A Heuristic to Select the Optimal Transformation Matrixes in Bioconvolving with Mixing Transform

Francesco Castro, Stefano Galantucci, Donato Impedovo and Giuseppe Pirlo

May 9, 2023

A heuristic to select the optimal transformation matrixes in Bioconvolving with Mixing transform

Francesco Castro, Stefano Galantucci, Donato Impedovo and Giuseppe Pirlo
University of Bari Aldo Moro, Bari

Abstract

Bioconvolving with Mixing transform is a cancelable biometric approach to protect biometric data and the user's privacy. This approach uses linear convolutions on biometric features to generate cancelable templates following the random transformation matrixes. This paper shows how the choice of the transformation matrixes impacts the protected system accuracy. Therefore, random matrix selection is not an optimal strategy. A heuristic algorithm is proposed to select the optimal transformation matrix that achieves the optimal protected system performance. The proposed heuristic is based on the minimum distance between the transformed mean template created by the EB-DBA and the transformed reference set. Two online signature verification systems have been protected by Bioconvonlving with Mixing transform to evaluate the proposed algorithm performance in terms of accuracy, False Negative Rate (FNR), and False Positive Rate (FPR). The experiments have been conducted on SVC2004, xLongSignDb, SUSig VisualSubCorpus, and SUSig BlindSubCorpus online signature datasets. The highest calculated Pearson index (r=0.87) shows a high correlation between the proposed heuristic and the system's accuracy. Therefore, the selected matrixes by the proposed heuristic allow for optimal system performance. The protected system accuracy improved to 11% using the selected transformation matrixes by the proposed heuristic compared to the random selection matrixes. Moreover, protecting the system using Bioconvolving, revised with the proposed heuristic, reduces accuracy at best by only 0.6 % compared to the unprotected system.

1 Introduction

Biometric data must be protected for two main reasons: individuals' privacy and the usability of the data itself. If the biometric data is compromised, it will not be able to be used anymore since it can be neither revoked nor renewed. After the biometric data acquisition and feature extraction process, the biometric template is created and stored in the database as reference data to authenticate the user. Biometric data can be recovered from unprotected biometric templates, leading to severe privacy and security issues [1]. Therefore, several strategies have been developed to protect biometric templates [2]. Cancelable biometrics is a proposed methodology that involves intentional and repeatable transformations of biometric signals, providing a comparison of biometric patterns in the transformed domain, thus making it extremely complex to retrieve the original data. This strategy observes the unlikability, revocability, renewability, and non-invertibility requirements of ISO/IEC 24745 on biometric information protection [3]. However, the performance of the system protected with a cancelable biometrics approach is generally much lower than the performance of an unprotected system [4]. Several strategies based on cancelable biometrics are developed to protect different biometric data. Bioconvolving (BCV) [5] is proposed to protect biometric templates of online signatures. Recently, BCV has been used to protect different biometric traits such as the face, iris, palm print, fingerprint, and ear [6].

The signature is considered a “no-invasive” biometric trait, so it is among the most accepted biometric data. The purpose of the handwritten signature is individual identification, to secure the individual’s endorsement in a document. The handwritten signature is collected using specific electronic devices (i.e., tablet, PDA, or smartphone) to capture different information about the movement of a specific pen on the device [7]. The information commonly collected are the horizontal and vertical pen coordinates, the timestamp, and the pen pressure [8].

BCV consists of dividing the biometric features of each user into a fixed number of parts, and then the feature parts are combined through linear convolution to obtain protected templates. The protected templates reside in the same original template domain. Therefore, the same matcher used for original templates can be used for protected templates. The Bioconvolving with Mixing transform (BCV-MT) is a security strategy to protect the biometric template, but the obtained performance is only sometimes satisfactory. This paper has shown how the choice of the transformation matrixes dramatically impacts system performance. Therefore, the random matrix selection in BCV-MT cannot be considered an optimal strategy, and then a heuristic algorithm to select the optimal transformation matrixes is proposed.

The main research contributions are:

- Show the impact of the transformation matrixes on the protected system accuracy through the matrix distribution analysis.
- Develop a heuristic algorithm to select the optimal matrixes through the minimum distance between mean template and reference set.
- Verify the correlation between the proposed heuristic and system accuracy.
- Submit a comparative analysis of the obtained results by the BCV-MT with the selected matrixes through the proposed heuristic and the BCV-MT with the random selection matrixes.
- Compare the results obtained by the systems protected through the proposed algorithm with the unprotected system results to show the proposed algorithm's effectiveness.

This paper is organized as follows. Section 2 summarizes the main cancelable biometric approaches. Section 3 details the BCV-MT technique and describes the proposed heuristic. Experiments and results are provided in Section 4, and the conclusions and future development are detailed in Section 5.

