Feedback based Adaptive matching algorithm for cancellable fingerprint templates

[1]T Sathishkumar, [2]G Prabhakara Rao, [3]P Arumugam
[1]Technical Officer D, [2]Scientific Officer H, [3]Scientific Officer F
[1]Security Electronics Section,
[1]Indra Gandhi Centre for Atomic Research, HBNI, Kalpakkam, Tamil Nadu, India

Abstract : The Cancellable fingerprint generation techniques extract features from the fingerprint, obtain a user key input from the user and transform the features to a non-invertible domain using the user key input and store the transformed features as templates. Non-invertible transformations are many to one transformation which means that distinct biometric feature sets could be transformed in to similar feature sets because of transformation by user key input. The matching algorithms generally match between enrolled and query bit strings and generate a score based on how different the two bit strings are to each other, like hamming distance. This simple approach to matching works well when both bit strings contain raw biometric features such as minutiae locations and angles. But matching cancellable templates using such bit string matching algorithms would not be successful because they don't take into account the many to one transformations caused by the user key. Specifically, the ability of the matching algorithm to separate between genuine and imposter diminishes and Equal Error Rate increases. A novel feedback based adaptive matching algorithm that matches features instead of bit-strings is presented in this paper. The proposed algorithm matches features in the cancellable templates that are generated by Delaunay triangulations. The algorithm includes a feedback based parameter which enables the matching algorithm to maximize separability and minimize EER by adapting itself to the templates in the population which is demonstrated by applying the algorithm on public database. The separability thus obtained is higher than other such algorithms and EER is comparable with other such algorithms.

IndexTerms - Cancellable fingerprints, Matching Algorithms, Separability, Equal Error Rate, Adaptive algorithm

I. Introduction

Fingerprint is a biometric that has been adopted for authentication in a wide range of applications [1]. The wide use of fingerprint as a biometric has opened a whole set of security challenges and different modes of attack as detailed in [2], [3]. Fingerprint regeneration from templates [4], [5] has been demonstrated with reasonable accuracy and can be used for injection attacks. The compromise of template via one application affects directly the security of other applications where the user has given the same fingerprint. The compromise also leads to permanent inability of using the same fingerprint elsewhere in future [6]. Hence template protection schemes gain prominence. Encryption algorithms cannot be used for protecting fingerprint templates because matching has to happen in the decryption domain, at which point an attack may happen. Further the uncertainty of biometrics does not mend well with exact nature of cryptography [7].

The cancellable biometric technique introduced in [6] provides a means for achieving template protection. The cancellable fingerprint is generated by taking the raw features of the fingerprint and transforming it based on the user key input in a non-invertible manner and storing the transformed features in template. Given the transformed features and the user key, the adversary will not be able to regenerate the raw features of the fingerprint due to the non-invertible nature of the transformation. Several works in literature exists detailing various methods of generating cancellable templates. [7, 8, 9, 10, 11, 12, 13, 14]. Delaunay triangles based cancellable template generation has been proposed in [8], [9].

For matching, the query fingerprint is also transformed using the same user key and matching happens in the transformed domain between enrolled and query templates. The different approaches for development of matching algorithm are detailed in [1]. For matching cancellable templates, correlation based or minutiae based matching algorithms will not work because the minutiae information is not stored directly in the template. Generating bit strings is the general approach in generating cancellable templates [11]. Matching of bit strings is then done by measuring the Euclidean or Hamming distance between the bit strings. This approach does not look to match the biometric features rather the bit strings generated from non-invertible transformations of the biometric features are matched. Because of non-invertible transformation applied during generating bit strings, the templates lose their uniqueness. Hence matching based on distance measures does not work well. The authors propose a novel feedback based matching algorithm to measure the degree of match and generate a score. In the matching algorithm, a feedback parameter is introduced which enables the system to adjust itself according the population of cancellable templates. Since cancellable template is derived from two components; user key and fingerprint, the feedback parameter inherently analyses the two components of the population and adjusts the algorithm accordingly.

The paper is organized as follows. Section 2 lists the related works on cancellable fingerprint templates and their matching algorithms. Section 3 explains the cancellable fingerprint generation based of biometric features and user key input. Section 4 describes the novel feedback based adaptive matching algorithm that is being proposed. Section 5 describes how the matching algorithm adapts itself to the user population by means of the feedback parameter and how the feedback parameter can be calculated. Section 6 lists the experimental results that were obtained by the algorithm on publicly available databases. Section 7 provides the conclusion and the future direction for the work.

II. LITERATURE WORKS

The format for standard fingerprint templates is given by ISO-19794-2 [15]. Since raw biometric features like minutiae locations and angles are stored in standard templates, the information could be used for re-creating the images with reasonable accuracy [3], [4]. The recreated images can be used for injection attacks. Since biometric features cannot be changed, the compromised biometric would be termed unusable for that particular user. The need for cancellable biometrics is first introduced in [6]. The attacks that are possible on any biometric system are detailed as fish bone model in [2] and [3].

Early attempt to generate cancellable templates is presented in [16] using local texture features and a two factor key. Local minutiae features were used for biometric template generation along with user pin in [17]. Pair polar coordinates are used for cancellable template generation in [7] wherein the authors map the minutiae points to random sectors and then matching algorithm works by matching the points above a threshold. Zoned minutiae pairs and modulo operation are used to generate cancellable templates in [18]. The paper proposes generating binary bit strings from the fingerprints and matches the bit strings using a Euclidean distance based L2 norm. A fuzzy vault approach for template protection is adopted in [10]. Hadamard transform based approach is detailed in [11]. Though the Hadamard matrix is invertible, a partial matrix is chosen to achieve non-invertible transformation and matching is again based on Euclidean distance normalized with L2 norm. Minutiae based bit strings for cancellable template generation is proposed in [13] wherein tuple based quantization is used to generate the bit strings. Matching score between two bit strings are then obtained as intersection of the two bit strings. Instead of extracting features directly from minutiae points, Delaunay triangle construction on the minutiae points was suggested in [8] and [9]. In [8], they proceed to construct 1 D bit strings out of the features of the Delaunay triangles and apply a transformation based on user key to generate the final template. Matching of bit strings is through calculating distance between the bit strings. In [9], the local features which are transformed are quantized and those features in the same range are assigned the same label. Then the matching proceeds by calculating number of matching triangles between the enrolled and query templates.

