World Applied Sciences Journal 29 (11): 1427-1433, 2014
ISSN 1818-4952

DOI: 10.5829/idosi.wasj.2014.29.11.221

# An Efficient Face Detection in Multi-Face Images Using Support Vector Machine

[1]Sanjeev Kumar, [2]Asutosh Kar and [1]Mahesh Chandra

[1]Department of ECE,, BIT, Mesra, Ranchi, India
[2]Department of E&TC, IIIT, Bhubaneswar, India

**Abstract:** This paper presents a robust algorithm for face detection in still gray level images. The structure and characteristics of the human nose is used to find possible face regions. Line detection filters are employed for this purpose; furthermore from among the several candidates detected in an image, a trained Support Vector Machine is used to correctly identify a human face. The proposed method is robust to deal with illumination problems. The accuracy of this method is higher than 90%, if tested for less than 10 faces in a simple background with adequate illumination. Owing to its simplicity it can be transferred from a PC to embedded device, making it a potential for customized and miniature systems. There are many unwanted elements in a picture which are commonly known as noise and should be removed from an image for further processing. Median filter is normally used to reduce noise in an image and for preserving useful details in the image. Adaptive filtering is more selective which helps for preserving edges and other high frequency parts of an image. Adaptive median filter is applied on the noisy image and again passed through SVM.

**Key words:** Face Detection · Gabor filters · Support Vector Machine · Adaptive Median Filter

## INTRODUCTION

Face detection is one of the most researched regions of image processing. It is a real difficult task to build a system which can identify faces just as a human being can. Though there has been significant development in this aspect, yet there is no perfect solution for all cases. It has become a part and parcel of our life as it finds its uses in areas like surveillance system, digital monitoring, PC, camera, social networking, cell phones and the like. Simply put, a face detection method can be defined as follows [1]. Given an image I, find all occurrences of faces and the extent of each face in I. This definition implies that some form of discrimination must be made between faces and all other objects. Though it may seem easy at first it poses its own set of difficulties. Variations in lighting conditions can make face images appear substantially different. Presence of additional features such as beards, mustaches and glasses can augment the global structure of the face such as the jaw line and mask local features such as corners of the mouth. Additionally, the large amount of intra-class variation among all faces makes it difficult for a reliable face detection mechanism.

Over the past decade, many approaches for improving the performance of human face detection have been proposed, which are categorized as the follows:

- Knowledge-based method: This method is aimed at finding invariant features of a face within a complex environment, thereby localizing the position of the face. Relationships among the features helpfully determine whether a human face appears in an image or not [2].
- Feature invariant approaches: Invariant features, unresponsive to different positions, brightness and viewpoints, are utilized in this approach to detect human faces. A statistical model is usually built up for describing the relations among face features and the presence of the detected faces. Such face features, for instance, are Facial Features [3], Texture [4] and Skin Color [5].
- Template matching method: A template cohering with human face features is used to perform a pattern-matching operation based on the template and an input image. Shape template [6] and Active Shape Model [7] are common examples of this method.

**Corresponding Author:** Sanjeev Kumar, ECE, Department, BIT, Mesra, Ranchi, India.
Tel: +09204137947.

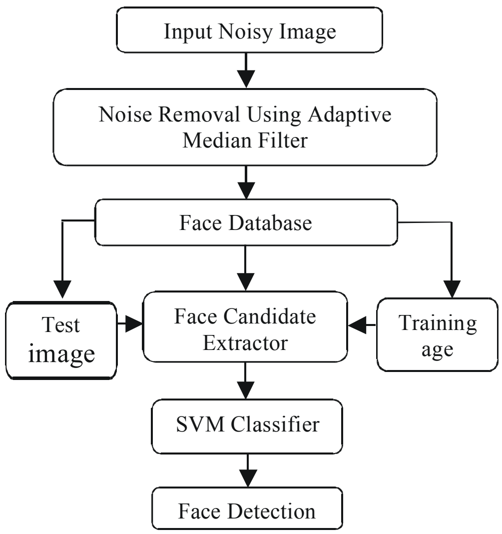


Fig. 1: Block diagram of the proposed method

- Appearance-based method: This method, such as Eigen face [8], Neural Network [9] and Hidden Markov Model [10], employs a series of face images to train and establish a face model for the face detection.

