International Journal of Electrical and Electronic Engineering & Telecommunications

ISSN 2319 – 2518 www.ijeetc.com
Vol. 3, No. 3, July 2014

Research Paper

FINGER PRINT RECOGNITION USING SCALE INVARIANT FEATURE TRANSFORM

H S Mohana[1]*, Rajithkumar B K[2], Navya K[2], H S Prabhakara[3] and G Shivakumar[1]

*Corresponding Author: H S Mohana, ✉ hsm@mcehassan.ac.in

Finger print recognition and matching will play a very pivotal role in forensic investigations. In this paper, we propose a novel method for finger print recognition. Its basic idea is to use a course to fine strategy based on Scale Invariant Feature Transform (SIFT) feature. To recognize a test sample, our method contains the three main steps: the first step identifies a certain number of finger prints from training samples, depending on the Horizontal Matching key points. If first steps verified successfully, we perform vertical SIFT Matching based on vertical key points. If first and second methods get verified successfully then it moves on to the third step. Here it verifies the finger prints by Histogram matching method. The first two steps, our method succeeds in greatly reducing the computational complexity, and avoiding the interference of those samples that may cause error recognition to a certain extent. The third step enhances the robustness of our method. This approach uses three way matching method, hence it achieves 98.9% accuracy. Experiments on different finger prints from standard database confirms that our method obtain high recognition accuracy and has a good robustness.

Keywords: Image processing, SIFT algorithm, Euclidean distance, MATLAB

INTRODUCTION

Finger print authentication refers to the automated method of verifying a match between two human finger prints (Wang *et al.*, 2013). Fingerprints are one of many forms of biometrics used to identify individuals and verify their identity. The analysis of fingerprints for matching purposes generally requires the comparison of several features of the print pattern (Mohana *et al.*, 2014). These include patterns, which are aggregate characteristics of ridges, and minutia points, which are unique features found within the patterns. It is also necessary to know the structure and properties of human skin in order to successfully employ some of the imaging technologies. In our

[1] Department of IT, Malnad College of Engineering, Hassan, Karnataka, India.
[2] Department of ECE, Malnad College of Engineering, Hassan, Karnataka, India.
[3] Department of ISE, Malnad College of Engineering, Hassan, Karnataka, India.

Proposal we first store some fig prints in database and depend on user interaction we perform matching process of fingerprints based on SIFT algorithm. SIFT calculate Localize stable key points in scale space and it Perform feature detection in fig prints based on relative to canonical scale and orientation. The key points matching between two fig prints are estimate based on Similarity metric found in Euclidean distance and edge points in fig print (Daniel Cabrini and Hauagge Noah Snavely, 2012).

RELATED WORKS

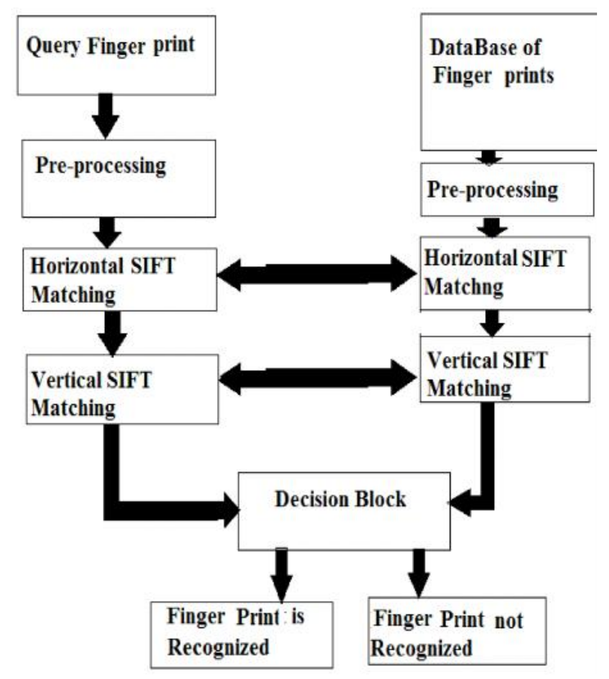
The review of the literature pertaining to the present topic is presented to the readers. In Wang *et al.* (2013) authors concentrates on the face Recognition based on SIFT algorithm using MATLAB. In this paper, we extended that work and applied that algorithm for finger print images. In Mohana *et al.* (2014) authors concentrate on the Matching of Kannada characters using SIFT,here we take some matching techniques. In Mohana *et al.* (2014) authors use 2D Sift methods for image mosaic based on SIFT, here we extended that work and applied that algorithm for verifies finger prints in our paper. In Mohana *et al.* (2012) authors use Gaussian filter for filtering for pre-processing, the Gaussian filter smoothing the image and here we adopt same pre-processing technique.

