IJETCR Open Access

ISSN: 2348 - 2117

International Journal of Engineering Technology and Computer Research (IJETCR)
Available Online at www.ijetcr.org
Volume 2 Issue 7; Page No. 74-80

Simultaneous Facial Feature Tracking and Facial Feature Recognition

Prof. B.B.Gite[1], Utkarsha Saswadkar[2], Prerna Om[3], Abhijeet Singh[4], Akshay Surwase[5]

[1]Head, Singhad Academy of Engineering, Pune, Dept of Computer Engineering, Pune University, Survey no 40, Kondhwa-Saswad Road, Pune, Maharashtra, India-411048
bbgite.sae@sinhgad.edu
[2]Student, Dept of Computer Engineering, Singhad Academy of Engineering, Pune University, Survey no 40, Kondhwa-Saswad Road, Pune, Maharashtra, India-411048
usaswadkar@gmail.com
[3]Student, Dept of Computer Engineering, Singhad Academy of Engineering, Pune, India, Survey no 40, Kondhwa-Saswad Road, Pune, Maharashtra, India-411048
prernaom21@gmail.com
[4]Student, Dept of Computer Engineering, Singhad Academy of Engineering, Pune, India, Survey no 40, Kondhwa-Saswad Road, Pune, Maharashtra, India-411048
kasrajan000@gmail.com
[5]Student, Dept of Computer Engineering, Singhad Academy of Engineering, Pune, India, Survey no 40, Kondhwa-Saswad Road, Pune, Maharashtra, India-411048
Akshaysurwase19@gmail.com

ABSTRACT
This paper presents the real time system interface between computer and human. We will be able to interact with the computer using facial expressions in real time with this fast and affordable technology. It will replace the mouse with the human face. Interactions take place using the eyes and the nose. We make use of Six Segment – Rectangular (SSR) filter, Integral Image and SVM for recognizing the facial features.

Key Words: Face detection, Face tracking, SSR filter, Integral Image, Support Vector Machine algorithm

INTRODUCTION:

Physical disability and Computers:

People with a physical disability may not be able to use a mouse, so they may be interacting with the computer either using a keyboard, one key-press at a time, or with voice-activated software. A possible solution could be the use of Voice-activated software.

Voice-activated software:

It is software that allows a computer to recognize what a user is saying and react accordingly.

Working:

The software works by identifying the sound patterns that the user produces and associating each pattern with a particular entry in its database.

For example,

Voice recognition software used for dictation can recognize that a particular sound pattern equates to its database entry for 'hello.' When the software receives that pattern through a microphone input, the software accesses the database and returns 'hello' in the text that it is typing for the user.

However, this software faces the following issues:

- Voice recognition software works best only after the software has gathered data about each user's speech patterns. It makes many mistakes but becomes effective with time.
- Since the software begins with a database of pre-programmed sound patterns, actual user speech could vary.
- The pronunciation of a given word can change overtime.
- The software requires good quality of the microphone or else the sound patterns may be poor.
- Ambient noise can alter the sound pattern.

Hence, we require a technology that could provide a solution to all these problems!

A possible solution could be Facial feature recognition and Expression tracking.

But, why Face?

This is so because human face is dynamic in nature and has a great degree of variability.

Moreover, we possess similar facial structure– a forehead, two eyebrows, two eyes, one nose and a pair of lips!

Therefore, the aim of this paper is to replace the mouse with the human face to interact with computer.

This technology could be a boon for people with physical disabilities who face problems in communicating with the computer. This technology will help disabled people to communicate through their eyes and nose movements. A camera is used detect and capture the face movements.

1. FEATURES

The features of this technology include:

- The system provides an efficient way to control mouse operations wirelessly, without the help of mouse hardware.
- It has same accuracy as compared to the traditional mouse.
- The system provides the facility of click and select.

2. FACE DETECTION

Face detection is a computer technology that uses digital images and determines the locations and sizes of human faces. This is a milestone in computer technology as it detects face and ignores anything else. Face detection is a very active but challenging research area in the computer application.

Out of large techniques in this field, for our technology, we refer the following approaches:

i. Feature-based Approach

This approach deals with finding the facial feature co-ordinates. (For example: Co-ordinates concerning the position of the eyeballs and nose tip). Verification is done by examining locations and distance of the co-ordinates from each other. This approach is well known for its pixel accuracy and speed.

ii. Image-based Approach

This approach scans the image of the user. This is done by a window that looks for faces at different locations within its scope.

The following method is used for face detection:

Haar-Face Detection Method

This face detection algorithm uses classifiers that work with Haar-like features. This is done using prior classifier training in which thousands of sample views of a face are taken. After training, the classifier applies it to an input image.

