique which is based on color detection. In this techniques based on the color distribution the segmentation of hand from background will take place in a real time. This technique provides to main benefits: The process of tracking is fast as the segmentation process is performed simultaneously in a specified area surrounding the hand. This technique is highly robust under different lightning conditions. To check the performance of the implemented technique a number of experiments have been performed. The implemented technique will be useful in various real time interactive applications, such as gesture recognition, augmented reality, virtual reality etc.

Keywords: Skin color, hand segmentation, human computer interaction, real-time tracking.

1. INTRODUCTION

In human computer interaction, keyboards, mice, joysticks etc are most commonly used devices for designing interactive applications. The disadvantage of the above devices is the absence of flexibility due to different spatial limitations. The interactive applications deals with interaction with different devices in a dynamic environment, therefore it becomes difficult for user to use mouse and keyboards which generated different limitations for interaction. Vision based techniques provides a solution for interaction in dynamic environment which uses one or more webcams and cameras for capturing users hand and body gestures and further processes and analyzes to generate a meaningful command or vocabulary. This vision based techniques helps in acquiring the user gestures in an easier manner which will play as an input for different interactive applications.

One of the key stages involved in these applications is to separate and track the area of interest (i.e. hand or body) from the complex background. For this purpose color information plays an important role for hand tracking. Zhang et al. [15] develops a gesture interface system, based on edge based method which tracks the tip pointer in a arbitrary quadrangle shaped panel. The tip pointer used is an intuitive input device.

The background needs to be simple for tracking the tip point (fingertip) in the quadrangle shaped panel. As the finger move fast the tracking lost the tip pointer and further re-initialization is done. The edge based tracking techniques are less robust in complex backgrounds and are computationally expensive. The vision based hand tracking technique which uses skin color [17] information is more robust and effective in complex environment. In vision based tracking color generates strong cue as it distributes in a small area of color space and more deviation in intensity than in color. The hand tracking techniques based on color can be categorized into two groups: parametric techniques and non parametric techniques.

A parametric technique shows the color concentration in parametric forms, e.g. Mixture of Gaussian [12, 10, 16]. Expectation-Maximization (EM) is a typical method used for probabilistic models. However, the hand color distribution parameters can deviate significantly under different environmental conditions like change in light intensity with different people. Therefore the major challenge is not having enough anterior knowledge for determining the order of Mixture of Gaussian.

A non parametric technique calculates the color dispersion from the training datasets using statistical probability models. Color histogram technique comes under non parametric approach. Kjeldsen and Kender [6] designs a structure based on histogram to segment the hand for designing a gesture based interface. The designed system works in a specific lighting condition of an office environment and assumes that the hand color was comparatively unique in the specific environment. Sigal et al. [11] shows a technique based on histogram which uses a second order Markov model to anticipate the development of the skin color histogram over time. Based on the segmentation and anticipations of the Markov model histograms get dynamically updated. The drawback of this method is the scene changes are not properly modeled and is only effectual in slow changing dynamic scenes. Therefore the histogram based tracking techniques perform well in conditions when histogram is quantified and adequate training data is available. Under non parametric technique Mean Shift is skin color segmentation method which is based on region matching [4]. An initial guess for a location is converged and scaled to the best match of color histogram similarity. Camshift is an advanced technique based on mean shift termed as continuously adaptive Mean Shift, uses the probability distribution for detecting the mode in an image by employing mean shift and the parameters of the target distribution is dynamically adjusted [1]. This camshift technique is fast and fairly accurate which can further improved by applying different color representation. Some popular hand