2 Related work

The cancelable approaches are based on the intentional and repeatable distortion of the biometric data with the purpose of creating protected biometric templates. Several efforts have used different non-invertible transformation strategies to create protected biometric templates. Harkeerat Kaur et al. [9] have used a polynomial transformation to map biometric information. Other non-invertible transformation strategies used a projection matrix to project the feature vector into another feature vector of fewer dimensions. The projection matrix is generated by the user-specific key [10], [11]. To improve system security, random permutations have been used in addition to non-invertible functions. The permutation key is used to shuffle the feature vector values before applying the non-invertible function [12 - 14]. Recent efforts have been focused on developing approaches to reduce the tradeoff between performance, security, and privacy. Some efforts have adopted various hybrid approaches to improve recognition performance [15]. The hybrid approaches combine the increased performance of Biohashing with the increased security provided by non-invertible transformation [16], [17]. Biohashing is used to combine biometric information with user-specific information to protect and increase the discrimination of biometric information. In the same direction, cancelable multimodal biometric systems have been developed. These systems combine different biometric traits with the purpose of improving recognition accuracy [18 - 23]. Recently, WenchengYang et al. have proposed an alternative to BCV to improve the system's performance and security [24]. In contrast to BCV [25], has been applied linear convolution between a feature part (b_i) selected by the feature vector $b = [b_1, \dots, b_n]$, and a support vector $\left(h_j\right)$ selected from a vector pool $h = [h_1, \dots, h_n]$. Therefore, the transformed feature has been obtained according to the following equation.

$$y = conv(b_i, h_j) \tag{1}$$

The value of j-index has been generated by a feature-guided index generation algorithm. Specifically, the hash function $hash(\cdot)$ is performed between a user-specific key (k_i)selected from the key vector $k = [k_1, \dots, k_n]$, and a feature part (b_i) as following equation.

$$j_{hex} = hash(b_i, k_i,' SHA') \tag{2}$$

where ‘SHA’ represents the Secure Hash Algorithm SHA-256. The output of Equation 2 is a hexadecimal, and it is converted into an integer to obtain the j-index value.

3 Materials and method

The following Section detailed the proposed algorithm to select the optimal transformation matrixes into the matrix selection domain. Before, the BCV-MT [25] approach and the feature-extraction is summarized.

3.1 BCV-MT

BCV-MT uses a non-invertible transformation adding the transformation matrixes to improve the template protection and renewability capability. In the first step is defined the key vector d of size (W+1) where d_0 and d_W are set to 0 and 100, respectively, and the other elements d_j are different integer values between 1 and 99. Next, each feature of length L is divided into a fixed number of parts (W) of length $L_j = \ b_j - b_{j-1}$, where

$$b_j = \left\lceil \frac{d_j}{100} \cdot L \right\rceil, j = 1, \dots, W \tag{3}$$

For example, given a feature vector a of size $L = 200, W = 2$ and a key vector $d = [0, 50,100]$ is computed the b_j as follows.

$$b_1 = \left\lceil \frac{50}{100} \cdot 200 \right\rceil = 100$$

$$b_2 = \left\lceil \frac{100}{100} \cdot 200 \right\rceil = 200$$

Consequently, a has been divided into 2 parts: $a' = \left[a_1, a_2, \dots, a_{b_1}\right]$ and $a'' = [a_{b_{1+1}}, a_{b_{1+2}}, \dots, a_{b_2}]$

In [25] is demonstrated that the recognition performance degradation is due to the increase of the parameter W. In the next step, the transformation matrix (C) has been created. It is composed of F rows and W columns. F is the online signature features number and W is the fixed number of feature parts. An example of a transformation matrix (C), for F=5 and W=2 is shown below.

$$\mathrm{C} = \begin{pmatrix} 3 & 1 \\ 4 & 5 \\ 1 & 2 \\ 2 & 4 \\ 5 & 3 \end{pmatrix}$$

Each column of C is obtained as a shuffle of the vector [1, …, F]$^{\mathrm{T}}$, and the i-th row $C[i, j], j = 1, \dots, W$ is employed to define the feature parts on which to apply the linear convolution $(*)$, to obtain the transformed features according to the following equation.

$$tf_i = (f_{(C[i,j]),P_1} * \dots * f_{(C[i,W]),P_W}), i = 1, \dots, F \qquad (4)$$

In this way, are generated F transformed features. Therefore, the original number of features is equal to the number of transformed features. The total number of transformation matrixes which can be generated is then equal to $(F!)^{(W-1)}$. It is the matrix selection domain for the proposed algorithm.

To perform deconvolution, an attacker must know at least one feature to obtain the original feature. So, the only way to achieve the original features is to perform a brute-force attack with an extremely high computational cost. In the BCV-MT [4], the transformation matrixes have been randomly generated through a random shuffle. A significant impact of the transformation matrixes on the system accuracy has been verified, as shown in Section 4. Therefore, the proposed algorithm has been developed to select the optimal transformation matrixes.

3.2 Feature extraction

Feature–extraction is an important process in implementing the proposed algorithm because it determines the feature number F and thus the matrix selection domain equal to $(F!)^{(W-1)}$.

The online signature features generally consist of horizontal and vertical pen coordinates (x, y), timestamp (t), and pen pressure (p). Therefore, the online signature is a sequence of elements: $(z_i), i = 1, \dots, M$ where each element consists of 4-tuple: $(x_i, y_i, t_{i,} p_i)$. The feature extraction process consists of representing an online signature in fixed feature domains. Five domains are obtained by the 4-tuple in the implemented feature – extraction. They are displacement (s), velocity (v), acceleration (a), path tangent angle (ta), and pressure (p). The five features have been extracted following the equations below.

- Displacement: $s_i = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$
- Velocity: $v_i = \frac{s_i}{(t_{i+1} - t_i)}$

- Acceleration: $a_i = \frac{v_i}{(t_{i+1} - t_i)}$
- Path tangent angle: $ta_i = \tan^{-1}\left(\frac{(y_{i+1} - y_i)}{(x_{i+1} - x_i)}\right)$
- Pressure: the pressure domain is an originally data, therefor no conversion was performed.

Further to the feature extraction, each online signature consists of five features (s, v, a, ta, p).

3.3 Proposed algorithm

The proposed algorithm is applied after the feature–extraction and before the BCV-MT. It is based on a heuristic to selects the optimal transformation matrixes in matrix selection domain in BCV-MT. The proposed heuristic is the minimum distance between a mean template (S_{EB}) and the users' genuine signatures $[g_1, .., g_M]$. The mean template is generated using EB-DBA algorithm [26] on the users' genuine signatures. The first step of EB-DBA is generating a Euclidean Barycenter sequence (EB) by the set of genuine signatures. The EB is the first computed average sequence and the input of the DBA algorithm [27]. DBA is an iterative algorithm used to refine the computed average sequence. Next, the computed mean template is transformed by BCV-MT using a matrix (T_k) of the selection domain $[T_1, \dots, T_N]$. The same matrix is used to transform the users' genuine signatures. For each matrix T_k the following steps are performed.

The mean template and the genuine signatures are transformed according the BCV-MT using the transformation matrix (T_k) according to the following equation.

$$tgf_i = (gf_{(T_k[i,j]),P_1} * \dots * gf_{(T_k[i,W]),P_W}), i = 1, \dots, F \tag{5}$$

The average distance (d_k) between transformed mean template (TS_{EB}) and transformed genuine signatures $[tg_1, .., tg_M]$ is calculated by Dynamic Time Warping (DTW) [28] as details to the following equation.

$$d_k = \frac{\sum_M^{z=1} DTW\ (TS_{EB},\ tg_z)}{M} \tag{6}$$

The computed average distance (d_k) is put into the distances vector $[d_1, \dots, d_N]$.

The distances vector $[d_1, \dots, d_N]$ includes all computed distances between the transformed mean template and transformed genuine signatures using all the transformation matrixes in the selection domain. The last step is selected the index (min) of the minimum value in the distances vector $[d_1, \dots, d_N]$. Consequently, the optimal transformation matrix is the matrix T_{min} in the selection domain $[T_1, \dots, T_N]$.

The selected matrix is the optimal transformation matrix to be used in BCV-MT to transform the users' biometric template.

Details of the proposed algorithm are described in Algorithm 1.

Algorithm 1:

Input: The set of user's genuine signatures $[g_1, .., g_M]$ and the matrix selection domain $[T_1, \dots, T_N]$.

Output: Transformation matrix selected.

```
1| S_EB = EB-DBA([$g_1, .., g_M$])  //compute mean template of genuine signatures
2| for k = 1 … N:
2.1|   TS_EB = Bioconvolving(S_EB, T_k) //transform mean template with T_k
2.2|   for z = 1 … M:
2.2.1|        tg_z = Bioconvolving(g_z, T_k)  //transform genuine signature with T_k
```

```
    2.2.2|    $dt_z$ = DTW($TS_{EB}$, $tg_z$)    //compute the distance between transformed mean template and the z-th transformed genuine signature
    2.2.3|    $[dt_1, \dots, dt_M] \longleftarrow dt_z$  //put the computed distance in the vector
  2.3|  return  $dt_z$
  2.4|  $d_k = \frac{\sum_M^{z=1}(dt_z)}{M}$    //compute the average distance
  2.5|  $[d_1, \dots, d_N] \longleftarrow d_k$  //put the average distance into the distances vector
3| return  $[d_1, \dots, d_N]$
4| min = index_min($[d_1, \dots, d_N]$)  //index of the minimum value in the distances vector
5| selected_matrix = $T_{min}$  //matrix with index ‘min’ in selection matrix domain
```

4 Experiments and results

To prove the validity of the proposed method, three different experiments have been performed. The reported experiments are conducted using 4 online signature datasets outlined below.

- SVC2004 in the version Task2 [29]
- xLongSignDb [30].
- BlindSubCorpus is a version of the SUSig dataset [31].
- VisualSubCorpus is a version of the SUSig dataset [31].

Each dataset is split into a training data set and a test data set. The training data set contains 7 genuine and 7 false signatures for each user. The test data set contains 3 genuine and 3 false signatures for each user. After the feature – extraction, each signature is defined by 5 features (F) and each feature has been divided into 2 parts (W). Consequently, the matrix selection domain is equal to $(F!)^{(W-1)} = 120$. The experiments employ two different online signature verification systems to prove the performance of the protected systems using BCV-MT revised with the proposed algorithm. The first system implements a multiple-template approach comparing a test sample with all the reference samples. The system computes the distances between the signatures using DTW and then classify them into genuine or false class by the SVM with gaussian kernel [32]. The second system implements single-template approach compares a test sample with a mean template generated from the reference samples. It is based on time-series averaging and local stability-weighted dynamic time warping [33]. The system uses a mean template and the references set to construct the local stability sequence (LS). It is used in the DTW cost function, thus implementing the LS-DTW algorithm. LS-DTW algorithm is used to compute the distances between the mean template and the reference signatures. The distances are classified using a SVM with gaussian kernel into genuine and false class. The conducted experiments and their results are detailed in the following.

4.1 Distribution of the transformation matrixes

The matrix distribution analysis is conducted to prove the impact of the transformation matrixes in BCV-MT. The protected multiple-template system accuracy using each matrix of the selection matrix domain and the matrixes frequency into each accuracy class has been calculated. The range $[0.5; 1]$ is divided into 13 equal intervals and each one represents an accuracy class. Figure 1 shows the bar charts of the matrix distribution analysis for each dataset.

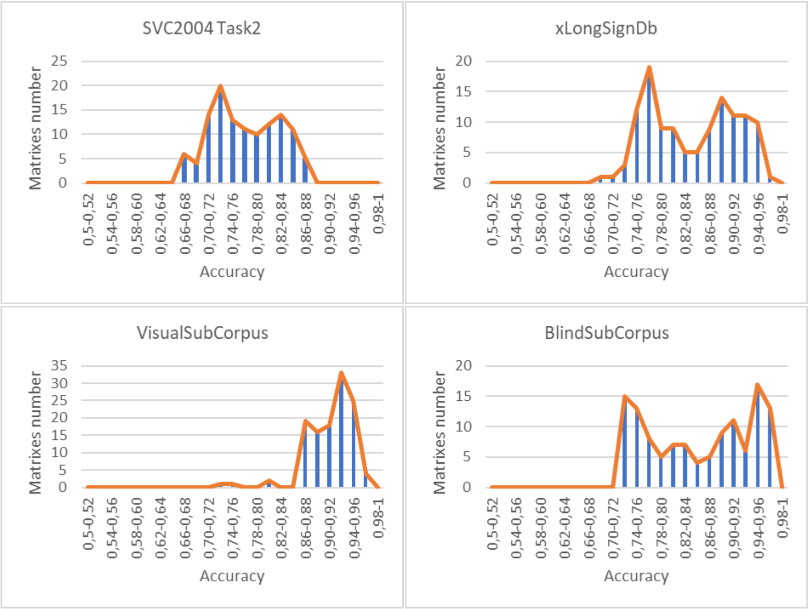

Fig. 1. Bar charts of the matrix distribution for each dataset.

Figure 1 shows that the number of matrixes is widely scattered, for each dataset used. It evidences that the system accuracy changes based on the transformation matrix used. Moreover, the variance and the average accuracy (ACCAverage) are calculated to prove the variability of the matrix distribution, and the results are reported in Table 1.

Dataset	Variance	ACCAverage
SVC2004 Task2	0.00299	0.770
xLongSignDb	0.00512	0.842
VisualSubCorpus	0.00153	0.911
BlindSubCorpus	0.00712	0.854

Table 1. Variance and average accuracy computed for each dataset.

Table 1 shows the highest variance on dataset BlindSubCorpus, xLongSignDb and SVC2004 Task2. Therefore, the random transformation matrix selection may not be the correct approach because the system accuracy is highly variable depending on the selected transformation matrix.

4.2 Correlation between distance and accuracy

The following experiment is conducted to prove the correlation between the computed distance and the system accuracy. The multiple-template system accuracy is calculated for each transformation matrix in the selection domain. The system accuracy is related to the distance calculated between the mean template and genuine signatures transformed for each matrix in the selection domain. This relationship is reported in the scatter plots in Figure 2 for each dataset.

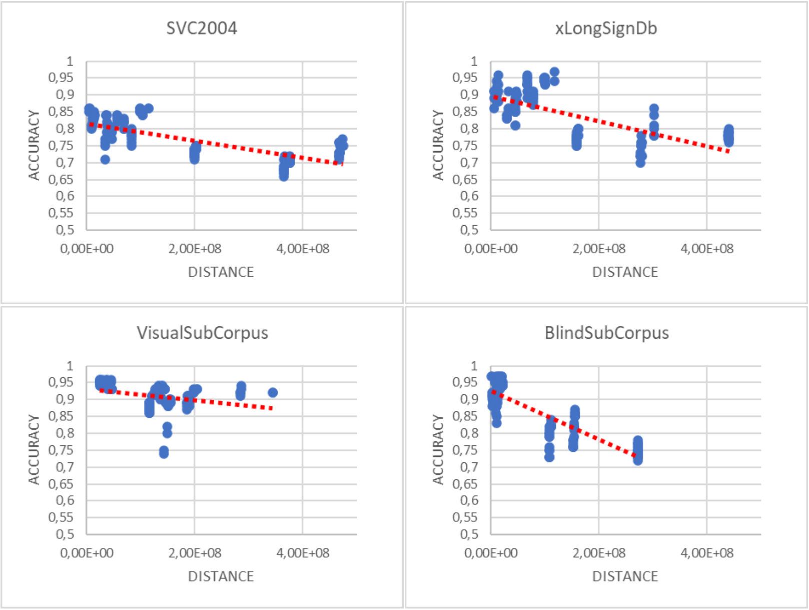

Fig. 2. Scatter plot on the correlation between distance and accuracy.

Figure 2 shows the accuracy degradation due to the increased distance calculated. The slope of the regression line in Figure 2 (red line) describes the incidence of the distance on system accuracy. Moreover, the Pearson correlation coefficient (r) to measure the correlation between distance and accuracy has been calculated. The regression line slope and the correlation coefficient are reported in Table 2.

Dataset	Regression line slope	r
SVC 2004 Task2	(-2.5) · 10 10	-0.73
xLongSignDB	(-3.7) · 10 10	-0.68
VisualSubCorpus	(-1.7) · 10 10	-0.32
BlindSubCorpus	(-7 .2) · 10 10	-0.87

Table 2. Regression line slope and Pearson coefficient (r) between distance and accuracy.

The Pearson coefficient shows a high linear negative relationship between the distance and accuracy for each dataset. The Pearson coefficient computed on the VisualSubCorpus dataset is lower than the other computed coefficient. Then, in VisualSubCorpus, the system accuracy does not decrease visibly with increasing distance, as shown in Figure 2 and Table 2. However, Figure 2 shows that the highest accuracy values are concentrated where the distance is lowest. Therefore, selecting the transformation matrix based on the minimum distance is a good heuristic for all used datasets.

4.3 Proposed algorithm performance and comparative analysis

In the last experiment, the multiple-template system and the single-template system are protected by the BCV-MT using the random matrix selection and the matrix selection by the proposed algorithm The random matrix selection is implemented using 4 transformation matrixes randomly generated and shown below.

$$\text{Matrix 1: }\begin{pmatrix}3 & 0\\ 4 & 4\\ 1 & 3\\ 0 & 1\\ 2 & 2\end{pmatrix}\quad \text{Matrix 2: }\begin{pmatrix}0 & 4\\ 2 & 1\\ 3 & 0\\ 4 & 3\\ 1 & 2\end{pmatrix}\quad \text{Matrix 3: }\begin{pmatrix}4 & 0\\ 2 & 1\\ 3 & 3\\ 1 & 4\\ 0 & 2\end{pmatrix}\quad \text{Matrix 4: }\begin{pmatrix}3 & 2\\ 0 & 3\\ 2 & 1\\ 4 & 0\\ 1 & 4\end{pmatrix}$$

Accuracy, False Negative Rate (FNR), False positive Rate (FPR), and delta average accuracy (ΔACC) are calculated to evaluate the results. The (ΔACC) computes the difference between the unprotected system accuracy and the protected system accuracy average. Table 3 shows the results.

SVC2004 Task2

	Single template				Multiple template			
Dataset	FPR	FNR	ACC	ΔACC	FPR	FNR	ACC	ΔACC
BCV-MT with Matrix 1	0.219	0.307	0.742	0.192	0.254	0.245	0.751	0.180
BCV-MT with Matrix 2	0.377	0.254	0.680	0.254	0.377	0.149	0.743	0.188
BCV-MT with Matrix 3	0.333	0.280	0.695	0.239	0.333	0.237	0.711	0.220
BCV-MT with Matrix 4	0.289	0.298	0.718	0.216	0.377	0.219	0.702	0.229
BCV-MT Proposed algorithm	**0.149**	**0.184**	**0.817**	**0.117**	**0.122**	**0.131**	**0.854**	**0.077**
Unprotected system	0.035	0.105	0.934		0.061	0.071	0.931	

xLongSignDb

	Single template				Multiple template			
Dataset	FPR	FNR	ACC	ΔACC	FPR	FNR	ACC	ΔACC
BCV-MT with Matrix 1	0.234	0.222	0.774	0.188	0.320	0.123	0.780	0.151
BCV-MT with Matrix 2	0.345	0.148	0.757	0.205	0.419	0.098	0.746	0.185
BCV-MT with Matrix 3	0.098	0.087	0.912	0.050	0.136	0.136	0.865	0.066
BCV-MT with Matrix 4	0.099	0.099	0.909	0.053	0.148	0.086	0.882	0.049
BCV-MT Proposed algorithm	**0.098**	**0.024**	**0.921**	**0.041**	**0.086**	**0.061**	**0.911**	**0.020**
Unprotected system	0.049	0.024	0.962		0.061	0.074	0.931	

SUSig VisualSubCorpus

	Single template				Multiple template			
Dataset	FPR	FNR	ACC	ΔACC	FPR	FNR	ACC	ΔACC
BCV-MT with Matrix 1	0.025	0.170	0.901	0.070	0.037	0.109	0.931	0.002
BCV-MT with Matrix 2	0.333	0.246	0.713	0.258	0.420	0.134	0.724	0.209
BCV-MT with Matrix 3	0.125	0.047	0.915	0.056	0.134	0.058	0.902	0.031
BCV-MT with Matrix 4	0.217	0.120	0.833	0.138	0.284	0.066	0.839	0.094
BCV-MT Proposed algorithm	**0.076**	**0.065**	**0.934**	**0.037**	**0.068**	**0.076**	**0.932**	**0.006**
Unprotected system	0.052	0.018	0.971		0.061	0.075	0.938	

SUSig BlindSubCorpus

	Single template				Multiple template			
Dataset	FPR	FNR	ACC	ΔACC	FPR	FNR	ACC	ΔACC
BCV-MT with Matrix 1	0.344	0.138	0.764	0.157	0.425	0.092	0.747	0.187
BCV-MT with Matrix 2	0.241	0.144	0.812	0.109	0.270	0.115	0.814	0.120
BCV-MT with Matrix 3	0.149	0.098	0.882	0.039	0.229	0.058	0.863	0.071
BCV-MT with Matrix 4	0.178	0.109	0.860	0.061	0.247	0.063	0.841	0.093
BCV-MT Proposed algorithm	**0.068**	**0.034**	**0.951**	-0.030	**0.034**	**0.028**	**0.968**	**-0.034**
Unprotected system	0.126	0.042	0.921		0.061	0.078	0.934	

Table 3. Performance comparison between the BCV-MT with random matrixes selection, BCV-MT with the selected matrixes by the proposed algorithm and the unprotected system.

Table 3 shows that the performance of protected systems using BCV-MT revised by the proposed algorithm is better than the BCV-MT with random matrix selection. Moreover, the ΔACC shows how the accuracy variation between the protected system accuracy with BCV-MT revised by proposed algorithm and the unprotected system is not drastically reduced, range from 0.6% to 11%. Therefore, the selected matrixes are among the best matrixes in the matrix selection domain. The proposed algorithm has been executed on the Azure Virtual Machine of Fsv2 series with the following datasheet. CPU Intel Xeon Platinum 8370C with turbo clock speed of 3.4 GHz, RAM 16 Gb and O.S. Linux. The average execution times (in milliseconds) of the proposed algorithm for each user are reported in Table 4.

Dataset	Execution time (ms)
SVC 2004 Task2	33200
xLongSignDB	42930
VisualSubCorpus	33970
BlindSubCorpus	26221

Table 4. Execution time in milliseconds for select the optimal matrix for each user.

Table 4 shows fast execution time. Therefore, the proposed algorithm can be implemented in real-world scenarios. Moreover, the proposed algorithm is used only in the enrollment phase, and then the cost of execution time does not impact the verification system's usability.

5 Conclusions and future work

In this work, a heuristic approach has been introduced to improve the performance of the online signature verification systems protected using BCV-MT. In BCV-MT, the transformation matrixes significantly impact the system performance. The matrix distribution analysis showed that a random matrix choice could compromise the protected system performance. To solve this issue, the proposed heuristic selects the optimal transformation matrixes through the minimum distance parameter between the transformed mean template and the transformed genuine signatures. This parameter showed a high correlation with the system accuracy. Consequently, using BCV-MT revised with the proposed algorithm is an optimal strategy to protect the biometric template, drastically reducing system performance. A limitation of the proposed heuristic is the increase of the matrix selection domain. The size of the domain grows factorial as features increase and exponential as features part increase. Therefore, a significant number of features or feature parts would expand the matrix selection domain, and then the proposed algorithm would become computationally costly in terms of time. However, the increase of the feature parts involves the reduction of the recognition performance [25]. Moreover, the proposed algorithm is used only in the training phase, and the cost can be acceptable. The proposed heuristic could be implemented in different biometric systems to protect different biometric data in future work. In fact, BCV has also been used to protect biometric traits such as the face, iris, palm print, fingerprint, and ear [6]. Finally, the proposed heuristic could be improved to automatically generate the optimal transformation matrix based on the input biometric features.

Acknowledgement

Francesco Castro is a PhD student enrolled in the National PhD in Artificial Intelligence, XXXVIII cycle, course on Health and life sciences, organized by Università Campus Bio-Medico di Roma.

References

1. M. Gomez-Barrero, J. Galbally, A. Morales, and J. Fierrez, "Privacy-Preserving Comparison of Variable-Length Data With Application to Biometric Template Protection," *IEEE Access*, vol. 5, pp. 8606–8619, 2017, doi: 10.1109/ACCESS.2017.2691578.
2. A. Thawre, A. Hariyale, and B. R. Chandavarkar, "Survey on security of biometric data using cryptography," in *2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC)*, 2021, pp. 90–95. doi: 10.1109/ICSCCC51823.2021.9478120.
3. M. Lutsenko, A. Kuznetsov, A. Kiian, O. Smirnov, and T. Kuznetsova, "Biometric cryptosystems: Overview, state-of-the-art and perspective directions," in *Lecture Notes in Networks and Systems*, vol. 152, 2021, pp. 66–84. doi: 10.1007/978-3-030-58359-0_5.
4. D. Impedovo and G. Pirlo, "Automatic Signature Verification in the Mobile Cloud Scenario: Survey and Way Ahead," *IEEE Trans Emerg Top Comput*, vol. 9, no. 1, pp. 554–568, 2021, doi: 10.1109/TETC.2018.2865345.
5. E. Maiorana, P. Campisi, and A. Neri, "Bioconvolving: Cancelable templates for a multi-biometrics signature recognition system," in *2011 IEEE International Systems Conference, SysCon 2011 - Proceedings*, 2011, pp. 495–500. doi: 10.1109/SYSCON.2011.5929064.
6. E. Abdellatef *et al.*, "Fusion of deep-learned and hand-crafted features for cancelable recognition systems," *Soft comput*, vol. 24, no. 20, pp. 15189–15208, Oct. 2020, doi: 10.1007/s00500-020-04856-1.
7. M. Diaz, M. Ferrer, D. Impedovo, M. Malik, G. Pirlo, and R. Plamondon, "A Perspective Analysis of Handwritten Signature Technology," *ACM Comput Surv*, vol. 51, p. 39, Aug. 2018, doi: 10.1145/3274658.
8. M. Martinez-Diaz, J. Fierrez, R. P. Krish, and J. Galbally, "Mobile Signature Verification: Feature Robustness and Performance Comparison," *IET Biom*, vol. 3, May 2014, doi: 10.1049/iet-bmt.2013.0081.
9. H. Kaur and P. Khanna, "PolyCodes: generating cancelable biometric features using polynomial transformation," *Multimed Tools Appl*, vol. 79, no. 29, pp. 20729–20752, 2020, doi: 10.1007/s11042-020-08734-8.
10. M. Deshmukh and M. K. Balwant, "Generating Cancelable Palmprint Templates Using Local Binary Pattern and Random Projection," in *2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, 2017, pp. 203–209. doi: 10.1109/SITIS.2017.43.
11. W. Yang, S. Wang, M. Shahzad, and W. Zhou, "A cancelable biometric authentication system based on feature-adaptive random projection," *Journal of Information Security and Applications*, vol. 58, p. 102704, 2021, doi: https://doi.org/10.1016/j.jisa.2020.102704.
12. A. B. J. Teoh, S. Cho, and J. Kim, "Random permutation Maxout transform for cancellable facial template protection," *Multimed Tools Appl*, vol. 77, no. 21, pp. 27733–27759, 2018, doi: 10.1007/s11042-018-5956-y.
13. G. S. Walia, T. Singh, K. Singh, and N. Verma, "Robust multimodal biometric system based on optimal score level fusion model," *Expert Syst Appl*, vol. 116, pp. 364–376, 2019, doi: https://doi.org/10.1016/j.eswa.2018.08.036.
14. Y. Lai, Z. Jin, B.-M. Goi, T.-Y. Chai, and W.-S. Yap, *Iris Cancellable Template Generation Based on Indexing-First-One Hashing*, vol. 9955. 2016. doi: 10.1007/978-3-319-46298-1_29.
15. P. Sharma, G. S. Walia, and R. Rohilla, "Recent advancement in cancelable biometric for user recognition: A brief survey," in *Proceedings of the 2020 9th International Conference on System Modeling and*

Advancement in Research Trends, SMART 2020, Dec. 2020, pp. 137–145. doi: 10.1109/SMART50582.2020.9337107.

16. H. Kaur and P. Khanna, “Random Slope method for generation of cancelable biometric features,” *Pattern Recognit Lett*, vol. 126, pp. 31–40, 2019, doi: https://doi.org/10.1016/j.patrec.2018.02.016.
17. Y. Liu, J. Ling, Z. Liu, J. Shen, and C. Gao, “Finger vein secure biometric template generation based on deep learning,” *Soft comput*, vol. 22, no. 7, pp. 2257–2265, 2018, doi: 10.1007/s00500-017-2487-9.
18. N. Kumar and M. Rawat, “RP-LPP : a random permutation based locality preserving projection for cancelable biometric recognition,” *Multimed Tools Appl*, vol. 79, no. 3, pp. 2363–2381, 2020, doi: 10.1007/s11042-019-08228-2.
19. A. K. Jindal, S. R. Chalamala, and S. K. Jami, “Securing Face Templates using Deep Convolutional Neural Network and Random Projection,” in *2019 IEEE International Conference on Consumer Electronics (ICCE)*, 2019, pp. 1–6. doi: 10.1109/ICCE.2019.8662094.
20. G. Walia, S. Rishi, R. Asthana, A. Kumar, and A. Gupta, “A Secure Multimodal Biometric System based on Diffused Graphs and Optimal Score Fusion,” *IET Biom*, vol. 8, Jul. 2019, doi: 10.1049/iet-bmt.2018.5018.
21. H. Kaur and P. Khanna, “Random Distance Method for Generating Unimodal and Multimodal Cancelable Biometric Features,” *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 3, pp. 709–719, 2019, doi: 10.1109/TIFS.2018.2855669.
22. F. E. Abd El-Samie *et al.*, “Efficient implementation of optical scanning holography in cancelable biometrics,” *Appl Opt*, vol. 60, no. 13, p. 3659, May 2021, doi: 10.1364/ao.415523.
23. Z. Jin, J. Y. Hwang, Y. Lai, S. Kim, and A. B. J. Teoh, “Ranking-Based Locality Sensitive Hashing-Enabled Cancelable Biometrics: Index-of-Max Hashing,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 2, pp. 393–407, 2018, doi: 10.1109/TIFS.2017.2753172.
24. W. Yang, S. Wang, J. J. Kang, M. N. Johnstone, and A. Bedari, “A linear convolution-based cancelable fingerprint biometric authentication system,” *Comput Secur*, vol. 114, p. 102583, 2022, doi: https://doi.org/10.1016/j.cose.2021.102583.
25. E. Maiorana, P. Campisi, J. Fierrez, J. Ortega-Garcia, and A. Neri, “Cancelable Templates for Sequence-Based Biometrics with Application to On-line Signature Recognition,” *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 40, no. 3, pp. 525–538, 2010, doi: 10.1109/TSMCA.2010.2041653.
26. M. Okawa, “Template Matching Using Time-Series Averaging and DTW With Dependent Warping for Online Signature Verification,” *IEEE Access*, vol. 7, pp. 81010–81019, 2019, doi: 10.1109/ACCESS.2019.2923093.
27. F. Petitjean, A. Ketterlin, and P. Gançarski, “A global averaging method for dynamic time warping, with applications to clustering,” *Pattern Recognit*, vol. 44, no. 3, pp. 678–693, 2011, doi: https://doi.org/10.1016/j.patcog.2010.09.013.
28. A. Sharma and S. Sundaram, “On the Exploration of Information From the DTW Cost Matrix for Online Signature Verification,” *IEEE Trans Cybern*, vol. 48, no. 2, pp. 611–624, 2018, doi: 10.1109/TCYB.2017.2647826.
29. D.-Y. Yeung *et al.*, *SVC2004: First International Signature Verification Competition*, vol. 5. 2004. doi: 10.1007/978-3-540-25948-0_3.
30. J. Galbally, M. Martinez-Diaz, and J. Fierrez, “Aging in Biometrics: An Experimental Analysis on On-Line Signature,” *PLoS One*, vol. 8, p. e69897, Jul. 2013, doi: 10.1371/journal.pone.0069897.

31. A. Kholmatov and B. A. Yanikoglu, “SUSIG: an on-line signature database, associated protocols and benchmark results,” *Pattern Analysis and Applications*, vol. 12, pp. 227–236, 2008.
32. K. K. Tseng, X. X. An, and C. Chen, “Online handwritten verification algorithms based on DTW and SVM,” *Journal of Internet Technology*, vol. 21, no. 6, pp. 1725–1732, 2021, doi: 10.3966/160792642020112106014.
33. M. Okawa, “Time-series averaging and local stability-weighted dynamic time warping for online signature verification,” *Pattern Recognit*, vol. 112, Apr. 2021, doi: 10.1016/j.patcog.2020.107699.