International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org
Volume 3, Issue 5, September-October 2014 ISSN 2278-6856

Implementation of an Improved HCI application for Hand Gesture Recognization

Miss. Medha Joshi[1], Prof. Sonal Patil [2]

[1]PG Student, Department of Computer Science and Engineering, G.H.Raisoni Institute of Engineering and Management, Shirsoli Road, Mohadi, Jalgaon. 425002, Maharashtra

[2]Asstt.Proffesor, Department of Computer Science and Engineering, G.H.Raisoni Institute of Engineering and Management, Shirsoli Road, Mohadi, Jalgaon. 425002, Maharashtra

Abstract

Human Computer Interface techniques provided us with interfaces which are able to interact virtually and help in gradually replacing the traditional input devices such as mouse and keyboard. This technique also increased the interaction types with the computer. The hand gesture recognition system can be used for interfacing between computer and human. In this paper we have specified a computer vision based improved HCI i.e. a mouse, which can control and command the cursor of a computer or a computerized system using a simple web-camera and hand gestures. A simple web-camera is used to track the hand movements made by the user. In order to move the cursor on the computer screen the user simply moves the hand on a surface within the viewing area of the camera. The video generated by the camera is analyzed separately on each frame using computer vision techniques based on the skin detection. After proper processing the events which are generated are used to move the cursor according to hand movements on the computer. The project is all about mouse cursor movement and click events based on skin detection technique. The system is a real time system hence this computer vision based mouse does not require any predefined datasets for recognizing the meaning of the hand gestures made by the user in front of the camera. The improved HCI is developed in a very cost effective way

Keywords: Human Computer Interface, computer vision technique, web-camera, hand gesture recognition, skin detection technique.

1. INTRODUCTION

Recently, Computer is used by many people either for their work or in their spare-time. As the computer technology continues to grow up, the human computer interaction importance is increasing enormously. Every new device can be seen as an attempt to make the computer more intelligent and making humans able to perform more complicated communication with the computer. For making the communication easy between computer and human different types of special input and output devices have been designed over the years for that purpose. Now-a-days most of the mobile devices are using a touch screen technology. However, this technology is still not cheap enough to be used in desktop systems. Hence an alternative way for touch screen in a cost effective compare to touch screen is to create a virtual human computer interaction device such as mouse or keyboard using a webcam and the computer vision techniques [1]. The idea behind is to make a computer understand speech, facial expressions and human gestures and develop a user friendly human computer interfaces (HCI). With the development and realization of virtual environment, the people‘s interacting requirements cannot be satisfied to a great extent by the current user-machine interaction tools and methods including mouse, joystick, keyboard and electronic pen. As the Virtual reality technologies which give humans the feeling of being involved in computer world, it has been a popular research field. At present, the traditional Graphical User Interface (GUI) is the main human-computer interaction mode where the mouse is the primary means of computer operations. The mouse is only 2 degrees of freedom input device, difficult to fulfill the people’s requirement as skillfully as compared to the hand operating which are practiced in natural life to human-computer interaction to minimize the cognitive burden of the interaction, and improve the efficiency of computer operations [2]. With different types of improved HCI techniques, it is now possible to interact with computer beyond the traditional input devices interactions. Gesture and Gesture recognition terms are heavily encountered in human computer interaction systems, but question arises in mind that what do you mean by term “Gesture”?

Gestures are the non-verbally exchanged information where particular messages are communicated by the visible bodily actions. At a time a person can perform innumerable gestures. Gestures allow individuals to communicate a variety of feelings and thoughts i.e. gesture acts a medium of communication for non-vocal communication in conjunction with or without verbal communication which is intended to express meaningful commands. The important aspect of hand gesture recognition system is the psychological aspects of gestures based on hand. As human gestures being major constituent of human communication and are perceived through vision, it is a subject of great interest for computer vision researchers which serves as an important means for human computer interaction. A statement can only be limited to a particular domain of gestures; due to its wide variety of applications it is very difficult to settle on a specific useful definition of gestures. The meaning and the significance associated with different gestures have much impact of the language and culture i.e. they differ very much with cultures, language and environment having less or invariable or universal meaning for single gesture [6]. Scientifically gestures are categorized into two different categories: static and dynamic. A dynamic gesture is intended to change over a period of time whereas a static gesture is observed at the spurt of time. A waving hand means goodbye is an example of dynamic gesture and the stop sign is an example of static gesture [5]. To interpret all the static and dynamic gestures over a period of time is necessary to understand the full message. This complex process is called gesture recognition. In section 2 the proposed system is described. The system architecture and the working of the system are described in section 3. The section 4 describes the result of the system and conclusion is described in section 5.