PROPOSED ALGORITHM

The proposed method consists of the following steps see Figures 1 and 2.

1. Collect fig prints of different persons and stored in a database of 24x42 pixels each
2. Select a Query Finger print image

Figure 1: A Block Diagram of the Proposed Algorithm

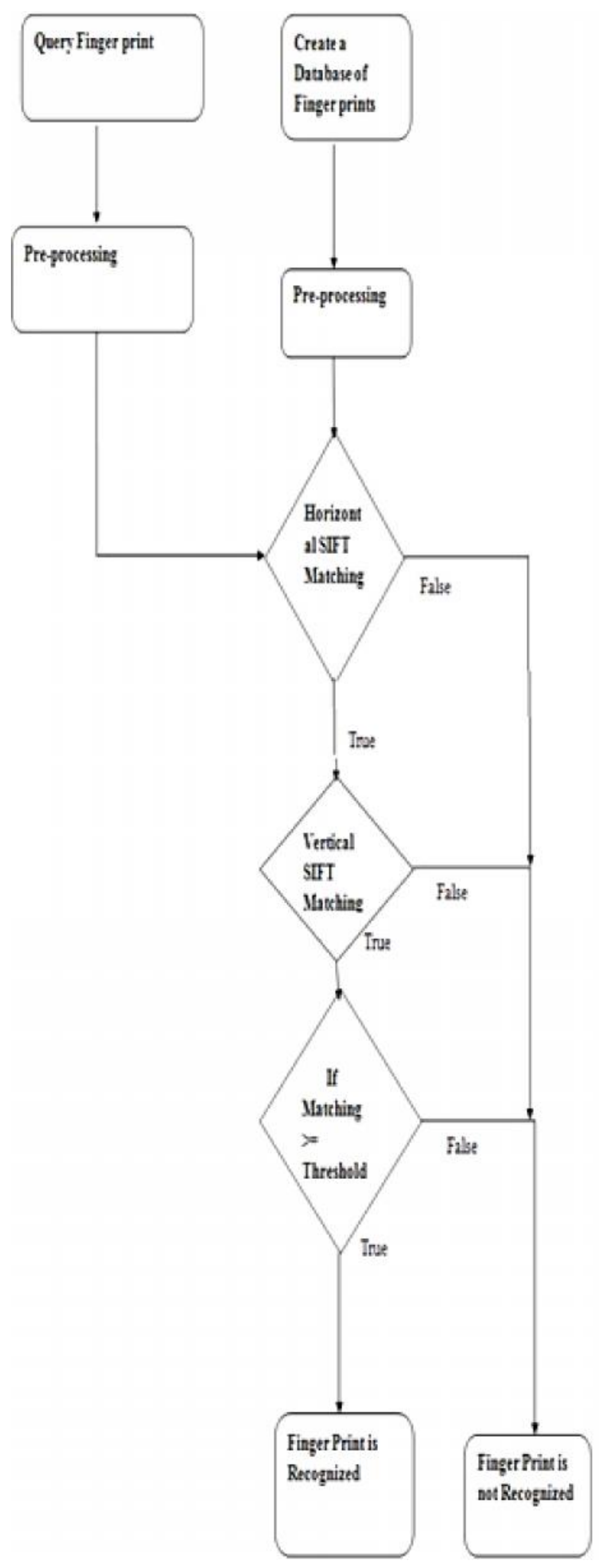


3. Pre-processing

   In this we perform

   a. Removing noise using Gaussian filter.
   b. Resizing pre-processed image into fixed pixel size and dimensions.
4. Horizontal SIFT Matching between Query image with Database.
   a. Here in this it first calculate key points between Query image with Database horizontal.
   b. The key point calculation is based on Euclidian distance and edges point in images.
   c. If like key points exists in horizontal direction, it marks horizontal line between images.
   d. It record matching percentage and if matching percentage is greater than

Figure 2: Flow Chart of the Proposed Algorithm



95% then it conclude that two images are matched.

5. Vertical SIFT Matching between Query image with Database.
   a. Here in this it first calculate key points between Query image with Database Vertical.
   b. The key point calculation is based on Euclidian distance and edges point in images.
   c. If like key points exists in Vertical direction, it marks Vertical l line between images.
   d. It record matching percentage and if matching percentage is greater than 95% then it conclude that two images are matched.
6. Decision Block
   a. In this Block it makes the decision based on pre-defined threshold value.
   b. Here the threshold value fixed to 95%.
   c. If Horizontal and vertical Matching value is greater than 95% in between two images, then it decide query fig print is belongs to database and declares it is authorized, else it decide query fig print not belongs to database and declares it is unauthorized.

