# Automated Motion Based Tracking in Real Time for Human Computer Interaction

R V Nataraj
School of Computing
Shanmugha Arts, Science, Technology & Research Academy (SASTRA) DEEMED UNIVERSITY
Thanjavur, Tamil Nadu
INDIA
rvnataraj@rediffmail.com

A Uma Makeswari
School of Computing
Shanmugha Arts, Science, Technology & Research Academy (SASTRA) DEEMED UNIVERSITY
Thanjavur, Tamil Nadu
INDIA
aum@cse.sastra.edu

*Abstract* – **Visual tracking is one of the major areas of research in computer vision and robotics. Object motion in a video sequence carries a lot of information and can be used in several applications such as Human Computer Interaction, security surveillance, robotics navigation and traffic monitoring. This paper presents distinguished, motion based real time vision system that provides user an alternate way to interact with the computer more naturally and effectively. With the help of video camera, this system detects and tracks the user's motion in the camera's field of view and translates the movements into mouse pointer movements on the computer screen. This system tracks the user's motion automatically and does not need any initialization. The prototype of the system has been implemented and tested in real time under various and varying lighting conditions.**

## I. INTRODUCTION

In recent years there have been efforts [2][3][4][5][6] to replace the traditional human computer interfaces like mouse and keyboard with intelligent interfaces using computer vision. Such intelligent interfaces are used in wide variety of applications: Human computer interaction, virtual reality, aid for people with severe disabilities and Interactive games.

Main advantages of using visual input are that visual information makes it possible to communicate computers at a distance i.e. no physical contact is needed with the equipment to be controlled. Also traditional human computer interfaces necessitate good manual dexterity. This makes life ordeal for the physically challenged especially quadriplegia. Also touch screen, track ball and track pad in laptop poses several problems. They are very cumbersome to operate. Since track ball needs to be operated using fingers, it rouses perturbing feeling. Though the paramount purpose of this technology is to assist physically challenged in operating the computer by assuaging the physical intervention, it also gives extra degree of control for normal users in operating a computer. With this motion based human computer interface one can develop intriguing interactive games that doesn't need any keyboard, joystick or folkloric mouse. A video camera can be setup to focus the user's movement so that one can activate menu options and interact with the game environment.

This paper consists of following. A brief description of related work is contained in section 2. Section 3 gives brief overview of the system and section 4 describes the proposed method for locating and tracking the moving objects. Advantages of the system are described in section 5. Experimental results are provided in section 7 and section 8 concludes this paper.

## II. RELATED WORK

There have been several approaches to control the mouse pointer through head movement, eye movement based on different principles. Some of these approaches are briefly discussed in this section.

Early research in Human computer interaction enabled users to access the computer by moving their eyes [3]. Five electrodes were placed around the eyes. The electrodes sensed the eye movement and transmitted it to the computer. Some systems uses head operated infrared transmitter for controlling mouse pointer [4]. Infrared technology also requires the line of sight between sender and receiver. And infrared-based human computer interaction requires some wearable devices for the transmission of infrared rays and thus poses severe inconvenience to the user.

Wearing helmets and electrodes might cause uneasiness and fretfulness to the user. With recent advances in camera technology and increasing computing power, current research in human computer interaction is based on visual hand/face gesture recognition and tracking systems using video cameras.

Another approach is to use the video camera to track the movement of the user in real time and translates them into mouse pointer on the screen [2]. Body features such as the tip of the user's nose or finger can be tracked. The system [2] involves two computers: the vision and user computer. Vision computer does the visual tracking part and the user computer runs a special driver and other application software the user wishes to use. The feature to be tracked is selected by clicking on the feature. When the user/supervisor initially clicks on the feature a square is drawn around the feature and the sub-image within this square is used as a template to determine the position of the feature in the next image frame.

Another approach is eye-head controlled human computer interface [5]. Most of the methods that have been described uses template matching algorithms. Template matching algorithms especially in the area of human computer interaction suffers from several disadvantages. One of the disadvantages is template matching algorithms suffers from computational costs. The computational costs are directly proportional to the template size i.e. larger the template sizes higher the computational costs. Also template matching is sensitive towards geometrical variations of the object being tracked. When using same template through a sequence of images where the object scales, twists, translates, it is natural that the template ultimately becomes obsolete. Another disadvantage is that the template needs to be initialised.

