Asian Journal of Information Technology (): - , 2016
ISSN: 1682-3915

Websites Authentication Based on Face Recognition

[1]Alaa Haidar Mohammad
[1] Department of Master in Web Science, Syrian Virtual University, Damascus, Syria.
Email: alaa85a@gmail.com

Supervisors: [2]Bashar Mohammad, [3]Bassel Alkhatib
[2] Department of Software and Information Systems, Faculty of Information Engineering, Tishreen University, Lattakia, Syria
Email: bashhwu@gmail.com

[3] Faculty of Information Technology Engineering-Damascus University, Syria & Faculty of Informatics and Communication Engineering at Arab International University, Syria
Email: b-khateeb@aiu.edu.sy, basselk@scs-net.org

Abstract: Nowadays, the number of websites is growing rapidly coupled with the hacking of the accounts which based on the traditional login method that depends on username and password, so it is important to improve the security of these sites by developing a method of verifying the identity of the users, we propose a face recognition system to achieve this target. In this paper, we review a method to detect faces and eyes depending on skin color then depending on Viola-Jones algorithm. After that, we propose our system which detects faces, eyes, and glasses by using the combination of skin color and Viola-Jones, then we normalize the detected face image. After that we select facial features and configure a face template. This template is stored in the case of registering a new user or compared with pre-stored templates in the case of login. Our experiment reveals that the Detection of 110 images from the FERET database provides 100%, 90% and 69% accuracy in terms of face Detection, eyes Detection and glasses Detection, respectively. The recognition of 80 images from the ORL database (two images for each user) using 320 images related to 40 users (eight for each user) for algorithm training provides 93.75%, 92.5% and 88.75% accuracy in terms of using Eigenface, Fisherface and LPP, respectively.

Keywords: Websites authentication, face detection, face recognition, glasses detection, skin color, Viola-Jones, face background removing.

Introduction

Over the last twenty years, the biometric technology has received significant attention due to its potential for a wide variety of applications in forensics, secure access, and prison security. Nowadays, face detection and recognition applications which have become a popular area of research in computer vision. Our choice of the face recognition system to verify the user when he tries to access website is because it can be used by many internet users, where almost there is no laptop without web camera, in addition to the evolution of the means of internet communication today and the fact that the existing of fast contact with the camera has become axioms for most internet users, While most other biometric identification systems need special equipment (fingerprint reader, Iris special camera....) (Bhatia et al., 2013).

In this paper, we propose a face recognition system to increase the security of the traditional login process that depends on username and password. When a user signs up as a new user the website displays an option to active face security. If the user selects this option a flash interface appears, then our system captures the image of the user automatically using the web camera. We use hybrid face detection algorithm that could detect faces, eyes and glasses in images with different complex backgrounds. In this hybrid algorithm, we use two face detection techniques: Pixel-Based skin color detection and Viola-Jones Facial features detection. By using this hybrid algorithm, we integrate the accuracy of Viola-Jones facial features detection algorithm and the speed of Pixel-Based skin color detection technique. If the system detects a face and two eyes of the user, the user can press a button to save his first photo in a special database, then the user presses the repeat button to allow the system to capture a second image (we save two photos for each user when he signs up as a new user). After face detection, we process the detected face by transforming it into grayscale, and applying a special mask to delete the lower-right corner and the lower left

Corresponding Author: [1] Faculty of Informatics Tishreen University, Department of Master in Web Science, Syrian Virtual University, Damascus, Syria.

corner, after that we select facial features (depending on the face recognition algorithms which are PCA, LDA and LPP) and configure a special template of this face, this template is stored in the case of a new user registration, so when a user tries to log in, a flash interface appears and the system captures the photo and configure a special template of the detected face, then the system compares this template with pre-stored templates in the case of login to compute the similarity measurement using Euclidean distance measure from the input image Fig. 1. Also, we try a method to determine the user identity with acceptable accuracy by using the image of his detected face without any additional information (username and password) Fig. 2. There are always some limitations in reality that affects the flaw of the system, for instance, different camera alignments, low-resolution images, different face expressions and facial reflection which can cause inaccuracy in the retrieving process. These scenarios are handled through several research approaches. In our research, the system recognizes only the faces with a frontal view which are captured by web camera and stored in our special database.

The rest of this paper is organized as follows: Section 2 provides a related work with our study. In section 3, we review some of the techniques used in face detection and our hybrid algorithm for face detection. In Section 4, we review two methods to isolate the detected face. In section 5, we review the techniques we used to recognize faces on our website. In section 6, we show the stages of face detection and face recognition on our website. In section 7, we show some of the experiences we have had on the detection and recognition algorithms and their results. Finally, we conclude our paper in section 8.

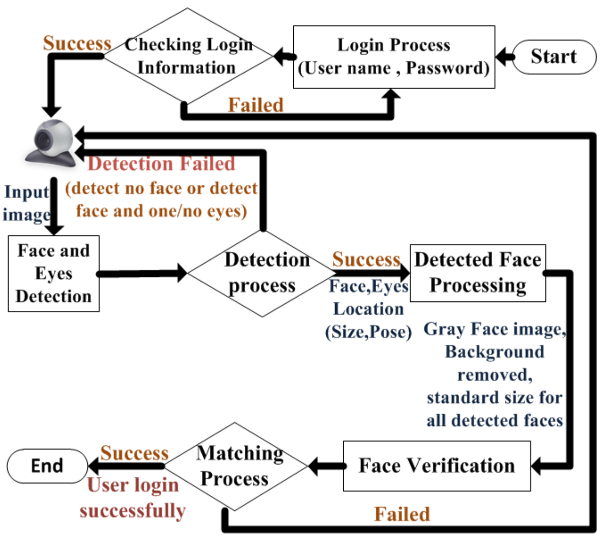

Fig. 1. The stages of facial recognition within our system (user verification)

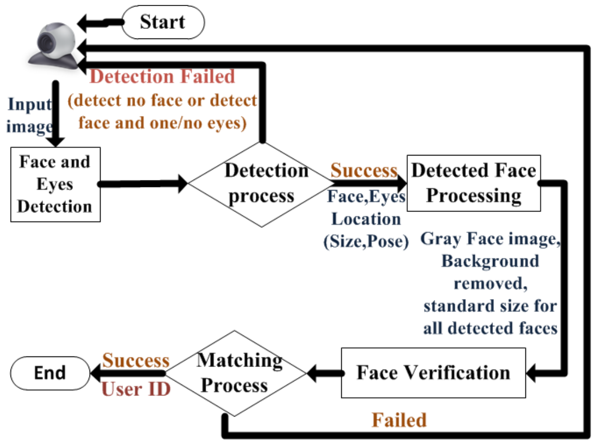

Fig. 2. The stages of facial recognition within our system (user identification)

Related work

Face detecting and recognizing is one of the vast and rapidly growing research areas, and numerous studies and new algorithms annually appear in this area. A lot of companies have invested large sums in developing face detection and recognition algorithms, but a lot of these algorithms are kept confidential and protected by patents, and they mostly deal only with the detection or recognition stage and not as an integrated system started by capturing a photo then detecting the face and finally identifying the detected face. So it is helpful to find studies of a complete system to detect and recognize faces with acceptable accuracy, with the mention of all detection and recognition algorithms stages. In the paper presented by Mehrnaz and Shahram in 2010 (Niazi et al., 2010), they applied Viola-Jones algorithm on the entire image to find face candidates, then applied skin color detector based on the color space HSV to define the candidate regions which have faces. A paper presented by Ijaz and Hadi in 2012 (Khan et al., 2005), they applied skin color detector based on the color space RGB on the entire image to find skin regions, then applied Viola-Jones algorithm on the detected regions, after that they applied Viola-Jones left eye detector on the left upper part of the detected face area which its width equal to a half of the detected face area width, then they applied Viola-Jones right eye detector as the same way. The Facebook team developed a system to identify faces named DeepFace, provides 97.25% accuracy (Taigman et al., 2014). The system aligns face photo and produces a frontal picture of the person (the person appears looking forward), by using a 3-D model based on the average of the frontal face appearance. Then the system uses artificial-intelligence technology known as Deep Learning to find a numerical description of the frontal face appearance, so if there are two different images with numerical descriptions which are similar enough, so DeepFace can decide that the images belong to the same person. In our work, we can detect faces and eyes in the colored and gray images in different light conditions and with partial rotation, and we use a method to remove face background and an algorithm to detect glasses (See Fig. 14), as a result, we try to build an integrated system to detect and recognize faces can work on our website.

Face detection algorithms

In this section we will review some of the techniques we used in face detection then we review the hybrid algorithm we have adopted to detect faces, eyes, and glasses.

Face detection based on skin color: First, we select skin pixels depending on the YC_bC_r color space, then apply Robert Cross Edge detection algorithm (Juneja et al., 2009) to isolate the head from the background, after that, we filter face candidates to faces /not faces by using two methods: The first one is relying on a set of conditions such as human head dimensions, the second one is using Eye Map to make sure that the face candidates represent faces and to determine more precisely the dimensions of the face. The developed algorithm can detect rather large faces (Close enough to the camera), and assumes all the faces are vertical and have a frontal view.

Color segmentation: We relied on YC_bC_r color space. In this color space, luminance information is stored as a single component Y, and chrominance information is stored as two color-difference components (Cb and Cr). Therefore, the luminance information can be easily ignored. The detection window for skin color was determined based on the Histogram of C_b and C_r component, obtained using 231 training skin images Fig. 3. The results are $C_bMax=125$, $C_bMin=91$, $C_rMax=169$ and $C_rMin=130$.

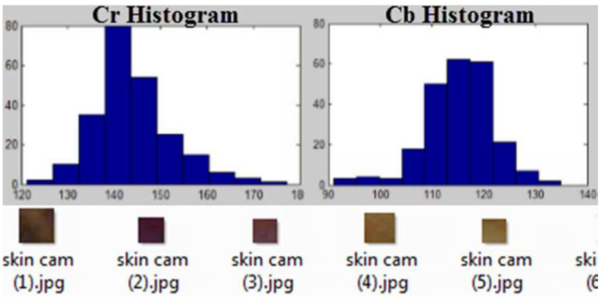

Fig. 3. C_b and C_r histogram

Isolate the Skin Area Depending on Special Conditions: We select face candidate by choosing a relatively large area from the detected skin areas, assuming that the captured image has a frontal view of the face. Then we specify the width of the face candidate according to the width in the mouth level (so the face width will not rely on the width of shoulders in case they are visible or the width of a head area in case the hair color is the same as skin color) Fig. 4.

Determine face location depending on eye map: First, we build two eye maps (Hsu et al., 2002), the first one is from the chrominance component based on the observation that high C_b values and low C_r values are found around the eyes, and the second one is from the luminance component based on the observation that the eyes usually contain both dark and bright pixels in luminance component and they can be found by grayscale morphological operators like dilation and erosion. Second, we combine these two maps into a single eye map. Third, we filter the potential eyes, according to several measurements, including the dimensions of these areas, the relative positions of these areas one on the right and one on the left and the distance between them. Finally, we determine more precisely the dimensions of the face by using eyes location and the conditions depending on the scheme of facial features Fig. 5. All stages are shown in Fig. 4.

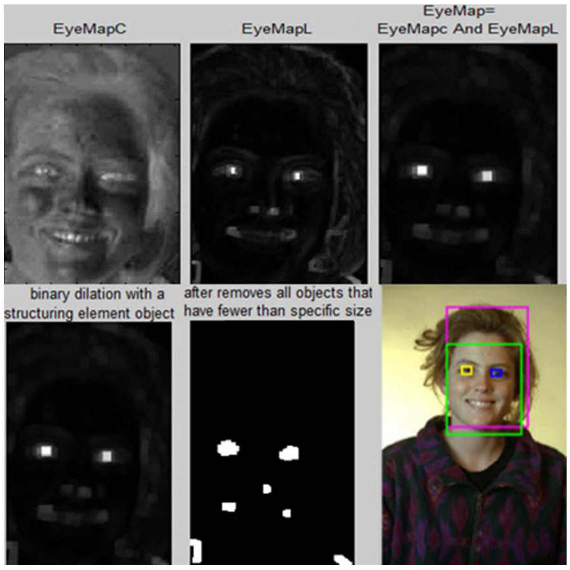

Fig. 4. Face site depending on eye map, in the last photo a comparison between face site depending on the special conditions (pink rectangle) and face site depending on the eye map (green rectangle)

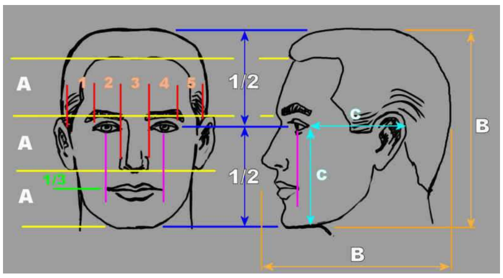

Fig. 5. The map of facial features

Face detection based on Viola-Jones: It was developed by Paul Viola and Micheal Jones and known as Viola-Jones algorithm (Viola et al., 2004). This algorithm includes main points which make it able to build an effective face detector.

- Using features, where the Viola-Jones algorithm doesn't work directly with the image pixels, but uses a set of features called Harr-like features to classify images. The most common reason for using features is that the features can be used to encode the knowledge of the field of Custom (for example, facial features), which is difficult to learn using the pixel-based system. The second reason is that according to the detection system (Viola-Jones) the feature-based system works much faster than a pixel-based system.
- The integral image which is a new image representation. It can be calculated by simple operations at the level of the pixel, after calculating integral image, we can calculate any of the features used by the detector at a fixed time, regardless of location or direction. The integral image is in the form of a matrix with dimensions equal to the original image dimensions + 1, and each pixel in the integrated image contains the sum of all pixels which are located above and left of this pixel in the original image. The sum of pixels within Harr-like features (A) (See Fig. 6) Can be calculated based on four matrixes as follows: L4 + L1- (L2 + L3), here values L1, L2, L3,4L belong to the integral image.

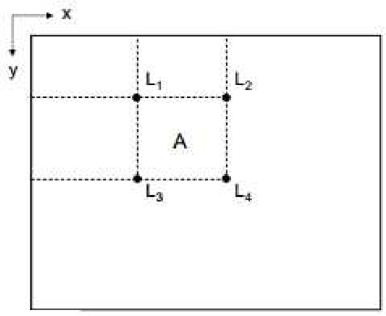

Fig. 6. Calculating features using the integral image

- Using a simple and effective classifier built by selecting a small number of important features of a wide range of potential features is by using AdaBoost training algorithm (Guo et al., 2001). Viola-Jones uses it because in any sub-window in the picture the total number of Haar-like features is very large (larger than the number of pixels within this window), so in order to determine the features that indicate the presence of a face, the algorithm works to identify a small set of facial features and integrate a set of weak classifiers to form a stronger classifier (See Fig. 7).

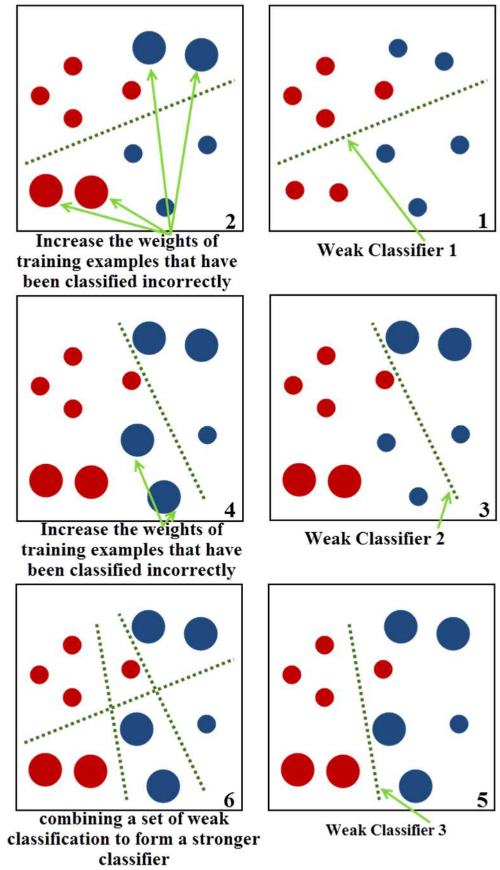

Fig. 7. Stages of AdaBoost algorithm

At the beginning, AdaBoost selects two features which can be noted and processed easily. The first feature relies on the fact that the eye area is usually darker than the cheeks and nose areas. The second feature relies on the fact that the eye area is darker than the bridge of the nose area Fig. 8.

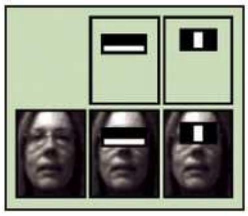

Fig. 8. The first two features tested by AdaBoost algorithm

- Combining effective classifiers in a Cascade Structure, which improves the performance of the detector. The main idea is "a smaller number of sub-windows required to be classified, leads to increasing the efficiency of the detector". The test image contains a large number of sub-windows which should be passed over a series of classifiers to be tested in turn to detect faces. But most of these sub-windows don't contain a face, so we can use a simple initial classifier (based on two features, (See Fig.9)) to reject the majority of sub-windows. Before applying the complex classifiers on the rest of sub-windows, we consequently decrease the False Positive ratio. If a sub-window is rejected by any classifier, it will classify as "not face" and no more processing will be conducted on it (See Fig. 9).

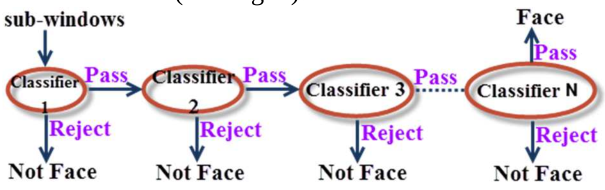

Fig. 9. Representation of a the detection cascade

Viola-Jones implementation: First, we apply Viola-Jones face detector on the input image, and then we apply the Viola-Jones left eye detector and Viola-Jones right eye detector on this image. The result is as shown in Fig. 10.a, we can improve the results by applying left eye detector and right eye detector on the detected face area only Fig. 10.b, in both cases, we note good results in face detection, but not in eyes detection.

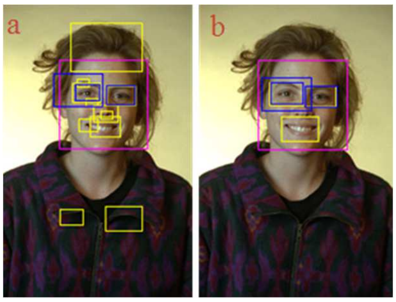

Fig. 10. Viola-Jones implementation example, face detector (pink rectangle), left eye detector (yellow rectangle), right eye detector (blue rectangle)

Hybrid face detection algorithm: The first method "skin color" has good accuracy in face detecting and medium accuracy in eyes detecting and relatively high speeds, while the second method "Viola-Jones" has high accuracy in face detection and medium accuracy in eyes detecting and it consumes relatively a long time for processing. Based on this, we proposed a method that works on integration between the two former methods (after improving the way of using Viola-Jones in eyes detection) to reduce the face detection total time and for better definition of the detected face dimensions. Then take advantage of this integration in the glasses detection on the face.

Improve the way of using Viola-Jones in eyes detection: After applying Viola-Jones algorithm to detect faces within the input image, we determine the left upper part of the detected face area which its width is equal to two-thirds of the detected face area width and apply left eye detector on this area. Then we go back and determine the right upper part of the detected face area width which its width is equal to two-thirds of the detected face area width and apply right eye detector on it (See Fig. 11).

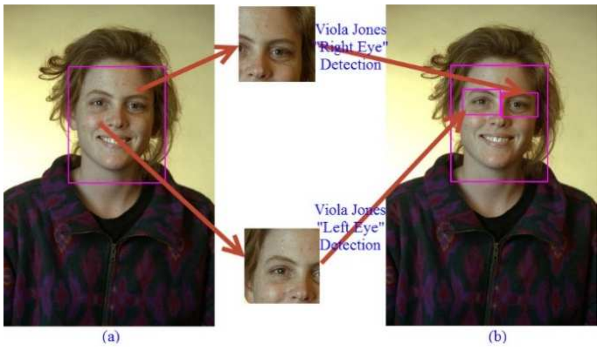

Fig. 11. Improve the way of using Viola-Jones in eyes detection

Hybrid algorithm for face detection: First, we apply the skin color detection algorithm to the input image to isolate areas that may represent faces, then we apply the Viola-Jones algorithm (after improving the way of using Viola-Jones in eyes detection) to identify the faces of the candidate areas. Finally, we apply a statistical method to detect glasses Fig. 13. This hybrid algorithm is used effectively if there are specific areas of the skin within the image (background color is different from the color of the skin, and a few discrete skin areas are within the image). While If there are several overlapping areas of the skin covering a large area of the image, then we apply Viola-Jones algorithm directly on the entire picture, to achieve greater accuracy in the results, and less time in the processing Fig. 12.

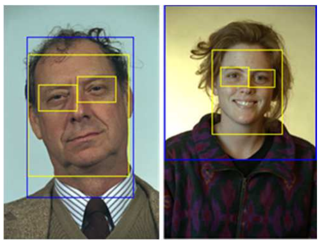

Fig. 12. Hybrid algorithm implementation examples, the result of detection based on skin color (blue rectangle), the result of detection based on Viola-Jones (yellow rectangles)

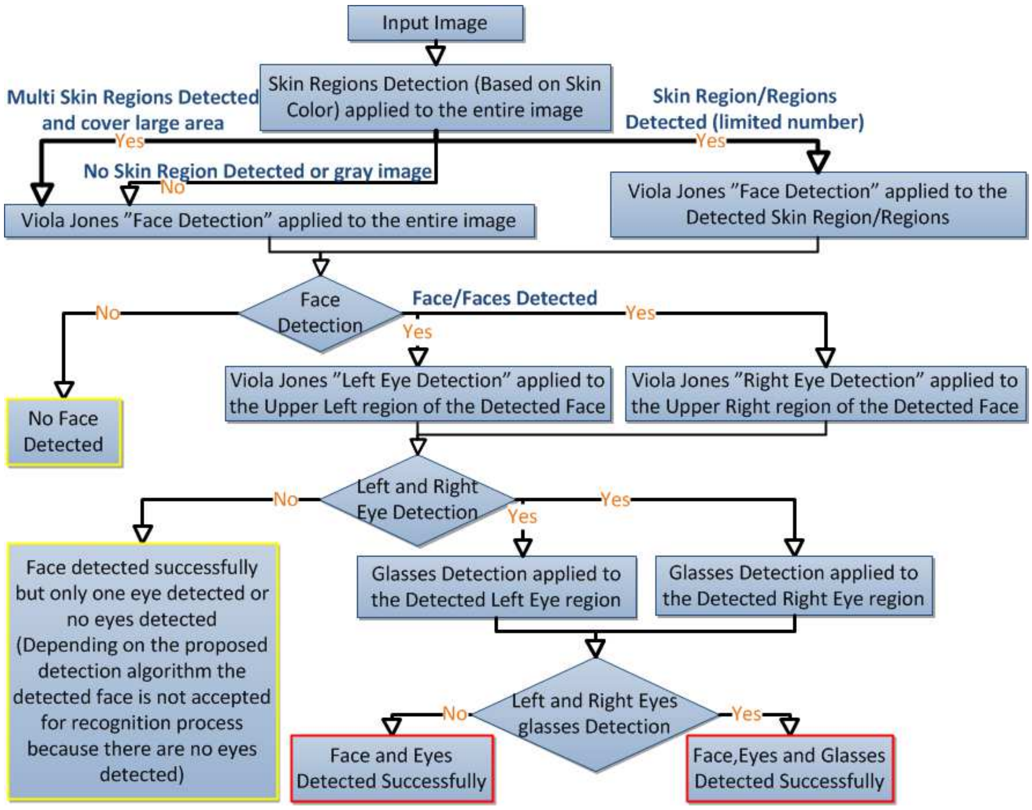

Fig. 13. The steps of the hybrid algorithm to detect faces, eyes and glasses

Statistical glasses detection method: First, we apply Robert Cross Edge detection on the cropped detected eyes pictures, then we find the sum of pixels in black and white areas within the outer perimeter that its width is equal to a fifth of the cropped picture width, with the exception of the upper side of the area of each cropped detected eyes pictures (This ratio was chosen to achieve a good inclusion of the glasses frame parts when count pixels, we exclude the supreme area because it may be included in many pictures eyebrows which can be treated like glasses frame). Finally, we compare these two values for each cropped detected eyes pictures as follows: $(Lcount1 * 3.5 > Lcount2)\&\&(Rcount1 * 3.5 > Rcount2)\&\&(\mathrm{Lcount1} < \mathrm{Lcount2})\ \&\&(\mathrm{Lcount1} < \mathrm{Lcount2})$, here Lcoun1 and Lcount2 are the sum of black and white pixels respectively within the left eye perimeter, Rcoun1 and Rcount2 are the sum of black and white pixels respectively within the right eye perimeter, 3.5 is an experimental value. If this condition is true for both cropped eyes pictures there will be glasses (See Fig. 14).

The problem of this method is the possibility of giving wrong results in the case of low-resolution or excessive makeup on the eyes, or in the case of wearing glasses that do not have the frame (Lens installed from its upper part only) (See Fig. 15).

