SAFEGUARDING CONFIDENTIALITY BY FINGERPRINT COMBINATION

[1]Sonali S. Gahukar, [2]Gargi S. Phadke

[1,2] *Department of Instrumentation Engineering*

Ramrao Adik Institute of Technology, Navi Mumbai, (India)

ABSTRACT

Individual's identity fraud in our society now, is reaching unprecedented proportions. Modern security is a battle between high security and low friction. There is an increasing emphasis on the emerging of automatic personal identification applications. Due to this, biometrics-based verification, especially fingerprint-based identification, is receiving a lot of attention. The oldest and extensively used form of biometric identification is the fingerprints. But problem in single fingerprint identification and verification is that, the data is easily available to the imposters. They can easily reconstruct original fingerprints back from the features stored in the database. In this Paper, the perspective of mixing two different fingerprints at the image level has been explored. These are acquired from two different fingers, in order to create a new fingerprint. The minutiae positions from one fingerprint and the orientation from another fingerprint images are extracted. On the basis of this extracted information, a combined template is fused and stored in a database. For the authentication, the system demands two query fingerprints from the same two fingers which were utilized during the enrolment process. And the comparison of these two templates is done for authentication.

Keywords: Combination, Fingerprints, Matlab, Minutiae, Orientation, Security

I INTRODUCTION

The general goal of the research of which this report is part, is 1. To extract minutiae points and orientations from first fingerprint and second fingerprint respectively and 2. To combine these two images into a virtual identity and either save it in database when it is a new entry or authenticate it with the fingerprints provided for verification. The main goal of this project is to preserve privacy of personal details. An approach involves extraction of specific information from two different fingerprints after combination, and comparing it with information already stored in the database of an individual using Matlab. The figure 1 explains the building blocks of proposed algorithm.

Props used for privacy protection like passwords can be stolen, assumed, or forgotten and also the tokens may be misplaced or stolen. But then the biometric authentication systems attempt to relate verification directly to the user protection. The use of fingerprints for identification has been employed in law enforcement for about a century. A much broader application of fingerprints is for personal authentication. For instance to access a network, a bank machine, a car, a home, electronic door locks, mobile phones, laptops or financial applications such as online payment. Fingerprint detection may be used to unlock a device, sign in to apps, and authenticate purchases using Google Play and some third-party apps, identity documents like passports or scanners at supermarkets etc. The problem with single fingerprint verification is that they aren't as secure as you might think, it raises some interesting privacy issues but worst of all, once your biometric information has been compromised you can't change it. In this paper a combination of two different fingerprints is proposed to create a new virtual identity, so that when database is attacked, the original fingerprints will never be available directly as a combination template stored in the database [1]. This method can decrease the risk of single fingerprint encryption. This system has following advantages: (a) it can be used to create a virtual identity from two different fingers and (b) it can be used to make the information present in an individual's fingerprint image ambiguous prior to storing it in a central database [2].

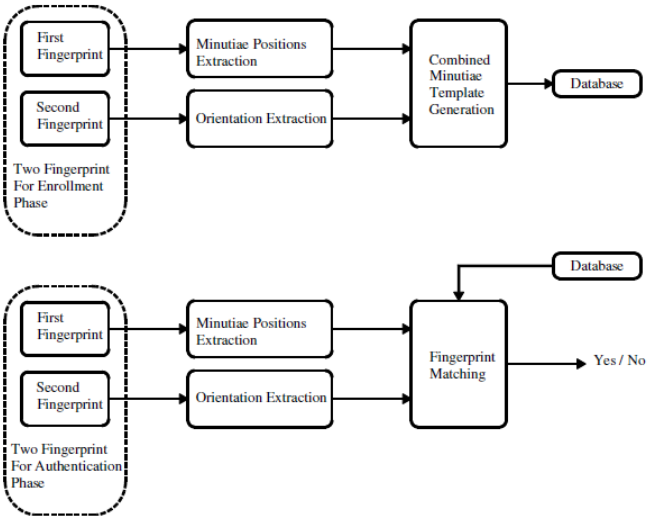

First Fingerprint
Second Fingerprint
Two Fingerprint For Enrollment Phase
Minutiae Positions Extraction
Orientation Extraction
Combined Minutiae Template Generation
Database
Database
First Fingerprint
Second Fingerprint
Two Fingerprint For Authentication Phase
Minutiae Positions Extraction
Orientation Extraction
Fingerprint Matching
Yes / No