Since template-matching algorithms tends to lose the objects when the appearance changes, it results in drifting away from targets. Tracking with fixed templates may be reliable over very short duration but it copes poorly over longer durations. So, for applications that require long term tracking such as vision based human computer interaction template-matching algorithms poses several difficulties.

Some systems [5][7] uses image thresholding technique. In this similarity measures are computed in the binarized image that is separated by a proper threshold value. For such systems, it is imperative that a proper threshold value should be chosen.

This paper presents robust, motion based tracking method to construct a novel vision based human computer interface controlled by user's movement.

## III. SYSTEM OVERVIEW

Automated motion based tracking system involves just a single computer equipped with a video camera. A schematic plan of the system is shown in Fig. 1. Automated tracker tracks the user's movement based on the user's motion in the camera's field of view and translates them into the mouse pointer movement on the computer screen.

The user computer shown in Fig. 1. receives the live video stream and displays it. A computer camera is used to take the live video of the user. The camera can be placed above or below the monitor or anywhere the user wishes to have. Once the system is started, the tracker locates the moving objects in the camera's field of view. The moving objects could be the face of the user or his/her fingers or any other object the user wishes to move. The tracking part (tracker) of the system determines the position of the moving object in the image frames and stops tracking automatically when no moving object is found. The coordinate of the moving object i.e. the position of the moving object with respect to the camera's field of view is used to move the mouse pointer on the computer screen.

Many programs require mouse clicks to select objects on the computer screen. Mouse clicking could be implemented either by using hardware interface or software interface. One of the methods is to implement mouse click operation by using a special switch or dedicated hardware that can be operated by the user comfortably. In this method, user can generate mouse clicks after having moved the mouse to the correct needed position. Hardware interface i.e. by using a separate physical switch does not suffer from "Midas Touch" problem. "Midas" is a character in Greek mythology. Midas was granted the power of turning everything he touches into gold. "Midas Touch" in this context is that everything the mouse pointer touches is selected.

Another method is to use software interface. The tracker can be set to generate mouse clicks based on the dwelling time of the moving object at a particular position. Software interface is easy to implement and non-intrusive. But it might suffer from "Midas Touch" problem and it is based on the dwelling time chosen. So, obviously, appropriate correct dwelling time need to be determined to

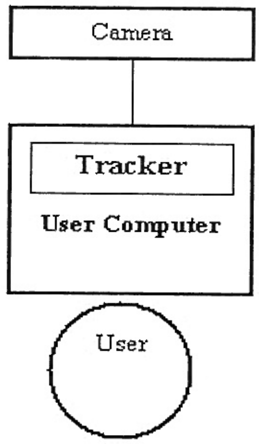


Figure 1. Overview of the system

generate mouse clicks. If the dwell time is too short, chances increase that cells are unintentionally selected. If the dwell time is too long, chances decrease even for intentional selection because of the possible slight drift in the object or feature movement.

## IV. TRACKING ALGORITHM

Motion tracking is often important in several types of applications requiring time varying image analysis. Applications may be object tracking, motion understanding, surveillance system, robotics guidance or facial feature tracking. Also the holistic performance of this system is highly depends on the tracking algorithm used. Since even a small error in tracking could jeopardize the whole system performance, tracking had been deemed more importance.

In this system tracking is done based on the user movement in the camera's field of view. Whenever user moves his head or any other feature in the camera's field of view it is tracked and subsequently the coordinate position of the feature is determined. Also the direction in which the feature is moving and the magnitude of movement are also determined. Based on the direction and magnitude of the movement, the mouse pointer on the screen is moved accordingly. The proposed algorithm for automatically tracking the moving feature in the camera's field of view is shown in Fig. 2. T, given in Fig 2, gives the threshold value for correlation. The system has tested with different values of T, empirically, and it has been found that performance is good when T is set to 0.8. The variables left and right given in the algorithm denote the boundary of the feature from left and right direction respectively and described in Fig. 3. The variable top denotes the upper boundary of the feature and the variable bottom denotes the lower boundary of the feature.