In general, method (ii), (iii) and (iv) are more complex than method (i); yet the more features are used in method (i), the more complicated it is.

Most real-time systems take the feature-based approach, combining it with various kinds of fast face candidate extraction methods. These algorithms can be put into a two-stage framework. In the first stage, regions that may contain a face are marked. This stage focuses attention to face candidates. In the second stage, the possible regions, or, face candidates, are sent to a “face verifier”, which will decide whether the candidates are real faces [11]. Different methods put emphasis on different stages. An extreme is that if the face verifier is powerful enough to discriminate between face patterns and non-face patterns in nearly all cases, the candidates selection stage may be omitted [9, 12]. In this case, the algorithm can move through the image from left to right and from top to bottom, treat each sub-region in the image as a face candidate. On the contrary, if in the first stage most non-face regions are eliminated and all face regions are selected, the face verifier might be dramatically simplified or even omitted. Noise in an image is certainly playing a vital role in decreasing the details of that image. So that should be eradicated to get back the details and to help in further processing on that image. There are several types of noise like Gaussian, Rayleigh, Gamma, Uniform, Exponential, Salt and Pepper, etc. Salt and pepper noisy can be visually identified as it looks like salt and pepper. Usually median filter [13] is the best solution for salt and pepper noise but when the degree of noise elements increases it becomes difficult for the normal median filter technique to deal with.

This paper proposes a combinational feature based approach which uses the structure information about the nose region as a robust cue to identify possible face candidates in an image, after which a SVM classifier is trained to identify the face regions among the candidates to successfully detect them. The noise in the image is filtered by the adaptive median filtering. The framework of the entire proposed technique is shown in Fig. 1.

**Discussion of Existing Methodology:** There are many existing algorithm on which SVM is working. They are like Integral Image and Rectangle Features. Initially Integral Image was proposed at Viola *et al*. [13]. For the original image $u(p,q)$, the integral image is defined as follow:

$$ii(p.q) = \sum_{p' \le p} . \sum_{q' \le q} u(p', q') \qquad (1)$$

Using the following pair of recurrence:

$$u(p,q) = u(p,q-1) + ii(p,q) \qquad (2)$$

$$ii(p,q) = ii(p-1,q) + u(p,q \qquad (3)$$

(Where $u(p,q)$ is the cumulative row sum (p,-1) = 0 and $ii(-1,q) = 0$) the integral image can be computed in one pass over the original image. Using the integral image any rectangular sum can be represented in four array references as shown in Fig. 2. The sum of the pixels in rectangle A is concentrated at position 1 in the integral image. Similarly the value at location 2 is $A+B$, at location 3 is $A+C$ and at location 4 is $A+B+C+D$. The sum within $D$ can be calculated as 4+1 - (2+3).

In Viola and Jones’s work, two rectangle features selected are fruitful. One feature evaluates the difference in intensity between the region of the eyes and a region across the upper cheeks. The feature realized on the observation that the eye region is often darker than the cheeks. The other feature studies the intensities in the eye regions to the intensity across the bridge of the nose. Sawettanusorn *et al*. have developed the SSR filter based on the hints from these two rectangles [14].Three rectangle features are used to make an eye for face candidates, which are based on the idea mentioned above. A rectangular window is scanned on the input image to

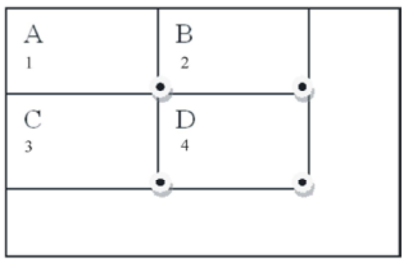


Fig. 2: The sum of pixels within rectangle D can be computed with for array references