2. PROPOSED SYSTEM

The proposed system is a real time application system which is depended on real time video processing. This system will replace mouse, one of the traditional input device so that by using simply the hand gestures the user will be able to interact naturally with their computer. The basic block diagram of the overall proposed system is as shown below in the figure.

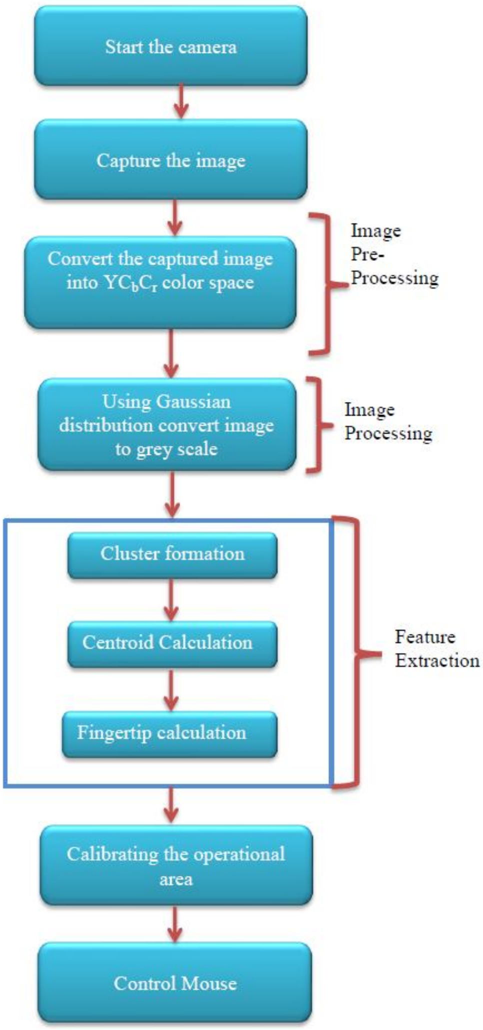


Figure 2 Proposed system flowchart

The above flowchart of the proposed system is shown for the single frame which is captured. The system is a real time system so the flowchart is a continuous process for each and every frame that is captured by the web camera

3. SYSTEM ARCHITECTURE AND WORKING

The following figure shows the setup required for the proposed system

Figure 3 System Architecture

The USB web camera used is INTeX IT-305WC is mounted on the system facing towards the user in order to capture the hand movements which are made by the user. The web camera can be connected to any of the USB port which is available free. The video captures the frames up to 30 fps and supports up to 16 mega pixels. As stated in the objective the proposed system is to find out the alternative for the traditional input device mouse. The working of the project is carried step by step. The camera drivers are installed for the proper working of the camera. The image of hand is captured as the camera is facing the user and the captured image is displayed on the camera window which is provided. The pre-processing of converting the captured image from RGB color space to YCbCr color space as the image captured is in RGB format is carried out. The common RGB representation of color images is not suitable for characterizing skin color as in the RGB space, the triple component (r;g;b) represents not only color but also luminance. In other words the RGB components are dependent to illumination conditions. For this reason skin detection with RGB color space can be unsuccessful when the illumination conditions change. Luminance may also vary across a person's hand due to the ambient lighting and it is not a reliable measure in separating skin from non-skin region. Luminance can be removed from the color representation in the chromatic color space. Therefore the input image which is RGB format is converted to YCbCr color model. Here Y is the luminance component and Cb and Cr are the blue and red chrominance components. The transformation simplicity and explicit separation of luminance and chrominance components makes this color space attractive for skin color modeling. The transformation of the RGB to YCbCr is done as follows [4]

After converting image to YCbCr color space the obtained image is transformed to gray scale image using Gaussian distribution is done as follows.

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.2990 & 0.5870 & 0.1140 \\ -0.1687 & -0.3313 & 0.500 \\ 0.500 & -0.4187 & -0.0813 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

After converting image to YC_bC_r color space the obtained image is transformed to gray scale image using Gaussian distribution is done as follows.