If the output of the classifier is ‘1’ then the input is likely to show a face. For all other inputs, the classifier shows ‘0’.

Every location on the image can be searched and checked for a face. This is done because the classifier is designed in such a way that it can be easily resized to search for a particular location in an image. Doing so is more efficient than resizing the entire image.

Following algorithms will be used for Eye Detection:

i. Template-Matching Algorithm

Template-Matching is a well-known method for object detection. In our template matching method, a standard eye pattern is created manually. Given an input image, the correlation values with the standard patterns are computed for the eyes. The existence of an eye is determined based on the correlation values.

This approach has the advantage of being simple to implement. However, it may sometimes be inadequate for eye detection since it cannot effectively deal with variation in scale, pose and shape. Therefore, we refer another algorithm called Adaptive EigenEye Method.

ii. Adaptive EigenEye Method

Adaptive EigenEye Method is based on the well-known method EigenFaces. Since this method is used for eye detection, we call it as “EigenEye Method”. The main idea is to decompose eye images into a small set of characteristic feature images called eigeneyes, which may be thought of as principal components of the original images.These eigeneyes function as the orthogonal basis vectors of a subspace called eyespace. However, the eigenface method is not scale invariant. To provide the scale invariance, we can resize the eye-database once with the information gathered by the face detection algorithm we can provide scale-invariant detection using only one database.

3. FACE TRACKING

Facial Tracking involves computer vision that is able to identify a face within its camera view and track the face’s movement from one location to the next. Following algorithm is used for tracking faces:

Continuously Adaptive Mean-Shift Algorithm

Adaptive Mean Shift algorithm is used for tracking human faces. It is based on robust non-parametric technique. This technique is used to find the mode of probability distributions called the mean shift algorithm. This mean shift algorithm modifies the faces in the video and deals with the problem of dynamically changing colour probability distributions.

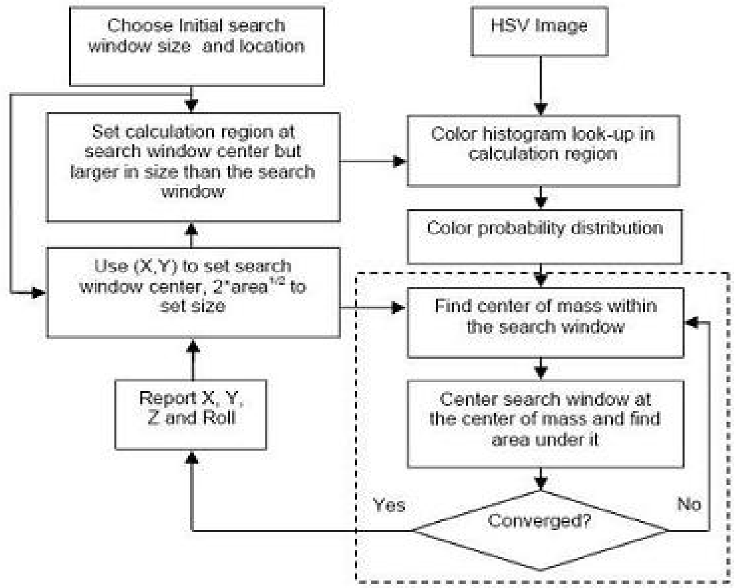

Figure 1: Block diagram of Continuously Adaptive Mean-Shift Algorithm

4. SSR FILTER

The input image is scanned by a rectangle. This rectangle is the SSR Filter that uses the concepts of bright-dark relation for the area Between-The-Eyes(BTE). It is divided into six segments which are as follows:

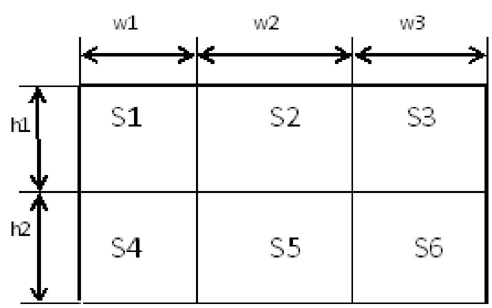

Figure 2: Six segmented rectangular filter.

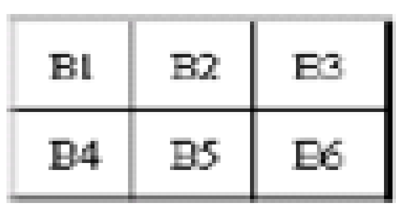

Figure 3: The proposed rectangular filter divided into six segments.

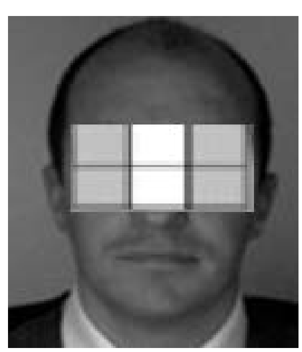

Figure 4: This SSR Filter shows that nose area is brighter than right and left eye area

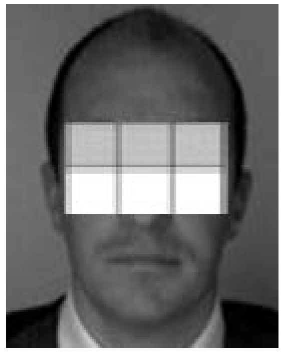

Figure 5: This SSR Filter shows that the Eye area is relatively darker than cheekbone area.

In the SSR Filter, average pixel value of a pixel Si is denoted as Si. These pixel values in each segment are computed initially. The values are then compared with each other. After comparison, it must satisfy certain conditions. Let us understand these conditions with an example.

Let us assume that one eye and an eyebrow are within the segment S3, and then we can say that:
S1 < S2 and S1 < S6 …………. (1)
S3 < S2 and S3 < S6 …………. (2)

These conditions suggest that the eye area is always darker than the nose area and the cheekbone area. Hence, when these conditions are satisfied, we call the image to be a face candidate.

5. INTEGRAL IMAGE

We can make the calculation of SSR filter fast and easy by computing an intermediate representation for image. This intermediate representation is called as integral image.

The pixel value for an integral image (x,y) is given by:

$$\sum_{x' \leq x, y' \leq y}^{n} i(x', y')$$

Therefore, the integral image can be defined as:

$ii(x, y) = \Sigma\, i(x', y')$
$x \leq x, y' \leq y$

To compute an integral image in one pass over an image, following pair of recurrences are used:

$s\ x, y = s\ x, y - 1 + i\ x, y$ ………….. (3)
$ii(x, y) = ii(x - 1, y) + s(x, y)$ ………(4) where $s(x, y)$ is the cumulative row sum and $s(x, -1) = 0$, $ii(-1, y) = 0$.

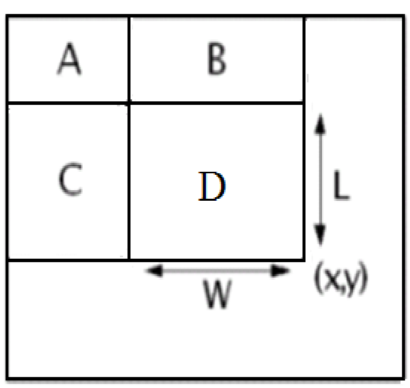

Figure 6: Integral Image

The sum of pixels within rectangle D(sr) can be computed using any four array references shown in Figure 6.

Therefore,

Sr = (ii(x,y)+(x-W,y-L))-(ii(x-W,y)+ii(x,y-L)) ...(5)

6. SUPPORT VECTOR MACHINE

Support Vector Machine (SVM) is a maximum margin classifier. Here, the hyper plane is a plane that separates positive samples from negative ones. According to a theorem in learning theory, to achieve minimal classification error the hyper plane should be in the maximum margin of the training sample. The Samples of data that are closest to the hyper plane are called support vector. SVM is used to verify the Between-The-Eyes (BTE) template.

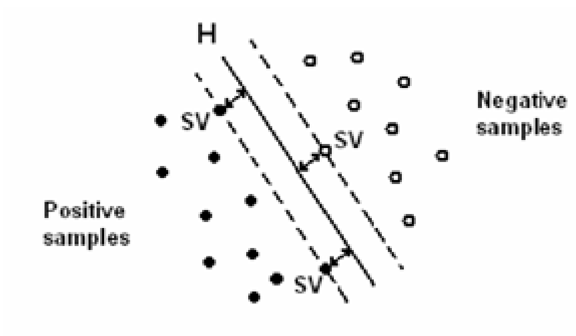

Figure 7: Hyper plane with the maximal margin.

i. Training pattern for SVM

(S1+S3) and (S1+S6) are points extracted from the SSR filter. These points correspond to left and right eye of the candidate.

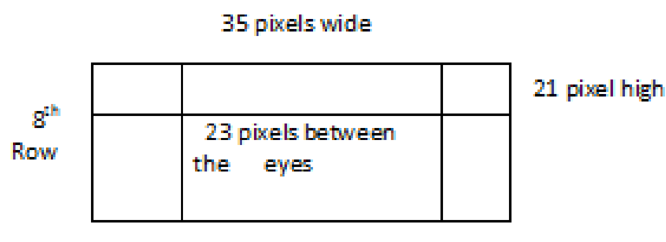

Figure 8: Extracting the training template

Each BTE is computed by extracting the 35pixel wide by 21pixel high template. The distance between the eyes is 23 pixels. Hence, the initial Scale Rate (SR) is computed by dividing the distance between left and right pupil of the candidate with 23.Therefore, now the template has a size of 35 x SR x 21. Thus, the template which has size and alignment of the training template is now horizontally obtained.

ii. Eye and Nose Tracking

BTE template is used to detect the eye movements using pattern matching. A reference BTE template is used as a reference point to achieve better eyes tracking result. The SVM algorithm, SSR filter and Integral image are used to find the eye pupil. The nose tip is found by measuring or checking the intensity using integral image.

