Modeling of an Intuitive Gesture Recognition System

Nithish R
Department Of Computer Science And Engineering
Federal Institute Of Science And Technology,
Ernakulam, Kerala

Unnikrishnan T A
Department Of Computer Science And Engineering
Federal Institute Of Science And Technology,
Ernakulam, Kerala

ABSTRACT

This paper presents a brief study of the various techniques that are used to model the intuitive gestures, in particular the gestures involving the hands. Here we study mainly three techniques: Haar Classifiers, Hidden Markov Model & Vision Based Tracking Using Color Markers. We also design a gesture based interaction system using vision based tracking using color markers to interact with the computer. The system uses both static & interactive gestures.

General Terms

Human Computer Interaction, Pattern Recognition, Gesture Based Computing.

Keywords

Gesture Recognition, Haar Classifiers, Hidden Markov Model, Vision based tracking using color markers, Gesture Based Interface for Human Computer Interaction.

1. INTRODUCTION

Gesture recognition is an active area of research in the field of image processing. Gestures are being used for a wide variety of human computer interfaces for a natural and intuitive form of interaction. Humans use a wide variety of gestures in their day to day life like “Hello” (Raise of the hand), “Good Bye” (Waving of the hands), etc. So by using these day to day natural gestures to interact with the computer, the user will feel at home with the system instead of using the mechanically designed keyboard and mouse. A set of natural & intuitive gestures can be designed. By using suitable gesture recognition techniques, the gestures can be recognized by the computer and the user can have a natural form of interaction with the system. The gestures can be in the form of static hand postures or interactive in the form of drawings. All these gestures can be recognized by capturing these actions from the user using a camera and then applying suitable gesture recognition techniques.

Gesture based interfaces are extremely useful in applications like Gaming where the user would be able to interact with the game in a natural and immersive way with the help of hand gestures instead of using the keyboard and mouse for interacting with the game. This has already been seen in some of the commercial gaming consoles like the Nintendo Wii, PlayStation Move from Sony and Kinect from Microsoft. The use of gestures would also be useful in drawing and simulation systems. It would also be of use in helping the people with disability in speaking in communicating with others by using the sign language as input gesture and obtaining its equivalent speech as output from the system. It also has immense applications in interactive learning systems where the students can interact with a given model in different ways which the teacher may not be able to show to the class. It can again be used to control the secondary devices in a system conveniently as illustrated by Song et al. [12]. Some of the other interfaces using gesture recognition include the Sixth Sense device developed by Mistry et al. [8] and the interface to control music playback using gestures developed by Henze et al. [9].

In this paper, we discuss three of the commonly used methods for modeling the gestures. We deal with hand gestures predominantly as hand offers the most user friendly way to perform the gestures. The hand has got over twenty five degrees of freedom. It also makes it difficult to identify and track these gestures. We have also developed a system which can be used by the users to interact with the computer intuitively through the use of gestures.

2. EXISTING TECHNOLOGIES

2.1 Haar Classifiers

Haar Classifiers are mainly used to detect and classify the static hand postures. It is a statistical approach. This method involves an initial training process where the system is trained by providing a set of both positive and negative samples. The positive samples involve the hand gesture whereas the negative samples do not contain the hand gestures. It may also have wrong hand gestures.

The training involves selecting the features from the samples based on some algorithm like the Adaptive Boost. The haar-like features are selected rather than the raw pixel data because these haar features can store general information regarding the gestures. It can also classify the gestures into distinct groups more effectively. The haar-like features are described using the ratio between the dark and light regions in an image. They are also much faster than pixel based calculations. They are also much more robust to lighting and noise conditions as they compare the gray level difference which would affect the entire image according to Chen et al. [1]. The haar-like features for the training are selected manually by an expert who can identify the gestures.

The training may involve multiple layers of classifiers each classifier eliminating some of the negative samples. This is because by using a single classifier, we cannot recognize the hand gestures entirely. Each of the classifier is designed in such a way that they eliminate some of the sure negative samples, i.e., the negative samples are eliminated only if the classifier is confident that it is a negative sample. In such a scenario, the initial classifiers would permit more number of false positives. But it also eliminates some of the negative samples which reduce the processing to be done by later classifiers. The later classifiers on the other hand can give more attention to the samples which have a much higher chance of being positive rather than processing the negative samples again.