Mean calculation

$$m = E\{x\} \quad (2)$$

$$\text{Value of} \quad x = (C_b, C_r)^T \quad (3)$$

Calculating the covariance matrix

$$C = E\{(x-m)(x-m)^T\} \quad (4)$$

$$P(C_b, C_r) = exp\{-0.5(x-m)^T C^{-1} (x-m)\} \quad (5)$$

Where m is the mean of x, C is the covariance matrix [4] With this Gaussian model, the probability of any pixel belonging to the skin in an image can be obtained in the following manner. Given a pixel with a color value (C1, C2), the probability that this pixel is a skin, is calculated as follows [4]

Probability

$$P(C1, C2) = \exp[-0.5\,(x-m)^T C^{-1}(x-m)] \quad (6)$$

The skin color samples were collected and from that it is seen that the skin color vary widely, because of intensity and not because of chrominance. A survey was made by collecting the skin samples of all possible types and from that we concluded that: 30% of the world population has a black skin color, 30% of the world population has a white skin color and 40% of the world population has brown skin color. These calculations are used in Gaussian mixture model. In this case we have calculated the three probabilities for each pixel, the first for the Gaussian model of black skin (p_ black) , the second for the Gaussian model of brown skin (p_ brown), and the third for the Gaussian white skin (p_ white) . After calculating the three probabilities i.e. we have summed all this probabilities by multiplying it with the percentage of occurrence. The formula is as follows.

$$p(x) = (0.3 * p\ black) + (0.4 * p\ brown) + (0.3\ p\ white) \quad (7)$$

This Gaussian Mixture model has transformed a color image which is transformed from RGB color into YCbCr color space to a grayscale image where the value of each pixel corresponds to its probability of belonging to the skin. This Gaussian model can detect and convert more skin pixel as compared to the single Gaussian model. With a dynamic thresholding (where the threshold is the mean of the grayscale image), images in gray level can then be converted into binary images showing areas of skin and non- skin regions. Since the regions of skin are brighter than other parts of the image, these regions can be segmented by thresholding the rest. It then produces a binary image on which the "1" represent the skin color pixels and "0" for other pixels. To extract the features it is necessary to locate the hand. For localization of hand we find boundary contours of the hand in the image. The scanning of the obtained image is started from left to right. The first white encounter is treated as the left side of the hand. Then we start the scanning of image from right to left. The first white pixel from this side again which is encounter is set as the right side of the hand. Now the scanning is performed in horizontal direction within the vertical boundaries which are defined earlier from left to right and top to bottom. The first encounter white pixel is set as top of the hand. As the hand extends from the bottommost part of the image, there is no cropping required for locating the end of the hand [3]. After localization of hand in the image, we implemented edge detection technique for finding the edges of the hand where the sharp change in the brightness is considered. For detection of boundary edges of hand object, after scanning the image pixels whose value changes rapidly from 0 to 1 are extracted. After finding the edges of hand calculating the centroid is done via image moment, which is the weighted average of pixel's intensities of the image. The centroid can be calculated by first calculating the image moment using this formula.

$$M_{ij} = \sum_x \sum_y x_i y_j\, I(x,y) \quad (8)$$

where Mij is image moment, I(x, y) is the intensity at coordinate (x, y).

$$\{\bar{x}\,\bar{y}\} = \left\{\frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}}\right\} \quad (9)$$

where $\bar{x}, \bar{y}$ are the coordinate of centroid and M_{00} is the area for binary image. After the centroid calculation the peaks which are used to represent the tip of the fingers are to be detected on the entire boundary matrices of hand object segmented earlier. The vertical hand image and horizontal hand image are processed differently for finger region detection. In horizontal image, we consider the x coordinate of the boundary matrices. When after the increment we get the x-boundaries starts decreasing we mark this point as a peak value. In vertical image, we only consider the y coordinates of the boundary matrices. When we get the values of y-boundaries starts increasing after the decrement in the y-boundaries values it is fixed as a peak value or a peak. After marking the detected peaks the highest peak in the hand image is detected using the Euclidean distance which is used to calculate the distance between all the tip of the fingers (peaks detected) and centroid. The formula for calculating Euclidean distance is [3]

$$E.D(a,b) = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2} \quad (10)$$

where 'a' represents all the boundary points and 'b' is the reference point which is centroid itself. Euclidean distance is calculated in order to map the circle. The figure 3 shows the extracted features