tracking techniques based on color classification are available [13, 14]. Wu et al. proposed color segmentation technique which uses a color distribution in the HSI color space for an image using self-organizing map (SOM) and each output neuron checks to a color cluster [13]. The disadvantage of this technique is color segmentation technique may fail as the lighting conditions changes dramatically. Ong and Bowden [9] propose a clustering technique which clusters the training data into similar shapes and builds a classifier tree for detection based on skin color distribution. The k-means clustering algorithm is used for clustering images in database by which a tree structure of boosted cascades is build. This classifier tree is further used to recognize particular gestures. This technique is useful is conditions where both training and testing databases have similar backgrounds. Some researchers used face detection techniques for hand tracking, as face and skin color have same skin color and the face can be detected automatically using different computer vision techniques [2, 7, 8]. In [2] Nickel and Stiefelhagen designs a gesture interface system for interaction which uses both face and hands. Manders et al. proposed a hand tracking technique based on camshift with joint probability function [7]. In these methods tracking of hand takes place once the face is detected which is not practically possible in all real time interactive environment. This paper implements an efficient hand color classification technique using randomized lists. This technique consists of two stages: training and tracking. In the first stage a region of interest is specified on hand for obtaining the trained data. The trained data will be categorized into number of color clusters using randomized list based on hand color. In the second phase the real time hand segmentation is done from the background using the trained randomized list. The implemented technique is fast and efficient under different illuminating conditions.

The paper is further organized under different sections: Section 2 describes the mechanism of the color categorization. Section 3 shows the steps for real-time hand segmentation technique. Section 4 shows the experimental results to check the validation of implemented technique. Section 5 shows the conclusion and future work with section6 covering the references.

2. COLOR CATERGORIZATION

In $L^*a^*b^*$ color model the segmentation of skin color not only depends on segmentation techniques but also relies on the distribution of different color model. Generally in non-parametric skin-color tracking techniques, the HSI color model is implemented as the color distribution which is more concentric that in the RGB color model. The $L^*a^*b^*$ color model is used in the present implemented technique which is different from RGB and HIS color model. In $L^*a^*b^*$, the L^* is denotes to lightness and a^* , b^* denotes to coordinates of chromaticity. It is proportional nearly with visual perception, which intends to equal distances in the color model corresponding to equal perceived color deviation. Values of the L^* , a^* and b^* are computed by the following equations [3]:

$$L^* = \begin{cases} 116 \left(\frac{Y}{Y_n}\right)^{1/3} - 16 & \text{if } \frac{Y}{Y_n} > 0.008856 \\ 903.3 \left(\frac{Y}{Y_n}\right)^{1/3} - 16 & \text{if } \frac{Y}{Y_n} > 0.008856 \end{cases} \quad (1)$$

$$a^* = 500 \left[f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right) \right] \quad (2)$$

$$b^* = 500 \left[f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right) \right] \quad (3)$$

where

$$f(t) = \begin{cases} t^{1/3} & t > 0.008856 \\ 7.787 * t + \frac{16}{116} & \text{otherwise} \end{cases} \quad (4)$$

in which X_n , Y_n and Z_n denotes the CIE XYZ tristimulus values of a perfect reflecting diffuser. In the present implemented technique, they are set as 250.155, 255.000, and 301.410, respectively. X, Y, and Z are the tristimulus values which are computed from R, G, B data of each pixel based on the following equations:

$$X = 0.607 * R + 0.174 * G + 0.2 * B \quad (5)$$

$$Y = 0.299 * R + 0.587 * G + 0.114 * B \quad (6)$$

$$Z = 0.066 * G + 1.116 * B \quad (7)$$

2.1 Skin-Color Distribution

The intension of skin-color distribution based on hand segmentation technique is to separate the hand color distribution into different color bunches using randomized list data structure. Generally in most color distribution based tracking a technique, the number of bunches should be specified in advance. Therefore the success of the distribution technique depends on the defined number of bunches. In the present work, the classifier is defined by randomized list data structure. Through implementing a randomized list data structure, the number of bunches will be automatically obtained, thus making the technique more efficient.