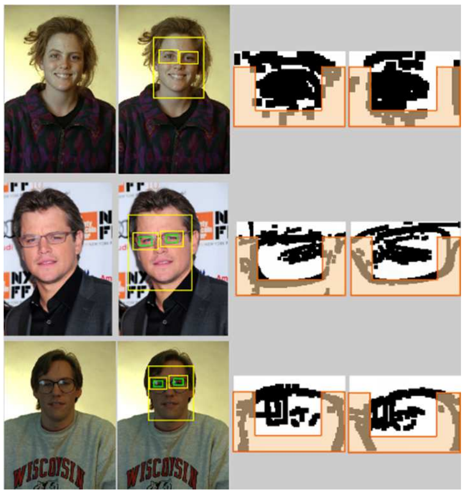

Fig. 14. Statistical glasses detection method implementation examples, skin color result (blue rectangle), Viola-Jones result (yellow rectangles) glasses detection (green rectangle)

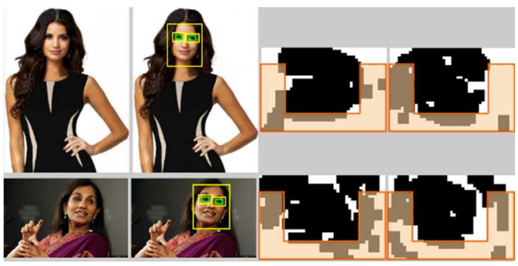

Fig. 15. False positive examples of glasses detection method

Processing of detected face

After face detection, we process the detected image to remove face background, and outfitted to the third stage which is the face recognition. We will display two methods to isolate the detected face, the first method removes face background based on the edge detection, and the second method isolates the face area based on eyes location.

Remove face background based on the edge detection: We apply Robert Cross Edge detection algorithm to select connected objects in the face image, after that, we scan the first and the last quarter of the image and remove all connected objects within this area. The problem of this method is the ability of the edge detection algorithm to detect a connected object in face background, especially, if the background color is similar to the color of the human skin Fig. 16.

Fig. 16. Remove face background based on the edge detection example

Isolate the face area depending on eyes location: We truncate the face area based on eyes location by following the rule "the height of the adult human face / the width of the adult human face = 1.2" Fig. 5, the result is human face height = 1.2 * face width, here face width is equal to the distance between the two external angles of detected eyes regions. Finally, we apply the special mask to delete the lower-right corner and the lower left corner Fig. 17. This method is more reliable than the previous method and gives good results regardless of lighting or face background, because of its primary dependence on the eyes location.

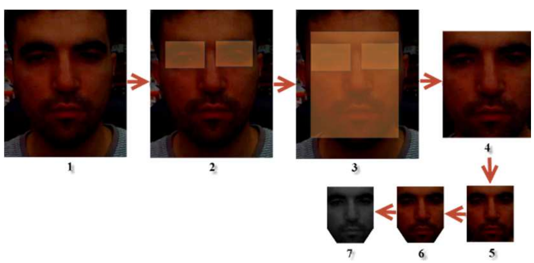

Fig. 17. Isolate the face area based on the eyes location example, then Apply special mask to isolate the face area well

Face recognition

We use three algorithms for face recognition which are Principal Component analysis (PCA), Linear Discriminant analysis (LDA), Locality Preserving Projections (LPP). We will give a brief explanation for each algorithm below, because, there are many studies explaining each algorithm in detail.

Face recognition based on principal component analysis (PCA): It is one of the most important technologies used to face recognition. It aims to reduce the large dimensions of the space data (training set), to smaller dimensions which represent the most important features for face recognition. eigenfaces are a set of eigenvectors used in the computer vision problem of human face recognition. The approach of using eigenfaces for recognition was developed by Sirovich and Kirby (Kirby et al., 1990), and is used to classify the faces by Matthew Turk and Alex Pentland (Turk et al., 1991), their way depends on exploiting distinctive nature of the eigenfaces weights for face representation. This technique is based on Principal Component Analysis (PCA).

Face recognition based on linear discriminant analysis (LDA): Linear Discriminant Analysis is a statistical method often used for dimensionality reduction and classification of data. It was invented by the great statistician Sir R. A. Fisher, who successfully used it for classifying flowers in 1936. Then this method was also recognized by Belhumeur, Hespanha, and Kriegman and so they applied a Discriminant Analysis to face recognition (Belhumeur et al. 1997). The difference between PCA and LDA is that the PCA finds a linear combination of features that maximizes the total scatter across all classes (across all images of all faces), and it is one of the important ways to represent data, but it doesn't consider any classes and so a lot of discriminative information may be lost when ruling out some components, and this can yield bad results, especially when it comes to classification. In order to find a combination of features that separates best between classes, the Linear Discriminant Analysis maximizes the ratio between-class scatter (SB) to within-class scatter (SW). SW is known as Intra-Personal, which represents variations in appearance of the same person because of the differences in lighting and facial expressions, while SB is known as Extra-Personal and represents variations in appearance because of the differences in identity (person).

Face recognition based on locality preserving projections (LPP): Locality preserving projection (LPP) is a manifold learning method used widely in pattern recognition in computer vision (He et al., 2005). The locality Preserving Projections differs from Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA), which preserves the Euclidean structure of face space, while locality Preserving Projections (LPP) finds an embedding that preserves local information, and obtains a face subspace that best detects the essential manifold structure. As a result, LPP should be seen as an alternative to Principal Component Analysis (PCA).

Development and implementation

In this chapter, we show the stages of face detection and face recognition on our website.

Website review: When a user registers a new user the website displays an option to activate face security, if the user selects this option a flash interface will appear, and then our system captures the image of the user automatically across the web camera see Fig. 18.

