



International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)

www.iasir.net

A FILE AUTHENTICATION SYSTEM USING HAND GESTURE PASSCODES

Karthik, K.Varalakshmi², Dr.S.Ravi³

¹⁻²Department of Computer Science Iyer¹and Engineering, SRM University, Kattankulathur – 603 2002.

³ Department of Computer Science and Engineering, Pondicherry University, INDIA.

Abstract: Traditionally, sign language is commonly used as communication language for auditory handicapped people. In some cases, it is used to assist voice communication. There is also a trend towards using hand gestures as a perspective way of communication between humans and robots. Therefore, many hand gesture recognition researches are proposed. In addition to voice and controller pads, hand gestures can also be an effective way of communication between humans and robots or even between auditory handicapped people and robots. To be an effective sign recognition system, it should be glove-free, fast, small database and accurate. In this paper, we propose a hand gesture recognition system which performs real time recognition. A file authentication process is done by acquiring passcodes which are shown by sign language or gestures. A two digit passcode is assigned to a file. The input is captured using a simple web camera and processed using labelling algorithm and if a match occurs, the file can be viewed by the user and if a mismatch occurs, access is denied.

Keywords: pass code, authentication, smart cards, gesture recognition, labelling algorithm, biometric

I. INTRODUCTION

Computer authentication methods have been widespread since the personal computer was developed in the 1970s. Many authentication methods model physical methods that have been in use for centuries, such as identity cards, visual authentication and passwords. Breaches in computer authentication methods or in physical access controls comprise some of the most devastating attacks against information technology systems. There are numerous advantages of authentication systems which are used to identify the user of a home, ATM or a security clearance computer system. The main purpose of these systems is to validate the user's right to access the system and information, and protect against identity theft and fraud. While there are still drawbacks with some systems, as the business world and the electronic marketplace become more complex, the advantages of authentication are ever more obvious. These traditional smart card technologies are not sufficiently reliable to satisfy the security requirements of electronic transactions, because they lack the capability to differentiate between a genuine individual and an impostor who fraudulently acquired the access privilege (Hong & Jain, 1998). It is therefore relevant that new methods of identification, verification and authentication are emerging based on biometric characteristics such as voice, face, eye (iris-pattern and retina-scan), fingerprint, hand-veins, etc.

Biometric measures are highly personal and unique to a person and cannot easily be transferred to other persons. These characteristics make biometrics advantageous for higher levels of security. Biometric techniques have been in use for as far back as the 19th century in law enforcement departments. Fingerprints are well established as of late, but systems employing fingerprints are now easily impressed by con artists.

A. Authentication Techniques

1. Passwords

The use of a password and username is the most common form of authentication used and is generally considered the cheapest and most convenient method. This requires the user to provide and remember a key containing a specified amount of alpha and numeric characters in relation to a username or e-mail address which must be correct at each login. Users are prompted for a user identifier, email address or user name and a password object that are authenticated to a database or ACL (access control list). Once the user is identified, she is given access to certain areas of a system, as stored in an authorization manifest. Websites, wireless networks, single-user machines and email technologies frequently use user names and passwords to authenticate users. Passwords are vulnerable to guessing, brute-force (trying every possible password combination) and theft attacks.

2. PKI and Smart Cards

PKI (Public Key Infrastructure) provides a cryptographically secure method of authenticating computer users. Users are given two cryptographic keys: a public key and a private key, which are used in the authentication process. When a user tries to authenticate to a system, he presents his public key, which is validated by the server, then an encrypted challenge is presented by the server, which only the user can decrypt with the private key. Since the user's private key is not shared with anyone else, the computer is mathematically certain of the user's identity. Smart cards are the most common method of storing secret keys for authentication.

3. Knowledge-Based Authentication

Banks and other financial institutions often use KBA (Knowledge Based Authentication) to verify a user based on something he knows. Secret questions/answers, pin numbers, and challenge words are common forms of KBA, requiring the user to provide an easily remember-able but unique answer to the challenge question.

