AN ADVANCED SCALE INVARIANT FEATURE TRANSFORM ALGORITHM FOR FACE RECOGNITION

Mohammad Mohsen Ahmadinejad*
Department of Computer ScienceUniversity of Kerala , India
Email:Mohsen.ahmadi64@yahoo.com

Elizabeth Sherly
Indian institute of Information Technology and Management-Kerala,Kerala, India
Email:sherly@iiitmk.ac.in

Abstract - In computer vision, Scale-invariant feature transform (SIFT) algorithm is widely used to describe and detect local features in images due to its excellent performance. But for face recognition, the implementation of SIFT was complicated because of detecting false key-points in the face image due to irrelevant portions like hair style and other background details. This paper proposes an algorithm for face recognition to improve recognition accuracy by selecting relevant SIFT key-points only that by rejecting false key points. In the new proposed Haar-Cascade SIFT algorithm (HC-SIFT), the accuracy in face recognition has been increased from 52.6% to 75.1% from SIFT to HC- SIFT algorithm.

Keywords:SIFT algorithm, Haar-Cascade, HC-SIFT, face recognition system, keypoint extraction.

1. Introduction

Developing face recognition system using local image features which are unaffected by pose variations and partial occlusions is a major field of image processing. The features will be invariant in illumination, image scaling, translation, affine transformation etc. Key-points allow local geometric deformation by presenting blur image gradient in multiple orientation and multiple scales. But, the features must also be capable enough to identify specific objects among many alternatives and should spot the interclass variance. The most unwavering problem in computer aided object recognition is the lack of success in finding such relevant image features.

In this paper an improved SIFT algorithm is proposed for face feature generation and classification combining SIFT with Haar cascade Transform for face recognition. This approach transforms an image into a large collection of local features. SIFT keys derived from the face area of an image are used in a nearest-neighbor approach for classifying the faces. SIFT key points are extracted from every face database, and while testing a face, the test image features are extracted and compare with each of the trained images and hence the most matching result is obtained.

2. Related work

Reliable and accurate matching of image features is a basic problem in computer vision applications. Scale-Invariant Feature Transform (SIFT) was published by Lowe in (1999) and upgradFed in (2004) which was used to describe and detect local image features. SIFT is an outstanding feature descriptor, because it is invariant to linear scaling, orientation changes, and partially invariant to affine distortion and illumination changes[1][2].

Abdel-Hakim et al. In there paper(2006) have proposes a novel colored local invariant feature descriptor to solve problem of many objects which, can be misclassified if their color contents are ignored. They used proposed approach which, builds the SIFT descriptors in a color invariant space instead of using the gray space to represent the input image. Their evaluation results shows which proposed method (CSIFT) was more robust than the conventional SIFT with respect to color and photometrical variationsImage alignment becomes worse in the object recognition scenario, where the goal is to align different occurrences of the same object class[3].

Complex object representations (2006) have been developed to manage with the variations of object shapes and appearances. However, these systems still require objects to be salient, alike, with restricted amount of background clutter[4].

Cheung *et al.* they presents(2007) a fully automated multimodal medical image matching technique which, their method have extended the concepts used in the computer vision SIFT technique for extracting and matching distinctive scale invariant features in 2D scalar images to scalar images of arbitrary dimensionality. The extension have involved for us hyperspherical coordinates for gradients and multidimensional histograms to create the feature vectors. Then the features have used successfully to determine accurate feature point correspondence between pairs of medical images (3D) and dynamic volumetric data (3D+time)[5].

Juan et al. In their paper(2009) compaired three robust feature detection methods: Scale Invariant FeatureTransform (SIFT), Speeded Up Robust Features (SURF) and Principal Component Analysis (PCA)–SIFT. They used KNN (K-Nearest Neighbor) and Random Sample Consensus (RANSAC) to the three methods in order to analyze the results of the methods in recognition. They used KNN to find the matches, and RANSAC has used to reject inconsistent matches from which the inliers can take as correct matches. The performance of methods have compared for scale changes, rotation, blur, illumination changes and affine transformations. Their experiments shows SIFT presents its stability in most situations although it's slow,SURF is the fastest one with good performance as the same as SIFT. PCA-SIFT[6].

Li, Qiaoliang, et al. in their proposed method(2009) improved the match performance of images which experimental results for multidate, multispectral, and multisensor remote images have indicated that the proposed method improves the match performance compared to intensity- and SIFT-based methods in terms of correct-match rate and aligning accuracy[7].