The training process is quite lengthy if done manually so Francke et al. [2] proposed a method Active Learning, in which the system is trained initially using some samples and then the future samples are automatically generated by the execution of the already trained system. These samples can then be processed as needed and supplied for training.

Now during the running of the gesture recognition program, the gestures are recognized by comparing each frame with the trained set of haar-like features. Whenever a frame matches with the trained features, the gestures are recognized.

The main advantage with the use of Haar Classifiers is its accuracy in detecting the gestures provided the training is extensive. If the training is not extensive, the results would not be accurate. For the extensive training, the positive and negative samples have to be with varying backgrounds, users, etc. Also a lot of samples need to be provided. For such an extensive training, a lot of time is required as the features have to be selected manually by an experienced operator. It is not very much suited for interactive gesture recognition though.

2.2 Hidden Markov Model

Hidden Markov Model is mainly used to recognize the interactive gestures. The interactive gestures refer to the gestures like drawing which are basically a set of static hand postures which are strung together over consecutive frames. The real problem with identifying the interactive gestures lies in the fact that the gestures can be of varying lengths and they can be started whenever a user desires, i.e., they do not have any predefined start time or position in the frame. This model can be considered as a collection of finite states which are connected by transitions. Each state has also got an associated transition probability which determines the probability of the transition to any other state.

The gestures can be specified by using examples or description. By using examples, the gestures are modeled by giving examples of the different gestures possible in the system and training the system. The models will then contain the details regarding the gestures. In description, each of the gesture is described formally using a gesture description language. Of the two, the example method is more flexible.

In training multi-dimensional hidden markov models are generated with the parameters depending on the training data provided to the system. Each model would have a specific number of states associated with it. These models are then used to recognize the gestures.

The main steps involved while training are preprocessing, feature extraction and classification according to Eickeler et al. [6]. In the preprocessing, the motion is detected by computing the image difference with the adjacent frames. Based on the motion, a feature vector can be generated. The feature vector can be generated by making the features more prominent which also makes the recognition easier. Some of the methods for feature recognition can are templates, zoning, geometric features, etc. The templates are one of the simplest methods of computing the features. The template represents the input data in raw form. It holds the paths that make up a gesture. In zoning method, the images are divided into zones and the paths followed in the zones can be used to obtain the feature vector. In geometric features method, geometric transformations are used to reduce the features in templates. This feature vector can then be used to classify the gestures.

The gestures can then be recognized by processing all the gesture models in parallel. An initial state having a transition to all the gesture models' initial states can be created. Also a final state for the recognition can be created which has a transition from each of the gesture models' final states in the model proposed by Yang et al. [4]. Now the gestures can be recognized by comparing their states with the gesture models by progressing through the feature vectors of the gesture models.

In the study by Eickeler et al. [6], the problem with the positioning of the gesture was resolved by using a region of interest (ROI). By using a ROI, the hand can be segmented and we can then process just the motion of the hands. Now the positions become relative to the position of the hands resolving the problem of gesture positioning in the frame.

The problem with the varying timing associated with the start of a gesture can be resolved by using a separate gesture which indicates the start of the gesture from the user. It can also be resolved by taking a parameterized input from the user.

The Hidden Markov Model can appropriately model the dynamic gestures. Using this model, the dynamic gestures can be recognized with a high degree of success as demonstrated by various applications [4, 5, 6]. The training is time consuming as different examples have to be manually marked. This method has limited success with identifying static hand postures though. It is really complex when both static and interactive gestures are used together and the system slows down.

2.3 Vision Based Tracking Using Color Markers

The major overhead involved in any gesture identification model is the feature extraction and classification. The efficiency of any gesture identification model highly depends on the algorithms used for feature extraction. Gesture recognition using color markers can be an effective way to improve the performance of gesture identification system as they eliminate the need of a complex algorithm for feature extraction.

Vision-based automatic hand gesture recognition has been a very active research topic in recent years with motivating applications such as human computer interaction (HCI), robot control, and sign language interpretation. The general problem is quite challenging due a number of issues including the complicated nature of static and dynamic hand gestures, complex backgrounds, and occlusions. Color markers are used on the finger tips and an associated algorithm is used to detect the presence and color of the markers, through which one can identify which fingers are active in the gesture. The possible permutations of these color markers can be used to generate sufficient number of hand gestures for interaction.