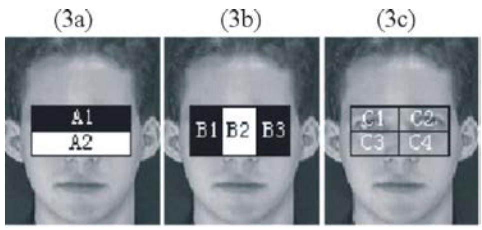


Fig. 3: Rectangle Features

look for face candidates by using three rectangle features. Fig. 3 shows the rectangle features. The parameter $H_{A1}$, $H_{A2}$, $H_{B1}$, $H_{B2}$, $H_{B3}$, $H_{C1}$, $H_{C2}$, $H_{C3}$, $H_{C4}$ indicate the sum of pixel value of region $A1$, $A2$, $B1$, $B2$, $C1$, $C2$, $C3$ and $C4$ respectively. Rectangle features are computed rapidly by using integral image. The rectangle features are used to identify the eye-pair-like regions based on three characteristics of gray distribution and geometry in face region.

- The eye region (containing eyes and eyebrows) $A1$ is relatively darker than the cheekbone area containing nose) $A2$, as shown in Fig. 3 (a), where

$$H_{A1} < H_{A2} \tag{4}$$

- The between-eye area (containing nose) $B2$ is brighter than the left area $B1$ and right eye area $B3$, as shown in Fig.3 (b), where

$$H_{B2} > H_{B1} \tag{5}$$

$$H_{B2} > H_{B3} \tag{6}$$

- Since symmetry is one of the criteria used for classification, this technique is good for faces that are frontal or nearly frontal with minimal rotation effects. Therefore, left eye region $C1$and right eye region $C2$ is symmetrical, also, the left cheekbone region $C3$ and right cheekbone region $C4$ is symmetrical, as shown in Fig.3 (c), where

$$th1 < H_{C1} / H_{C2} < th2 \tag{7}$$

$$th1 < H_{C3} / H_{C4} < th2 \tag{8}$$

Computing the three rectangle features in each window scanned on the image. When expression (4), (5), (6), (7) and (8) are all satisfied simultaneously, the region is considered to be an eye-pair-like region.

## MATERIALS AND METHODS

Since the discussed algorithm primarily focuses on grayscale images, it becomes imperative to include a preprocessing stage for "improving" images. This is achieved using a modified Homomorphic filtering technique for obtaining better illumination compensation [15] and adaptive median filtering. Adaptive Median Filter is a better technique to reduce the high density noise elements i.e. if the density of salt and pepper noise is more than 2 then the normal median filter fails to get rid of that noise. Once the image is rescued from the noise, the image is sent to the SVM trainer for the purposes of identification.

Adaptive median filter changes size of $S_{xy}$ (the size of the neighborhood) during operation. Notation used for describing the algorithm of adaptive median filter.

$Z_{min}$ = Minimum gray level value in $S_{xy}$
$Z_{max}$ = Maximum gray level value in $S_{xy}$
$Z_{med}$ = Median of gray levels in $S_{xy}$
$Z_{xy}$ = Gray level at coordinates (x, y)
$S_{max}$ = Maximum allowed size of $S_{xy}$

**Algorithm:**

Level

A: $A1 = Z_{med} - Z_{min}$
$A2 = Z_{med} - Z_{max}$

if A1 > 0 AND A2 < 0, go to level B
else increase the window size
if window size < $S_{max}$, repeat level A
else output $Z_{xy}$

Level

B: $B1 = Z_{xy} - Z_{min}$
$B2 = Z_{xy} - Z_{max}$

if B1 > 0 AND B2 < 0, output $Z_{xy}$
else output $Z_{med}$

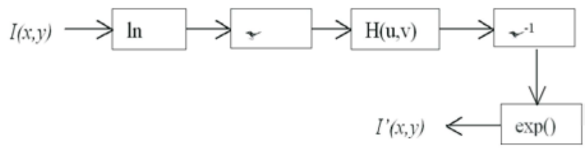


Fig. 4: Block diagram of a basic Homomorphic Filter (HF)

The basic process of Homomorphic filter is represented in Fig. 4. The Homomorphic filter utilizes the concept of Illuminance-Reflection model, which states that an image can be considered as a two-dimensional function of the form $I(x,y)$, whose value at spatial coordinates $(x,y)$ is a positive scalar quantity and the physical meaning of this is determined by the source of the image. Since this paper focuses on grayscale images, it can be said when an image is generated from a physical process, its values are proportional to energy radiated by a physical source. Alternately, an image can be thought of as an array of measured light intensities and is a function of the amount of light reflected of the objects in the scene.

This intensity is a product of illumination (the amount of source illumination incident on the scene being viewed) and reflectance (the amount of illumination reflected by the objects in the scene).If we denote illumination as $L(x,y)$ and reflectance as $R(x,y)$ , then an image $I(x,y)$ can be expressed as,

$$I(x,y) = L(x,y).R(x,y) \tag{9}$$

Thus it can be noted that while illumination results from the lighting conditions present when the image is captured and can change when lighting conditions change, but reflectance results from the way the objects in the image reflect light and is determined by the intrinsic properties of the object itself, which can be safely assumed to remain same in this theoretical analysis. Further, it is noted that illumination varies slowly in space (slow spatial changes fllow spatial frequency) while reflectance can change abruptly (high spatial frequencies). So the problem of eliminating apparent changes in facial appearance with the change in lighting conditions can be done by enhancing the reflectance while reducing the contribution of illumination. This is achieved by separating the two components from (9) and then treating` the resulting image in frequency domain with a high pass filter.

For better enhancement the regular approach was replaced with a modified technique employing separate filters for separate regions to give a better result. The modified approach is represented in Fig. 5. In this approach the large image is segmented in two halves, both horizontally and vertically and each half is separately processed and combined to obtain a better output than a single Homomorphic filter. This processed image is then used for detecting faces. This image is now referred to as the normalized image. Once the image is normalized, it is sent to the SVM trainer for identification purpose. For more efficient performance an optimal Gabor filter is used to mark certain regions of the image where a face structure is likely to be present. These regions are called as face candidates. Moreover a generic SVM is trained using the test images. To train the classifier, two set of images are needed. One set contains an image or scene that does not contain the object, in this case a facial feature, which is going to be detected. This set of images is referred to as the negative images. The other set of images, the positive images, contain one or more instances of the object. For training facial features at least 50 negative images with at least a mega-pixel resolution were used for training. These images consisted of everyday objects, like paperclips, natural scenery, photographs of forests and mountains. Now this trained SVM is given the face candidate locations to identify the actual face structures among them and on successful identification it is marked with green boxes.

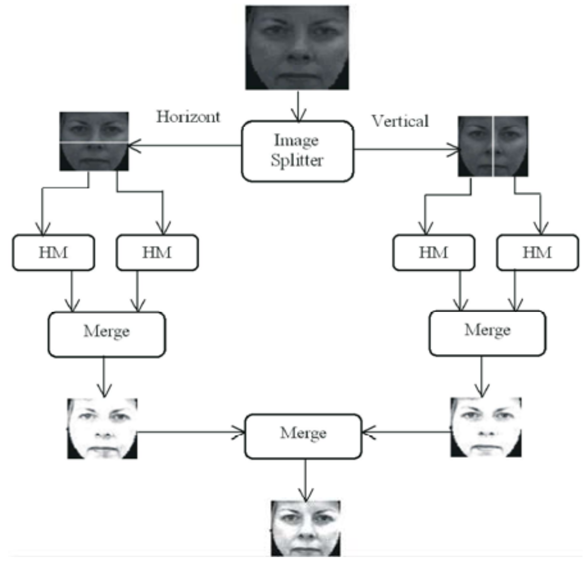


Fig. 5: Block diagram for illumination compensation using modified Homomorphic Filter (HF)

The Algorithm of this method is presented in a step wise manner in Table 1. It can be noted that there are two steps independent of each other, identification of possible face regions (Step 2) and training of the SVM classifier (Step3). For a better time efficiency these two processes can be executed in parallel to give a better performance.

Table 1: Algorithm for Detection

| | |
|---|---|
| 1. | Normalization of the image using modified Homomorphic filter |
| 2. | Identification of possible Face Regions from the normalized image |
| 3. | Training of the SVM classifier |
| 4. | Identifying the actual faces from the possible face regions |
| 5. | Marking the actual faces with green boxes for successful identification |

## RESULTS

In order to verify the functioning of the discussed algorithm, 130 images collected randomly from our surrounding including 70 face and 60 non face images along with some images from "FERET face database", were used as training images for the SVM classifier. The method was implemented in MATLAB and simulations carried out in simple and complex background to obtain the effectiveness of the program.

For a real world evaluation, the following image from our surrounding was taken as input. The test image is then sent to the adaptive median filter for preprocessing. The output filtered image is shown in Fig. 7. The image is then sent to the modified Homomorphic filter for preprocessing. The output normalized image is shown in Fig. 8 and is used for detecting faces. After normalization, the SVM classifier is trained in face and non-face images. These images are obtained from the FERET database. This trained SVM is now given the normalized image for detection purposes. The SVM is now produced the filtered image for detection purposes. The face detection in noisy image is shown in Fig. 9(a) and that in the filtered image is shown in Fig. 9(b).

Since the performance of the classifier cannot be decided based on only a single test image, few more variety of images were included for obtaining a more comprehensive result under different situations. These images are publicly available images and they were grouped as per the scenario in which they were taken. A particular face image with different intensity levels was used to determine the robustness of this approach. The results of these are shown in Fig. 10(a), 10(b), 11(a), 11(b), 12(a) and 12(b).

A group picture with only face part has been taken as input and face detection was carried out on it. So this result is shown for noisy and filter images with the faces detected in green boxes in Fig. 9(a) and 9(b).

A comparative performance study is now made to analyze the robustness of the system which is depicted in Table 2 is shown below. The overall hit ratio is found as 95 while the miss ratio is 2.925. Another comparison can be made on the detection rates based on the feature being used for identification. Table 2 show the comparison based on the feature and their respective success ratio.

Fig. 6: Test Image Input from surrounding

Fig. 7: Image after Filtration

Fig. 8: Image after Normalization

Fig. 9(a): Face detection in noisy image

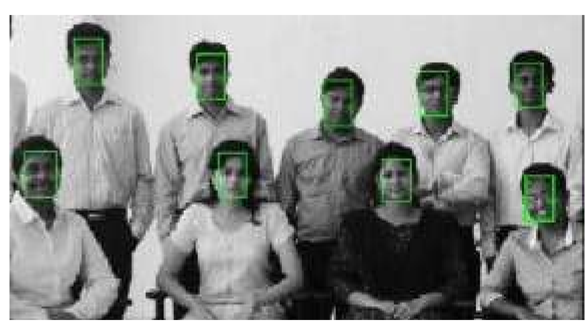

Fig. 9(b): Face detected in filtered and normalized

Table 2: Results of Face detection under different conditions

| Figure No | Image Size | No. of Faces | Type of image | Face Detected | Missed Detection | False Hits | Hit Ratio | Miss Ratio |
|---|---|---|---|---|---|---|---|---|
| 9(a) | 353X182 | 9 | NOISY | 2 | 7 | 0 | 22.22% | 77.78% |
| 9(b) | 353X182 | 9 | FILTERED | 9 | 0 | 0 | 100% | 0.00% |
| 10(a) | 125X91 | 15 | NOISY | 3 | 12 | 0 | 20% | 80% |
| 10(b) | 125X91 | 15 | FILTERED | 12 | 3 | 0 | 80% | 20% |
| 11(a) | 150X66 | 7 | NOISY | 1 | 6 | 0 | 14.28% | 85.72% |
| 11(b) | 150X66 | 7 | FILTERED | 7 | 0 | 0 | 100% | 0.00% |
| 12(a) | 150X151 | 8 | NOISY | 0 | 0 | 1 | 0 | 12.5% |
| 12(b) | 150X66 | 7 | FILTERED | 6 | 2 | 0 | 75% | 25% |