Shaikh et al.in their paper have proposed a methodology for image stitching process which, have combined various feature detection and extraction algorithms.. They summarized 3 robust feature detection and extraction algorithms, namely SIFT, SURF and MSER which, combined set of key points has used for image transformation. Multiple feature extraction techniques can be used for image stitching which can build seamless panorama image according of the proposed method[8].

In this paper we have advocated SIFT algorithm initially and then proposed modified SIFT algorithm to improve the quality and to extract relevant information for recognizing the original image. Though SIFT algorithm is invariant to feature attributes, but depending on the image and its application, improvement in SIFT algorithm is required. Here Haar-cascade classifier in face models is used for extracting the face area from the image.

3. SIFT ALGORITHM

The various steps involved in SIFT algorithm is constructing a Scale space to find out the features, applying Gaussian to remove the noise, and to find the key points and remove unwanted key points. Finally features are extracted using Brute Force matcher algorithm. Various steps in the algorithm are as follows:

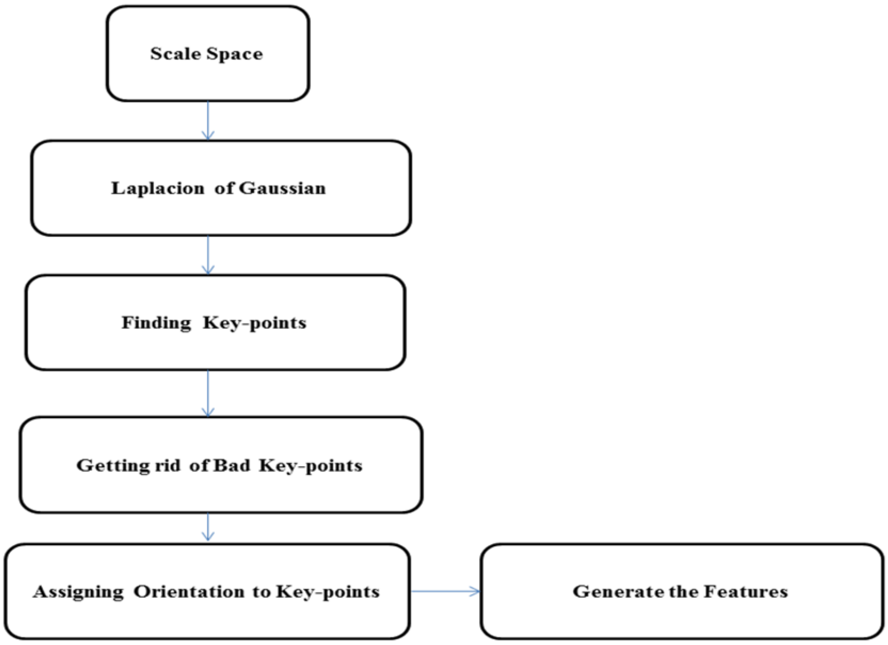


Fig. 1. This diagram shows a SIFT algorithm step

3.1. Constructing a Scale Space

Constructing a scale space is the initial step for finding SIFT features. Here we have taken the image and generated progressively blurred copies. Then the original image is resized to half of its size and again blurred images, and the process continues. The number of octaves, octave number of blurred images in each octave, and the amount of blur depends on size of the image. Figure 1 shows an example for blurred images in the first octave.

Mathematically, the blur can be obtained by Gaussian operator and are shown in equations (1, 2).

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{1}$$

$$L(x, y, \sigma) = G(x, y, \sigma) * L(x, y) \tag{2}$$

L = Blurred image

G = the Gaussian blur operator

I = Image to get blurred

x, y = coordinates

σ = the amount of blur

Fig. 2. Scale space in octaves

σ is the amount of blur in first image. Then, the amount of blur in the next image will be k*σ. Here ' k ' is selected as$\sqrt{2}$. The value of σ in different octaves are given below

Octave ↓ / Scale →				
0.707107	1.00000	1.414214	2.00000	2.828427
1.414214	2.00000	2.828427	4.00000	5.656854
2.828427	4.00000	5.656854	8.00000	11.313708
5.656854	8.00000	11.313708	16.00000	22.627417