METHODOLOGY AND IMPLEMENTATION

Pre-Processing

In this we first removing the noise using Gaussian filter

Gaussian Filtering

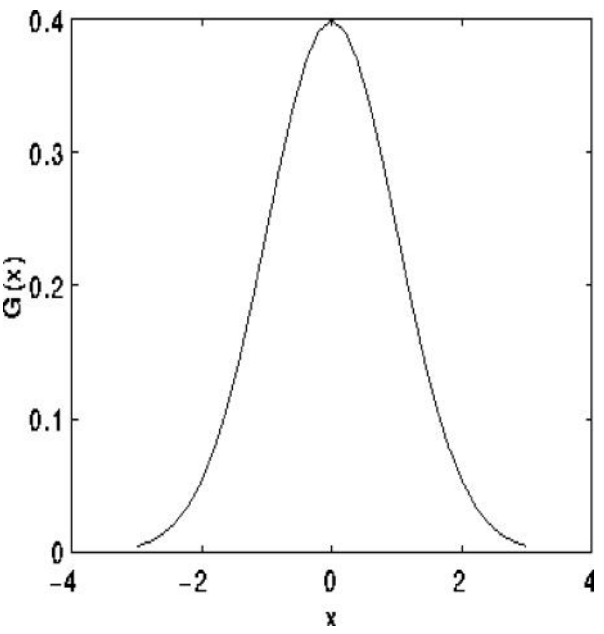
Gaussian filtering is used to blur images and remove noise and detail.

In one dimension, the Gaussian function is

$$G(x) = \frac{1}{\sqrt{2®\pm}} e^{-\frac{x^2}{2\pm^2}} \quad ...(1)$$

where ± is the standard deviation of the distribution. We have also assumed that the distribution has a mean of zero (i.e., it is centered on the line $x = 0$). The distribution is illustrated in Figure 3.

Figure 3: 1-D Gaussian Distribution with Mean 0 and ± = 1



SIFT

SIFT key points of objects are first extracted from a set of reference images and stored in a database. An object is recognized in a new image by individually comparing each feature from the new image to this database and finding candidate matching features based on Euclidean distance of their feature vectors. From the full set of matches, subsets of key points that agree on the object and its location, scale, and orientation in the new image are identified to filter out good matches. The determination of consistent clusters is performed rapidly by using an efficient hash table implementation of the generalized Hough transform. Each cluster of 3 or more features that agree on an object and its pose is then subject to further detailed model verification and subsequently outliers are discarded. Finally the probability that a particular set of features indicates the presence of an object is computed, given the accuracy of fit and number of probable false matches. Object matches that pass all these tests can be identified as correct with high confidence.

Key Stages

Scale-Invariant Feature Detection

Lowe's method for image feature generation transforms an image into a large collection of feature vectors, each of which is invariant to image translation, scaling, and rotation, partially invariant to illumination changes and robust to local geometric distortion. These features share similar properties with neurons in inferior temporal cortex that are used for object recognition in primate vision. Key locations are defined as maxima and minima of the result of difference of Gaussians function applied in scale space to a series of smoothed and re sampled images. Low contrast candidate points and edge response points along an edge are discarded.