Fig. 1 Fingerprint Fusion for Privacy Safeguarding

1.1 Problems in single fingerprint identification

With the extensive applications of fingerprint techniques in authentication systems, protecting the privacy of the fingerprint becomes an important issue. Conventional encryption is not adequate for fingerprint privacy protection because decryption is necessary before the fingerprint matching. This obviously exposes the fingerprint to the imposter. Hence, establishing protection techniques explicitly for fingerprints is the necessity of the time [3]. Lots of problems are created, because majority of the existing techniques make use of the key for the fingerprint privacy protection. They may also be more vulnerable when both the key and the protected fingerprint are hijacked together.

Yanikoglu et al. first introduced the combination of two separate fingerprints into a single identity. This unique characteristic is either in the image level or in the feature level. He performed the combination of two different fingerprints to create a new identity. He did this by combining the minutiae positions extracted from the two different fingerprints. The locations of original minutiae of each fingerprint were protected in this new identity. However, it is easy for the attacker to recognize this new identity since it contains more minutiae positions compared to than that of the two original fingerprints [4].

Othman and Ross presented the method in which an input fingerprint image is mixed with another fingerprint that is from a different finger, so as to create a new mixed image that produces tricky identity of the original fingerprint which becomes difficult to recognize. Mixing fingerprints produces a new unit that looks like a reasonable fingerprint. And then hence, (a) it can be practiced by traditional fingerprint algorithms and (b) a burglar may not be able to make a decision if a given print is mixed or not. Here each fingerprint is separated into two components, the spiral and continuous components, for mixing two fingerprints. Subsequent to pre-aligning of the two components of each fingerprint, the constant component of one fingerprint is united with the spiral component of the other fingerprint image so as to create a mixed fingerprint [5][6].

This Paper is ordered as follows. Some research about Combined fingerprints is briefed in Section I. In Segment II, the Proposed Fingerprint Recognition system is explained. Section III shows results and discussions. Section IV, determines conclusions.

II THE PROPOSED FINGERPRINT RECOGNITION SYSTEM

In the proposed system, there are two phases. For the enrolment stage, two fingerprints from two different fingers of an individual are acquired. Then follows the process of extraction of the minutiae positions from first fingerprint and the orientation from second fingerprint. Then, a combined minutiae template is created on the basis of minutiae positions and the orientations. Subsequently, this combined template is stored in a database. In the authentication stage, two query fingerprints are requisite from the same two fingers. It follows the same steps of the enrolment stage, extraction of the minutiae positions from the first fingerprint followed with orientations from second fingerprint. This extracted information is then compared with the corresponding template stored in the database against that specific name. The authentication will be successful when the enrolment and authentication templates match [3].

Fingerprint identification system consists of three steps. First it collects 2D images of fingerprints, then it extracts specific information from them and finally, resemblance of certain fingerprint information is measured by comparing it with information already stored in the database.

2.1 Fingerprint Image Enhancement

This is one of the most critical stages of fingerprint recognition because identification of poor quality images is a most difficult problem. Fingerprint Image enhancement is required to make the image clearer for further evident operations. For increasing the contrast between ridges and furrows and also for connecting the false broken points of ridges, these enhancement methods turn out to be very useful for maintaining a higher accuracy of fingerprint recognition. This is due to the fingerprint images captured from sensors or other medium are not guaranteed with perfect quality. Prior to feature extraction, fingerprint enhancement is essential to suppress various noises, and improve the clarity of ridge structures.

For the improvement of the texture component, a multiscale patch-based sparse representation method has been used here. Dictionaries are constructed with a set of Gabor elementary functions to capture the characteristics of fingerprint ridge structure, and multiscale patch-based sparse representation is iteratively applied to rebuild high-quality fingerprint image. The first step for fingerprint enhancement by means of sparse representation is to create a redundant dictionary. To achieve good representation, dictionary should characterize all kinds of image structures and details. The local fingerprint patch usually forms a 2D sinusoidal-shaped wave with well-defined orientation and frequency.