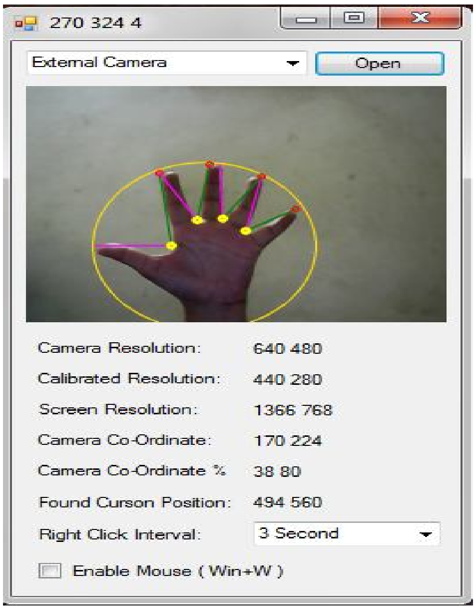


Figure 4 Extracted features

The resize the input image is carried out for mapping between the camera co-ordinates to screen co-ordinates as the camera co-ordinates and screen co-ordinates vary a lot. The mapping of the co-ordinates is done using a simple technique by finding out the ratio. The technique is as described below:

Step1: Assuming a Web Camera Resolution

Step 2: Give the margin to get more prominent workable capture area.

Step 3: Calculate Cab x & Cab y for the calibrated area as follows:

$$\text{Cab x} = \text{Captured X} - \text{Left Margin} \quad (11)$$

$$\text{Cab y} = \text{Captured Y} - \text{Top Margin} \quad (12)$$

[Subtracting the margin the calibration captured resolution is obtain which will be the calibration area of the camera]

Step 4: Finding the percentage of Calibrated co- ordinates in calibrated area as follows

$$\%\ of\ X\ in\ Calibrated\ area = (Cab\ x * 100)/(Total\ Calibration\ area\ X) \quad (13)$$

$$\%\ of\ Y\ in\ Calibrated\ area = (Cab\ x * 100)/(Total\ Calibration\ area\ Y) \quad (14)$$

Step 5: To find display screen co-ordinates % Cab x and % Cab y are used as follows

$$M\_x = (X\ co-ordinates\ of\ display\ screen * \%\ Cab\ x)/100 \quad (15)$$

$$M\_y = (Y\ co-ordinates\ of\ display\ screen * \%\ Cab\ y)/100 \quad (16)$$

Step 6: The X & Y co-ordinate of Display screen is achieved in term of Mx and My. These co-ordinates are the actual position of cursor on the projected screen.

Step 7: Utilized these Mx& My as per the application.

Where,

Captured X = X co-ordinates of web camera resolution.

Captured Y= Y co-ordinates of web camera resolution.

Cab x = X co-ordinates of Calibrated operational area.

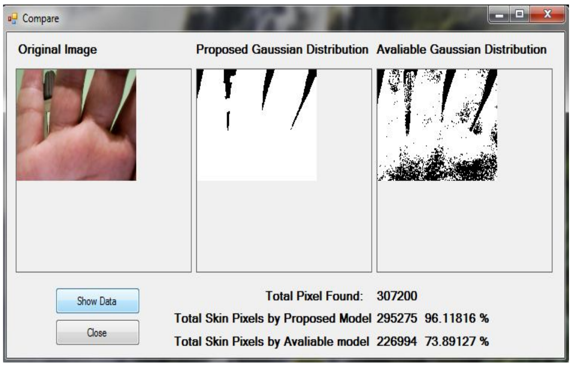
Cab y = Y co-ordinates of Calibrated operational area.

After calibrating the screen the action like right click, double click, left click are performed due to which the system is able to replace the mouse.