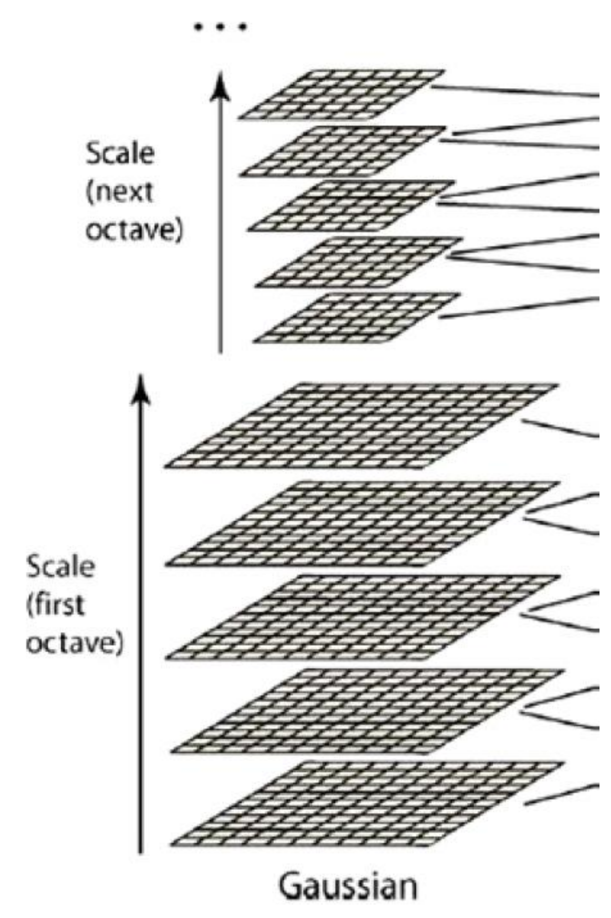
Feature Matching and Indexing

Indexing consists of storing SIFT keys and identifying matching keys from the new image. Lowe used a modification of the k-d tree algorithm called the Best-bin-first search method that can identify the nearest neighbors with high probability using only a limited amount of computation. The BBF algorithm uses a modified search ordering for the k-d tree algorithm so that bins in feature space are searched in the order of their closest distance from the query location. The nearest neighbors are defined as the key points with minimum Euclidean distance from the given descriptor vector. The probability that a match is correct can be determined by taking the ratio of distance from the closest neighbor to the distance of the second closest. Lower rejected all matches in which the distance ratio is greater than 0.8, which eliminates 90% of the false matches while discarding less than 5% of the correct matches. To further improve the efficiency of the best-bin-first algorithm search was cut off after checking the first 200 nearest neighbor candidates. For a database of 100,000 key points, this provides a speedup over exact nearest neighbor search by about 2 orders of magnitude, yet results in less than a 5% loss in the number of correct matches.

Scale-Space Extrema Detection

This is the stage where the interest points, which are called key points in the SIFT framework, are detected. For this, the image is convolved with Gaussian filters at different scales, and then the difference of successive Gaussian-blurred images is taken. Key points are then taken as maxima/minima of the Difference of Gaussians (DoG) that occur at multiple scales. Specifically, a DoG image $D(x, y, \pm)$ is given by

Figure 4: Scale Space



Horizontal SIFT Matching and Vertical SIFT Matching

In Horizontal SIFT Matching and Vertical SIFT Matching, the SIFT is usually used to extract local features of an image, and it has several advantages: SIFT features keep invariant on rotation, scale scaling, and illumination change of images; SIFT features maintain a certain degree of stability on perspective changes, affine transformation, and robustness to the noise; SIFT features are uniqueness and informative.

The feature extraction procedure of Horizontal SIFT Matching and Vertical SIFT Matching Can be described as follows.

Step 1: Construct the DOG scale-space:

$$D(x; y; \pm) = (G(x; y; k) - G(x; y; \pm))*I(x; y) = L(x; y; k\pm) - L(x; y; \pm) \quad ...(2)$$

where $G(x; y; \pm) = 1$

$$L(x; y; \pm) = G(x; y; \pm) * I(x; y) \quad ...(3)$$

Step 2: Get the key points: We get the key points at the scale space extreme in the difference of Gaussian function convolved with the image.

Step 3: Assign an orientation and gradient modulus to each key point:

$$m(x; y) = (L(x + 1; y) - L(x - 1; y))^2 + (L(x; y + 1) - L(x; y - 1))^2 \quad ...(4)$$

A Scale Invariant Feature Transform Based Method

$$^{-}(x; y) = \tan^{-1} (L(x; y + 1) - L(x; y - 1)\ L(x + 1; y) - L(x - 1; y)) \quad ...(5)$$

Step 4: Construct the descriptor of SIFT features:

We sample within an 8*8 neighborhood windows centered on the key points, and divide the neighborhood into four 4 * 4 child windows as shown in Figure 4. Then we calculate the gradient orientation histogram with eight bins in each child window and get a 128-dimensional vector Called descriptor.