Gabor functions possess both frequency and orientation selective properties. These functions also have best joint resolution in both spatial and frequency domains as well. They capture the periodic, yet non-stationary characteristics of fingerprint pattern and form a very spontaneous representation of fingerprint images. Thus, Gabor basis functions are used to model the local patch of fingerprint images. The 2D Gabor functions have the general form as :

$$h(x,y,\theta,f) = exp[-\frac{1}{2}[\frac{(x_\theta^2)}{(\delta_x^2)} + \frac{(y_\theta^2)}{(\delta_y^2)}]]cos(2\pi f x_\theta + \phi_0) \qquad (1)$$

$$x_\theta = xcos\theta + ysin\theta \qquad (2)$$

$$y_\theta = -xsin\theta + ycos\theta \qquad (3)$$

where θ represents the orientation of the normal to the parallel stripes of Gabor function; f is the frequency of a sinusoidal plane wave; δx and δy are the space constants of the Gaussian envelope along x and y axes, respectively; Φ0 is the phase offset of Gabor function [7].

In Multi-Scale Iterative Sparse Representation, the patches are processed in raster-scan order, from left to right and top to down, for rebuilding of the whole fingerprint. The patch size here plays important role to determine the quality of reconstructed fingerprint image. A large patch can suppress the noise well but it smoothes the fingerprint ridge details such as ridge ending and bifurcations, which are vital fine level features for detection. Contrariwise, a small patch can safeguard the ridge details but it is sensitive to noise. Hence, the sparse representation through multi-scale patches is used to retrieve from this problem, [7].

2.2 Minutiae Extraction

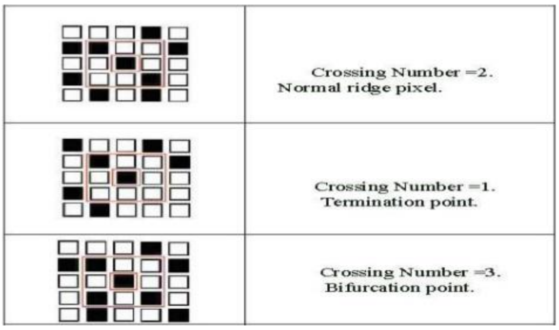
After a fingerprint image has been improved, features can be extracted from it. Most commonly used methods are as follows:

2.2.1 Fingerprint Ridge Thinning

After the fingerprint image is enhanced, it is then subsequently converted to binary form, and submitted to the thinning algorithm which trims down the ridge thickness to one pixel wide. Ridge Thinning is used to eliminate the redundant pixels of ridges until the ridges are just one pixel wide. In each scan of the full fingerprint image, the algorithm marks down disused pixels in each small image window (3x3). In the end it removes all the marked pixels after numerous scans. Binarization is completely enforced because then only pixels with maximum gray intensity value are left. Further to eliminate some H breaks, the thinned ridge map is then sieved by morphological operations. Due to the large number of spurious features produced, the accurate extraction of minutiae from image would not be possible. Exploiting a series of image enhancement stages prior to thinning is effective in facilitating the trustworthy extraction of minutiae.

2.2.2 Minutiae Marking

The Crossing Number (CN) concept is the most regularly used method for minutiae extraction. This method makes use of the thinned image where the ridge flow pattern is eight-connected. By scanning the local neighborhood of each ridge pixel in the image using a 3x3 window, the minutiae are extracted. The crossing number value is then calculated, which is defined as half the sum of the differences between pairs of adjacent pixels in the eight-neighborhood. The ridge pixel can then be categorized as a ridge ending, bifurcation or non-minutiae point. Usually, for each 3x3 window, if the central pixel is 1 and has exactly 3 one-value neighbors, then the central pixel is a ridge branch. And if the central pixel is 1 and it has only 1 one-value neighbour, then this central pixel is categorised as a ridge ending. Therefore, if crossing Number is 1, 2 and 3, then minutiae points are categorized as Termination, Normal ridge and Bifurcation respectively. For each and every extracted minutiae point, the following information is recorded: x and y coordinates, orientation of the associated ridge segment, and the type of minutiae (ridge ending or bifurcation) for further operations.


Crossing Number =2.
Normal ridge pixel.

Crossing Number =1.
Termination point.

Crossing Number =3.
Bifurcation point.