The algorithm given in Fig. 2 determines the boundary of the moving object in real time video stream. At any time the algorithm considers only two image frames (the current image frame and the previous image frame). The image data is segmented into rows and columns. Step 4 in fig. 2 correlates the corresponding columns between current and previous image frame and the cross correlation value is stored in array w. Steps 6 and 7 scans the computed cross correlation value and if the value is less than 0.8 it is concluded that pixel data sets has changed which gives the vertical boundary (left line and right line in fig. 3) of the moving object. Similarly steps 5, 8, 9 operate on the rows and gives the horizontal boundary (top line and bottom line in fig. 3) of the moving object. The horizontal magnitude and direction of the movement of the object is computed with the help of the position of left and right line. Similarly the vertical magnitude and direction of the movement of the object is computed with the help of top and bottom line position. The cross correlation between pixel data sets is calculated using Zero Mean Normalized Cross Correlation and is given in equation (1).

$$ZMNCC(X,Y) = \frac{\sum_i (X[i]-mx)*(Y[i]-my)}{\sqrt{\sum_i (X[i]-mx)^2 * \sum_i Y[i]-my)^2}} \quad (1)$$

where x is the set of image pixel data from image $I_i$, y is the set of image pixel data from image $I_{i-1}$, mx is the mean of data set x, my is the mean of data set y.

The range of the Zero Mean Normalized Cross Correlation is [ -1, 1] with 1 being the perfect match and 0 indicates no correlation at all. –1 indicates that the data set is mirrored i.e. –1 would be produced for data x={1,2,3,4,5} and y={5,4,3,2,1}

## V. ADVANTAGES

Automated motion based visual tracking has got many advantages when compared to the normal visual tracking. In automated visual tracking whenever an object starts moving in the video sequence, it is tracked. In other words, moving objects in the field of view of camera are tracked in the image frames. Some of the advantages of automated tracker especially in the area of human computer interface are follows.

Human computer interface with automated motion based tracking is very simple to initialise and in most cases initialisation is not needed at all. Since the face or object movement is tracked automatically, the user need not have to place his lived-in feature in the feature window as done in [7]. Also the user need not have to click on the feature to be tracked as done in [2]. This makes the task very simple for people with severe disabilities and they can access the system at ease and can greatly benefit from automated motion based tracking. Also people with disabilities might find it difficult to access the computer system using the same feature being tracked all the time. This might generate severe pain. Automated motion based tracker eliminates all these disadvantages.

For each time

1. Store previous image frame to $I_{i-1}$
2. Store current image frame to $I_i$
3. Store 0 to left, right, top, bottom
4. For k = 1 to imagewidth
   Compute the cross correlation for Column k between $I_{i-1}$ image frame and $I_i$ image frame
   Store the correlation coefficient to w[k]
5. For k = 1 to imageheight
   Computer the cross correlation for row k between $I_{i-1}$ image frame and $I_i$ frame
   Store the correlation coefficient to h [k]
6. For n = 1 to imagewidth-1
   Compute the absolute difference between w [ n ] and w[ n + 1 ]
   If (w [ n ] – w [ n + 1]) > T
   Left = n and proceed to next loop.
7. For n = imagewidth to 2
   Compute the absolute difference between w [ n ] and w[ n - 1 ]
   If (w [ n ] – w [ n - 1]) > T
   Right = n and proceed to next loop.
8. For n = 1 to imageheight-1
   Compute the absolute difference between h [ n ] and h[ n + 1 ]
   If (h [ n ] – h [ n + 1]) > T
   Bottom = n and proceed to next loop.
9. For n = imageheight to 2
   Compute the absolute difference between h [ n ] and h[ n - 1 ]
   If (h [ n ] – h [ n + 1]) > T
   Top = n and exit the current loop.
10. Compute the intersection points between top, bottom, left and right.
   The square drawn with the above 4 intersection points gives the moving object.