Fig. 3. σ in each octave

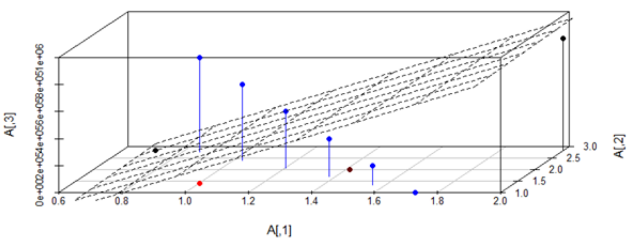


Fig. 4. Schematic representation of σ in each octave

3.2. Laplacion of Gaussian(LOG)

While finding Laplacian of Gaussian, an image can be blurred a little and a second order derivative can be calculated on it. This locates corners and edges on the image. These corners and edges are effective for finding key-points. But the second order derivative is extremely sensitive to noise. The blur smoothes it out the noise and stabilizes the second order derivative.

Calculating all those second order derivatives is computationally intensive. In order to solve the issue, Laplacian of Gaussian is generated between two consecutive scale or difference of the Gaussian using scale space.

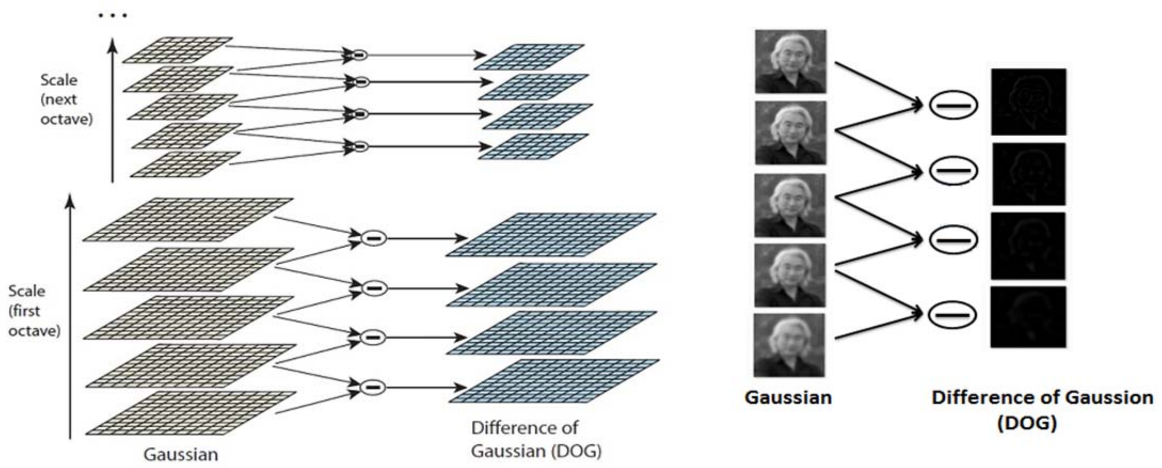


Fig. 5. DoG extraction from the octaves

The Difference of Gaussian (DoG) is approximately equal to Laplacian of Gaussian

3.3. Finding Key-points

Finding key points is a two part process:

(1) locate of maxima/minima in DoG

(2) find sub-pixel maxima/minima

(1)Locate maxima/minima

In this step, we iterate through each pixel and check each pixel with all neighbors in above and below image and also with current image which have found in DOG.

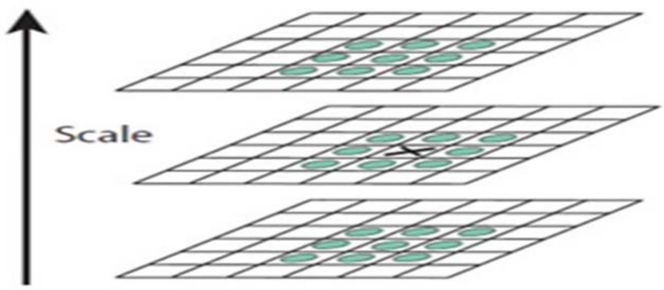


Fig. 6. checking local maxima/ minima

Total of 26 comparisons have been performed for finding maxima and minima. Here the lowermost and topmost scale key-point is not detected, so that the number of comparison is reduced.

(2) Find Sub pixel maxima/minima

Using the available pixel data, sub-pixel values are generated. This is done by the Taylor expansion around the approximate key-point expansion of the image.

The formula is shown in equation (3).

$$D(x) = D+\frac{\partial D^r}{\partial X} X + \frac{1}{2} X^r \frac{\partial^2 D}{\partial X^2} X \qquad (3)$$

These sub-pixel values increase chances of matching and stability of the algorithm.