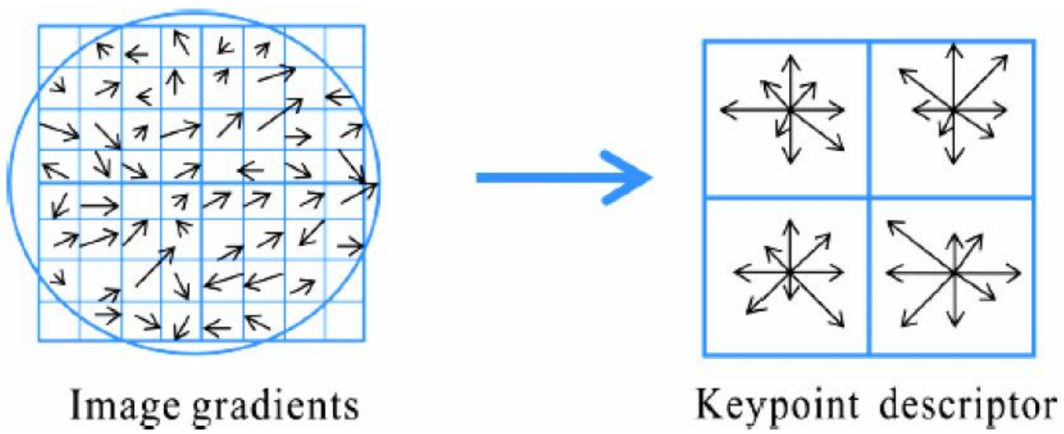


Figure 5: Get Descriptors from the Gradient Orientation Histogram

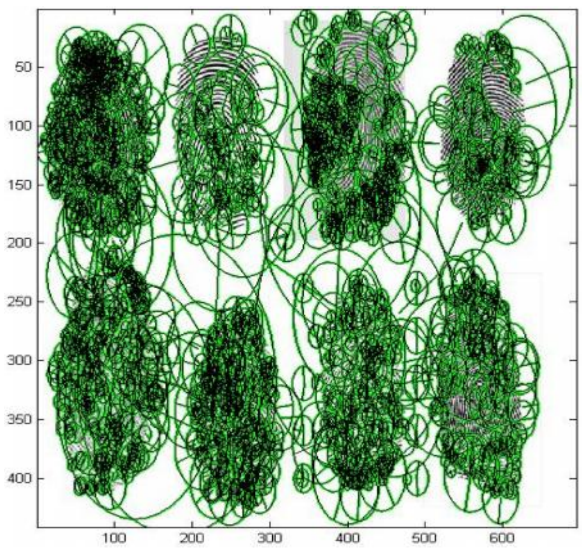


Figure 6: A Show for the Descriptors of SIFT Extracted

We can extract different numbers of descriptors from different images. Take images of one subject in FERET face database as an example, the descriptors of SIFT features extracted in the images can be shown as a series of points in Figure 5.

RESULTS

Step 1: Collect a different finger print of different persons and Stored in a database shown in Figure 7.

Step 2: Select a query finger Print image.

Step 3: Pre-preprocessing.

The captured images contains some noises so here we remove that noise using

Figure 7: Database of Finger Prints

Figure 8: Query Finger Print

Figure 9: Filtered Image

Gaussian filter. The main advantages of this filter is it remove noise and as well as smoothing the images. The filtered image is shown in Figure 9.

Step 4: Horizontal SIFT Matching.

In this we first calculate key points between Query images with Database horizontal shown in Figures 10 and 11.