Table 3: Results of Face detection for different facial features

| Facial Feature | Identification Rate |
|---|---|
| Mouth | 67% |
| Eyes | 93% |
| Nose | 95% |

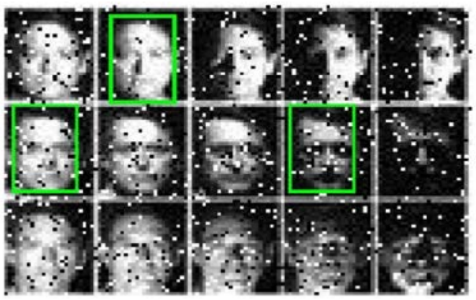

Fig. 10(a): No. of Faces in a single noisy image with same face but varying levels of illumination

Fig. 10(b): Face detected in filtered and normalized

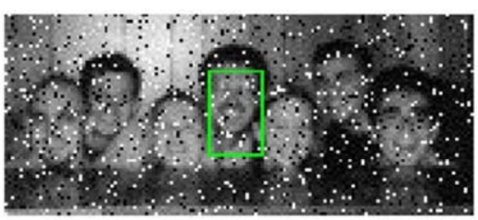

Fig. 11(a): Face detection in sample noisy image with faces only

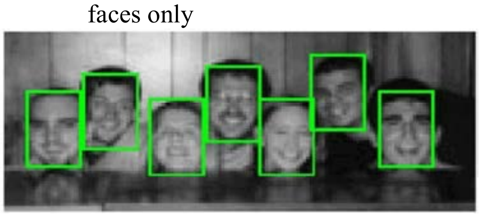

Fig. 11(b): Face detected in filtered and normalized

Fig. 12(a): Results of Face detection in different situations

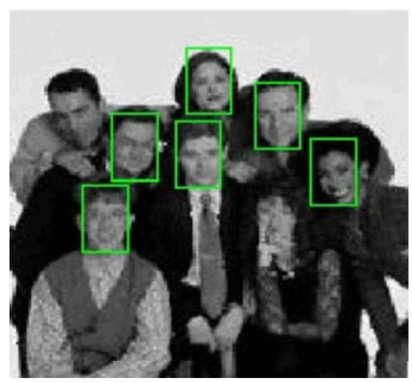

Fig. 12(b): Face detected in filtered and normalized

This method provides an increment in identification rate over existing techniques and due to its low memory and processing requirement it has potential for being employed in embedded identification systems.

Here in Fig.12(a) there is a false detection which is known as non-face. So in noisy image there is a chance of false detection but in filtered and normalized image as depicted in Fig. 12(b) the problem has gone. Now the numbers of faces detected in both noisy and filtered images are given in Table 2 for a comparative study.

Another comparison can be now made on the detection rates based on the feature being used for identification. Table 3 show the comparison based on the feature and their respective success ratio.

This method provides an increment in this identification rate over existing techniques employed and due to its low memory and processing requirement it has potential for being employed in embedded identification systems.

## CONCLUSION AND FUTURE WORK

Face detection is an important aspect for various fields of study such as face recognition, expression detection, video monitoring, status authentication and others for which till date it remains an important research field. In this paper it has been shown that the efficiency of using the nose structure as an identifying feature for face detection improves the success rate. The drawback of this paper lies in the fact of its dependence on availability of frontal face images. Future scope lies in improving the identification ratio, either by combining several feature extraction algorithms [16] or combining several classifiers [17] or combining the above two methods [18]. Improvement in this algorithm in high noisy environment can be made by removing both impulse noise and High Density Additive White Gaussian Noise. Impulse noise and High Density Additive White Gaussian Noise can be removed by the algorithms discussed in paper [19] and [20]. Improvements can also be made in developing methods to identify faces from angles or even when the face remains obscured. Future research will focus on looking for more effective approaches for detecting faces more accurately.