The inconvenience of placing markers on the user's hand is negotiated by the cost effectiveness and flexibility offered by this approach. As with the other two methods we have discussed here, the approach is capable of identifying both static and interactive gestures. No image samples need to be stored and the gesture identification involves some purely two-dimensional geometrical calculations.

The tracking can also be done in three dimensions using sensors and/or depth of field cameras as demonstrated in [10, 11].

3. PROPOSED GESTURE BASED INTERFACE

3.1 Working

The proposed gesture based interface for interaction with the computer involves the use of different color markers on the fore finger & thumbs of both the hands. These colors are tracked using a webcam attached to the computer. Based on the tracking of the gestures, different gestures are recognized. Once the gestures are recognized, the corresponding action is performed on the computer. Some of the gestures we have used in the system include pinch to zoom, photo frame,etc. The system consists of three major modules for the gesture bassed interface: color tracking, gesture identification, and triggering operation on the computer.

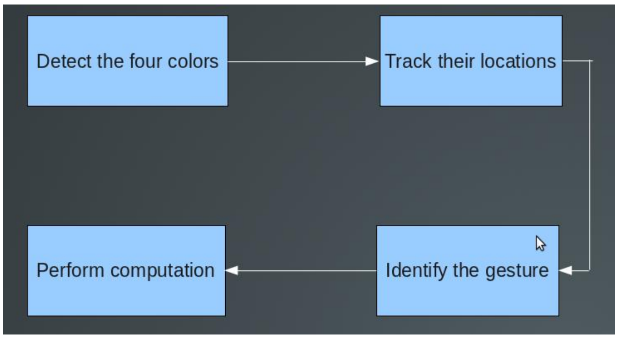
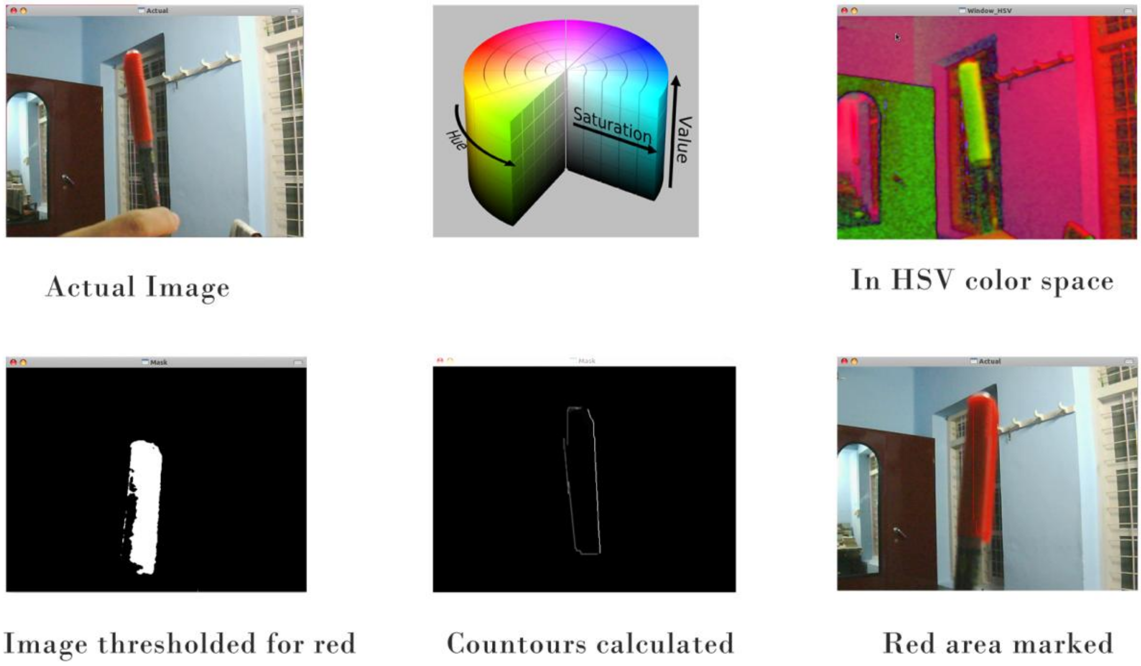


Fig 1: Block diagram of the proposed System

3.2 Color Identification

The system uses four colors: Red, Green, Yellow and Blue. Since the tracking is in real two-dimension, the color tracking module locates the position of each of the four colors as their (x,y) coordinate pairs. Each of the captured frames is processed independently. For color detection, it is found that HSV (Hue Saturation Value) color space is more convenient than the default RGB (Red Blue Green) color space. This is expected because in an RGB color space for each combination of red, blue and green channels we get a different color. When it comes to HSV color space, once the Hue value is fixed, we choose a specific color (say red) and the Saturation determines how light or dark is the chosen color. Value determines the intensity of the color. It should be noted that the OpenCV library uses a different HSV scale. By convention, Hue, Saturation and Value vary from 0 to 255. This is different from the OpenCV HSV color space where Hue value varies only from 0 to179. So the Hue values should be adjusted to fit in to this scale for using them with OpenCV.