Fig. 2 Bifurcation and Termination

2.3 Orientation Extraction

The thinned image values, 1's and 0's which are obtained from the binary image, are stored in a temporary variable. The parameter calculation of the orientation represents the corresponding local area orientation around the origin. The algorithm searches for a 1 in the variable containing the thinned image details. The scan of the image takes place with the help of a 3x3 matrix generated. As soon as it finds a 1, the algorithm calculates eight different values. These eight different values are the orientation with respect to the original point. These eight different values are stored in a variable so as to keep a note of the orientation angle and are used as an additional feature to the minutiae details. The position of scan depends on the position of Minutia Termination and Minutia Bifurcations. When a 1 is found in the temporary variable, the algorithm stores the value and starts looking for the next 1 in the nearby vicinity. When the next 1 is found in the vicinity, the co-ordinates of the same are recorded and the angle is calculated. The algorithm does not stop here; it still looks for more 1's since the algorithm is designed to extract six points on the ridge. Once all the values are calculated it is then saved in another variable. This calculation provides us with a unique information or so-called feature that is used for matching purpose when a query image is inputted [3].

III RESULTS AND DISCUSSIONS

Fingerprint identification system processes in three steps. First it collects 2D images of fingerprints, then it extracts precise information from them and finally, resemblance of certain fingerprint information is measured by comparing it with information already stored in the database. The block diagram of combined fingerprint detection system at database entry phase is shown in Figure 3. The detailed description of the same is as follows:

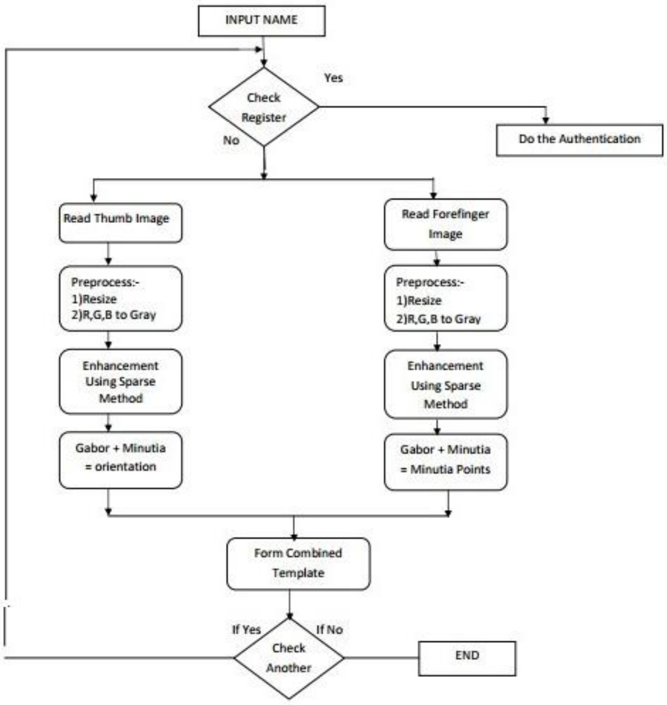


Fig. 3 Database Entry

3.1 Reading Index Fingerprint Image

By using image acquisition devices like scanners or other fingerprint reader devices, the proposed procedure begins the capture of the index fingerprint image. This image is read from the devices and then further processing is to be performed on that image. Figure 4 below shows the index fingerprint image, on which further processing is to be executed.

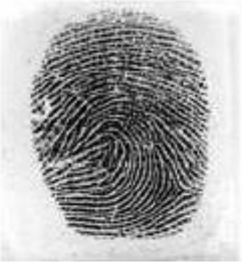


Fig. 4 Index fingerprint Image

3.1.1 Preprocessing

The detection of minutiae requires some preprocessing steps to be performed on original input fingerprint image. As shown below in Figure 5, the original image is primarily converted to Gray scaled image.

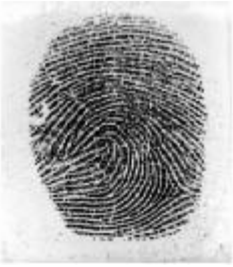


Fig. 5 Conversion of RGB image to Gray scaled image

3.1.2 Enhancement using Sparse Method

In image processing, after conversion to gray scale, sparse method has been used for Image improvement. Gabor basis functions are used to obtain a superior fingerprint image. Basically, to eliminate the noise and preserve true ridge/valley structures, these filters are used as band pass filters. Figure 6 below shows the image obtained after Sparse method.

