Received: 10 February 2022 | Revised: 4 March 2022 | Accepted: 9 April 2022

IET Biometrics

DOI: 10.1049/bme2.12075

The Institution of Engineering and Technology WILEY

REVIEW

Hybrid biometric template protection: Resolving the agony of choice between bloom filters and homomorphic encryption

Amina Bassit[1,2] | **Florian Hahn**[2] | **Raymond Veldhuis**[1,4] | **Andreas Peter**[2,3]

[1]EEMCS Faculty, