Evolutionary Methods for Generating Synthetic MasterPrint Templates: Dictionary Attack in Fingerprint Recognition

Aditi Roy, Nasir Memon, Julian Togelius
New York University Tandon School of Engineering, Brooklyn, NY, USA
ar3824@nyu.edu, memon@nyu.edu, julian@togelius.com

Arun Ross
Michigan State University, East Lansing, MI, USA
rossarun@cse.msu.edu

Abstract

Recent research has demonstrated the possibility of generating "Masterprints" that can be used by an adversary to launch a dictionary attack against a fingerprint recognition system. Masterprints are fingerprint images that fortuitously match with a large number of other fingerprints thereby compromising the security of a fingerprint-based biometric system, especially those equipped with small-sized fingerprint sensors. This work presents new methods for creating a synthetic MasterPrint dictionary that sequentially maximizes the probability of matching a large number of target fingerprints. Three techniques, namely Covariance Matrix Adaptation Evolution Strategy (CMA-ES), Differential Evolution (DE) and Particle Swarm Optimization (PSO), are explored. Experiments carried out using a commercial fingerprint verification software, and public datasets, show that the proposed approaches performed quite well compared to the previously known MasterPrint generation methods.

1. Introduction

A growing number of mobile devices like laptops, tablets or smartphones are utilizing automated fingerprint verification for user authentication. Often, the sensors embedded in these devices capture only a limited portion of the whole fingerprint and, therefore, multiple such *partial fingerprints* are acquired and stored for the same finger during enrollment [4]. Further, a user is often permitted to enroll the partial fingerprints corresponding to multiple fingers in an attempt to increase the usability of the system. The user is deemed to be successfully authenticated, if the input print acquired during authentication matches any one of the stored partial prints obtained during enrollment.

The vulnerability of such a partial fingerprint-based user authentication system was recently investigated by Roy et al. in [9]. Their work showed that it is possible to launch a dictionary attack with substantial success by using a set of carefully chosen "MasterPrints". Masterprints are partial fingerprint impressions that fortuitously match with a high proportion of other fingerprints corresponding to different subjects. Therefore, an adversary can generate a set of fingerprint spoofs inscribed with Masterprints in order, for example, to unlock the smartphone of a large number of subjects.

In [9], two approaches were proposed to generate MasterPrints, namely "Sampled MasterPrint" (SAMP) and "Synthetic MasterPrint" (SYMP). In the first approach, the Masterprints - known as SAMPs - were sampled from a fixed fingerprint dataset. In this case, each fingerprint in the dataset was matched with all other fingerprints in the dataset, and the ones with the highest Impostor Match Rates were selected as Masterprints. In the second approach, Masterprints were created synthetically by applying a first-order hill-climbing algorithm where the SAMPs from the first approachwere used as the initial seed in the hill-climbing process. However, this approach suffers from two major drawbacks. First, a rectangular grid of 9×9 pixel cells was used to prevent the system from creating multiple minutiae points too close to each other. Also for the minutia orientation, the $[0, 2\pi)$ range was quantized into 16 equally spaced intervals. These design decisions drastically reduced the number of possible minutiae in the search space. Second, local search algorithms such as hill-climbing sometimes either tend to result in local minima or take a long time to converge to an acceptable result.

These observations motivate the application of global search algorithms with minimum parameter tuning for generating synthetic MasterPrints. Evolutionary algorithms that provide global optimization of numerical, real-valued

problems for which exact and analytical methods do not apply, are ideal candidates. This paper presents three techniques, namely Covariance Matrix Adaptation Evolution Strategy (CMA-ES), Differential Evolution (DE) and Particle Swarm Optimization (PSO) for generating SYMPs. The methods were evaluated using an optical fingerprint dataset as well as a capacitive fingerprint dataset. Results show that much more effective MasterPrints can be generated using the proposed algorithms.

The rest of this paper is organized as follows. Section 2 introduces the metric used for evaluation of the dictionary attack accuracy. Section 3 describes the three techniques for generating synthetic MasterPrints. Then, in Section 4 we present detailed results and present our conclusions in Section 5 of the paper.

2. Measure of Security for Fingerprint-based Verification System

To quantify the efficacy of dictionary attacks, Bonneau et al. proposed *Marginal success rate* (λ_β) [2] in the context of password-based authentication system. Marginal success rate computes the probability that an attacker can correctly guess an unknown password x in β attempts. It is defined as follows:

$$\lambda_\beta = \sum_{i=1}^{\beta} p_i. \tag{1}$$

where the i^{th} password in an ordered database of N passwords has a probability of occurrence p_i and $p_1 \geq p_2 \geq \ldots \geq p_N$. However, as mentioned in [9], this metric cannot be used in biometric-based authentication as multiple biometric inputs can be jointly accepted with a probability based on the False Match Rate (FMR) of the matcher. So, the Marginal success rate has been redefined as follows in the context of the biometric authentication system.

Consider a population of N subjects $\mathcal{U} = \{U_1, ..., U_N\}$ in a fingerprint dataset $\mathcal{F}$. Let, T be the number of fingerprint templates for each subject in the dataset. Then, the dataset can be denoted as $\mathcal{F} = \{F_t^i | i \in \{1, \ldots, N\}, t \in \{1, \ldots, T\}\}$. A match with the i^{th} subject is declared if a MasterPrint ($\mathbb{M}$) matches with any one of the T templates corresponding to the subject U_i. Since a MasterPrint can match multiple subjects, the attack accuracy of a MasterPrint $\mathbb{M}$ is measured in terms of its "Independent Imposter Match Rate" ($IIMR$). $IIMR$ is formally defined as follows, where θ represents the matching threshold:

$$IIMR = \frac{1}{N} \sum_{i \in \mathcal{U}} (1 - \prod_{t \in T_i} (1 - \phi(\mathbb{M}, F_t^i))).$$

where

$$\phi(\mathbb{M}, F_t^i) = \begin{cases} 1, & \text{if } S(\mathbb{M}, F_t^i) > \theta \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Here, $S(\mathbb{M}, F_t^i)$ represents the match score between MasterPrint $\mathbb{M}$ and t^{th} template of subject U_i.

When multiple attempts are made by using a sequence of β distinct MasterPrints $\mathcal{M}^\beta = (\mathbb{M}_1, ..., \mathbb{M}_j, ..., \mathbb{M}_\beta)$, then the imposter match rate of a MasterPrint $\mathbb{M}_j$ needs to be computed by matching with only those templates corresponding to the subjects that were not already matched by other MasterPrints $\mathbb{M}_1$ - $\mathbb{M}_{j-1}$. The set of subjects matched with MasterPrint $\mathbb{M}_j$ is denoted as $\mathbb{U}_j$ ($\mathbb{U}_j \subset \mathcal{U}$). Then, Sequential IMR corresponding to j^{th} MasterPrint $\mathbb{M}_j$ in a sequence is defined as below:

$$SIMR_j = \frac{1}{N} \sum_{i \in \mathcal{U} - \mathbb{U}_{\mathbb{M}_{1,\ldots,j-1}}} (1 - \prod_{t \in T_i} (1 - \phi(\mathbb{M}_\text{J}, F_t^i))). \tag{3}$$

The modified Marginal Success Rate ($\lambda_\beta^{'}$) of a MasterPrint sequence $\mathcal{M}^\beta$ can then be defined as:

$$\lambda_\beta^{'} = \sum_{j=1}^{\beta} SIMR_j. \tag{4}$$

This metric is used in all our experiments to compute the attack efficacy of a sequence of β MasterPrints.

3. MasterPrint Generation

The proposed approaches for MasterPrint generation try to find the best set of synthetic MasterPrints from a search space by maximizing their $IIMR$ over a training dataset. Since searching over the entire space is intractable, local search techniques are applied to find a good solution. In this work, three different evolutionary algorithms are used to generate minutiae-based SYMPs. In the rest of this section, each of these techniques is presented in detail.

3.1. Pre-processing

To generate a synthetic MasterPrint, a predefined number (m) of MasterPrints (SAMP) are first sampled from the training dataset. To select the SAMPs, $IIMR$s for all the candidate prints are computed and the SAMPs with maximum $IIMR$s are chosen. This set of SAMPs is used to initialize the search space of the evolutionary algorithms. Each SAMP is represented as a parameter vector which describes a fingerprint template in terms of a set of minutiae. Each minutia is represented by a 2-D location and orientation. If a SAMP has n minutiae, then it is represented as a $3n$-dimension vector.

However, it has been shown in [13] that evolutionary methods frequently get stuck in local optima with very low fitness due to various reasons. First, the input to the algorithm may be a very high-dimensional vector and/or the state description may be ill-chosen. The authors found that

the presence of certain "poisonous" irrelevant inputs induces local minima in the fitness landscape-evolution and their removal may be required to find an optimal solution. In our context of MasterPrint generation, deletion of some minutiae points is required to get an optimum template.

A *masked descriptor* is used to indicate whether a minutia point will be considered during matching or not. If the value of the bit-mask is greater than 0.5, the corresponding minutia is deemed to be useful and used during matching. Otherwise matching is done without considering that particular minutia. Thus, the masked SAMP, which also includes the masked descriptor, is a $4\mathbb{N}$-dimensional vector, where $\mathbb{N}$ is the maximum number of minutiae allowed in a partial print template. In this study, $\mathbb{N}$ was set to 30 based on an empirical evaluation. During initialization of a $SAMP$ template, the bit-masks corresponding n minutiae ($n <= \mathbb{N}$) present in that fingerprint were set to 1. For the other ($\mathbb{N} - n$) minutiae, the position, orientation, and bit-mask were set to 0.

The inclusion of a bit-mask makes it possible to exclude any minutia from the fingerprint template during matching while maintaining a fixed sized feature vector. The initial set of m SAMPs computed by the process described above is used as the input to each of the algorithms to compute SYMPs outlined in the following sections.

3.2. Covariance Matrix Adaptation Evolution Strategy (CMA-ES)

The first approach to generate SYMPs is based on Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [6], which is often used for non-linear non-convex optimization problems in continuous domains as a reliable and robust optimizer for both local and global optimization. It has been shown to decrease time complexity by converging to an optimal solution in a few generations [5]. One other advantage is that it does not require tedious parameter tuning as the internal parameter selection strategy is automated.

First, m SAMPs are used to initialize the mean M as the weighted sum of the feature vectors in which the best members, in terms of fitness $IIMR$ value, are assigned higher weights. In each generation, CMA-ES executes the following steps until the maximum number of iterations is reached or the $IIMR$ of the best sample attains a predefined maximum value:

1. Population Generation, Selection and Recombination: The search for the global optima starts with the generation of λ offsprings by sampling a multivariate normal distribution around the mean M. Then, the selection is made by keeping μ best samples. The recombination of the evolutionary process is achieved by calculating the new mean vector for the current generation as the weighted average of best μ samples. The mean vector is updated in a way to maximize the likelihood of successful candidate solutions.

2. Covariance Matrix Adaptation: A covariance matrix represents pairwise dependencies between the variables in the distribution. The covariance matrix adaptation (CMA) is a method to update the covariance matrix of the distribution such that the likelihood of previously successful search steps is increased. The dependencies between the variables in the distribution are updated to learn a second-order model of the underlying objective function.

CMA-ES records two paths of the time evolution of the distribution mean, called search or evolution paths. These paths contain information about the correlation between consecutive steps. One path is used for the covariance matrix adaptation and helps a much faster variance increase in favorable directions. The other path is used to conduct an additional step-size adaptation described next.

3. Step size Adaptation: This adaptation tries to capture consecutive movements of the distribution mean. The step-size adaptation effectively prevents premature convergence yet allowing fast convergence to an optimum.

At the end of the iterations, the best offspring is returned as SYMP to be the best solution achieving highest $IIMR$ on the training dataset.

3.3. Differential Evolution (DE)

The second approach uses Differential Evolution (DE) [11, 12] that is a floating point encoded evolutionary algorithm for global optimization. The strength of the algorithm lies in its simplicity, speed, robustness, and requirement of negligible parameter tuning.

DE starts with an initial population, which is the set of the best SAMPs found from the training dataset. The feature vectors are also known as "target vector" in DE. The three basic steps of DE are as follows: in the "mutation" stage, a new "mutant vector" is generated for each target vector by adding the weighted difference between two randomly selected vectors from the previous iteration. The method of creating the mutant vector differentiates between the various DE schemes. The DE/current-to-best/1 [11, 12] mutation strategy is applied in the current work.

This mutated vector's parameters are then mixed with the parameters of the corresponding "target vector" from the previous iteration to yield a "trial vector". This parameter mixing stage is termed as "crossover". There are two types of crossover schemes, namely "Exponential" and "Binomial". We used the "Exponential" technique. Finally in the "selection" stage, if the $IIMR$ of the trial vector is better than the IIMR of the target vector, the target vector is replaced with the trial vector.

This evolutionary cycle is repeated for every sample of the population to generate a new population. Successive generations are produced until meeting the predefined termination criteria. The best target vector of the final generation is selected as the output SYMP.

3.4. Particle Swarm Optimization (PSO)

The third and last approach is Particle Swarm Optimization (PSO) [3, 7], where the population of potential solution candidates is represented by a set or swarm of particles m moving in a D-dimensional, real-valued search space of possible solutions. Every particle has a position vector encoding a candidate solution to the problem and a velocity vector directing its movement.

At the beginning, the selected SAMPs are used to initialize the particles, and the velocities are initialized randomly. In every iteration, the fitness value of each particle is evaluated by computing the $IIMR$. The particles are updated by using two "best" fitness values. The first one is the best fitness value obtained so far by the particle, the corresponding position is termed as "personal best". The other best value is the best fitness value for the whole swarm, which is called the "global best". After finding the two best values, each particle updates its velocity and position towards the best known positions in the search-space. This is expected to move the swarm towards the best solutions.

In each iteration, if the $IIMR$ of the current particle improves from its previous best $IIMR$, the current best particle is replaced with the new particle. Moreover, if the $IIMR$ of the current particle is better than the previous global best $IIMR$, the global best particle is also replaced with the new particle. The particles are modified till the maximum number of iterations is reached, or the $IIMR$ attains a predefined maximum value. Finally, the particle with maximum $IIMR$ is selected as the $SYMP$.

4. Experimental Results

In this section, we evaluate the performance of the proposed approaches. Our aim is to evaluate how well the MasterPrints match with target subjects as a function of the number of allowed attempts, the number of impressions per finger, etc. Experiments were carried out using optical as well as capacitive fingerprint dataset. The proposed approaches were also compared with the baseline hill-climbing method in [9]. Depending on the dataset used, an average improvement of 8-15% was achieved with respect to the baseline method.

4.1. Dataset Description

The Authentec AES3400 dataset [8] and FVC 2002 DB1-A dataset [1] were used in the experiments in order to facilitate comparison with the work by Roy et al. [9]. The first dataset consists of 8640 capacitive fingerprint images from 720 fingers, each having 12 impressions. The fingerprints in this dataset are partial in nature. The other dataset contains 8 full fingerprints of size 388×374 from 100 subjects, for a total of 800 fingerprints. The 8220 partial prints of size 150×150, created by [9] from this dataset and made available to us, were used for evaluation. Training and test sets were produced by dividing each dataset into two disjoint groups each containing 50% of the fingers. The training datasets were used to generate the synthetic Masterprints, while the test sets were used for evaluating their attack efficacy.

4.2. Experiment Design

The commercial fingerprint verification software Verifinger 6.1 SDK was used in this work. The experimental setup was similar to [9] in order to compare the proposed approaches with the baseline hill-climbing approach. Evaluation was performed using three different threshold settings corresponding to False Match Rate (FMR) values of 1%, 0.1% and 0.01%.

We performed "finger-level comparison" to compute the MasterPrint attack success rate using synthetic MasterPrints. Specifically, the Masterprint dictionary was constructed by determining a sequence of 5 distinct MasterPrints that sequentially tries to increase the probability of a successful match.

The attack accuracy of a MasterPrint was measured in terms of $SIMR$ that computes the number of fingers against which the MasterPrint was successfully matched. Since the subjects in the two datasets used in this work had impressions from only one finger, the term "subject" and "finger" are used interchangeably in the rest of the paper. We report the evaluation results using the metric Marginal success rate $\lambda_{\beta}^{'}$ of Eq. 4 that measures the attack efficacy in β attempts using distinct MasterPrints.

Detailed results are described below.

4.3. SYMP Generation

SYMPs were generated from the training datasets by applying the methods described in Section 3. The initial population was created from the 75 best-performing MasterPrints sampled from the training dataset. Thus, we ensured sufficient dissimilarity in the initial population. In CMA-ES, 30 offsprings (λ) were generated in each iteration. In PSO, we started with a large inertia weight of 0.9 for an initial bias towards the global search and decreased it linearly to a minimum value of 0.4 through different iterations to facilitate local explorations [10]. A new population was generated until the $IIMR$ did not change or changed negligibly over 100 iterations. It was observed that the $IIMR$ stabilized after 250–400 iterations.

4.4. Results on FingerPass Capacitive Dataset

Figure 1 shows the minutiae distribution of 5 SYMPs generated by three proposed methods. Figure 1(a) shows the minutiae distribution of 5 best SAMPs from the initial population $\mathbb{S}$. It can be observed that most of the minutiae

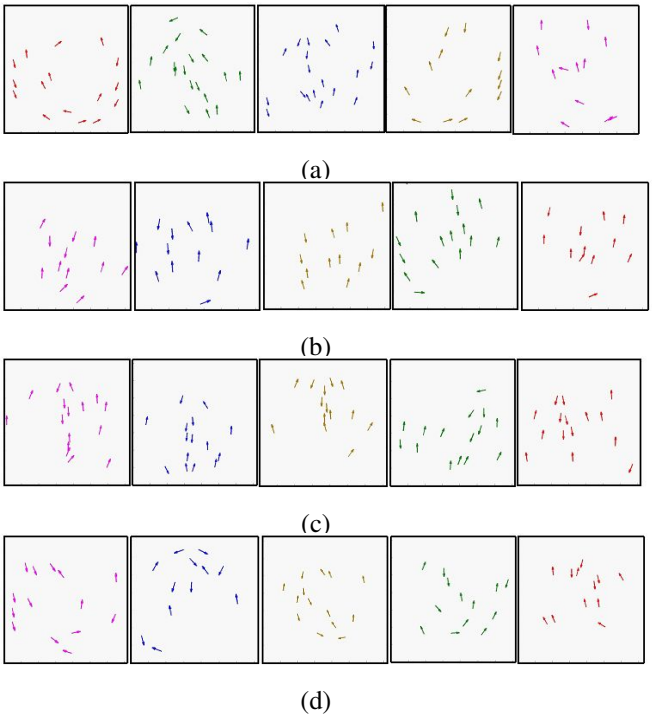


Figure 1: Minutiae location in 5 MasterPrint templates from the FingerPass DB7 dataset: (a) SAMP (b) CMA-ES SYMP (c) DE SYMP (d) PSO SYMP.

points present at the edges of the templates are automatically excluded from the synthetic templates. Since these minutiae are mostly spurious in nature, they have a lower contribution to matching performance. Therefore, they are not considered by most of the synthetic templates generated by the three proposed methods. Another observation is that the 5 SYMPs tried to create more minutiae in the core region. Since almost all the partial prints in the capacitive dataset contained the core area, the imposter match probability is highly affected by the minutiae from this region.

4.4.1 Performance with Different Number of Impressions per Finger

At the three different FMR settings, $SIMR$ values were calculated for each set of the 5 SYMPs corresponding to the 5 test trials. The average $SIMR$ is then reported over these 5 trials. Further, the average $SIMR$ was computed for a different number of fingerprint impressions per finger to study the effect of increasing stored impressions.

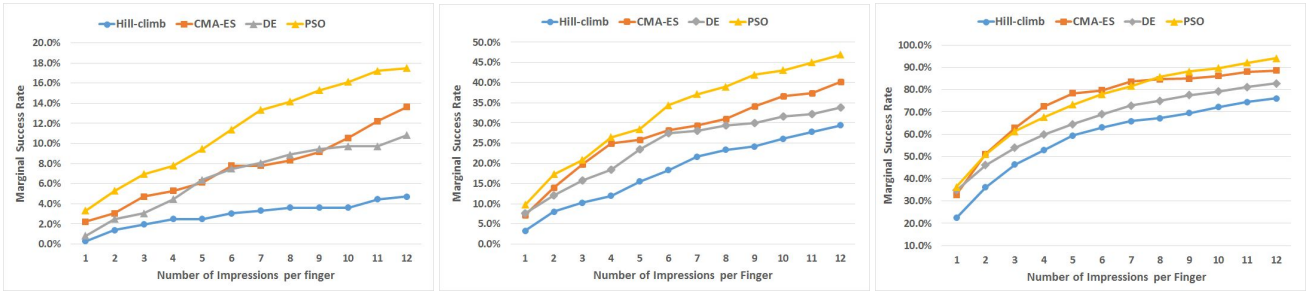
Figure 2 shows the combined average $SIMR$ or $\lambda_5^{'}$ of the SYMPs generated by CMA-ES, DE, and PSO techniques compared to the SYMPs created by the baseline hill-climbing method as a function of the number of partial fingerprints per finger in three different FMR settings. It can be observed that the SYMPs generated by the three proposed methods performed significantly better than the SYMPs generated by the hill-climbing algorithm, irrespective of the number of impressions per finger and the FMR setting. Among the three proposed techniques, PSO performed the best with a three-fold improvement over the baseline hill-climbing method at the highest security setting (FMR = 0.01%). The $SIMR$ using the PSO SYMPs increased to 17.50%, from a baseline performance of 4.72%, when using 12 impressions per finger (see Figure 2(a)). Further, it can be observed that the CMA-ES SYMPs performed better than the DE SYMPs, though not as good as PSO SYMPs. Using CMA-ES SYMPs, $SIMR$ improved by a factor between 2 and 3 compared to the hill-climb SYMPs. For example, at a 0.01% FMR, using 12 impressions per finger, the $SIMR$ increased to 10.83% compared to 4.72% using the hill-climb method. At lower security settings, CMA-ES performance was comparable with PSO, as can be seen in Figure 2(c).

Due to the high variance in the results, we ran a 1 way ANOVA on the four groups to test for statistical significance in the difference between the groups. For the FingerPass DB7 dataset, at FMR = 0.1%, using 8 impressions per finger, this test resulted in F = 11.87 with a p-value much lower than 0.05. Running Tukeys HSD post-hoc test shows more specifically that the difference between the hill-climb SYMP and the CMA-ES and PSO SYMPs are significant at p ¡ 0.05. This implies that we can be over 95% certain that the newly proposed CMS-ES and PSO SYMPs are better than the Hill-climb SYMP. However, it was found that the difference between the DE SYMP and hill-climb SYMP is not very significant thereby indicating the poor performance of the DE technique.

Analysis of the minutiae distribution of the SYMPs generated by the three techniques shows that the PSO SYMPs captured the minutiae distribution of the core region better than the other two techniques (see Figure 1). This distribution might explain their better matching rate. The minutiae points in DE SYMPs are more dispersed, thus leading to a lower match rate.

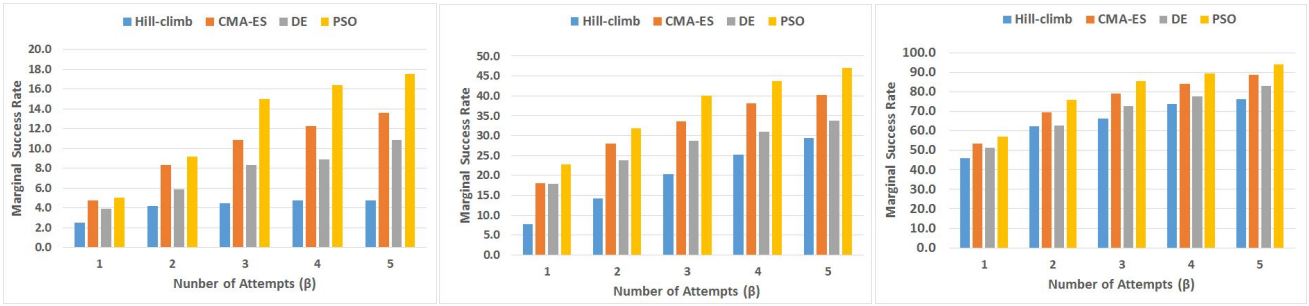
4.4.2 Performance with Different Number of Attack Attempts

Next, we study how the chance of a successful attack increases with the number of allowable attempts. Figure 3 summarizes the results for different security settings using SYMPs generated by the 4 techniques when the number of impressions per finger was fixed at 12. It can be observed that at a 0.01% FMR, with only one allowable attempt, the hill-climbing SYMPs from [9] matched with $\sim$ 2% of the subject population while the PSO SYMPs matched with 5% of the subject population. With increasing number of attempts β, PSO SYMPs consistently performed best. However, SYMPs generated by CMA-ES and DE also per-



(a) FMR = 0.01% (b) FMR = 0.1% (c) FMR = 1%

Figure 2: Marginal success rate ($\lambda_5^{'}$) variation as a function of the number of impressions per finger using the 5 sequential SYMPs from the FingerPass DB7 dataset. The SYMPs generated by CMA-ES, DE and PSO methods show considerable improvement in performance than the SYMPs generated by the hill-climbing approach.



(a) FMR = 0.01% (b) FMR = 0.1% (c) FMR = 1%

Figure 3: Marginal success rate ($\lambda_\beta^{'}$) variation as a function of the number of attempts β using 12 impressions per finger from the FingerPass DB7 dataset.

formed quite better than the hill-climbing SYMPs for any value of β. At five allowable attempts setting, PSO based MasterPrints were able to successfully attack $\sim 95\%$ of the subject population when FMR was set to 1%. These results demonstrate the effectiveness of the carefully designed MasterPrints in performing dictionary attack on capacitive fingerprint datasets.

4.5. Results on FVC2002 Optical Dataset

Figure 4 shows the minutiae distribution of the SYMPs generated from the FVC 2002 DB1-A dataset. Similar to the capacitive dataset, here also the five best SAMPs and SYMPs generated by the three proposed methods are shown. One important difference from the capacitive SYMPs is that here the minutiae points are not distributed around the core region. The nature of partial fingerprints in the two datasets is the main reason behind such a difference. Since the partial fingerprints of the optical dataset were uniformly cropped from a full fingerprint, the minutiae points are also distributed over full fingerprint locations. Further, it may also be noted that minutiae distribution in SYMPs is more centralized than the SAMPs. Similar to capacitive DE SYMPs, here also the minutiae points in DE SYMPs are more scattered than PSO or CMA-ES SYMPs.

4.5.1 Performance with Different Number of Impressions per Finger

To evaluate the performance, the average $SIMR$ values were calculated by averaging over 5 trials (cross-validation) at different FMR values using a set of 5 SYMPs on the FVC 2002 DB1-A dataset. Figure 5 shows the variation in the average $SIMR$ or Marginal success rate ($\lambda_5^{'}$) when the number of full fingerprints per finger is increased from 1 to 8. Here too, the sequential SAMPs created by the CMA-ES, DE and PSO techniques performed far better than the hill-climbing SYMPs of [9] . At a 0.01% FMR, the $SIMR$ the hill-climbing SYMPs ranged from 7.0% to 26.0% with 1 - 8 impressions per finger, while the $SIMR$ of the SYMPs created by CMA-ES ranged from 11.0% to 34.0%, DE SYMPs ranged from 10.0% to 31.0%, and PSO SYMPs ranged from 9.0% to 33.0%. On the optical dataset, the performance of the MasterPrints generated by CMA-ES method was found better than the PSO MasterPrints. At a 1% FMR, it was possible to attack all the subjects/fingers using a set of 5 PSO or CMA-ES generated SYMPs, even when each finger had only 2 full impressions.

As was done with the FingerPass DB7 dataset, the four types of SYMPs were also analyzed with a 1 way ANOVA for the FVC 2002 DB1-A dataset at FMR = 0.1% using 12

impressions per finger. This resulted in F = 6.01 with a p-value much lower than 0.05. This is a lower F-value than that of the FingerPass DB7 dataset but still significant. Running Tukeys HSD post-hoc test shows that both the CMA-ES and PSO SYMPs are significantly different than the hill-climb SYMP at p ¡ 0.05. As in the case of the FingerPass DB7 dataset, here too the DE SYMPs were not found to be significantly different than hill-climb SYMP.

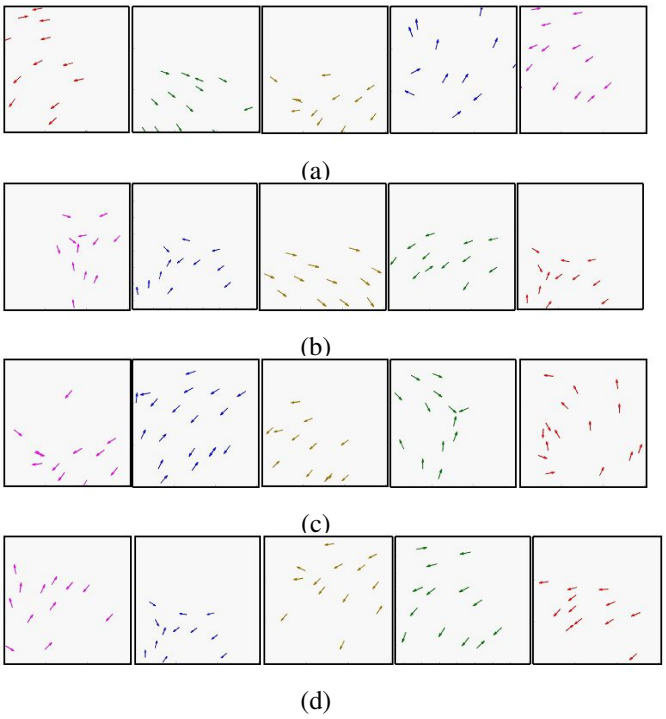


Figure 4: Minutiae location in 5 MasterPrint templates from the FVC 2002 DB1-A datase: (a) SAMP (b) CMA-ES SYMP (c) DE SYMP (d) PSO SYMP.

4.5.2 Performance with Different Number of Attack Attempts

Next, the average SIMR values were computed using an increasing number of SYMPs as the number of allowable attempts was increased from 1 to 5. Figure 6 shows variation in the Marginal success rate ($\lambda^{'}_{\beta}$) of SYMPs generated by CMA-ES, DE, and PSO techniques compared to the SYMPs created by the baseline hill-climbing method as a function of the number of attempts (β) in three different FMR settings using 8 impressions per finger. It can be observed from the figure that as the number of attempts increased, the chance of successful attack increased drastically. For example, at a 0.01% FMR, $\lambda^{'}_{\beta}$ rose from 7.0% in 1 attempt to 33.0% in 5 attempts, while using PSO generated SYMPs. At lower security setting (FMR = 1%), it was possible to successfully match all the subjects/fingers in only 2 attempts if a finger is represented by 8 impressions. It should also be noted that in each case, CMA-ES SYMPs performed best, closely followed by PSO SYMPs. This result is different from the capacitive dataset, where PSO SYMPs performed best.

5. Conclusions

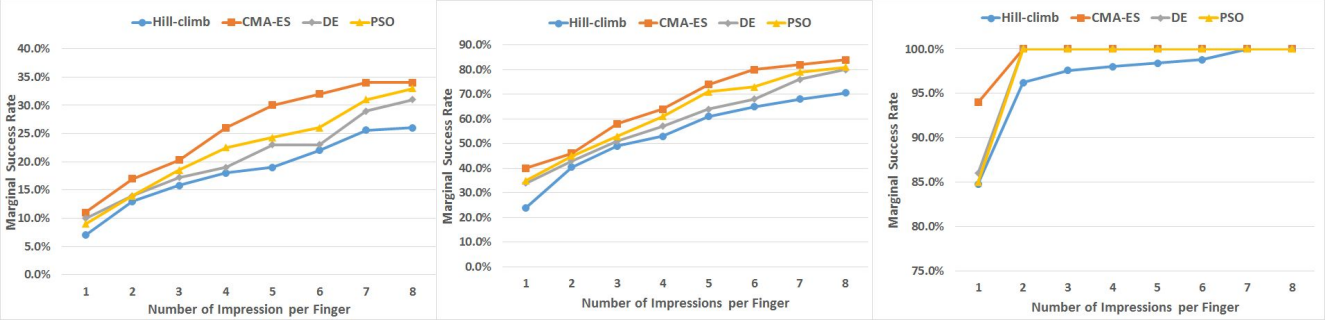
This article presents novel approaches for generating MasterPrints synthetically by applying evolutionary search algorithms with minimum parameter tuning. Since searching the entire space is intractable, local search techniques like CMA-ES, DE, and PSO were applied to find a better solution. Evaluation of the proposed approaches on the optical FVC 2002 DB1-A dataset and the capacitive FingerPass DB7 dataset were carried out. The main findings of this work are as follows:

- The work shows that it is possible to create synthetic MasterPrints that are far superior in attacking unknown subjects compared to the MasterPrints generated by the baseline hill-climbing method of [9]. With a dictionary of 5 MasterPrints, it was possible to attack 47% users in the FingerPass DB7 dataset and 84% users in the FVC dataset, at an FMR of 0.1%. On the capacitive dataset, the average improvement using the proposed techniques over the baseline method across all FMR settings was $\approx$ 15%, while on the optical dataset it was $\approx$ 8%.

- The CMA-ES and PSO techniques performed better than the DE method on both the datasets. On the capacitive dataset, PSO outperformed other methods; whereas, on the optical dataset, CMA-ES performed the best. The fundamental difference in the nature of partial fingerprints in the two datasets could be the reason behind this. When the partial prints have more minutiae in the core region, PSO performed best by capturing such a minutiae distribution. On the other hand, when the minutiae were more scattered, CMA-ES performed the best.

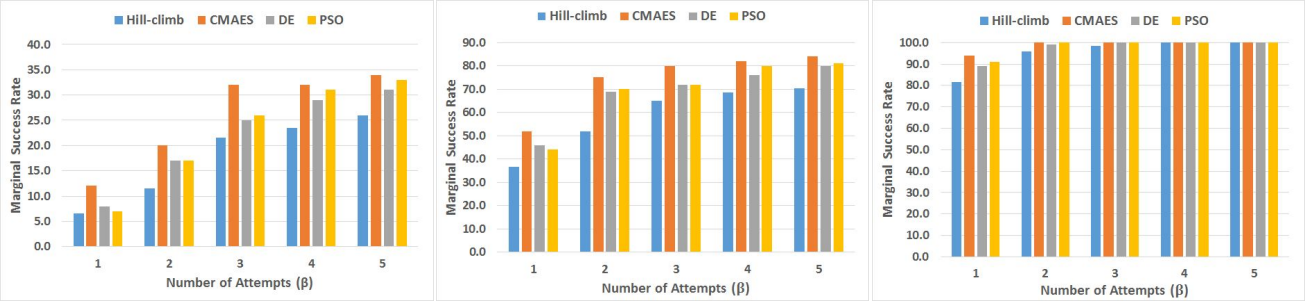
- Detailed analysis of the minutiae distribution of the SYMPs generated by the three proposed methods reveals that less important minutiae points (like the ones present at the edges of the templates) are automatically excluded from synthetic templates. New minutiae are added in the regions of the fingerprints that increase the chance of a match.

All the synthetic MasterPrint generation techniques presented in this article worked at the “template-level” by performing minutiae manipulation. In our future work, we would like to create MasterPrints at the “image-level” that can be used to launch a spoof-attack.



(a) FMR = 0.01% (b) FMR = 0.1% (c) FMR = 1%

Figure 5: Marginal success rate ($\lambda_5^{'}$) variation as a function of the number of impressions per finger using the 5 sequential SYMPs from the FVC 2002 DB1-A dataset.



(a) FMR = 0.01% (b) FMR = 0.1% (c) FMR = 1%

Figure 6: Marginal success rate ($\lambda_\beta^{'}$) variation as a function of the number of attempts β using 8 impressions per finger from the FVC 2002 DB1-A dataset.

References

[1] Fvc2002 database. `http://bias.csr.unibo.it/fvc2002/databases.asp`. Accessed June 2017.

[2] J. Bonneau, S. Preibusch, and R. Anderson. A birthday present every eleven wallets? the security of customer-chosen banking pins. In *Financial Cryptography and Data Security*, pages 25–40. Springer, 2012.

[3] R. C. Eberhart, J. Kennedy, et al. A new optimizer using particle swarm theory. In *Proceedings of the sixth international symposium on micro machine and human science*, volume 1, pages 39–43. New York, NY, 1995.

[4] B. B. Han, C. A. Marciniak, and W. C. Westerman. Fingerprint sensing and enrollment, Apr. 3 2014. US Patent App. 14/244,143.

[5] N. Hansen, S. D. Müller, and P. Koumoutsakos. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evolutionary computation*, 11(1):1–18, 2003.

[6] N. Hansen and A. Ostermeier. Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation. In *Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*, pages 312–317. IEEE, 1996.

[7] K. James and E. Russell. Particle swarm optimization. In *Proceedings of 1995 IEEE International Conference on Neural Networks*, pages 1942–1948, 1995.

[8] X. Jia, X. Yang, Y. Zang, N. Zhang, and J. Tian. A cross-device matching fingerprint database from multi-type sensors. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 3001–3004. IEEE, 2012.

[9] A. Roy, N. Memon, and A. Ross. Masterprint: Exploring the vulnerability of partial fingerprint-based authentication systems. *IEEE Transactions on Information Forensics and Security*, 12(9):2013 – 2025, 2017.

[10] Y. Shi and R. C. Eberhart. Parameter selection in particle swarm optimization. In *International Conference on Evolutionary Programming*, pages 591–600. Springer, 1998.

[11] R. Storn and K. Price. *Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces*, volume 3. ICSI Berkeley, 1995.

[12] R. Storn and K. Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.

[13] J. Togelius, T. Schaul, J. Schmidhuber, and F. Gomez. Countering poisonous inputs with memetic neuroevolution. In *International Conference on Parallel Problem Solving from Nature*, pages 610–619. Springer, 2008.