## REFERENCES


1. Yang, M., D. Kriegman and N. Ahuja, 2002, "Detecting faces in images: a survey," IEEE Trans. Pattern Anal. Mach. Intell., 24(1): 34-58.
2. Jerome M. Shapiro, 1993. "Embedded Image Coding Using Zero tress of Wavelet Coefficients", IEEE Transaction on Signal Processing, 41(12): 3445-3462.
3. Leung, T.K. ,M.C. Burl and P. Perona, 1995. "Finding Face in Cluttered Scenes Using Random Labeled Graph Matching", Proc. Fifth IEEE Int'l Conf. Computer Vision,, pp: 637-644.
4. Dai, Y. and Y. Nakano, 1996. "Face-Texture Model Based on SGLD and Its Application in Face Detection in a Color Scene", Pattern Recognition, 29(6): 1007-1017.
5. Yang, J. and A. Waibel, 1996. "A Real-Time Face Tracker", Proc. Third Workshop Applications of Computer Vision, pp: 142-147.
6. Craw, D. Tock and A. Bennett, 1992. "Finding Face Features," Proc. Second European Conf. Computer Vision, pp: 92-96.
7. Lanitis, C.J. Taylor and T.F. Cootes, 1995. "An Automatic Face Identification System Using Flexible Appearance Models", Image and Vision Computing, 13(5): 393-401.
8. Turk, M. and A. Pentland, 1991. "Eigen face for Recognition", J. Cognitive Neuroscience, 3: 71-86.
9. Rowley, H., S. Baluja and T. Kanade, 1998. "Neural Network-Based Face Detection", IEEE Trans. Pattern Analysis and Machine Intelligence, 20(1): 203-208.
10. Rajagopalan, K. Kumar, J. Karlekar, R. Manivasakan, M. Patil, U. Desai, P. Poonacha and S. Chaudhuri, 1998 "Finding Faces in Photographs", Proc. Sixth IEEE Intl Conf. Computer Vision, pp: 640-645.
11. Wu, J.X. and Z.H. Zhou, 2003. "Efficient face candidates selector for face detection," Pattern Recognition, 36(5): 1175 - 1186.
12. Osuna, E., R. Freund and F. Girosi, 1997. "Training support vector machines: an application to face detection," In Proc. IEEE Conf. Computer Vision Pattern Recognition, pp: 130-136.
13. Viola, P. and M. Jones, 2001. "Rapid object detection using a boosted cascade of simple features," Proc. of IEEE Conf. CVRP, pp: 511-518.
14. Sawettanusorn, O., Y. Senda and S. Kawato, 2004. "Detection of face representative using newly proposed filter," Journal of Signal Processing, 8(2): 137-145.
15. Delac, K., M. Grgic and T. Kos, 2006. "Sub-Image Homomorphic filtering for Improving Facial Identification under Difficult Illumination Conditions", International Conference on Systems, Signals and Image Processing, pp: 95-98.
16. Nain, N., P. Gour, N. Agarwal, R.P. Talawar and S. Chandra, 2008. " Face recognition using PCA and LDA with singular value decomposition using 2DLDA", In Proceedings of the World Congress on Engineering, 1: 1-7.
17. Yang, T. and V. Kecman, 2010. "Face recognition with adaptive local hyper-plane algorithm", Pattern Analysis and Applications, 1: 79-83.
18. Tulyakov, S., 2008. "Review of classifier combination methods", Studies in Computational Intelligence (SCI), pp: 361-386.
19. Shyam Lal, Sanjeev Kumar & Mahesh Chandra, 2012. "Removal of High Density Salt & Pepper Noise Through Super Mean Filter for Natural Images" International Journal on Computer Science Issue (IJCSI), 9(3): 303-309.
20. Shyam Lal, Mahesh Chandra & Gopal Krishna Upadhyay, 2012. "Efficient Algorithm for Removal of High Density Additive White Gaussian Noise", World Applied Sciences Journal, 16(12): 1722-1733.