Figure 10: Horizontal Matching Method Descriptor

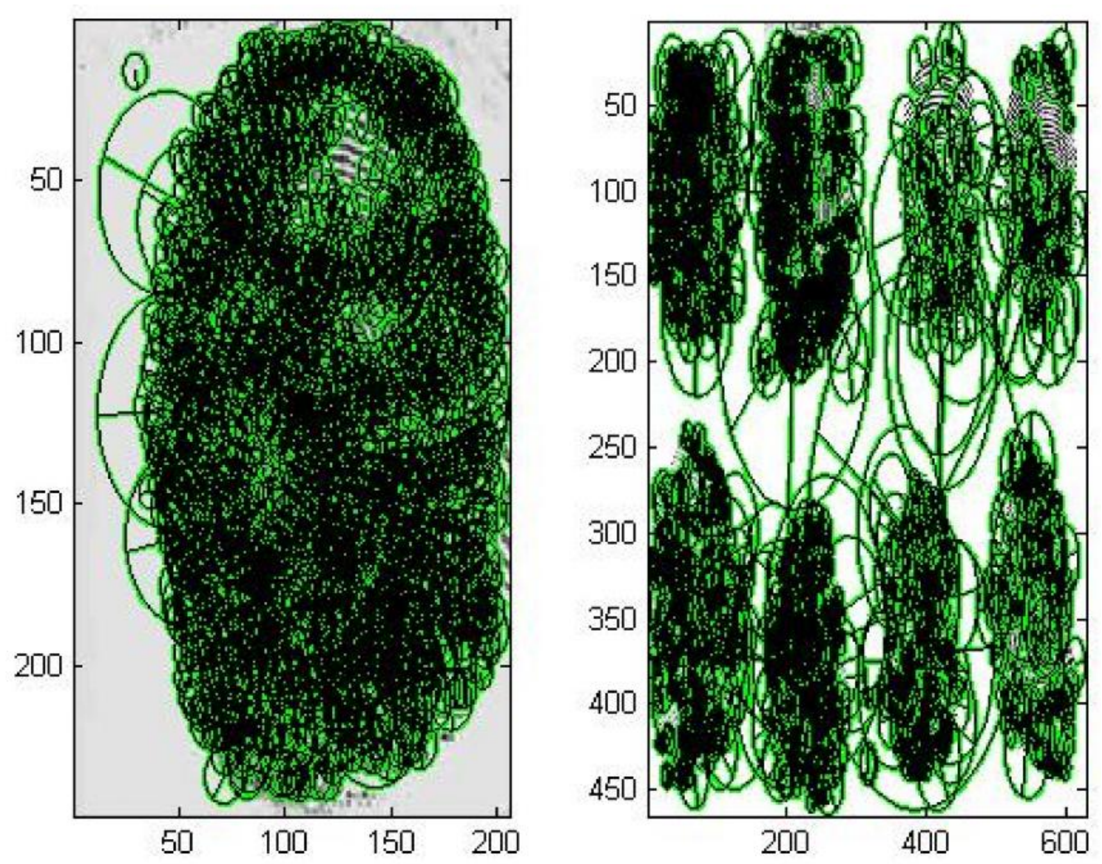
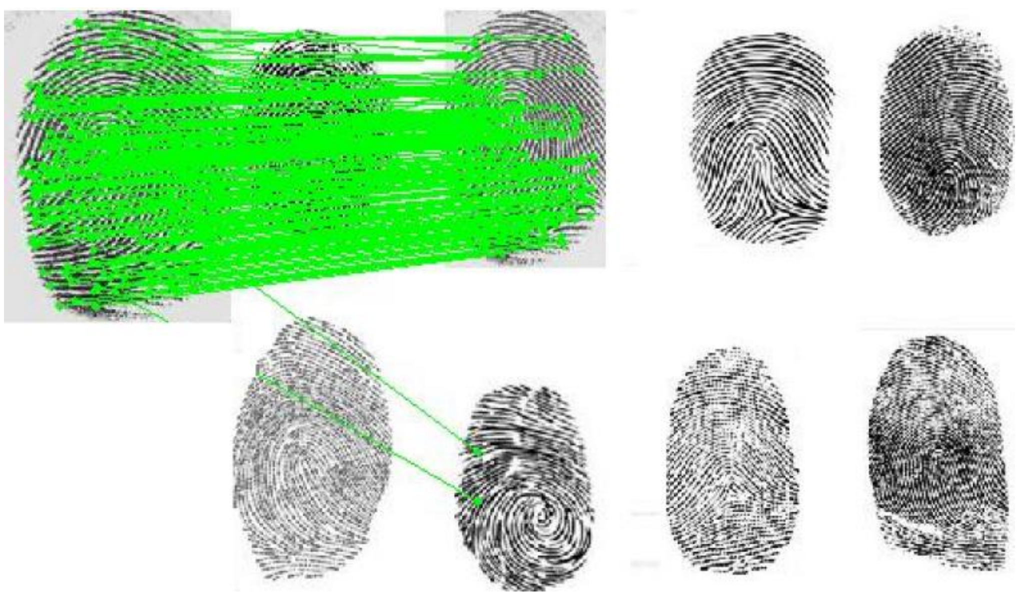


Figure 11: Horizontal SIFT Matching



In the above Figures 10 and 11, the query image matches horizontal with database image and matching Factor is greater than 95% hence we conclude that query image is authorized and we move to vertical method for future verification.

Step 4: Vertical SIFT Matching

In this we calculate key points between Query images with Database in vertical.

In the above Figures 12 and 13 the query image matches vertical with database image and matching Factor is greater than 95% hence we conclude that query image is authorized and we move to decision block.

Step 5: Decision block

In this we again check two images matching by calculate its Histogram shown in Figures 14 and 15.