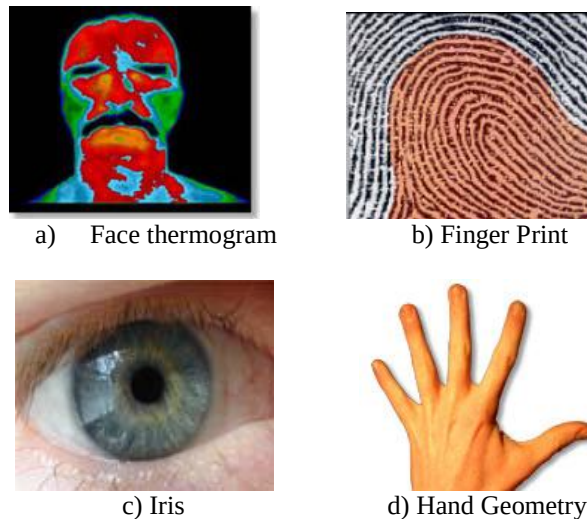


Figure 1 – Various Biometric Authentication Methods

4. Biometrics

Biometrics is an authentication tool to verify a user's identity using some physical aspect. Fingerprint scanners, facial recognition, voice print recognition, retinal and iris scans are all widespread forms of biometric authentication. Users can occasionally bypass or spoof biometric authentication by exploiting a weakness in the underlying technologies, for example, using a picture of the victim to bypass facial recognition.

In chapter 2, we shall see the literature survey of the various authentication systems and their characteristics and we shall also see the system architecture, proposed modules and their explanations, simulation of the implemented systems in the following chapters.

II. LITERATURE SURVEY

In recent years research on gesture recognition has intensified quite tremendously. Hand gesture recognition applications though, have been used mainly for gesture based man machine interfaces (Palvovic, Sharma & Huang, 1997), automatic translation of gesture based natural languages, i.e. sign language (Treisch & Von der Malsberg, 2002) and robotics where robots learn how to handle objects in their own environment by watching human subjects manipulating them.

As for personal authentication using hand gestures, there has not been a lot of research done to investigate the uniqueness of hand gestures, that is, the hand shape tightly coupled with sequential hand movements. This caused an opportunity for research in categories such as securities, password-based access control, etc. There are two types of hand gestures that can be recognized, namely static hand gestures and dynamic hand gestures. A gesture is an expressive movement of a hand(s) to convey an idea or meaning. Therefore, to differentiate between static and dynamic gestures, a static hand gesture can be defined as a gesture which can be recognized by hand posture only or by what is sometimes referred to as hand shape. On the contrary a dynamic hand gesture is defined as a gesture which can be recognized by hand motion which is made up of changes in a sequence of hand postures. The hand motion may be interpreted in 3-D form.

Historically there has been development of different technologies, in this regard, with various features. Traditional features involve passwords, smart cards, Person Identification Number (PIN), etc. These features, however weak, have transcended from being primitive, to being applied in modern technologies. The main attributes for their perpetration are efficiency and ease of implementation in real-time.

A. Biometric Authentication

Biometrics user authentication identifies and verifies the identity of the user with measurements of unique characteristics stored in the programs server, eliminating the need for a password or identification card. Physiological statistics generally used are such things as fingerprint, facial recognition and iris or retina scan. Voice recognition, signature and keystroke scans are referred to as behavioral biometrics. All of the verifications associated with this authentication are highly individual to each user and are very difficult to steal or reproduce. In the field of biometric authentication, static hands received a lot of attention. Jain, Ross and Pankanti (1999) developed a hand geometry based verification system and applied it to a prototype web access system. The system employed features of the hand such as the length and width of fingers, the aspect ratio of the palm or fingers, the thickness of the hand, etc. omitting non-geometric features of the hand such as skin-colour. The experimental results of the system were satisfactory. They obtained 500 images from 50 users, where each user contributed 10 images. Only 360 out of the 500 images were used for testing the hand geometry system and the remaining 140 were discarded due to the incorrect placement of the user's hand. A further contribution was made by Bulatov et al. (2004) on static hand recognition. A document scanner was used to scan an image of the right hand and about 30 features were extracted. The extracted features were finger length (i.e. thumb, index, middle, ring and the last finger), finger width (all fingers), upper and lower finger circle radius, finger perimeter, etc. Tested on a database of 714 hand images from 70 people, Bulatov et al. concluded that hand geometry based recognition systems can be used for medium security applications.