Figure 9: Reference BTE template finds the nose tip.

7. IMPLEMENTATION

We require a webcam for this application. The user is at a distance of about 35 centimeters from the screen. When the user runs the application, the camera captures the user's image. This image is processed as a frame at the rate of 30frames/sec. This sequence of frames forms a video. The algorithms for face detection and tracking help us to find the face and eye co-ordinates, after which the BTE template is extracted. The co-ordinates extracted by the BTE are cross-checked or verified with the SVM. On confirmation, the nose tip co-ordinates are found and the video is kept constantly updated.

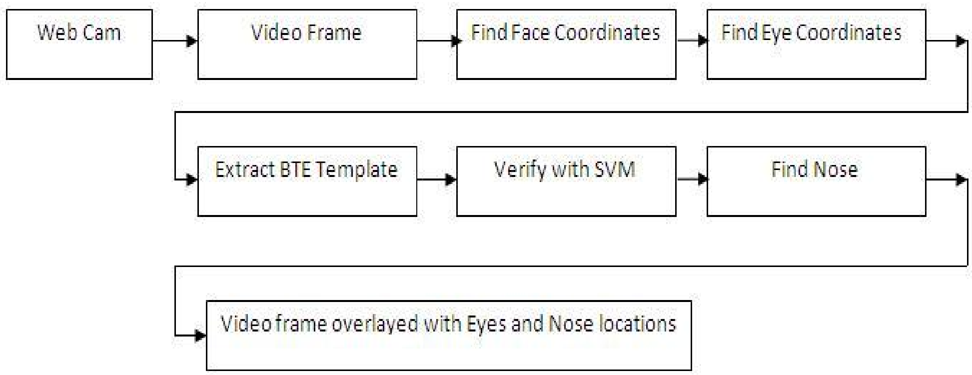

Figure 10: Data Flow Diagram

Hardware requirements include:

- 512 MB RAM
- Hard disk 40 GB
- 1.2GHz processor

Software requirements for the project include:

Java:

Java is an object-oriented programming language that lets developers 'Write Once Run Anywhere (WORA)'. This means that a code that runs on one platform need not be recompiled in another platform.

Java Media Framework:

The Java Media Framework (JMF) is a Java library that enables audio, video and other time-based media to be added to Java applications and applets.

This is an optional package, which can capture, play, stream, multiple media formats. It extends the Java platform.

JMF provides a platform-neural framework for handling multimedia.

8. APPLICATION

This technology could be used to interact with the computer without touching any physical device. Mouse operations such as click, double-click can be done using this technology without using the physical mouse. Hence, this technology is a boon to all physically disabled people who are currently dependent on other people for communicating with the computer.

9. FUTURE SCOPE

Future works may include:

- Improving the robustness against lighting conditions perhaps by using more sophisticated and expensive capturing devices such as infrared cameras that can operate in absence of light and give more accurate tracking results.
- Adding the double left click and the drag mode (enabling/disabling with the right double eye blink) functionalities.

10. CONCLUSION:

"Simultaneous Facial Feature Tracking and Facial Feature Recognition" is boon for physically disable people who are unable to use physical mouse. This efficient, real-time and affordable technology teaches them a new way to interact with computer, without having to depend on anyone else.

11. ACKNOWLEDGEMENT:

We would like to thank Prof. Mr. B.B. Gite for his guidelines in making this paper.

12. REFERENCES:

1. Erik Hjelm and Boon Kee Low, "*Face detection : A survey*".Computer Vision and Image Understanding 83, 236–274, 200.
2. Peter Tiňo "*Support Vector Machines*".School of Computer Science University of Birmingham.
3. PremKuchi, Prasad Gabbur, P. SubbannaBhat, Sumam David, "*Human Face Detection and Tracking using Skin Color Modeling and Connected Component Operators*".
4. Layal El-Afifi, MohamadKaraki and Joelle Korban, "*.Hands-free Interface.- A Fast and Accurate Tracking*

Procedure for Real TimeHuman Computer Interaction".

5. Patrick Naughton and Herbert Schildt, "*Java 2: The Complete Reference"*, Tata McGraw Hill, 1999.