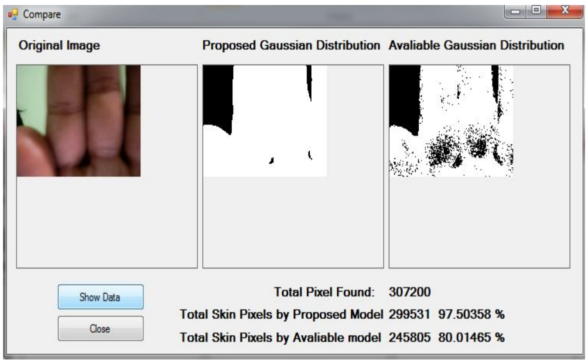
4 RESULTS

This section shows the comparative analysis of our proposed approach with the already existing approach. For experimental results we have taken reading for detecting skin pixel values under different illumination conditions and from the result it was observed that luminance has great impact in skin detection technique. The following shows the result of skin pixels of six conditions for different light conditions at different instances.

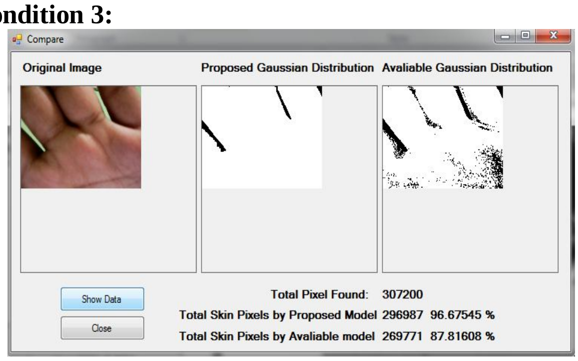
Condition 1:



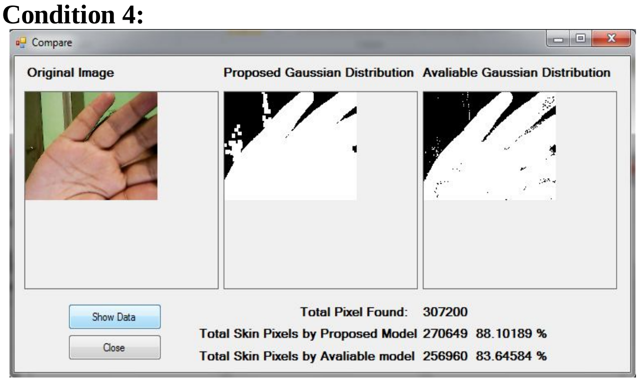
Condition 2:



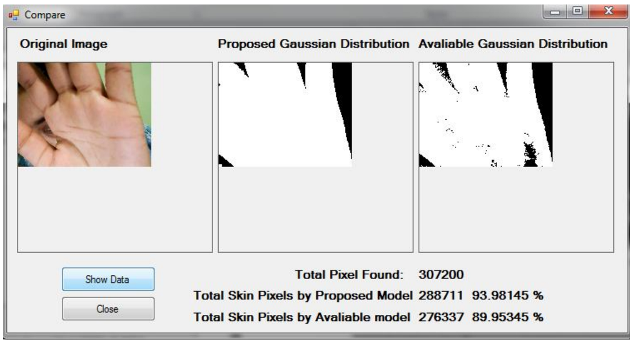
Condition 3:



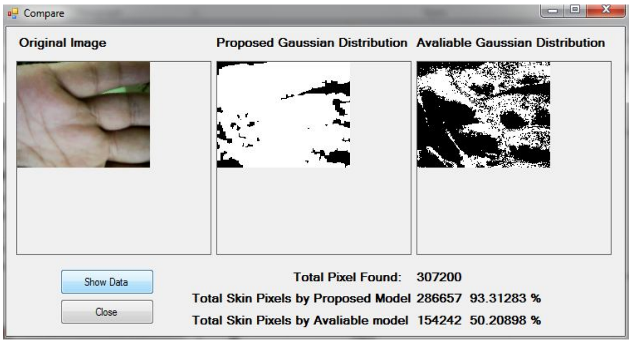
Condition 4:



Condition 5:



Condition 6:



The following table shows the percentage of the total skin pixels recognized by the proposed model and the existing model for six different conditions

Table1 Results for skin pixel detection

Conditions	Proposed Gaussian Model Value for skin pixel detection (%)	Available/Existing Gaussian Model Value for skin pixel detection (%)
Condition 1	96.118	73.891
Condition 2	97.504	80.015
Condition 3	96.675	87.816
Condition 4	88.102	83.646
Condition 5	93.981	89.953
Condition 6	93.313	50.209
Average	94.28217	77.58833

The graph for the above values is shown below where the y- axis represents the percentage of the skin pixels detected and x- axis represents the instances for which the values are calculated.