B. Appearance-Based Gesture Recognition

A lot of techniques on appearance-based hand gesture recognition have already been researched. Freeman and Roth (1995) used a histogram orientation based recognition system, whereby an input image was compared to another stored image, using orientation histogram matching. The only problem with this particular method is that some of the different gestures have very similar orientation histograms and it is only most appropriate for close-ups of the hand, as mentioned by the authors. Treisch and Von der Malsberg (2002) developed a method for the accurate estimation of the posture of a gesturing hand, by using elastic graph matching. In elastic graph matching, objects are represented as labeled graphs with an underlying two-dimensional topology. Nodes on the graph are labeled with a local image description and the edges are labeled with a distance vector.

Cui and Weng (2000) developed an algorithm which took into account hand shape and hand motion with hand segmentation tightly integrated into the framework. The hand segmentation is thorough in such a way that even complex backgrounds are masked off and only the hand image remained with a uniform background. Each hand gesture is converted into a sequence of hand images and then the images are compared with other sequences of images stored in a database to establish a match. They achieved a recognition rate of 93.2% for 161 testing sequences (805 images, each sequence was made up of 5 images), and 87% for the most expressive features (MEFs).

Yoon et al. (2001) proposed a system which used combined features of location, angle and velocity. Skin-color and motion are used to locate the hand candidate regions. The hand trajectory is produced by the connection of centroids of the moving hand regions using a hand tracking algorithm. The feature recognition procedure employs a k-means clustering algorithm for a HMM (Hidden Markov Model) codebook. They achieved impressive results of recognition rates 98.96% for the training data set and 93.25% for the testing data set.

A considerable contribution to appearance based recognition was made by Ramamoorthy et al. (2003) who developed a real-time recognition engine which could reliably recognize dynamic hand gestures, despite individual variations. The gestures are distinguished on the basis of hand shapes and the nature of motion. The system is based on HMMs for gesture recognition and a Kalman filter is used for tracking the hand. The system works in such a way that when a hand shape change is detected, the hand shape recognizer is invoked for identifying the new hand shape and the tracker is also initialized. The output of the tracker is thus recognized. The results were of recognition rates averaging above 90% for all users and all the different gestures. HMMs have been employed quite extensively in appearance-based hand gesture recognition. This has left space for the exploration of other methods that can perform pattern analysis. HMMs have also been applied successfully to the area of speech recognition.

C. Hand Gesture Applications

The most obvious application of hand gestures is sign language recognition which has been researched extensively. Sign languages are the most structured gestures made by humans. The meaning of a gesture is dependent on hand shape, direction and sequence which can be an alphabet, word or number. The majority of the initial literature is based on wired "data gloves" which the user must wear and they are quite expensive. The shortfall of most data glove literature is that they focus more on finger signing, whereby the user spells each word with finger signs corresponding to alphabetic letters of the particular language or as numbers as well.

Liu and Lovell (2005) developed a system that used active shape models to extract hand gestures. They used a combination of skin color probability distribution and canny edge detection to determine a reliable contour of the hand. The skin color is interpreted in the HSV color space and further the authors emphasize, as already established previously, that the hue (H) values of human skin color are within invariant ranges. After the hand is segmented and extracted from the original input image, the background is made uniform blue and landmarks are placed around the hand image edges in order to determine an optimal contour.

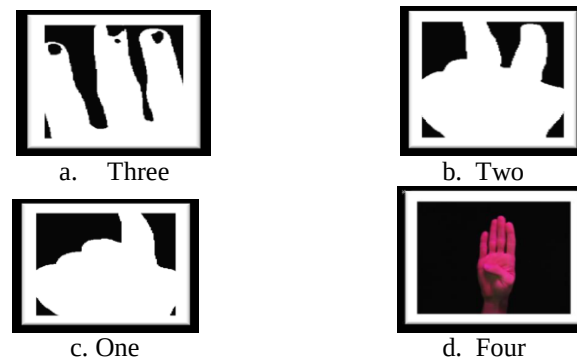


Figure 2 – Hand Gestures

Although Hand gesture recognition has many applications, some of them are still unexplored. One such application is where hand gestures can be used as passcodes for authentication. The proposed system is used to authenticate a file using hand gestures as passcodes. The file is assigned a two digit pin. Input gestures are recorded using a web camera. The input images are converted to a binary image and compared with the pin. If a match is found, the file can be accessed.