In the present implemented technique, each color bunch comprises of the pixels which consist of same features defined by four parameters: class C , weight vector ω , threshold λ and pattern count t . ω defines the set of weighted connections between the bunches and each of the input signals. λ describes a hyper-spherical region of influence around the bunches in the $L^*a^*b^*$ color model. t shows the number of times that a color bunch has reacted to the input color signals received through the training data. As adverted in previous section 2.1, the skin-color distribution is corresponded in the $L^*a^*b^*$ color model. The input vector, which is normally delimited by a feature vector for color distribution, corresponds a feature of the incoming pattern and is of importance to the classification problem. In the present implemented technique, it will significantly bear on the classification performance, such as robustness to the complex environment conditions. The input vector in the present implemented technique is delimited by a^* and b^* elements of

each individual pixel, which is corresponded in the L*a*b* color model. From the experiments that we have carried on, a* and b* elements of the training data is sufficient and effective for the color distribution problem. Therefore, the L* component, that constitutes the lightness value in the L*a*b* color model, is not used for tracking techniques. This builds the hand tracking technique robust under different complex environment conditions. The response for an input signal, i.e. an input vector of each individual pixel, a distance between the input vector and its comparable weight in each color bunch will be calculated based on the Euclidean distance function. In the first input signal demoed to the training procedure, a new color bunch will be generated first and this input signal will be laden by weight vector of this new color bunch, and the pattern count in this color bunch is set to 1.

Throughout training, an input signal will be incorporated in a color bunch if it falls into this bunch, i.e., its distance in this bunch is less than a pre-defined threshold λ . Afterwards, the pattern counter is incremented by 1. Otherwise, a new color bunch will be generated and the current input vector is placed as a weight vector of this bunch. The color distribution technique [17] covers of the following steps, as shown in Figure. 1.

1. Specify the specific region(s) in the hand to be trained.
2. Use the target region to find the training data set in the L*a*b* color model.
3. The input color signal X , compute the distance $d_i = \sqrt{\sum_{j=1}^2 (\omega_{ij} - x_j)^2}$. If it does not comes into any existing bunch, a new bunch i will be made and X is set as the weight vector the bunch i . This new bunch is then salted away at the terminal of a randomized list data structure.
4. The input color signal X , if it comes into a bunch i , the pattern counter of this available bunch is incremented by 1. The information of this bunch is farther informed in the list structure.

In a response for an input color signal X , colligated probability response model are usually used, in which each bunch will generate a probability value to find which bunch will be actuated for the incoming input signal. In the present implemented technique, a fast response mode is applied, through which the classifier calculates the distance established by a radial basis

function i.e. $d_i = \sqrt{\sum_{j=1}^2 (\omega_{ij} - x_j)^2}$ among an input signal and the weights in the color bunch, and then directly equates as this distance is to a lesser extent than the predefined threshold, that is:

$$p_i = \begin{cases} 1 & \text{if } d_i < \lambda_i \\ 0 & \text{if } d_i \geq \lambda_i \end{cases} \quad (8)$$

If d_i is less than or equal to a specific pre-defined threshold of a color bunch, the bunch will become active to trigger its colligated color class C . Otherwise, the bunch will not respond this input signal.

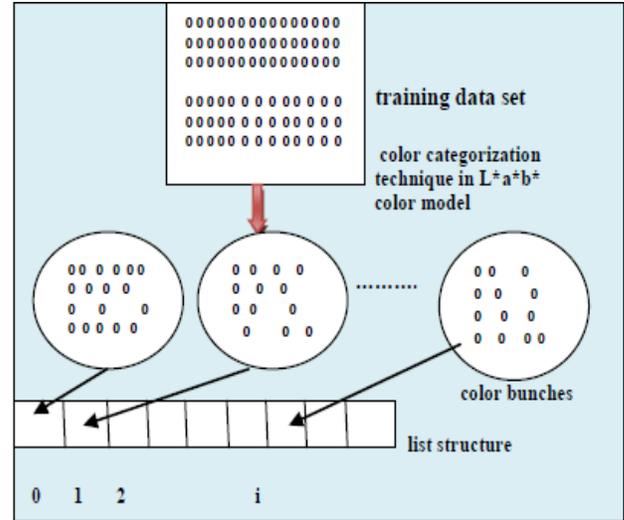


Figure 1. Color Categorization

3. REAL-TIME HAND SEGMENTATION ALGORITHM

3.1 Flowchart

In a live video for each incoming frame, the system will directly place the pixels which may refer to the hand utilizing the randomized lists-based classifier that is generated during the training procedure. In the next stage some of the post-processing processes, such as pixel labeling, group connectivity, are needed to accurately segment the hand in a frame from the background. The figure 2 shows the process for hand segment.

3.2 Initialization

The user has to keep the hand in front of the camera to specify one or more regions on the hand for obtaining the training data to generate the color distribution at the initial stage. The values are stored into a randomized data structure. This process has to be complied by the user once at the start of the application.

3.3 Hand Localization

Through an incoming frame from a real time video, the main issue is localizing the hand region accurately. In the first frame, the system obtains the complete color images that needs to be segmented and converted into L*a*b* color model. The randomized lists-based classifier takes the a* and b* elements of each pixel. Randomized list directly label the pixels belonging to the hand. By equation (8), pixels with value 1 as output represents the hand segmentation results. Once the pixels are labeled different techniques [13] are applied to group the labeled pixels into several connected regions. During this stage, different regions need to be segmented which consist of a regions from the background similar to hand color distribution. The hand region retains the labeled pixel with largest number in the region. Then a feature point defined as interaction point will be extracted which will be used as a input device. In the present implemented technique, apply the center of the tacked hand region. This interaction can be used as an input for interactive application. By the next incoming frame, the system obtains a

bounded region circumferent the tracked hand in previous frame, and further repeats the same procedure to segment the hand, thus making the segmentation faster.

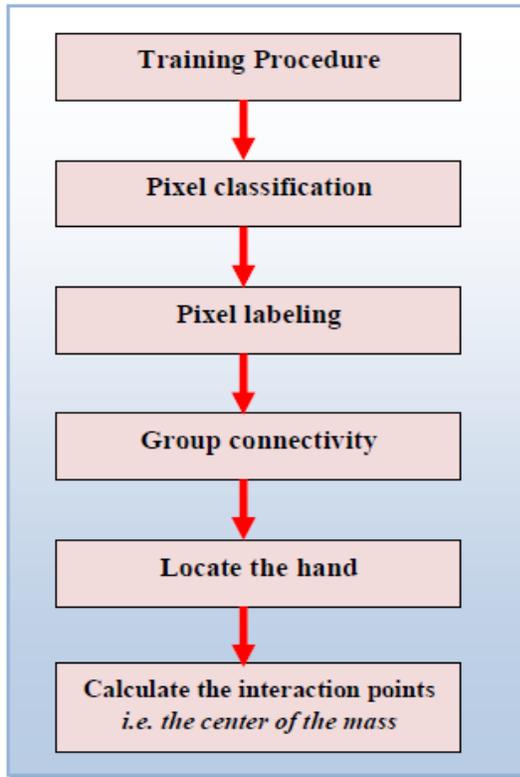


Figure 2. Flowchart of the hand segmentation

3.4 Refining the training data

As the user defines a region(s) on the hand for tracking, the selected region(s) may consist of some pixels which do not refer to the hand pixel as shown in figure 3 (a). In such conditions, some of the generated color bunch may match to those pixels which do not refer to the hand region. However, based on our observation in the experiments that we have done, the pattern numbers of these color bunch are frequently small. Therefore in the present implemented technique, these color bunches will be clipped according to a given threshold and the trained randomized lists will be re-arranged. Although the selected regions consist of some small regions of the background, the technique can still segment the hand from the background in much accurate manner.

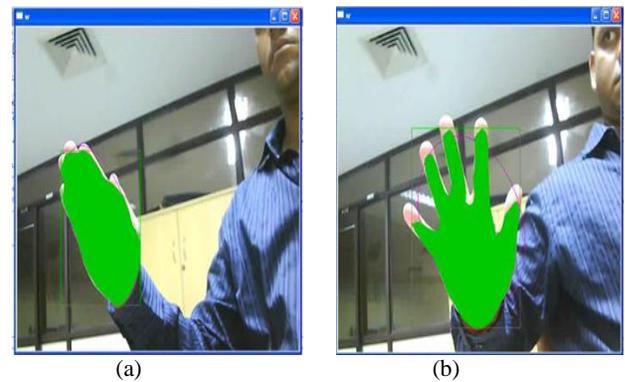


Figure 3. Example of refining the trained randomized lists

For example, in Figure. 3(a), the specified area consists of some parts of the background wall and furniture, as marked by the two blue circles. The hand can still be evenly tracked using the processed training results, as displayed in Figure. 3(b).