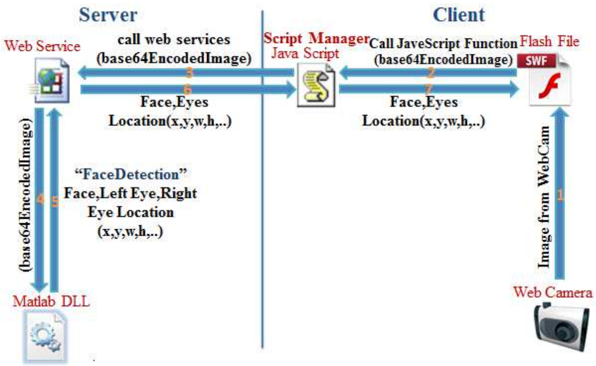

Fig. 18 Face detection method within the site

When a user tries to log in if he had chosen the option of face security the system will capture the photo and decide if the user is the owner of the current account as shown in Fig. 19.

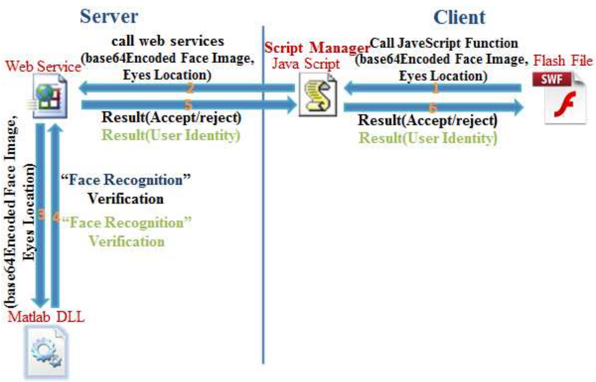

Fig. 19. Face recognition within the site

The following is a practical example of using our website Fig. 20 and Fig. 21.

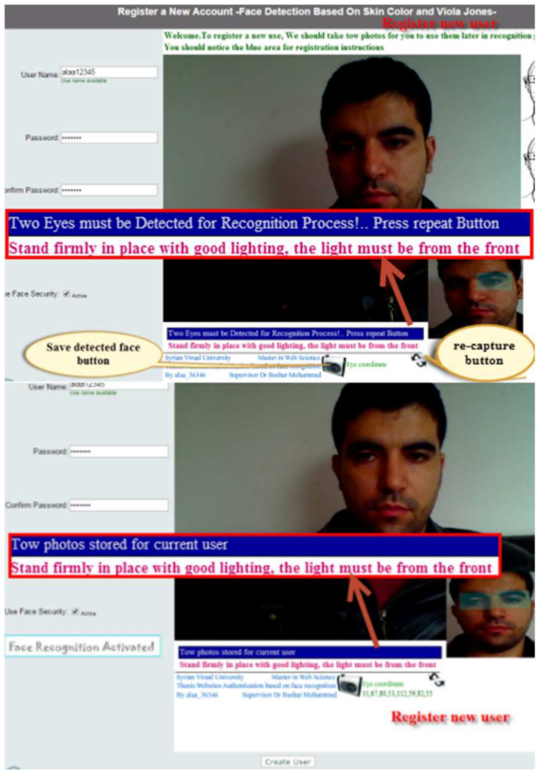

Fig. 20. Face detection on our website

Fig. 21. Face verification and identification on our website

Experimental results

We tested our algorithms on a computer with the parameters shown in table 1.

Table 1. System parameters.

Laptop parameters	
CPU	Intel (R) Core (TM) i5-2450M CPU 2.50GHz
RAM	6 GB
Operating system	Windows 7 64-bit
Simulation software	Matlab R2012a

Experimental results of applying face detection algorithms: We have evaluated detection algorithms on FERET databases which is a standard database of face images that aims to provide standard images to the algorithm developers, and to supply a sufficient number of images to allow testing of these algorithms. We use the following criteria for evaluation:

- True Positive (TP): the area has been identified as face correctly by the algorithm, which in fact contains a face.
- False Positive (FP): the area has been identified as face incorrectly by the algorithm, which in fact does not contain a face.
- True Negative (TN): the area has been correctly rejected by the algorithm because it does not contain the face, which in fact does not contain a face.
- False Negative (FN): the area was rejected incorrectly by the algorithm because it does not contain the face, which in fact contains a face.
- True Detection Rate (TDR): the number of faces detected correctly divide the total number of faces in the input image: TP / (TP + FN)
- Precision: the number of faces detected correctly divided the total number of detected faces: TP / (TP + FP)

Therefore, any algorithm will be more qualified whenever increasing True Positive (TP) and True Negative (TN), and decreasing False Positive (FP) and False Negative (FN).

The results of our detection algorithm experiments on 110 images from the FERET database (from the image "00060_931230_fa" to image "00146_941121_fa") are shown in table 2, table 3 and table 4. Some implementation examples in Fig .22 .

Table 2. The results of detection algorithms "face detection".

Method	TP	FP	FN	TDR	Precision
Skin Color	105	9	2	0.981	0.921
Viola-Jones	110	2	0	1	0.982
Skin Color+Viola-Jones	110	0	0	1	1

Table 3: The results of detection algorithms "eyes detection".

Method	TP	FP	FN	TDR	Precision
Skin Color	67	11	32	0.676	0.858
Viola-Jones	98	144	11	0.899	0.404
Skin Color+Viola-Jones	99	3	8	0.925	0.970

Table 4: The results of detection algorithms "glasses detection".