Figure 2. Algorithm for Motion based visual tracking

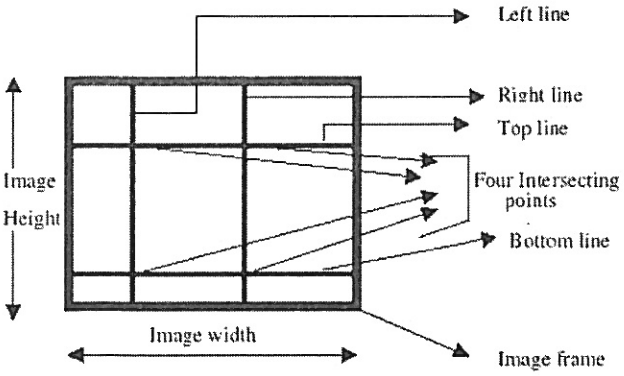


Figure 3. Automated Tracking

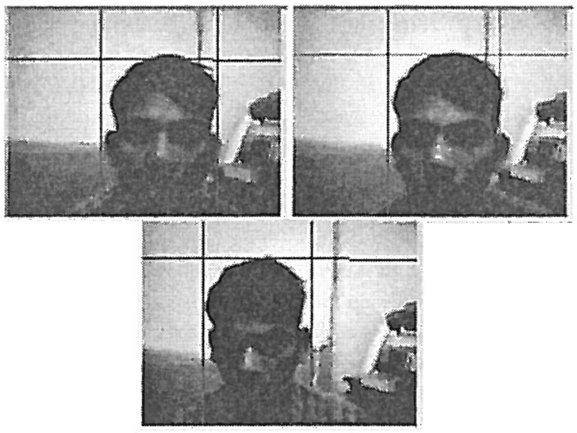

Figure 4. Snapshots of head movement tracking

Most of the vision based human computer interface implements absolute mouse movement. In absolute mouse movement, the x and y coordinates contains normalized absolute coordinates. With respect to image frame and the computer screen, each and every point in the image frame corresponds uniquely to an area in the computer screen. For example if the user wants to move the mouse pointer to the top left, he/she needs to move the feature being tracked to the top left in the image frame. But inherently all the mouse pointer control devices, for example mouse; track pad uses the relative mouse movement. In relative mouse movement the x and y coordinates specifies the relative motion from when the last position reported. Positive value moves the mouse right or down and negative values moves the mouse left or up. For example, negative value for x moves the mouse in the left direction and positive value for x moves the mouse in the right direction. Similarly positive value for y moves the mouse in the down direction and negative value for y moves the mouse in the up direction. So, when compared to absolute mouse movement relative mouse movement, obviously, has got many advantages. Using feature recognition followed by tracking, it is very difficult to implement the relative mouse movement because the feature is tracked continuously. In motion based automated tracker, the same feature need not be continuously tracked and this makes relative mouse movement possible to implement.

"Midas Touch" is one of the common problems in the vision based human computer interface. It is inherently associated with all feature-tracking systems which tracks a particular feature continuously. Since automated motion based tracker does not track any particular feature continuously, "Midas Touch" could be avoided.

## VI. MULTIPLE OBJECTS TRACKING

Currently the system does not track multiple objects. The system has been designed to track only one moving object. Multiple objects tracking have got several applications in the area of autonomous vehicle navigation and surveillance system. Generally it is not required for human computer interaction.