4. EXPERIMENTS

The implemented hand tracking technique has been implemented using dev C++, opencv libraries and number of experiments has been performed to test the robustness of the implemented technique. A normal web camera captures the image sequence from a real time environment. The user first concentrates a region on the hand by bring the hand in front of camera to obtain the trained data in the $L^*a^*b^*$ color model. The system will carry out the training procedure to obtain a color distribution using randomized lists with is further used for hand segmentation. This segmentation technique is used to segment the hand from the background in real-time, as the randomized lists-based classifier by automatically performing on a small region circumferent the hand. Figure. 4 show few hand tracking outputs in a real time environment. In the following experiments performed, the hand moved indiscriminately, by moving it very fast, or hand is moved in different directions and angles to check its robustness. Moreover, the variations in illumination conditions are also done during the experiment check. These outputs show that the hand can be robustly segmented and tracked in real-time under different environmental and illumination conditions.



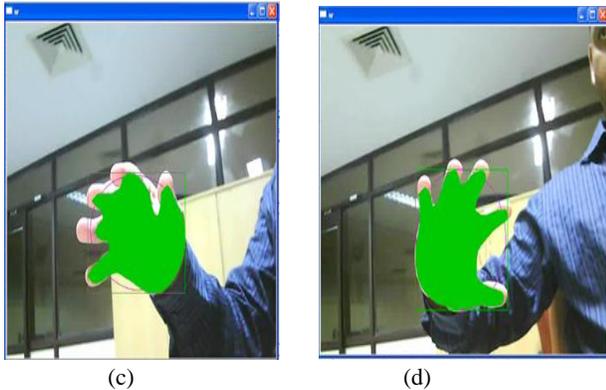


Figure 4. Examples of the real time tracking method

Beside the effectualness of the implemented technique, two major issues covered below to show the accurateness of the implemented technique:

(1) Rapid movement

Generally, the user gestures with his hand at a normal speed. But sometimes as when the user gestures by moving his hands at high speed then the tracking points may be disturbed. Thus, the former interaction point is stored and the user can move his hand close to the position of the former interaction point to simply re-track the hand without any manual restart.

(2) Lighting condition

In the present implemented technique, only the a^* and b^* elements in the $L^*a^*b^*$ color model are applied for training and tracking. It makes the tracking robust and efficient under various illumination conditions. Figure 5 generates some tracking outputs under different illumination conditions from a live video sequence. Generally in many real time applications, only partial hand is needed to be tracked to draw out the interaction point. Some examples like a virtual keyboard system, which need only a part of the hand is tracked for interaction application. It will be useful for at least one point of hand is visible for detecting and tracking as considering a worst case such as sudden illumination change which can be used for interaction purpose.

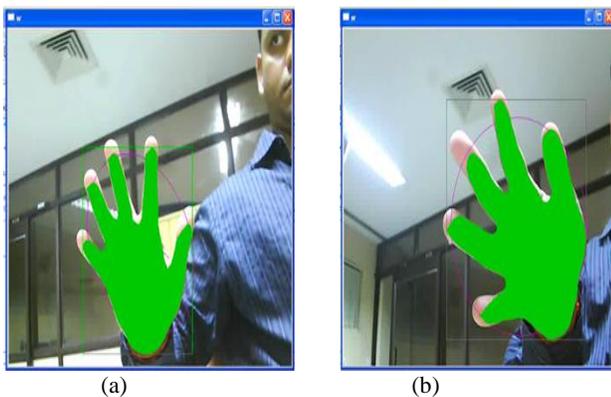


Figure 5. Tracking results under different lighting conditions

5. CONCLUSIONS

The main aim of the paper to implement and test a fast and effective real time hand tracking technique based on hand color segmentation and distribution using randomized lists. The technique is simple to run by moving the user's hand freely for interactive applications. Another advantage is that the implemented technique is its robustness under different illumination conditions. Few experiments have been performed to validate that the implemented technique is robust and stable. Moreover, the implemented hand tracking technique can be used further for designing interactive application like wireless mouse etc. Few examples where it can use are controlling video games in virtual reality or augmented reality applications. In the future work, more research and implementation will be performed to apply this technique for virtual reality and augmented reality game based applications.