Delaunay triangle based cancellable fingerprints seem to show great promise among the published works. In [8], generation of 1D bit string leads to lose of so much information extracted from the triangles. Further the matching algorithm just matches bit strings by calculating the distance between them. As a result, Equal Error Rate (EER) seems quite high. In [9], a more intuitive approach to matching is presented wherein the number of matching triangles is to be calculated. However, the quantization process based on an arbitrary parameter leads to lose of information. This reflects in the final result as increase in EER.

The work presented below adopts the feature set generation process based on Delaunay triangles from [8] and [9]. An additional feature that would be beneficial in matching called type of the triangle is introduced. For cancellable template generation, a user key input is got from the user and modulo operation is applied for non-invertible transformation. A novel matching algorithm that tries to match each triangle from query image with all the triangles in the enrolled image is presented here. A match score is generated for each match and the most perfect matching triangle from the enrolled image is selected. Based on a feedback parameter, which is derived from the population, number of matching triangles is calculated and a score is generated accordingly. The feedback parameter maximizes separability and EER, because the feedback parameter helps the algorithm in adapting to the population and the user keys chosen by the population.

III. CANCELLABLE FINGERPRINT GENERATION

Delaunay triangles can be constructed on a set of points such that no point comes inside the circumcircle of any triangle [19]. For extracting information from the minutiae points, Delaunay triangulation is chosen because of the structural stability [20]. It has to be noted that given as set of points, Delaunay triangulation is not unique and hence the elasticity of the fingerprint or missing minutiae would affect the triangulation. However, such effects tend to affect only the local triangles while other triangles tend to remain intact. The flow diagram for cancellable template generation is shown in figure 1.

During enrollment, from the given fingerprint image, minutiae points are extracted. Each minutiae point can be represented as

M_i={x, y, Ɵ, type}

Where,

x represents the X coordinate

y represents the Y coordinate

Ө represents the minutiae angle

type represents the type of the minutiae (Ridge Ending or Bifurcation)

From the set of minutiae points, Delaunay Triangles are constructed. From each Delaunay triangle, following information is extracted as feature sets.

$F.S_i$ = {d1, d2, d3, Ө1, Ө2, Ө3, a, t} where,

d1, d2, d3=sides of the triangle (arranged in ascending order)
Ө1, Ө2, Ө3=Minutiae angles at the vertices of the triangle (arranged in ascending order)
a= Area of the triangle
t=Type of the Triangle
t=1 if all the three vertices of triangle are all ridge endings
t=2 if two of the vertices of triangle are ridge endings, other vertex being a bifurcation
t=3 if only one of the three vertices of the triangle are ridge endings, other two are bifurcations
t=4 if all the vertices of the triangle are bifurcations

The reason behind arranging the sides of the triangles and minutiae angles in ascending order is during matching a triangle from query image with a triangle in enrolled image, corresponding features should get matched with each other. To match corresponding features, it becomes beneficial to maintain a similar arrangement of features in both enrolled and query feature sets. Hence ascending order of arrangement is preferred here. Any other arrangement scheme could be preferred as long it is applied both for enrolled and query features.
If there are N triangles in the enrollment fingerprint image, N feature sets will be generated.
F.S = {$F.S_1$, $F.S_{2,}$ $F.S_3$ ………….. $F.S_{N\}}$

For generating cancellable templates, the feature sets have to be transformed based on a user key in a non invertible manner. Modulo operation is chosen as the non invertible operation in this work. If the features are modulo operated with a user input, the features would not be recovered back even if transformed features and user input is available to the adversary. For doing this modulo operation an eight digit key can be got as input from the user. The choice of number of digits in the user key is arbitrarily chosen as eight. Lowering the number of digits would lead to lesser degrees of freedom and there might be cases where the majority of population will have the same user key. Increasing the number of user keys puts stress on the user's memory. Hence eight is chosen as an optimum number of digits to expect from the user. There is one more constraint on the user key inputs. Since the operation is going to be modulus, it would be meaningless to have 0 as digits in user key inputs. Hence users are requested to choose an user key of non zero 8 digits.

Let the i^{th} user's key be given as
$Userkey_i$={ Userkey[1], Userkey[2], Userkey[3], Userkey[4], Userkey[5], Userkey[6], Userkey[7], Userkey[8]}
The i^{th} Feature set is then transformed as follows.
d1 = d1 % (Userkey [1]*10 +Userkey [2]);
d2 = d2 % (Userkey [3]*10 +Userkey [4]);
d3 = d3 % (Userkey [5]*10 +Userkey [6]);
Ө1 = Ө1 % (Userkey [1]*10 +Userkey [2]);
Ө2 = Ө2 % (Userkey [3]*10 +Userkey [4]);
Ө3 = Ө3 % (Userkey [5]*10 +Userkey [6]);
a = a % (Userkey [7]*10 +Userkey [8]);

t remains unchanged

The transformed feature set is then stored as template.

Template = $\{F.St_1, F.St_2, F.St_3 \dots F.St_N\}$

Where F.St indicates transformed feature set.

Userkey is not stored anywhere in the system. Hence there is no way for the adversary to get the user key from the system database.