3.4. Getting rid of Bad Key-points

Finding bad key points is also a two part process.

(1) Removing low contrast features

(2) Removing edges

Removing low contrast features

If the magnitude of the intensity (i.e., without sign) at the current pixel in the DoG image (that is being checked for minima/maxima) is less than a specific value, it will get rejected.

Removing edges:

In this step the key-point two gradients are calculated. It is considered that as flat region if both the gradients are small and if one is small and other large, considered as edge region. If both gradients are large, it will be considered as the key-point.

3.5 Assigning Orientation to Key-points

In this step gradient directions and magnitudes are collected around each key-point. Then the most prominent orientation in that region is figured out which assigns this orientation(s) to the key-point.

The gradient and magnitude is calculated by equations (4, 5).

$$\mathrm{m(x,y)} = \sqrt{(L(x+1,y) - \mathrm{L}(x-1+y))^2 + \left(\mathrm{L(x,y-1)}\right) - \mathrm{L}(x-1+y))^2} \quad (4)$$

$$(x,y){=}tan^{-1}((L(x,y+1) - \mathrm{L(x,y-1)})/(\mathrm{L(x+1,y)} - L(x-1,y))) \quad (5)$$

The magnitude and orientation are calculated for all pixels which are around the key-point. A histogram is generated with 360 degrees of orientation are divided into 36 bins (each bin possesses 10 degrees). If the gradient direction at a certain point (in the “orientation collection region”) is 17.9 degrees, then it will go into the 10-19 degree bin. The amount that added to the bin is proportional to the magnitude of the gradient at that point.

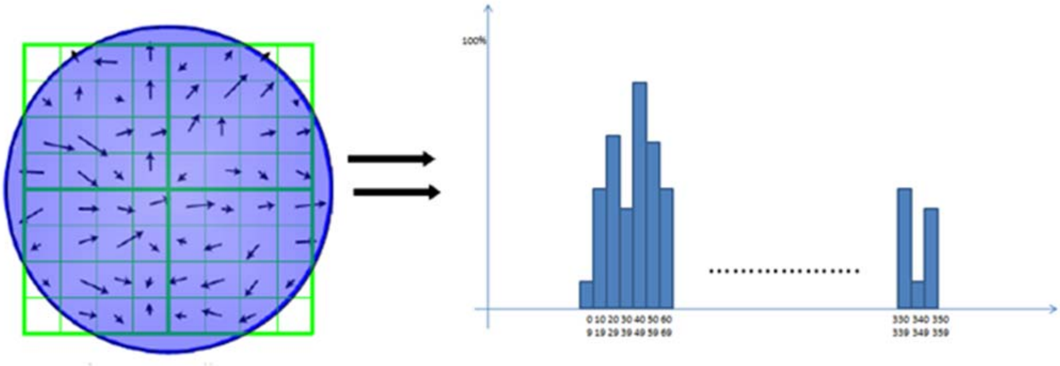


Fig. 7. The orientation assignment

3.6. Generate the Features

In the last step, we have taken a 16×16 window of “in-between” pixels around the key-point. Then split that window into sixteen 4×4 windows. From each 4×4 window histogram of 8 bins is generated, which each bin is corresponding to 0-44 degrees, 45-89 degrees. Gradient orientations from the 4×4 are put into these bins. This is done for all 4×4 blocks.

Key-points detected by this algorithm are shown in Figure (8)

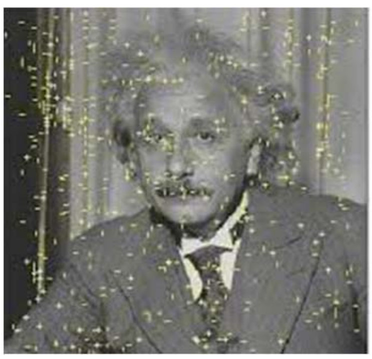


Fig. 8. Features Extracted (No. of key points detected is(8)

4. Proposed algorithm (HC-SIFT)

While using SIFT algorithm for face recognition, results often went to wrong detection due to the extraction of false SIFT key points from areas other than face as shown in Figure 12. These false key points affect the training process that resulted into poor performance.

The proposed method HC-SIFT algorithm uses pre-filtering method using Haar Cascade. In this method, only face models are extracted from an image, and ignore other irrelevant details from the image. Only relevant key-point of eyes, lips, mouth etc has been taken, so that the number of key-point will be less, and the detection become more accurate.