6. REFERENCES

- [1] Bradski, G. R. 1998. Computer video face tracking for use in a perceptual user interface, *Intel Technology Journal*, Q2, (1998).
- [2] Cabral, N. C. Carlos, H., Morimoto, H. and Marcelo, K. Z. 2005. On the usability of gesture interfaces in virtual reality environments, In *Proceedings of the Latin American conference on Human-computer interaction*, (2005), 100-108.
- [3] CIE. 1986. *Colorimetry*, second edition. Vienna, Austria. Publication CIE No. 15.2.
- [4] Comaniciu, D., Ramesh, V. and Meer, P. 2000. Real-time tracking of Non-rigid objects using Mean Shift, In *Proceedings of IEEE conference on Computer Vision and Pattern Recognition*, (2000), 142-149.
- [5] Jones M. and Rehg, J. 1998. Statistical color models with applications to skin detection, *Compaq Cambridge Research lab*, TP CRL 98/11, (1998).
- [6] Kjeldsen, R and Kender, J. 1996. Finding skin in color images. *International Conference on Automatic Face and Gesture Recognition*, (1996), 312-317.
- [7] Manders, C., Farbiz, M., Chong, J. H., Tang K, K. Y. , Chua, G., and Loke, M.H. 2008. Robust hand tracking using a skin tone and depth joint probability model, *International Conference on Automatic Face and Gesture Recognition*, (2008).
- [8] K. Nickel and R. Stiefelhagen. Pointing gesture recognition based on 3D tracking of face, hands and head orientation, In *Proceedings of the Fifth International Conference on Multimodal Interfaces*, pp. 140-146, 2003.
- [9] Ong, E. J and Bowden, R. 2004. A boosted classifier tree for hand shape detection, In *Proceedings of the 6th IEEE*

Conference on Automatic Face and Gesture Recognition, 2004. 889-894.

- [10] Y. Raja, S. J. McKenna and S. G. Gong. Colour model detection and adaptation in dynamic scenes, *5th European Conference on Computer Vision*, pp. 460-474, 1998.
- [11] L. Sigal, S. Sclaroff and V. Athitsos. Skin Color-based video segmentation under time-varying illumination, *IEEE Trans on Pattern Analysis and Machine Intelligence*, 26(7), pp. 862-877, 2004.
- [12] Y. Wu and T. S. Huang. Color tracking by transductive learning, In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 133-138, 2000.
- [13] Y. Wu, Q. Liu and T. S. Huang. An adaptive self-organizing color segmentation algorithm with application to robust real-time human hand localization, In *Proceedings of Asian Conference on Computer Vision*, pp. 1106-1111, 2000.
- [14] X. M. Yin and M. Xie. Hand image segmentation using color and RCE neural network. *Journal of Robotics and Autonomous Systems*, 34(4), pp. 235-250, 2001.
- [15] Z. Zhang, Y. Wu, Y. Shan and S. Shafer. Visual panel: Virtual mouse keyboard and 3D controller with an ordinary piece of paper, In *Proceedings of the Workshop on Perceptive User Interface*, pp. 1-8, 2001.
- [16] X. Zhu, J. Yang, and A. Waibel. Segmentation hands of arbitrary color, In *Proceedings of the IEEE Conference on Automatic Face and Gesture Recognition*, pp. 90-95, 2000.
- [17] M. Yuan, F. Farbiz, C.M. Manders and T. Yen. Robust hand tracking using simple color classification technique, In *International Journal of Virtual Reality*, 8(2), pp. 7-12, 2009.