III. PROPOSED WORK

In the proposed system framework is discussed. The input hand gestures are captured using a simple web camera. The passcode/pin to lock the file is also got from the user. The hand gesture images are converted to

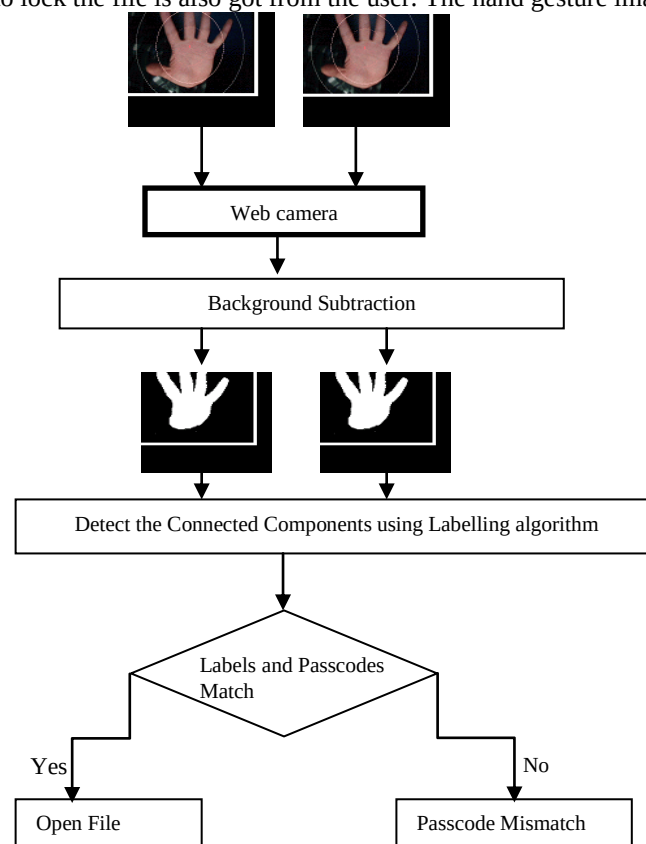


Figure 3 – System Architecture

binary images and then background subtraction is performed with the input images and a background image. The resulting image obtained from the subtraction is processed using labelling algorithm. We mark the

components we get and then number them so as to distinguish them from the non-required one. If the labelled components match with the given pin, the file opens and if there is no match, the system displays the message “Mismatch” and the file cannot be accessed. In this section, we shall illustrate the 3 working modules for the proposed authentication system. They are (a) Capturing the Input images , (b) Background subtraction, and (c) Matching and authentication .

A. Capturing the Input images

The hand gestures are captured using a webcam

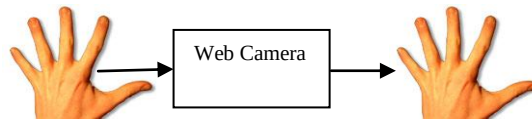


Figure 4- Image Capture

Procedure:

1. Show the input on a plain background
2. Capture the image

We require a plain background here so that it can be useful during background subtraction that we shall see in the next module.



Figure 5 – Capturing the image(s)

B. Background subtraction

The input hand gesture images are acquired by performing a background subtraction operation where the required gesture is extracted from the image and the background is omitted. Then they are converted to binary images.

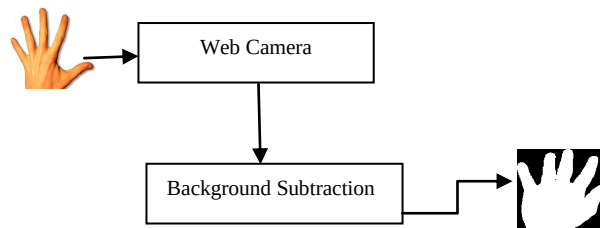


Figure 6 – Background Subtraction

Procedure:

1. Get the required gesture as input using a webcam
2. Capture the image and store it
3. Perform background subtraction to omit the background
4. Convert resulting image to binary



Figure 7– 1st input image after Background Subtraction

a. Background Subtraction Algorithm

1. The two input images are stored in variables *a* and *b*
2. They are converted from RGB to gray scale
(i) $a = \text{rgb2bw}(a)$
3. The image subtraction is done and stored in the variable *c1*
 $c1 = \text{imsubtract}(a, b)$
4. The result is then converted to binary image and stored in variable *d*
 $d = \text{im2bw}(c1)$
5. The size of the binary image is then stored in an array with *r* rows and *c* columns
 $[r \ c] = \text{size}(d)$
6. The following loop sets the values for converting the image to binary format
 for *i*=1 to *r*
 for *j*=1 to *c*
 if $c1(i, j) > 25$ – Threshold value
 $\text{Out}(i, j) = 255$ – value for white
 else
 $\text{Out}(i, j) = 0$ – value for black

We use median filters to adjust the salt and pepper noise in the image and also use the `imfill` command to fill up the required spaces.