Here Haar-Cascade classifier is applied to the face area to extract the relevant Difference of Gaussian (DoG) key points from the extracted face area from SIFT algorithm. The face extraction helps to avoid wrong key-points due to the background objects. The process are shown in Figure (8,9).

Fig. 9. Detecting Feature points from the face area

Fig. 10. (a) Fig. 10. (b)

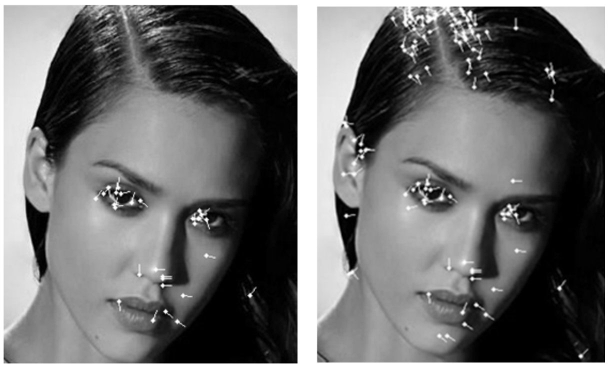


Fig. 10. (a): Pre-filtering method using Haar-cascade and face models. (No of key point: 28), Fig. 10. (b)without pre-filtering (No of key point: 118)

In Figure 10(a) we got 28 key-points and this 28 key-point provides better features for detecting the face. In Figure 10(b) we got 118 keypoint but most of them are not relevent for detecting face and might create false responses. The accuracy of detecting face is not solely depends on the number of key-point but rely on the relevant key-points.

5. HAAR-CASCADE classifier

Haar-like features (2006)rather than using the intensity values of a pixel, use the variation in contrast values between nearby rectangular groups of pixels. The contrast variances between the pixel sets are used to obtain relative light and dark areas. Haar features can easily be scaled by increasing or decreasing the size of the pixel group being examined.

The simple rectangular features of an image are calculated using an intermediate representation of an image, called the integral image . The integral image is an array containing the sum of the pixels' intensity values located directly to the left of a pixel and directly above the pixel at location (x, y) inclusive. So if A[x,y] is the original image and AI[x,y] is the integral image then the integral image is computed as shown in Equation (6)

$$AI[x,y] = \sum_{x'\leq x, y'\leq y} A(x',y') \tag{6}$$

The features rotated by forty-five degrees, like the line feature as introduced by Lienhart and Maydt, require another intermediate representation called the rotated integral image or rotated sum auxiliary image. The rotated integral image is obtained by finding the sum of the pixels' intensity values that are located at a forty five degree angle to the left and above for the x value and below for the y value. So if A[x,y] is the original image and AR[x,y] is the rotated integral image is then calculated as shown in equation (7).

$$AR[x,y] = \sum_{x'\leq x, x'\leq x-|y-y'|} A(x',y') \tag{7}$$

6. Training classifiers for facial featurs

The facial features such as mouth, eyes and nose computed using Haar-Cascade is given for training. To train the classifiers, we have taken features from matched key-points. Features include key-point location, magnitude (highest value of key-point) and orientation (angle of each key-point).Brute Force (B-F) matcher algorithm is used for comparing the features. The B-F matcher algorithm checks every descriptor from first image to every descriptor in the second image. The descriptors are compared with their vector parameters using Euclidean Vector difference. The best-match image is detected from the maximum number of matched descriptors in each pair.

7. Results

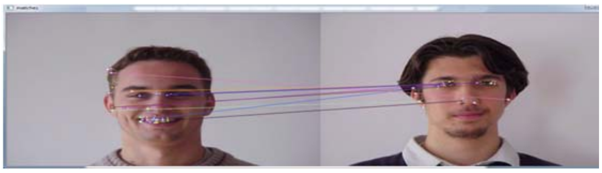
We have experimented image of different persons, images of one person smiling and not smiling, same image, images of different facial expression etc. The images are chosen from LFW and CVL Face Databases. The proposed HC-SIFT algorithm shows better result in all the cases and a comparison table on performance of various features of SIFT, PCI SIFT, SURF and HC-SIFT are given in table 1.

Table 1. performance on features of various methods

METHOD/ FEATURES	SIFT	PCA_ SIFT	SURF	Proposed Method(HC-SIFT)
Transformation	Good	Normal	Poor	Better
Illumination	Poor	Normal	Good	Normal
Blurring	Good	Poor	Poor	Better
Scale	Normal	Poor	Poor	Better
Rotation	Normal	Normal	Poor	Normal
Time	Normal	Normal	Good	Better
Overall Accuracy	Normal	Good	Normal	Better

The result of various test performed is shown below.