Figure 12: Vertical Matching Method Descriptor

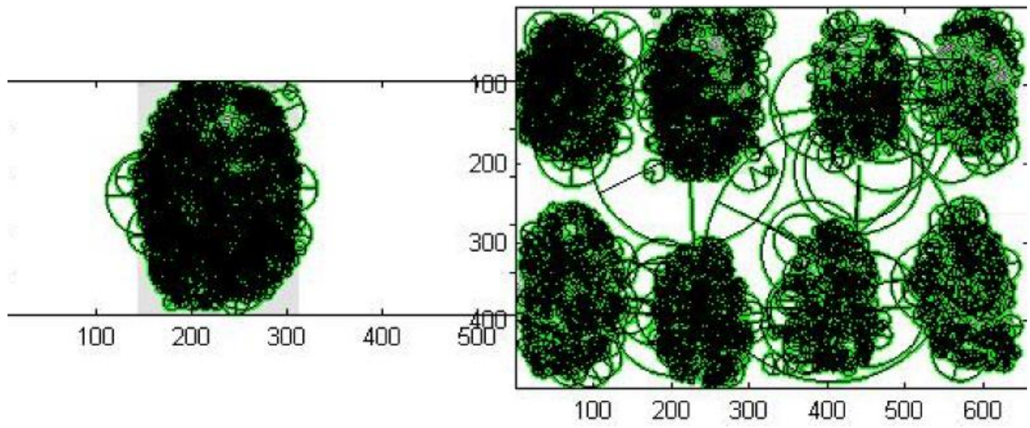


Figure 13: Vertical SIFT Matching

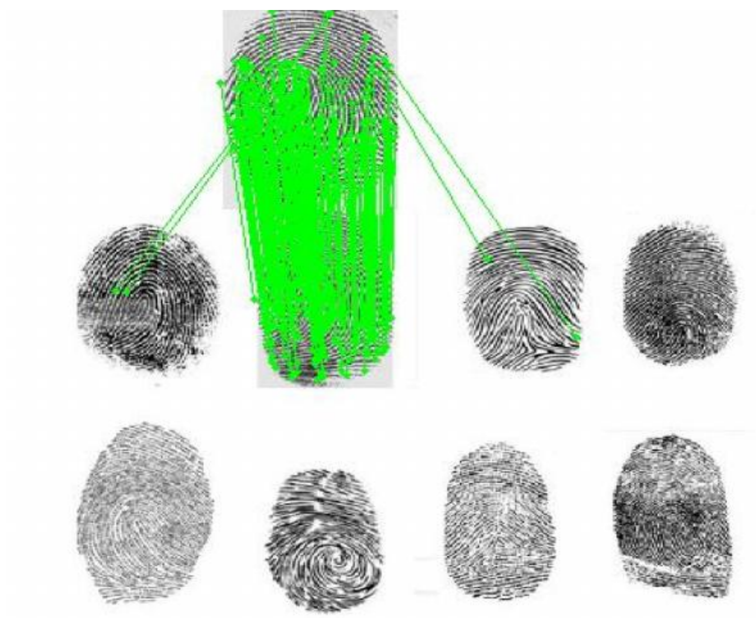


Figure 14: Histogram of Query Image

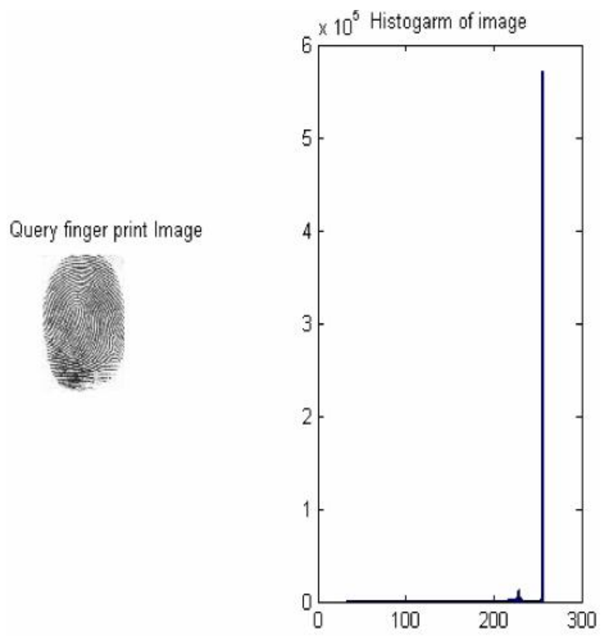
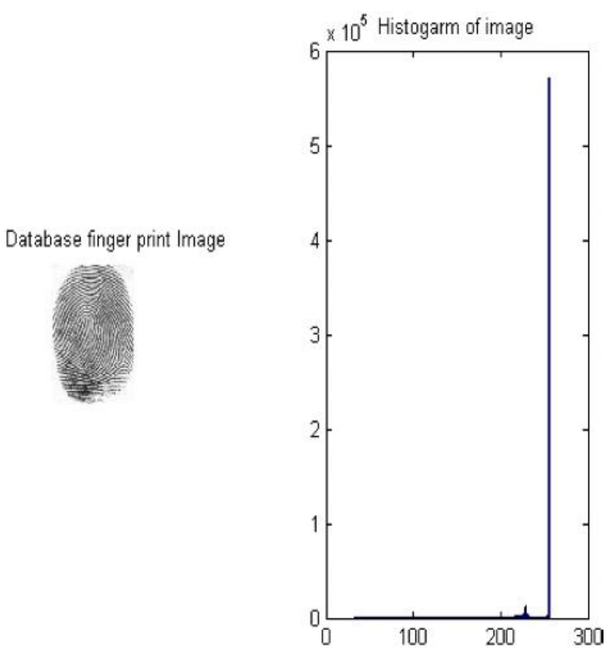


Figure 15: Histogram of Matched Database Image



From Horizontal SIFT Matching, Vertical SIFT Matching and Histogram we conclude that the query finger print is belongs to database so we declare the query finger print is authorized and by experimental results we conclude it achieve 98.9% perfect matching.