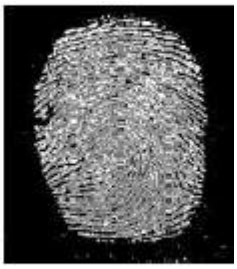


Fig. 6 Sparse method output

3.1.3 Minutiae Points Detection

Minutiae are the most important features of a fingerprint, hence comparison of one fingerprint with another can be carried out by means of them. Minutiae are the characteristic points in a fingerprint image used universally for further analysis and steps. Ridge endings and ridge bifurcations are the most frequently used minutiae. Figure 7 demonstrates minutiae points obtained for the original fingerprint image of Figure 4.

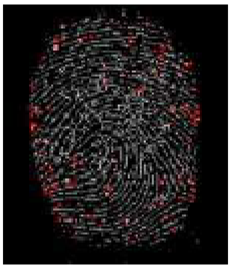


Fig. 7 Detected minutiae positions

3.1.4 Reading Thumb Fingerprint Image

Now the thumb fingerprint image is captured as shown in Figure 8 on which further processing is performed.

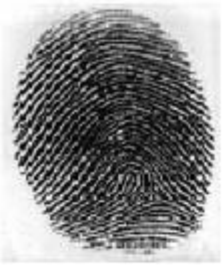


Fig. 8 Thumb fingerprint image

3.1.5 Preprocessing

Same procedure as above is done here of Conversion to Gray scaled image.

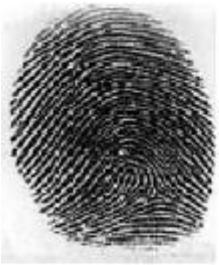


Fig. 9 Conversion of RGB image to Gray scaled image

3.1.6 Enhancement using Sparse method

Enhanced image of thumb finger is shown below in Figure 10.

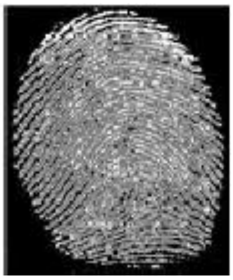


Fig. 10 Sparse method output

3.1.7 Orientation points detection

Orientation points of the thumb fingerprint are obtained as seen in Figure 11.

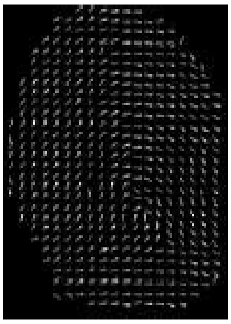


Fig. 11 Obtained orientation positions

3.1.8 Combined template

A combined template is created by executing AND operation on minutiae image and orientation image. Thus, the extracted minutiae points and orientations are combined together to form a Combined template. The image of combined template thus obtained is shown below in Figure 12.

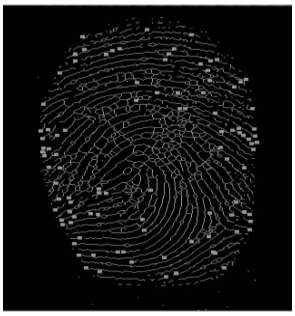


Fig. 12 Combined template

After this if there are still some more entries, we can continue with the same procedure. And if no or done with the entries, then this process can be terminated.

In Figure 13, the Authentication flow chart is represented. Here the process is same as the Database entry till the combined template formation. After that the features are extracted. These features are then used for authentication purpose. The authentication process involves the comparison of obtained combined template and the stored database combined template against a particular name provided. If they match, then this method approves the name and the fingerprints. Then further, other names and fingerprints combinations can also be checked for authentication.