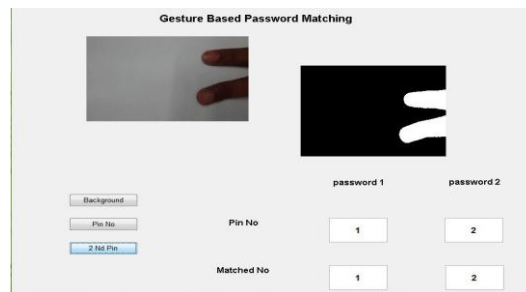


Figure 8 – 2nd input image after background subtraction

C. Matching and Authentication

After the subtraction process and conversion to binary image, we take the resulting image and process it using the labeling algorithm.

(i). Connected Component Labeling Algorithm

Connected-component labeling is an algorithmic application of graph theory, where subsets of connected components are uniquely labeled based on a given heuristic. Connected-component labeling is used in computer vision to detect connected regions in binary digital images, although color images and data with higher dimensionality can also be processed. When integrated into an image recognition system or human-computer interaction interface, connected component labeling can operate on a variety of information. We mark the components we get and then number them so as to distinguish them from the non-required one. 20

A graph, containing vertices and connecting edges, is constructed from relevant input data. The vertices contain information required by the comparison heuristic, while the edges indicate connected 'neighbours'. An algorithm traverses the graph, labeling the vertices based on the connectivity and relative values of their neighbors. Connectivity is determined by the medium; image graphs, for example, can be 4-connected or 8-connected.

Procedure:

1. Get the input images and pins from user
2. Perform background subtraction and get the required binary image.
3. Label the connected components
4. Check if labels and pins match
5. If yes, file opens
6. Otherwise display "password mismatch"

(ii). Algorithm: Match & Authenticate

1. Here we are using two passcodes *pin_no.1* and *pin_no.2*
2. The image subtraction is done and stored in the variable *c1*
 $c1 = \text{imsubtract}(a, b)$
3. The result is then converted to binary image and stored in variable *d*
 $d = \text{im2bw}(c1)$
4. The size of the binary image is then stored in an array with

r rows and *c* columns

[r c]=size(d)

begin

for *i*=1 to *r*

for *j*=1 to *c*

if the threshold value is greater than 25 *c1(i,j)> 25*

Out(i,j)=255 – show the connected components in white

else

Out(i,j)=0- show the connected components in black

end

[Out] = sub(f,b)

5.The subtraction operation is defined as a function and is called when needed. The image is plotted on the 3rd plane of RGB. The image after subtraction is stored in the variable *out*

subplot(223)

imshow(Out)

We use the *bwlabel* command to label the connected components in the binary image

[b cnt] = bwlabel(Out)

6.The values of the connected components are placed in an array containing variables *b* and *cnt* that stand for black and white respectively.

begin

if *cnt == 1*

'pin no is 1'

Else if *cnt == 2*

„pin no is 2'

Else if *cnt == 3*

'pin no is 3'

Else if *cnt == 4*

'pin no is 4'

These steps are repeated for the second input image also

[b cnt1] = bwlabel(Out);

We then take the values of *cnt* and *cnt1* and put them in an array *km*

km = [cnt cnt1];

5. if *km(1) == value of pin no.1 & km(2) == value of pin no.2*

open the file 'matlab commands.doc'

else

print 'password mis matched'

End

The salt and pepper noises that arise while processing are removed using median filters and gaps in the images are filled using the *imfill* command.