CONCLUSION

This paper proposes a SIFT features based method for finger print recognition and it reduce the size of data and computational speed is high. This provide a good tool for the peoples finger prints identification and it well suited for Forensic applications. SIFT method is robust. At the same time it can cope with some images with tilt and roll. It is effective and Matching can be verified by three methods, i.e., Horizontal, vertical and histogram matching methods. SIFT features respectively, which reduce the computational complexity and enhance the effectiveness for finger print recognition at the same time. The proposed method was tested on real-world stereo images from a robotic application and standard dataset images. The experimental results show that the feature matching can be 98.9% accuracy.

REFERENCES

1. Chariot A and Keriven R (2008), "GPU-Boosted Online Image Matching", in 19th Int. Conf. on Pattern Recognition, Tampa, Florida, USA.
2. Dai B, Zhang D, Liu H, Sun S and Li K (2009), "Evaluation of Face Recognition Techniques", *Proc. of SPIE*, Vol. 7489, pp. 74890M-1-74890M-7.
3. Daniel Cabrini and Hauagge Noah Snavely (2012), "Image Matching Using Local Symmetry Features", Cornell University Fhauagge.
4. Faraj Alhwarin, Danijela Ristiæ–Durrant and Axel Gräser (2010), "VF-SIFT: Very

Fast SIFT Feature Matching", Institute of Automation, University of Bremen, Otto-Hahn-Alle NW1, D-28359 Bremen, Germany.

5. Firedman J H, Bentley J L and Finkel R A (1977), "An Algorithm for Finding Best Matches in Logarithmic Expected Time", *ACM Transactions Mathematical Software*, pp. 209-226.

6. Geng C (2009), "Face Recognition Using Sift Features", Proc. of the 16th IEEE International Conference on Image Processing, pp. 3313-3316.

7. Heymann S, Miller K, Smolic A, Froehlich B and Wiegand T (2007), "SIFT Implementation and Optimization for General-Purpose GPU", in January, WSCG.

8. http://www.homepages.inf.ed.ac.uk/rbf/HIPR2/sobel.htm

9. http://www.mathwork.com

10. Ke Y and Sukthankar R (2004), "PCA-Sift: A More Distinctive Representation for Local Image Descriptors", Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 506-513.

11. KriÇzaj J, ÇStruc V and PaveÇsi´c N (2010), "Adaptation of Sift Features for Robust Face Recognition", Proc. of the 7th International Conference on Image Analysis and Recognition, pp. 394-404.

12. Luo J, Ma Y, Takikawa E, Lao S, Kawade M and Lu B L (2007), "Person-Specific Sift Features for Face Recognition", Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 593-596.

13. Majumdar A and Ward R K (2009), "Discriminative Sift Features for Face Recognition", Proc. of Canadian Conference on Electrical and Computer Engineering, pp. 27-30.

14. Mikolajczyk K and Schmid C (2005), "A Performance Evaluation of Local Descriptors", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 27, No. 10, pp. 1651-1630.

15. Mohana H S *et al.* (2012), "Machine Vision Based Non-Magnetic Object Detection and Removal on Moving Conveyors in Steel Industry Through Differential Techniques", *International Journal of Computer Vision and Image Processing (IJCVIP)*, Vol. 2, No. 3, Accessed (March 14, 2014), doi:10.4018/ijcvip.2012070105.

16. Mohana H S, Navya K, Srikanth P C and Shivakumar G (2014), "Stone In-Scripted Kannada Characters Matching Using Sift", IRF International Conference, April 5, Pondicherry, India.

17. Mohana H S, Rajithkumar B K, Sujatha B R and Bhavana M B (2014), "Extraction of Stone In-Scripted Kannada Characters Using Sift Algorithm Based Image Mosaic", *International Journal of Electronics & Communication Technology (IJECT)*, Vol. 5, No. 2, April-June.

18. Muja M and Lowe D G (2009), "Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration", in Int. Conf. on Computer Vision Theory and Applications.

19. Rafel C Gonzalez and Richard E Woods (2007), *Digital Image Processing*, 2nd Edition, Prentice Hall.

20. Se S, Ng H, Jasiobedzki P and Moyung T (2004), "Vision Based Modeling and Localization for Planetary Exploration Rovers", in Proceedings of International Astronautical Congress.

21. Wang Y Y, Li Z M, Wang L and Wang M (2013), "A Scale Invariant Feature Transform Based Method", *Journal of Information Hiding and Multimedia Signal Processing*, Vol. 4, No. 2.