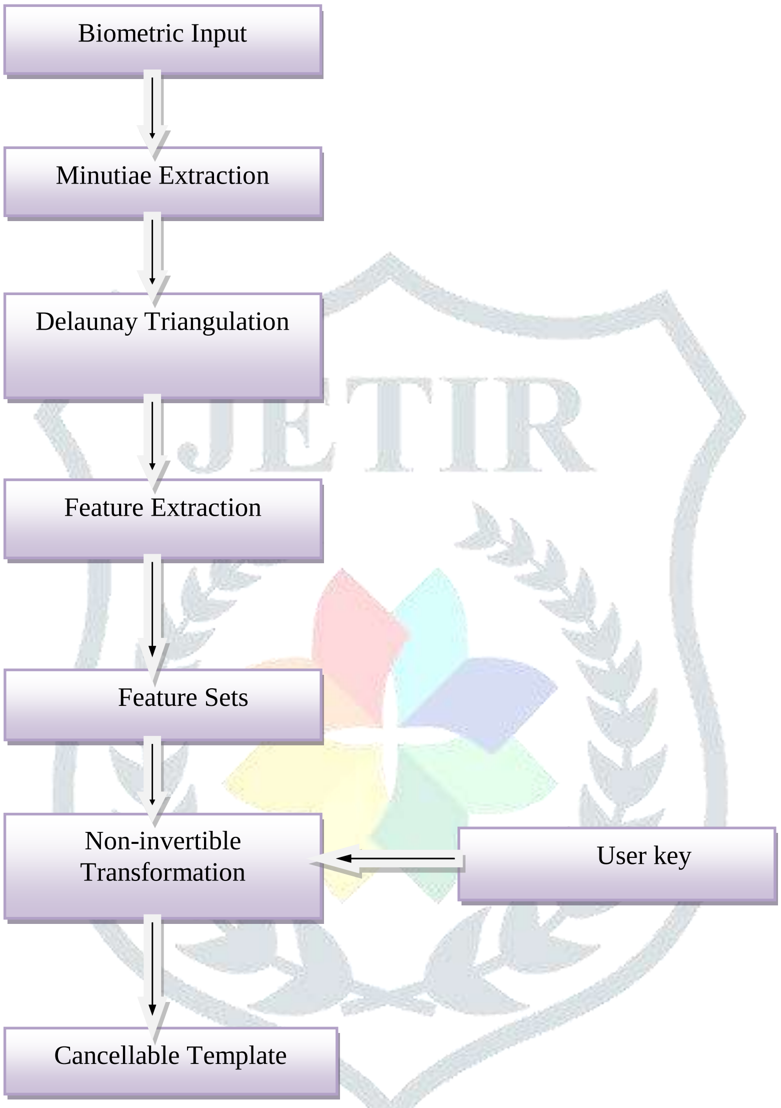


Fig. 1: Flow diagram for generation of cancellable template

IV. NOVEL FEEDBACK BASED ADAPTIVE MATCHING ALGORITHM

The matching between query and enrolled template is called as verification. During verification, the user presents the fingerprint along with userkey to generate the query template. The fingerprint presented is called as query fingerprint and transformed using userkey into query template. The matching algorithm has to perform verification and generate a score. The score can be either similarity score or difference score. The proposed matching algorithm here is designed to calculate difference score in the range of <0, 1>. A score of 0 indicates perfect match while a score of 1 indicates perfect mismatch. The block diagram for matching algorithm is presented in figure 2.

The proposed matching algorithm takes each triangle from the query template and matches the features with all the triangles from the enrolled template. Let M be the number of triangles in the query template and N be the number of features in the enrolled Template. The templates are nothing but transformed feature sets as given by

Query Template = {F.Sqt$_1$, F.Sqt$_2$, F.Sqt$_3$ … F.Sqt$_M$}
Where, F.Sqt$_i$ indicates the ith transformed feature set in query template.
Enrolled Template = {F.Set$_1$, F.Set$_2$, F.Set$_3$ … F.Set$_N$}
Where, F.Set$_i$ indicates the ith transformed feature set in enrolled template.

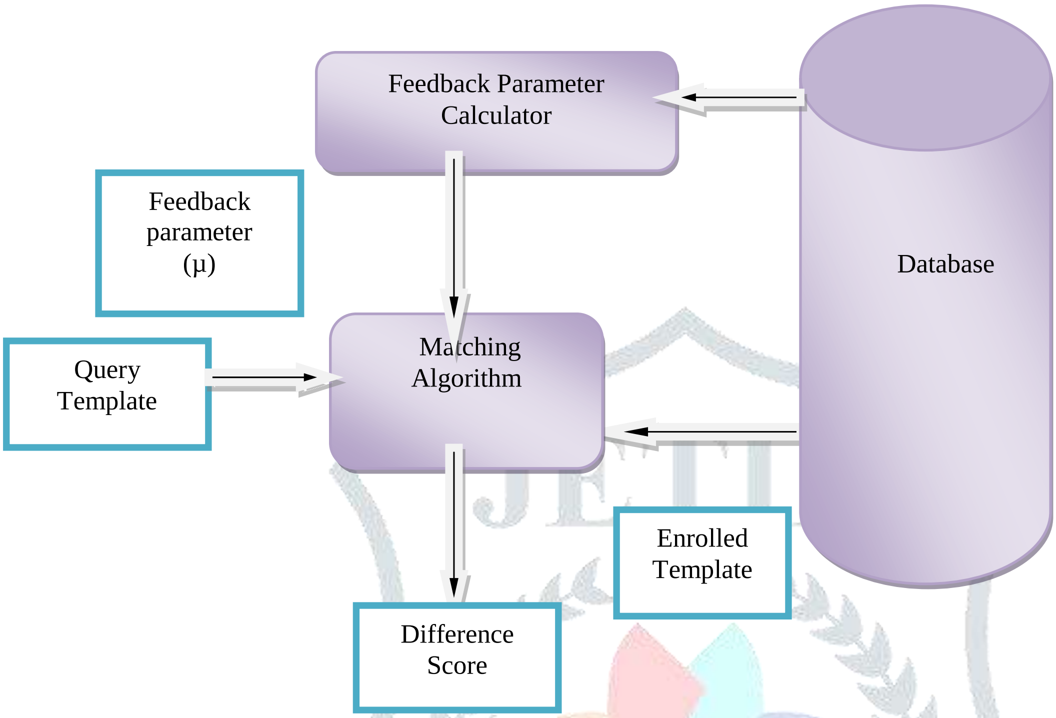


Fig 2: Feedback based Adaptive matching algorithm

A match score is generated for each match between triangles of enrolled and query templates.
Area difference = $|a_q-a_e|/\max(a_q,a_e)$
Minutiae difference = $|\Theta_q- \Theta_e|/\max(\Theta_q ,\Theta_e)$
There are three minutiae hence three Minutiae Differences namely,
Minutiae difference 1
Minutiae difference 2
Minutiae difference 3
Side difference= $|d_q-d_e|/ \max(d_q\ d_e)$
There are three sides hence three Side Differences namely,
Side difference 1
Side difference 2
Side difference 3
Type difference = 0 if ($|t_q-t_e|=0$) (The types of triangles are same)
= 1 if ($|t_q-t_e|=3$) (The types of triangles are completely different)
= 0.75 if ($|t_q-t_e| = 2$) (The types of triangles are different by 75 percent)
= 0.25 if ($|t_q-t_e| = 1$) (The types of triangles are different by 25 percent)

In the above a subscript of ‘q’ indicates that feature is from a triangle in the query template, a subscript of ‘e’ indicates that feature is from a triangle in the enrolled template. All the eight differences corresponding to eight features are then averaged to get a match score as shown below. | | indicates modulo operation to avoid negative values and max() indicates maximum function.

Match score= (Area difference + Minutiae difference 1 + Minutiae difference 2+ Minutiae difference 3 + Side difference 1 + Side difference 2 + Side difference 3 + Type difference)/8

The above calculation indicates the fraction by which each feature in the transformed query feature set varies from the corresponding feature in the transformed enrolled feature set. This is the reason why the features are arranged in ascending order during enrollment thereby making it convenient to compare corresponding features. Then these fractional differences are averaged to get the total difference. If two triangles match their match score would be low. If the triangles are very different, their match score would be very high. In any case, match score values are restricted between 0 and 1.

Each triangle in query template is matched with all the triangles in the enrolled template and for each matching, a score is generated. Hence for 1 triangle in query template, N scores will be generated. The minimum of all such scores indicates the closest match that the particular triangle in query template has been able to find in the enrolled template. This particular minimum score alone is taken and rest of the scores is ignored.

```
For each triangle in Query Template
        For each triangle in Enrolled Template
                Calculate Match score
        End
Pick the minimum of all the Match scores calculated for this particular triangle in query template
End
```

When the above operation is completed for all the triangles in the query template, there will be a list of M match scores wherein each score corresponds to the minimum of all the match scores obtained by comparing one triangle in query template with all triangles in enrolled template. This list of match scores is given below.

$M.S= \{M.S_1, M.S_2, M.S_3 \dots M.S_M\}$

Where $M.S_i$ indicates the match score obtained by taking the minimum of all the match scores by matching ith triangle in query template with all the triangles in the enrolled template.

Now a feedback based parameter (μ) is used to quantize the M.S into two levels either 0 or 1 as follows.

```
For each entry in M.S
        If $M.S_i <\mu$
                $M.S_i =0$;
        Else
                $M.S_i =+1$
        End
End
```

The feedback based parameter, is derived from the population distribution. Intuitively the feedback parameter gives the tolerance below which a particular triangle in the query template be considered able to find a match in the enrolled template and hence a value of 0 is assigned. If the match score is greater than this feedback parameter, it is deemed that there is no match for that particular query triangle in the enrolled template and hence a value of 1 is assigned. The final match score is calculated by taking the average of the match scores. If the number of matching triangles are more, there would be more 0s in the M.S else more 1s and accordingly final score will be less than or more than the threshold.

V. CALCULATION OF FEEDBACK PARAMETER

The feedback based parameter μ is calculated from the population as follows. Since the feedback parameter is intuitively the tolerance for triangles to vary, it could be assumed that triangles would normally vary by 10 % and hence μ=0.1 could be taken as the initial value. During enrollment, two impressions of the same finger of each user are taken. The user key is got from the user and the both the impressions are converted into templates by transformations. These two templates are matched using the matching algorithm presented in section 4 using initial μ =0.1. This would lead to the generation of the genuine score. As all the users enroll into the system, more genuine scores could be generated. As all users finish enrolling, the system would have accumulated as many genuine scores as number of users in the system.

In an offline manner, imposter scores are calculated by comparing one user's template against another. With the set of genuine and imposter scores a distribution graph is drawn and separability is calculated using d prime parameter as given below.

$$Separability = \frac{|mean(genuine)-mean(imposter)|}{\sqrt{\sigma(genuine)^2+\sigma(imposter)^2}} \qquad (1)$$

Where σ indicates standard deviation. Separability indicates the ability of the system to separate between genuine and imposters.

Then varying μ, the procedure is repeated again; this time enrollment again is not necessary, the genuine scores can be calculated simply changing μ in the matching algorithm. Imposter scores could be calculated similarly. As μ varies separability also varies and reaches maximum at particular value of μ. That value is chosen as the optimum feedback parameter because the algorithm is able to attain maximum separability at that value.

$$\frac{d(Separability)}{d\mu} = 0 \; at \; \mu = \mu\text{opt} \qquad (2)$$

Calculation of feedback parameter has the following advantages

- Since the cancellable templates are transformed using user key and the feedback parameter is derived from the cancellable templates, it inherently compensates for key distribution among the population. Even if the entire population has chosen the same user key, the algorithm would adjust itself to maximize its separability.
- The feedback parameter can be calculated in an offline manner whenever system performance degrades. As many new users get added on to the system or the existing users re-enroll themselves using different key which is the case in cancellable biometrics, the feedback parameter can be calculated offline in an automatic manner. No intervention is required.
- The problem of losing unique features of fingerprint due to the cancellable process is compensated to an extent by taking the feedback from the population. Specifically, by taking the feedback from the population, the algorithm adjusts itself to the templates it has rather than relying on the unique features of fingerprint to achieve performance

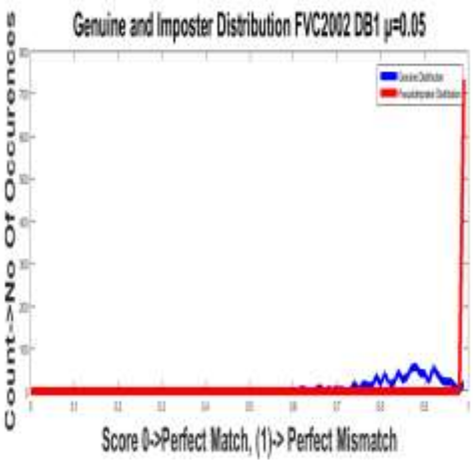
VI. EXPERIMENTAL RESULTS

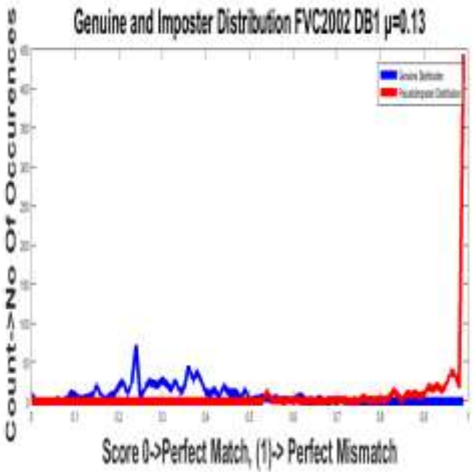
6.1 Separability

The following experiments were conducted on FVC 2002 DB1 DB2 and DB3 databases [21]. Each database consists of 100 fingers with 8 impressions of each finger. For enrollment and generating genuine scores, each impression is treated as enrollment impression and is matched with all other impressions of the same finger and the impression with least score is chosen as the enrollment pair. This matching is performed with initial μ=0.10. This would ideally be the case in real world environment wherein several impressions of same finger would be taken for enrollment and the best two would be selected. . This approach will lead to 8 scores per finger and hence 800 genuine scores. For generating imposter scores, each impression of a finger is matched against all impressions from the next finger and the match with highest score is taken as imposter score. This would generate 8 imposter scores per finger and 800 imposter scores in total.

For extracting minutiae from the fingerprint images, Fingerprint Minutiae Viewer (FpMV) software from NIST is used [22]. To ensure only high quality minutiae are taken for enrollment and matching, minutiae points with quality higher than 0.60 are considered. Once minutiae points are extracted, Delaunay triangles are constructed, and the feature sets are extracted from the triangles.

An ideal case in a population would be when each enrollment would be done with a different user key. This is evaluated by using different random numbers for each enrollment and applying modulo operation on feature sets to generate the template. Once the templates are generated, the feedback parameter is varied and genuine and imposter scores are generated. Distribution graphs are plotted for various values of μ for DB1 as shown below.





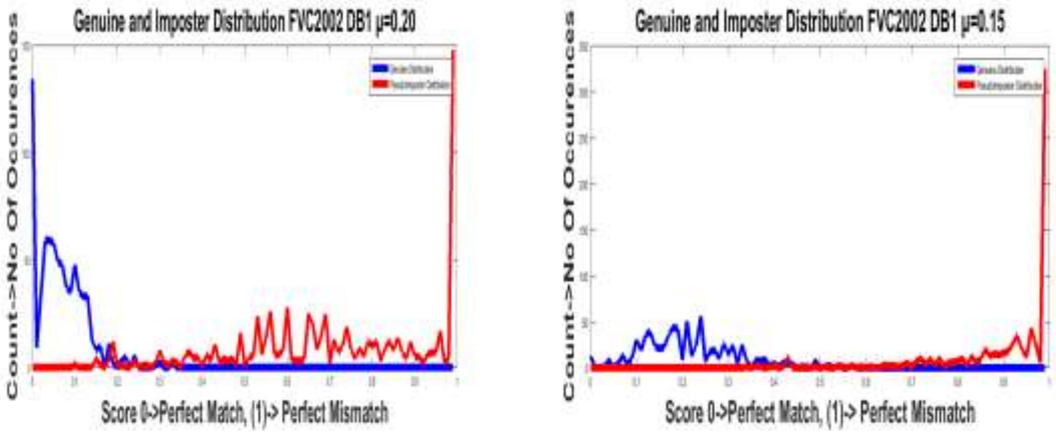


Fig. 3: Clockwise from top left. a) Distribution graph for μ=0.05 b) Distribution graph μ=0.13 c) Distribution graph for μ=0.15 d) Distribution graph for μ=0.20

Figure 3 illustrates the distribution graph for different values of the feedback parameter. When μ=0.05, the tolerance is very low and hence both genuine and imposter graphs are stacked up towards the imposter end of the graph. As μ increases, the genuine distribution separates itself from the imposter distribution. On further increase of μ=0.20, the tolerance becomes too high and the genuine side of the graph starts accommodating some imposter scores as well. The above movement of distribution graphs suggests that there is an optimum value of μ for which the genuine and imposter graphs are well separated. In this work, Separability is measured using the following two parameters d-prime [22] and Battacharya Coefficient [23]. Higher the d prime value indicates better separability. Battacharya coefficient lies between 0 and 1 and 0 indicate perfect separation of the distributions whereas 1 indicates same distribution. For various values of μ, these two parameters are calculated and tabulated in table 1.

Table 1: Separability parameters versus feedback parameter for FVC 2002 DB1 Different Key Scenario.

Feedback Parameter (μ)	d-prime	Bhattacharyya Coefficient
0.05	2.0589	0.5113
0.10	3.9974	0.1178
0.13	5.7621	0.0155
0.14	**6.1016**	**0.0095**
0.15	5.9933	0.0110
0.18	4.2488	0.0801
0.20	4.0137	0.0921

As can be seen from the bold row in table above, separability is maximum at a particular value of μ=0.14.

Figure 4 and Figure 5 illustrate the maximizing of separability.

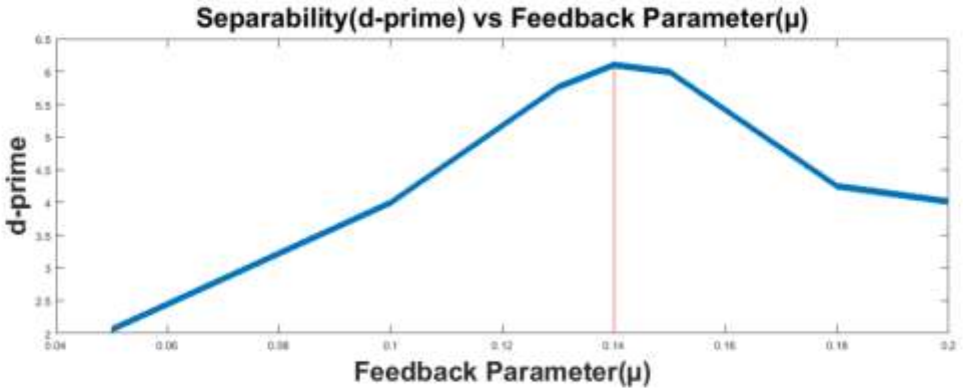


Fig 4: Separability (d-prime) vs Feedback parameter (μ) for FVC 2002 DB1

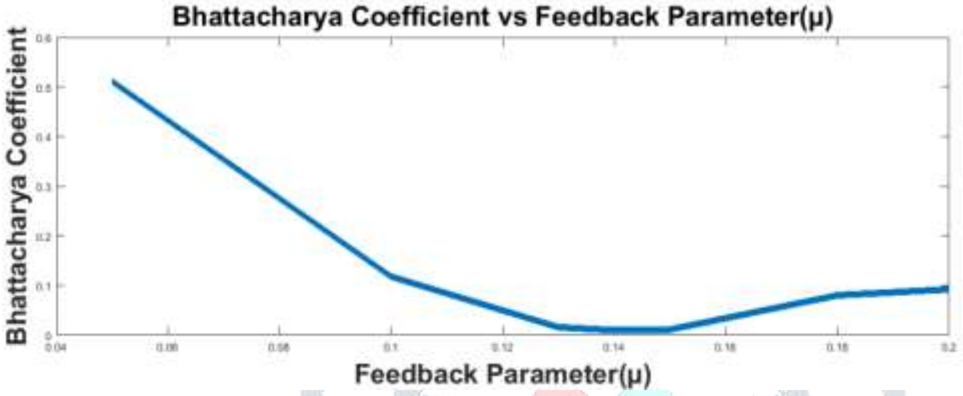


Fig 5: Separability (Bhattacharya Coefficient) vs Feedback parameter (μ) for FVC 2002 DB1

The algorithm is evaluated for the case when the entire user population possesses the same key. This is made possible by using the same key for all the users during enrollment. Intuitively, the separability is expected to decrease because the feature sets of all users will be transformed by the same user key. However, as the user key distribution has changed, the optimum feedback parameter would also change as illustrated in table 2.

Table 2: Separability parameters versus feedback parameter for FVC 2002 DB1 Same Key Scenario.

Feedback Parameter (μ)	d-prime	Bhattacharyya Coefficient
0.05	1.1373	0.8499
0.08	2.8407	0.3240
0.10	**3.5188**	**0.2111**
0.15	2.7060	0.3584
0.25	1.7859	0.4720

As seen from table 1 and table 2, because of the choice of user keys, the separability changes and the optimum feedback parameter changes as well. The separability drops when all the user keys are the same which is expected by intuition. The optimum feedback parameter which gives a measure of tolerance also drops from 0.14 to 0.10.

The same exercise is repeated for FVC 2002 DB2 and DB3 and the feedback and separability parameters for different and same key scenario are tabulated in table 3 and 4 respectively.

Table 3: Separability achieved for Different key scenario in FVC 2002 DB1, DB2 and DB3

Different Key Scenario

FVC2002	DB1	DB2	DB3
Feedback parameter (μ)	0.14	0.14	0.16
d prime	6.1016	5.2461	5.0792
Bhattacharya Coefficient	0.0095	0.0320	0.0392

Table 4: Separability achieved for same key scenario in FVC 2002 DB1, DB2 and DB3

Same Key Scenario			
FVC2002	**DB1**	**DB2**	**DB3**
Feedback parameter (μ)	0.10	0.09	0.13
d prime	3.5188	2.9764	3.2757
Bhattacharya Coefficient	0.2111	0.3149	0.2538

Performance comparison of the proposed algorithm with algorithms in literature is presented in table 5. The separability of the proposed algorithm for DB3 is higher in both same key and different key scenarios than other algorithms in literature.

Table 5: Separability Comparison of other algorithms and proposed method on FVC2002 DB3

Separability (d prime)	Same Key Scenario	Different Key Scenario
	FVC 2002 DB3	**FVC 2002 DB3**
Ahmad et al.[7]	1	-
Moujahdi et al.[24]	-	-
Sandhya et al (FS_INCIR) [8]	1.45	3.24
Sandhya et al. (FS_AVGLO)[8]	1.56	2.52
Proposed Method	**3.2757**	**5.0792**

6.2 EQUAL ERROR RATE (EER)

Equal Error Rate is defined as the error where False Acceptance Ratio (FAR) and False Rejection Ratio (FRR). Due to the cancellable nature of the algorithm, EER has suffered an increase in literature works. Figure 6 illustrates the Error Rate Curve (EER) drawn for FVC 2002 DB3 in same key scenario at optimum feedback value of μ=0.13. The EER achieved is 0.0756 which is comparable with other cancellable fingerprint algorithms available in literature.

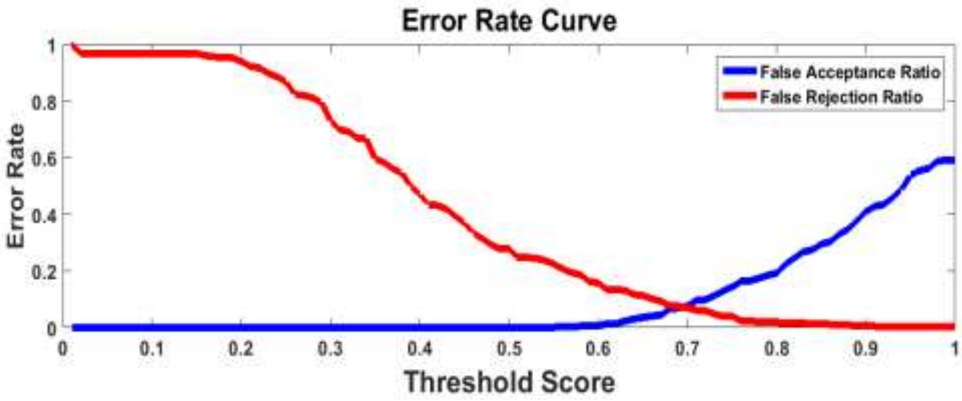


Fig 6: Error Rate Curve for Same Key scenario for FVC 2002 DB3 at μ=0.13

The EERs for FVC 2002 DB1, DB2 and DB3 for same key and different key scenarios is tabulated in table 6.It has to be noted that while the algorithms in literature take two out of eight impressions of each finger from database and calculates EER, the proposed algorithm was applied on all 8 impressions of the finger. All eight impressions of a finger are not alike. They have been purposely taken at different angles and image intensity also varies. In spite of that, the EER achieved by the proposed algorithm is comparable with algorithms in the literature. For different key scenario, 2.04%, 2.17% and 1.9% is obtained for DB1, DB2 and DB3 respectively.

Table 6: EER Comparison of other algorithms and proposed method on FVC2002 DB1, DB2, DB3

EER %	**FVC 2002**		
	DB1	**DB2**	**DB3**
Ahmad et al[7]	9	6	27
Wang and Hu [25]	3.5	4	7.5
Yang et al[9]	5.93	4	-
Sandhya et al. (FS_AVGLO)[8]	3.96	2.98	6.89
Proposed Method	**3.64**	**5.51**	**7.56**

6.3 REVOCABILITY ANALYSIS

Revocability is defined as the ability of the algorithm to differentiate between same fingers with different user keys. On a template compromise, the user will revoke his enrollment in the system and re-enroll again using a different user key. This revocability measure also assures the security of the algorithm against injection attacks using stolen fingerprint impressions but different user keys. As user key is not stored anywhere in the system, it cannot be stolen. To evaluate this parameter, the enrollment procedure is conducted again using different random user keys Such enrolled templates are called as pseudo imposter templates. Then earlier enrolled templates were matched against recently enrolled templates with different user keys. The distribution graph obtained is shown in figure 7.

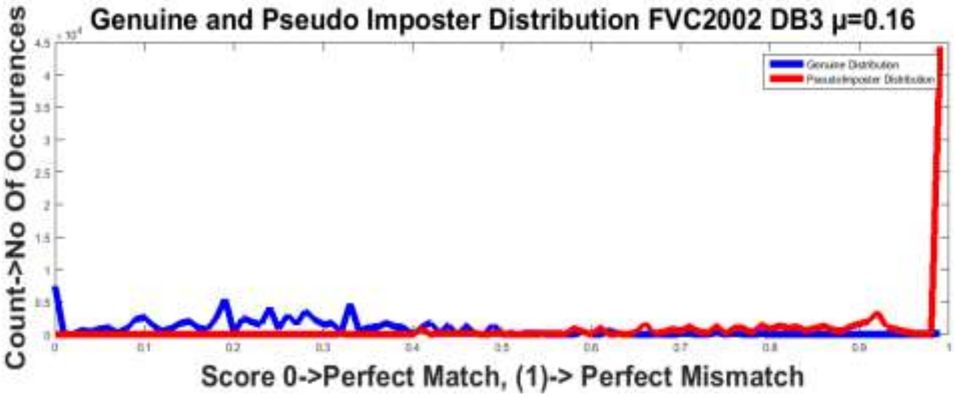


Fig 7: Distribution graph for Genuine and Pseudo Imposter Templates for FVC 2002 DB3 at μ=0.16

As can be seen from figure 7, the pseudo imposter distribution is well separated from the genuine distribution; hence by changing the user key, entirely different templates can be generated for the same fingerprint. The d-prime measure of separability that was achieved was 4.5499 which are lower than different key scenario but higher than same key scenario. The EER obtained was 0.024 and is considerably lower than same key scenario. This analysis proves that algorithm has good revocability and the matching algorithm would successfully differentiate between genuine and pseudo imposters.

6.4 SECURITY ANALYSIS

Since User key is not stored anywhere in the system, there is no way for the adversary to get the user keys of any user from the system. A brute force attack on the key would take 9^8= 43,046,721 attempts. Even with the possession of correct user key and the template, the feature sets cannot be arrived at accurately because of the non-invertible nature of the modulo operation. Since the templates does not contain any information regarding the position of the minutiae, hill climbing attacks which generate synthetic templates to attack the system would find it difficult without user key. Due to two parameters, say feature set and user key, are joined to create the template, a template level hill climbing attack will be very hard. If the pre-transformed feature set is somehow got from the channel of the system, this can be used for injection attack on the system. In that scenario, the system would fail as both user key and feature set is in the hands of an adversary. However, the adversary will still not be able to regenerate the finger print features and hence the user can simply revoke the enrollment and re-enroll into the system.