To identify a particular color, a lower bound and upper bound on the HSV values of the color to be tracked is calculated at first. The areas of the image, which has the color values in this range, are thresholded from the original image. Noise areas from this thresholded image are removed before the centroid of the thresholded area is calculated. Noise refers to any background areas which lie in the same HSV range. They need to be eliminated before the actual centroid is calculated for the color we track. By applying a set of image transformations, the centroid of the color markers is calculated and the locus of this centroid is continuously tracked for gesture identification.



Actual Image

In HSV color space

Image thresholded for red

Countours calculated

Red area marked

Fig 2: Illustration of detection of Red Color

3.3 Gesture Identification

To identify each of the gestures, they are classified in to two categories: static and interactive. We discuss their identification separately.

Static Gestures: They do not impose the movement of the colors to be identified uniquely. Rather they are detected using the relative position and presence or absence of the of the four color markers, creating sufficient number of exclusive permutations. The term 'relative positions' refers to the positions of the four color markers with respect to each other. None of the gestures have an association with their positions on the frames captured. If the condition for triggering a gesture holds for a sufficient number of frames, the corresponding operation is invoked. The static gestures are relatively easy to identify, but provide only a reduced set of functionalities which demands the system to support interactive gestures or a combination of both.

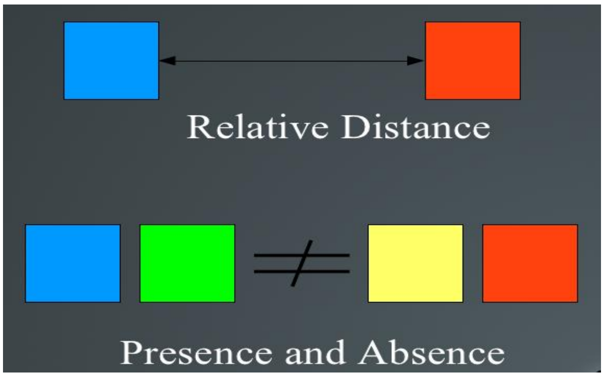


Fig 3: Identification of Static Gestures

Interactive Gestures: The user's interaction using color markers is monitored in each frame to identify each of the interactive gestures. They are dynamic in the sense that the movement of the colors is being checked for gesture identification. To avoid any dependencies to the positions of the colors on the frame, the change in x and y coordinate values with respect to the previous frame is being continuously monitored. With this algorithm, the point from where you start showing an interactive gesture becomes the initial reference point and the whole of the interactive gesture is checked for a known pattern from this reference point until the allocated number of frames gets expired.

Note that in most of the scenarios a combination of the static and interactive gestures will be necessary. A very clear incidence for the same being the technology uses a static gesture to trigger the monitoring of interactive gestures for launchers.

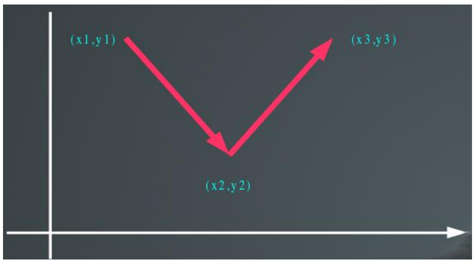


Fig 4: Identification of Interactive Gestures

3.4 Triggering Functionality

Once the gestures are identified, their respective functionality is invoked using system calls through terminal commands. In the proposed system, software based on the XLIB functions, Xdotool has been used to trigger mouse & keyboard operations. It can generate a set of keyboard and mouse events providing a wide range of functionalities for the system.

4. CONCLUSION

Each of the gesture recognition techniques that we studied had their own advantages, disadvantages and applications.

The Haar Classifiers are efficient in recognizing static hand postures. They require a lot of effort while training though. The training requires manual marking of the features of a large set of both positive and negative samples images. But once the training is done, the performance in evaluating static gestures is good. But using the same technique for recognizing interactive gestures makes the system complex.