Fig. 11. (a) Img1 Fig. 11. (b) Img2



No: of key-point in img1: 51
No: of key-point in img2: 12
No: of matching Key-points: 9
Match percentages is: 23.73%

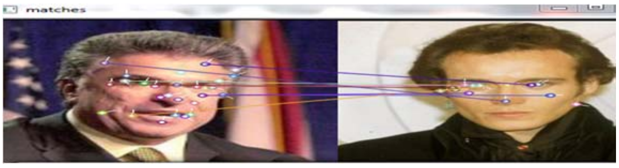
Fig. 12. (a) Img1 Fig. 12. (b) Img2

No: of key point in img1: 28
No: of key-point in img2: 32
No: of matching Key points: 21
Match percentages is: 75.0%

Fig. 13. (a) Img1 Fig. 13. (b) Img2

No: of key-point in img1: 15
No: of key-point in img2: 38
No: of matching Key points: 24
Match percentages is: 72.72%

Fig. 14. (a) Img1 Fig. 14. (b) Img2



No: of key-point in img1: 42
No: of key-point in img2: 20
No: of matching Key points: 11
Match percentages is: 26.19%

Fig. 15. (a) Img1 Fig. 15. (b) Img2

No: of key-point in img1: 24
No: of key-point in img2: 42
No: of matching Key points: 19
Match percentages is: 79.17%

Fig. 16. (a) Img1 Fig. 16. (b) Img2

No: of key-point in img1: 28
No: of key-point in img2: 43
No: of matching Key points: 20
Match percentages is: 82.17%

The performance of SIFT and HC- SIFT on key-points over face recognition is given in table 2.

Table 2. Performance of SIFT and HC-SIFT

Number of matches SIFT	Number of matches (HC-SIFT)	Accuracy of SIFT	Accuracy of (HC-SIFT)
12	9	12.24	23.73
32	21	67.12	75.0
38	24	63.82	72.72
42	19	70.45	79.17
43	20	73.36	82.17

8. Conclusion:

In this paper an improved SIFT algorithm (HC-SIFT) for face recognition is proposed and demonstrated. Though SIFT algorithm is invariant to many of the image attributes, but fails if the image size is large or not under the defined scope. So in this paper we have applied Haar-Cascade method to extract the relevant portion of image and then detected the key-points. Precision and recall for various test images having calculated, which shows as an average precision rate of 79.84 and recall as 77.5. The accuracy of the image recognition is increased from 52.6% to 75.1% from SIFT method to HC-SIFT algorithm.

REFERENCES

[1] Lowe D. object recognition from local scale-invariantfeatures[J], International Conference on ComputerVision(ICCV), 1999:1150-1157.
[2] David G et al.Distictive Image Features From Scale- nvariantKeypoints[J], International Journal of Computer Vision, 2004,60(2):91-110.
[3] Abdel-Hakim, Alaa E., and Aly A. Farag. "CSIFT: A SIFT descriptor with color invariant characteristics." Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. Vol. 2. IEEE, 2006.
[4] Bay, H., A. Ess, T. Tuytelaars, and L. Van Gool. 2006. Speeded-up robust features (SURF). Computer Vision ECCV (2006), Vol. 3951. Lecture Notes in Computer Science. p. 404-417.
[5] Cheung, Warren, and Ghassan Hamarneh. "N-sift: N-dimensional scale invariant feature transform for matching medical images." Biomedical Imaging: From Nano to Macro, 2007. ISBI 2007. 4th IEEE International Symposium on. IEEE, 2007.
[6] Juan, Luo, and Oubong Gwun. "A comparison of sift, pca-sift and surf."International Journal of Image Processing (IJIP) 3.4 (2009): 143-152.
[7] Li, Qiaoliang, et al. "Robust scale-invariant feature matching for remote sensing image registration." Geoscience and Remote Sensing Letters, IEEE6.2 (2009): 287-291.
[8] Shaikh, Taherim S., and Archana B. Patankar. "Multiple Feature Extraction Techniques in Image Stitching." International Journal of Computer Applications 123.15 (2015).
[9] Wilson, Phillip Ian, and John Fernandez. "Facial feature detection using Haar classifiers." Journal of Computing Sciences in Colleges 21.4 (2006): 127-133.