VII. CONCLUSION AND FUCTURE WORK

The cancellable templates have two components; say features from the fingerprint and a user key input. Due to non-invertible transformation of the features, and no constraint on the user keys chosen by the population, the matching algorithms lose their ability to differentiate between genuine and imposter templates. In such a situation, the proposed algorithm takes feedback from the population of templates and adjusts itself to maximize separability. Experiments were carried out on entire dataset of FVC2002 DB1, DB2 and DB3. Since all impressions were used for enrollment and matching, the results obtained are truly general and comparable with existing works in literature. In some cases, the proposed algorithm outperforms the existing works.

Since there is no constraint on the user keys chosen by the population the algorithm suffer the loss of separability. It has to be evaluated if it would be possible for the system to suggest user keys to the user during enrollment like how usernames are suggested in online enrollment systems. If different keys are used to enroll different users it would truly increase the separability of the algorithm and would make the templates more secure.

REFERENCES

[1] Davide Maltoni, Dario Maio, Anil K.Jain Salil Prabhakar “Handbook of Fingerprint Recognition” Second Edition, Springer ISBN: 978-1-84882-253-5

[2] Anil.K.Jain, Arun Ross, Sharath Pankanti “Biometrics: A tool for Information Security” IEEE transactions on information forensics and security, vol 1, no 2, june 2006

[3] Anil K. Jain, Karthick Nandakumar, Abhishek Nagar “ Biometric Template Security” EURASIP Journal on Advances in Signal Processing, Vol 2008, doi:10.1155/2008/579416

[4] Raffaele Cappelli, Alessandra Lumini, Dario Maio, Davide Maltoni “Fingerprint Image Reconstruction from Standard Templates” IEEE Transactions on pattern analysis and machine intelligence , vol 29, no 9, sept 2007

[5] Arun Ross, Jidnya Shah, Anil K. Jain “From Template to Image: Reconstructing Fingerprints from Minutiae Points” IEEE Transactions on pattern analysis and machine intelligence vol 29, no 4, april 2007

[6] Nalini K.Ratha, Sharat Chikkerur, Jonathan H.Connell, Rudd M.Bolle “Generating Cacelable Fingerprint Templates” IEEE Transactions on pattern analysis and machine intelligence, vol 29, no 4, april 2007

[7] Tohari Ahmad, Jiankun Hu, Song Wang “Pair-polar coordinate-based cancelable fingerprint templates” Pattern Recognition Elsevier, 44 (2011) 2555-2564

[8] Mulagala Sandhya, Munaga V.N.K. Prasad, Raghavendra Rao Chillarige “ Generating cancellable finger print templates based on Delaunay triangle feature set construction” IET Biometrics, ISSN 2047-4938, 2016 vol 5

[9] Wencheng Yang, Jiankun Hu, Song Wang, Jucheng Yang “Cacnelable Fingerprint Templates with Delaunay Triangle – Based Local Structures” Cyberspace safety and security Springer pp 81-91, Lecture notes in computer science, vol 8300, Springer, Cham, 2013

[10] Tran Khanh Dang, Quynh Chi Truong, Thu Thui Bao Le, Hai Truong “ Cancellable fuzzy vault with periodic transformation for biometric template protection” IET Biometrics ISSN 2047-4938

[11] Song Wang, Jiankun Hu “A Hadamard Ttransform-Based Method for the Design of Cancellable Fingerprint Templates” 6th International Congress on Image and Signal Processing, 2013, CISP 2013

[12] Zhao Song, Cao Hai Wang, Li Heng Jian “ Chos-based Renewable and Privacy Preserving Binary Palmprint Phase Templates for Cancellable Palmprint Recognition” IEEE Conference on mInformation Engineering and Computer Science , 2009

[13] Zhe Jin, Andrew Beng Jin Teoh, Thian Song Ong, Connie Tee “ Fingerprint template protection with minutiae based bit string for security and privacy preserving” Elsevier Expert Systems with Applications 2012

[14] Mayada Tarek, Osama Ouda, Taher Hamza “ Robust cancellable biometrics scheme based on neural networks” IET Bioemtrics, ISSN 2047-4938

[15] “ISO/IEC 19794-2 Information Technology Biometric data interchange formats- finger minutiae data” second edition 2011-12-15

[16] Sharat Chikkerur, Nalini K.Ratha, H.Connel, Raud M.Bolle “Generating Registration- free Cancelable Fingerprint Templates” IEEE Conference on Biometrics: Theory, Applications and Systems 2008

[17] Chulhan Lee, Jeung-Yoon Choi,Kar-Ann Toh,Sangyoun Lee, Jaihie Kim “ Alignment-Free Cancelable Fingerprint Templates based on local minutiae information” IEEE transactions on systems ,man and cybernetcis vol 37, no 4, august 2007

[18] Song Wang, Wencheng Yang, Jiankun Hu “ Design of Alignment-Free Cancelable Fingerprint Templates with Zoned Minutia Pairs” Pattern recognition ,Elsevier ,2017

[19] Jacob E Goodman, Joseph O. Rourke “Discrete and Computational Geometry”, ISBN:0-8493-8524-5

[20] Manuel Abellanas, Ferran Hurtado, Pedro A. Ramos “Structural tolerance and Delaunay Triangulation” Elsevier Information Processing Letters 1999

[21] Fingerprint Verification Competition 2002 “http://bias.csr.unibo.it/fvc2002/”

[22] MacMillan N, Creelman C (2005) “Detection Theory: A User’s Guide.” Lawrence Erlbaum Associates.

[23] Bhattacharyya, A. (1943). "On a measure of divergence between two statistical populations defined by their probability distributions". Bulletin of the Calcutta Mathematical Society. 35: 99–109. MR 0010358.

[24] Chouaib Moujahdi, George Bebis, Sanaa Ghouzali, Mohammed Rziza “ Fingerprint shell:Secure representation of fingerprint template” Elsevier pattern recognition letters 2014

[25] Song Wang, Jiankun Hu “Alignment-free cancelable fingerprint template design: A densely infinite-to-one mapping (DITOM) approach, Eslevier pattern recognition, 2012