Method	TP	FP	FN	TDR	Precision
Skin Color	-	-	-	-	-
Viola-Jones	-	-	-	-	-
Skin Color+Viola-Jones	16	0	7 5	0.695 0.761	1

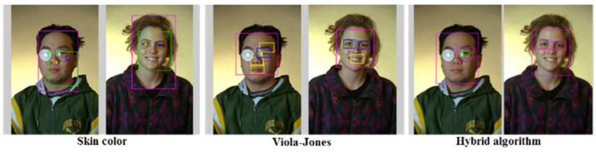

Fig. 22. Implementation examples

False Negative (FN) of glasses (the area is rejected incorrectly by the algorithm because it does not contain the glasses, which in fact contains glasses) there are seven cases, in five cases the eyes have been detected, but no glasses have been detected, and in two cases no eyes have been detected therefore the system don't apply glasses detection algorithm.

The average time consumed by detection algorithms is Skin color (0.504 s), Viola-Jones (1.158 s) and hybrid algorithm (1.056 s).

Experimental results of applying face recognition algorithms: We have evaluated detection algorithms on ORL databases which include a set of faces were taken in the laboratory, it is used in face recognition projects. We use the following criteria for evaluation:

- True Positive (TP) indicates the number of test images that have been recognized correctly.
- False Positive (FP) is the number of test images that have been recognized incorrectly.
- Precision = a number of images that have been recognized correctly/ total number of recognized images.

The results of our recognition algorithm experiments on 80 images (two images for each user) using 320 images from the ORL database related to 40 users (eight images for each user) for recognition algorithm training table 5:

Table 5. The results of recognition algorithms.

Method	TP	FP	Precision
Eigenface	75	5	0.937
Fisherface	74	6	0.925
Locality Preserving Projection (LPP)	71	9	0.887

We can notice that Eigenface algorithm is more effective than the LPP and Fisherface in the case of a relatively small group training.

Conclusion

In this study, we have designed a website depending on the user's face as an additional tool for security. When a user registers as a new user two images are captured for him, then when he tries to log in to the website one image is captured for him and used to verify identity. Also, we designed a method to determine the user's identity through the image of his face without entering any additional information.

Despite the good results we have obtained, there are some difficulties and problems facing our detection and recognition system as follows:

- There can be errors in the face recognition stage in the case of low-light or different facial expressions.
- Tests indicate that the detection stage is more accurate than recognition stage.
- Recognition process that requires the applying of multiple processing stages can cause increasing of the overall processing time, which is an important factor when we are working on the web so we must work to reduce the processing time as much as possible.
- Face Detection works only with the vertical faces which have a frontal view.

These are some points we can focus on to develop the system in the future:

- Improving face recognition algorithms to make them better is able to recognize faces in the case of low-light or different facial expressions.
- Developing the capacity of the glasses detection algorithm.
- Improvement the stages of face detection and face recognition to decrease the total processing time as much as possible.
- If the user has more than one account, then the user identification without entering any information will become less accurate, so here we can view several possibilities for the owner of the captured image instead of one possibility.

Acknowledgement

The authors would like to thank Kinh Tieu and Gunnar Ratsch for their helpful discussions on AdaBoost algorithm. In our study, we used some Matlab codes from external sources which are: LDA core written by Deng Cai, 2004, 2006. LPP core from http://www.pudn.com/, there is no mention of the author. We benefited from the previous codes in the configuration of "FaceRecognitionLPP.dll" and "FaceRecognition_Fisherface.dll" which we used on our website.

References

Bhatia, Renu, 2013. Biometrics and Face Recognition Techniques. International Journal of Advanced Research in Computer Science and Software Engineering, 3 (5), 1,5.

Niazi, Mehrnaz. and Jafari, Shahram, 2010. Hybrid Face Detection in Color Images. IJCSI International Journal of Computer Science Issues, 7 (5), 367–373.

Juneja, Mamta. and Sandhu, Parvinder. Singh, 2009. Performance Evaluation of Edge Detection Techniques for Images in Spatial Domain. International Journal of Computer Theory and Engineering, 1 (5), 1793-8201.

Hsu. Rein-Lien. and Abdel-Mottaleb. Mohamed, 2002. Face Detection in Color Images', IEEE, 24 (5).

Viola, Paul., Jones. and Mike, 2004. Robust Real-Time Face Detection. International Journal of Computer Vision, 57 (2), 137–154.

Turk, Matthew. A. and Pentland, Alex. P, 1991. Face Recognition Using Eigenfaces. IEEE Conf. On Computer Vision and Pattern Recognition, 586-591.

Belhumeur, Peter N. and Hespanha, Joao P., Kriegman, David J, 1997. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 19 (7).

He, Xiaofei. and Yan, Shuicheng., Hu, Yuxiao, 2005. Face Recognition Using Laplacianfaces. IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 27 (3).

Guo. Guo-Dong. and Zhang. Hong-Jiang, 2001. Boosting for Fast Face Recognition. 5F, Beijing Sigma Center, 96-100.

Khan, Ijaz. and Abdullah, Hadi., Shamian, Mohd, 2012. EFFICIENT EYES AND MOUTH DETECTION ALGORITHM USING COMBINATION OF VIOLA JONES AND SKIN COLOR PIXEL DETECTION. International Journal of Engineering and Applied Sciences, 3 (4).

Kirby, M. and Sirovich, L, 1990. Application of the Karhunen-Loève Procedure for the Characterization of Human Faces. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12 (1), 103-108.

Taigman, Yaniv. and Yang, Ming., Ranzato, Marc’Aurelio., Wolf, Lior, 2014. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. IEEE Computer Vision and Pattern Recognition (CVPR).