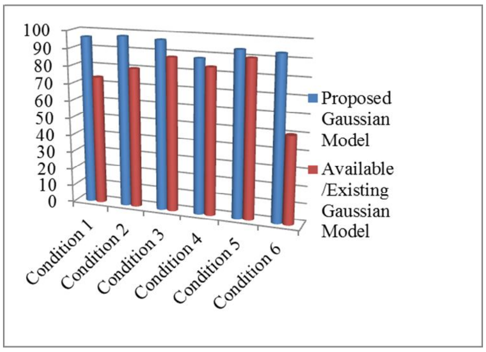


Figure 6: Comparison of the conditions

From the above results it is clear that the available/existing Gaussian model gives very poor result as compared to the proposed Gaussian Model for skin detection. Finding the average of the values for the proposed Gaussian model and Available/Existing Gaussian it is found that the efficiency of the proposed system skin detection rate is 94.282% and available /existing Gaussian Model is 77.588% for different luminance condition and at different instances.

5. CONCLUSION

A new technique has been proposed to control the mouse cursor and implement its function using a real time camera to recognize the hand gestures and control the computer/laptop according to those gestures.This system is based on computer vision algorithms and can do all mouse tasks such as left and right clicking, double clicking and starting the applications using the gestures like notepad, paint, word etc. A new virtual HCI vision-based interface is designed, which is sufficiently robust to replace a computer mouse and extend the interaction capabilities in a cost effective manner. This system realizes the function of the mouse gestures very well. This technique gives the skin detection rate up to 94% which can further be improved by eliminating the environmental challenges which is mostly a tough part. Further by reducing the luminance effect and the requirement of plain background to detect more skin pixels the system if updated will in turn give more accuracy. More features such as the zoom-in and zoom out can also be implemented to make the system more efficient and reliable. This system can also be further implemented in the mobile where using pointing devices like mouse is difficult.

References

[1] Abhik Banerjee, Abhirup Ghosh, Koustuvmoni Bharadwaj, Hemanta Saik (2014)"Mouse Control using a Web Camera based on Colour Detection" International Journal of Computer Trends and Technology (IJCTT) – volume 9 number 1, ISSN: 2231-2803, March 2014.

[2] Rong Chang, Feng Wang and Pengfei You(2010) ,"A Survey on the Development of Multi-touch Technology" IEEE 978-0-7695-4003-0/10 , DOI 10.1109/APWCS.2010.120 © 2010.

[3] Meenakshi Panwar, Pawan Singh Mehra (2011) ,“ Hand Gesture Recognition for Human Computer Interaction”, Proceedings of the 2011 International Conference on Image Information Processing (ICIIP 2011) 978-1-61284-861-7/11/$26.00 ©2011 IEEE

[4] Baozhu Wang, Xiuying Chang, Cuixiang Liu (2011) “Skin Detection and Segmentation of Human Face in Color Images” International Journal of Intelligent Engineering and Systems, Vol.4, No.1,pp 10-17.

[5] Sanjay Meena “A Study on Hand Gesture Recognition Technique” Masters theses ,Department of Electronics and Communication Engineering ,National Institute Of Technology, Rourkela , Orissa ,2011.

[6] Siddharth S. Rautaray, Anupam Agrawal “Vision based hand gesture recognition for human computer interaction: a survey” DOI 10.1007/s10462-012-9356-9, © Springer Science+Business Media Dordrecht 2012.

AUTHOR

Medha Joshi has received her degree in computer science and engineering form Sant Gadge Baba University, Amravati 2010. She is currently perusing her Master’s of Engineering in Computer Science and Engineering from G.H.Raisoni Institute of Engineering and Management, Jalgaon under North Maharashtra University, Jalgaon. Her area of interest in research is Image Processing.

Sonal Patil received B.E. degree in Computer Science & Engineering from S.S.B.T.’S COET,Bambhori from North Maharashtra University in 2008 and M.Tech In CSE From TIT,Bhopal from Rajiv Gandhi Prodyogiki Vishvavidyala in 2012.She is currently working as a Assistant Professor in G.H.Raisoni Institute Of Engineering & Management,Jalgaon. She has published 7 articles in National & International Jornals and 17 papers in National & International Conferences.Member of ISTE