The Hidden Markov Model is efficient in recognizing interactive gestures by properly modeling the gestures through training. The training involves extracting the feature vectors from the samples of the gesture with the help of a trained operator. Also the training is lengthy. But once the training is done, the model is quite accurate in identifying the interactive gestures. But this system too is not very conducive for identifying the static gestures.

As long as the surroundings does not create considerable amount of noise for color identification, Vision Based Tracking by the use of Color Markers gives excellent tracking and performance in normal conditions. The method although having some inconvenience in the use of color markers on the fingers, proves to be the best in space complexity and cost effectiveness as we are not storing any sample data as in the case of the other models.

The gesture based interface for interaction with the computer has been modeled using the Vision Based Tracking using Color Markers. The system provides good performance in normal lighting conditions but its performance in poor lighting conditions is low. But despite the drawback in low lighting conditions, the system provides the users with an interface to interact with the computer using intuitive hand gestures. The system is highly cost efficient and extensible.

5. ACKNOWLEDGMENTS

The authors gratefully acknowledge the contributions of all the students and staff members of the Department of Computer Science and Engineering of Federal Institute of Science And Technology for their support and motivation. We would like to pay our gratitude to our project in charge, Mr. Mahesh C & Ms Reshmi R, for their constant support and encouragement and for guiding us with patience and enthusiasm in all the stages.

6. REFERENCES

[1] Qing Chen, Nicolas D. Georganas, Emil M. Petriu, "Real-time Vision-based Hand Gesture Recognition Using Haar-like Features" presented at Instrumentation and Measurement Technology Conference – IMTC 2007 Warsaw, Poland, May 1-3, 2007.

[2] Hardy Francke, Javier Ruiz-del-Solar and Rodrigo Verschae, "Real-time Hand Gesture Detection and

Recognition using Boosted Classifiers and Active Learning", Department of Electrical Engineering, Universidad de Chile.

[3] Tin Hninn Hninn Maung, "Real-Time Hand Tracking and Gesture Recognition System Using Neural Networks" presented at World Academy of Science, Engineering and Technology, 2009.

[4] Tie Yang, Yangsheng Xu, "Hidden Markov Model for Gesture Recognition", The Robotics Institute, Carnegie Mellon University, May 1994.

[5] Mahmoud Elmezain, Ayoub Al-Hamadi, Jorg Appenrodt, Bernd Michaelis, "A Hidden Markov Model-Based Continuous Gesture Recognition System for Hand Motion Trajectory", Institute for Electronics, Signal Processing and Communications (IESK), Otto-von-Guericke-University Magdeburg, Germany, 2008.

[6] Stefan Eickeler, Gerhard Rigoll, "Continuous Online Gesture Recognition Based on Hidden Markov Models", Faculty of Electrical Engineering-Computer Science, Gerhard-Mercator-University Duisburg.

[7] Asanterabi Malima, Erol Özgür, and Müjdat Çetin, "A Fast Algorithm For Vision-Based Hand Gesture Recognition For Robot Control", Faculty of Engineering and Natural Sciences, Sabancı University, Tuzla, İstanbul,Turkey.

[8] Pranav Mistry, Pattie Maes, "SixthSense: a wearable gestural interface", SIGGRAPH ASIA Art Gallery & Emerging Technologies, Yokohoma, Japan, December 2009.

[9] Niels Henze, Andreas Löcken, Susanne Boll,T obias Hesselmann and Martin Pielot ,"Free-hand gestures for music playback: deriving gestures with a user-centred process", Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia, 2010.

[10] Van den Bergh, Michael and Van Gool Luc, "Combining RGB and ToF cameras for real-time 3D hand gesture interaction", Proceedings of the 2011 IEEE Workshop on Applications of Computer Vision (WACV), 2011.

[11] Catalin Constantin Moldovan and Ionel Staretu, "Real-time gesture recognition for controlling a virtual hand", Proceedings of 2011 International Conference on Optimization of the Robots and Manipulators (OPTIROB 2011), Romania, 2011.

[12] Yale Song, David Demirdjian and Randall Davis,"Tracking Body and Hands For Gesture Recognition: NATOPS Aircraft Handling Signals Database", Proceedings of the 9th IEEE Conference on Automatic Face and Gesture Recognition (FG 2011), March 2011.