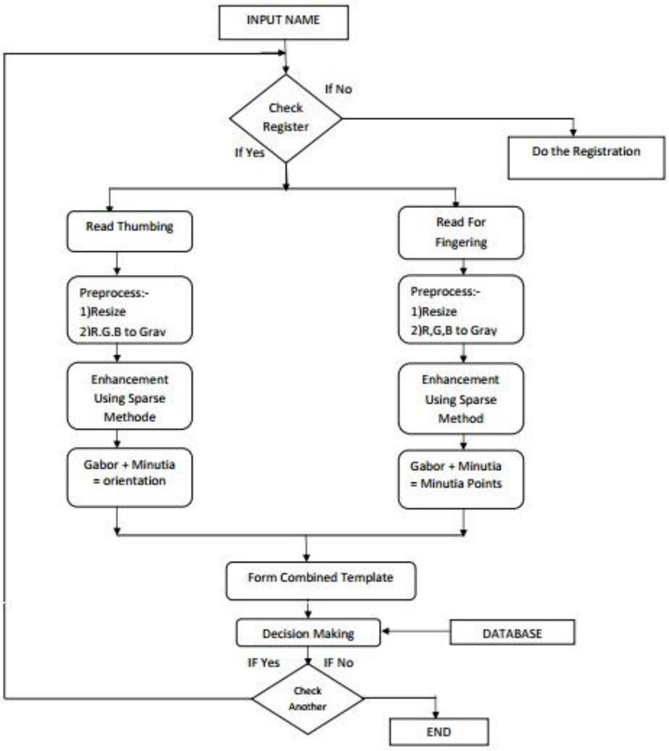


Fig. 13 Verification flow chart

3.1.9 GUI

The GUI is prepared in Matlab for all these processes in which we can see the results as shown in figure 14 below.

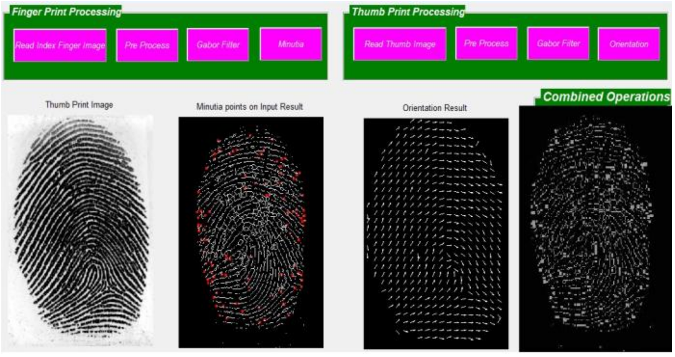


Fig. 13 Matlab GUI

IV CONCLUSION

The consistency of any automatic fingerprint recognition system robustly relies on the precision obtained in the minutiae extraction process. A number of factors damage the correct location of minutiae. Among them, low image quality is the one most influencing. For obtaining this we have used the sparse method for Image enhancement.

In this paper, we considered two fingerprints of two different fingers of a same individual. And subsequently we extracted minutiae from one fingerprint and orientation from the second fingerprint. Then we fused these two images and stored in a database. And also authenticated this combined template. Hence, this method gives a highly secured database where the imposters cannot split the combined fingerprint. Thus providing a Robust database for fortifying security of individuals.

The further work can be carried out after the combination on fortification of the system. Features can be extracted from the combined template and they can be compared during authentication. Decision tree can be implemented. Even trained neural networks can be attempted. This will further enrich the efforts in this paper.

REFERENCES

[1] Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, ``Handbook of Fingerprint Recognition", New York: Springer-Verlag, 2003.

[2] S. Li and A. C. Kot, ``A novel system for fingerprint privacy protection'', in Proc. 7th Int. Conf. Inform. Assurance and Security (IAS), pp. 262–266, Dec. 5–8, 2011.

[3] Sheng Li and Alex C. Kot, ``Fingerprint Combination for Privacy Protection", IEEE Transactions on Information Forensics and Security, Vol. 8, no. 2, pp. 350 – 360, Feb. 2013.

[4] Yanikoglu and A. Kholmatov, ``Combining multiple biometrics to protect privacy", in Proc. ICPR- BCTP Workshop, Cambridge, U.K., Aug. 2004.

[5] Ross and A. Othman, ``Mixing fingerprints for template security and privacy", in Proc. 19th Eur. Signal Proc. Conf. (EUSIPCO), Barcelona, Spain, Aug. 29–Sep. 2, 2011.

[6] A. Othman and A. Ross, “Mixing fingerprints for generating virtual identities,” in Proc. IEEE Int. Workshop on Inform. Forensics and Security (WIFS), Foz do Iguacu, Brazil, Nov. 29–Dec. 2, 2011.

[7] Manhua Liu, Xiaoying Chen and Xiaoduan Wang, ``Latent Fingerprint Enhancement via Multi-Scale Patch Based Sparse Representation”, IEEE Transactions on Information Forensics and Security, Vol. 10, No. 1, pp. 6-15, January 2015.