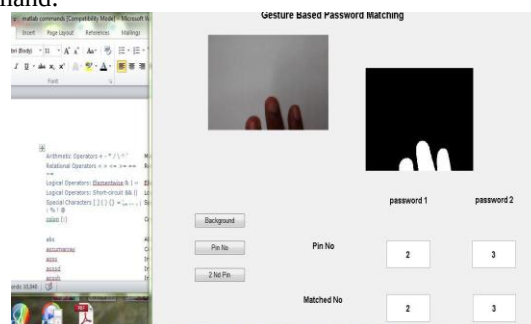


Figure 9 – Match & Authenticate

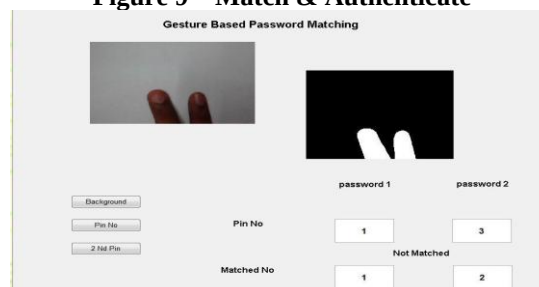


Figure 10 – Passcode Mismatch

IV. CONCLUSIONS

Gestures shown by the user were successfully interpreted using the connected component algorithm. The only advantage of using this method was it involved no complications like the Hue, saturation, values and skin tone detection that are most likely to vary due to lighting conditions. Therefore, for a simple web camera this method produced sufficient results and is capable of producing better results with high resolutions cameras as well. MATLAB has proved itself to be an efficient application. MATLAB was employed to perform the simulation of all the results. It has a lot of built-in functions that helps lessen the amount of time spent on writing the simulation code. However, there were some challenges that we had to come across while designing the system in MATLAB. First of all, for a very efficient result perfect lighting was required. Any disturbance in the lighting will hinder the output. Following that, preventing shadows from falling in the view of the camera was a tedious task. The shadows had a very big impact on the output as they were labeled along with other components creating confusion for the algorithm in determining which the required and non-required components are. In another case, if we had used concepts like Haar wavelet transform with Hue, Saturation values to extract skin tone and other features it would make the design more complex in MATLAB. Efficiency would be less as the results are more likely to be distorted due to improper lighting. All these problems can be eradicated by using appropriate filters to remove the shadows. Developing the codes in programming languages like OpenCV will have better performance and efficiency. Another advancement that can be made is getting the inputs dynamically i.e. setting a perfect interval for each input gesture to get recognized, comparing and determining the output thereby making the system fully automated. Databases can also be added and compared during authentication process.

V. REFERENCES

- [1] Leyuan Liu, Nong Sang, Saiyong Yang [2012] "A Low-Cost Hand Gesture Human-Computer Interaction System" pp: 299-300
- [2] Bulatov, y., Jambawalikar, S., Kumar,P. & Sethia, s [2004] "Hand Recognition Using Geometric classifiers." Proceedings of the 1st International Conference on Biometric Authentication (ICBA).
- [3] CUI, Y. & WENG.J [1999] "A learning-based prediction-and-verification segmentation scheme for hand sign image sequences." IEEE Transaction on Pattern Analysis Machine Intelligence, Vol 21, No. 8, pp. 798–804.
- [4] T.S Huang and V.I Palvovic, "Hand Gesture Modeling, Analysis, and Synthesis", Proc. Of International Workshop on Automatic Face-and Gesture Recognition, No. 95, pp 73-79
- [5] A. Ramamoorthy, N. Vaswani, S. Chaudhury, and S.Banerjee[2003]. "Recognition of Dynamic Hand Gestures", Pattern Recognition, No. 36, pp. 2069-2081,
- [6] V.I Palvovic, R. Sharma, and T.S Huang, "Visual Interpretation of Hand Gestures for Human Computer Interactions, a review", Department of Electrical and Computer Engineering, The Beckman Institute for Advanced Science and Technology, University of Illinois.
- [7] Jain, A.K., Ross, a. & S.Pankanti[1999]. "A Prototype Hand Geometry-based Verification System". 2nd Int'l Conference on Audio- and Video-based Biometric Person Authentication.
- [8] L.Y. Liu, N. Sang, R. Huang[2010], "Background subtraction using shape and color information," Electronics Letters, vol. 46, no.1, pp.41-43
- [9] Jain,A.K., Ross, A. & Prabhakar, S[2004]. "An Introduction to Biometric Recognition" IEEE Transactions on Circuits and Systems for Video Technology, Special Issue an Image- and Video-Based Biometrics, Vol. 14, No. 1.
- [10] Triesch,J. & Von Der Malsburg, C [2002] Classification of Hand Postures against Complex Background using Elastic Graph Matching. Imaging and Vision Computing, No. 20, pp. 937-943
- [11] Siddharth S. Rautaray, Anupam Agrawal , "Real Time Hand Gesture Recognition System For Dynamic Applications" Indian Institute of Information Technology Allahabad, India.