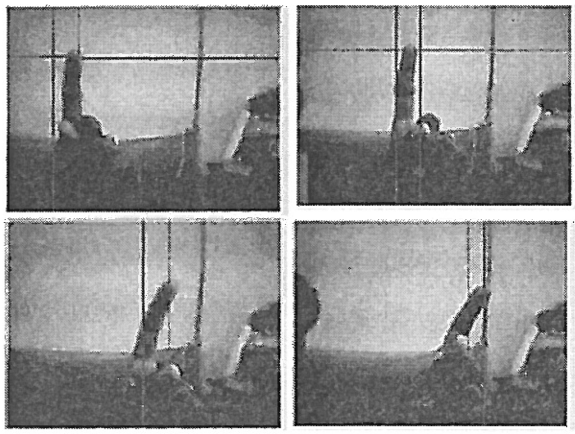

Figure. 5. Snapshots of finger tracking

## VII. EXPERIMENTAL RESULTS

All the experiments were conducted on a single processor system in real time with processor speed 800MHz equipped with Logitech QuickCam web camera. When the system is started, user can move their head or any other feature in the camera's field of view to control the mouse pointer on the computer screen. All video data was gathered under normal lighting conditions and no special lights were used. The image size used is of size 160 x 120 pixels and 320 x 240 pixels. Experiments were also conducted under varying lighting conditions. Even under varying lighting conditions the system was able to track the moving object properly. Since, at any time, the tracking algorithm considers only the last two frames (the current frame and previous image frame), it is quite insensitive to the small varying lighting conditions. The system might give erroneous results under severe continuous varying lighting conditions. Results of the motion based tracking algorithm tracking the movement of the head is given in Fig. 4. The lines shown in the snapshot gives the boundary of the moving object. Automated tracker tracking finger movement is shown in fig. 5.

## VIII. CONCLUSION

The Automated motion based tracking algorithm presented in this paper makes up an alternative communication method that can be operated by several kinds of people and is suitable for people with profound disabilities. The experiences with this hands free Human computer interaction are very inspiring. Because of its generality (ability to track any moving feature in the camera's field of view), the system can cater to the special needs of people with various disabilities. With this technology one can develop intriguing games that can be operated by just head movements. The current system does not track multiple moving objects and an algorithm is being designed to track multiple objects. Future work will incorporate stereo tracking to improve the tracking accuracy.

## IX. REFERENCES


[1] Christopher Fagiani, James Gips, Margrit Betke. "Evaluation of Tracking Methods for Human-Computer Interaction." in proceedings of the Sixth IEEE workshop on Application of Computer Vision (WACV' 02), Dec 2002.

[2] Margrit Betke, James Gips, Peter Fleming "The Camera Mouse: Visual Tracking of Body Features to Provide Computer Access for People with Severe Disabilities" IEEE TRANSACTIONS ON NEURAL SYSTEMS AND REHABILITATION ENGINEERING, VOL. 10, NO. 1, MARCH 2002

[3] J. Gips, P. Olivieri, and J.J. Tecce, "Direct Control of the Computer through Electrodes Placed Around the Eyes", Fifth International Conference on Human Computer Interaction, Orlando (1993). Published in Human-Computer Interaction: Applications and Case Studies, M.J. Smith and G. Salvendy (eds.), Elsevier, (1993) pp. 630-635.

[4] D. G. Evans, R. Drew, and P. Blenkhorn, "Controlling mouse pointer position using an infrared head-operated joystick," *IEEE Trans. Rehab.Eng.*, vol. 8, no. 1, pp. 107–117, 2000.

[5] Jong-Gook Ko, Kyung-Nam Kim and R.S. Ramakrishna, "Facial Feature Tracking for Eye-Head Controlled Human Computer Interface" in Proc. IEEE TENCON'99.

[6] O. Takami, N. Irie, C. Kang, T. Ishimatsu, and T. Ochiai, "Computer interface to use head movement for handicapped people," in Proc. IEEE TENCON'96, Digital Signal Processing Applications, vol. 1, 1996, pp. 468–472.

[7] R V Nataraj and P Raja Shaker, "Automated Visual Tracking in Real Time with Application to Designing Cost-effective, Hands Free Conversational Human Computer Interface" in *Proceedings of the 2004 International Conference on Systemics, Cybernetics, Informatics (ICSCI-2004),* vol. 2, pp. 148-151.

[8] K.Toyama. Look, Ma – no hands! Hands- free cursor control with real time 3D face tracking. In Proc. Workshop on Perceptual User Interfaces (PUI'98), San Fransisco, November 1998

[9] J. Gips, M. Betke, and P. Fleming, "The Camera Mouse: Preliminary investigation of automated visual tracking for computer access," in Proc. Rehabilitation Engineering and Assistive Technology Society of North America 2000 Annu. Conf. (RESNA), Orlando, FL, July 2000, pp. 98–100.

[10] Kristen Grauman, Margrit Betke, James Gips, Gary R. Bradski "Communication via Eye Blinks – Detection and Duration Analysis in Real Time" IEEE 2001 pages – 1010-1017.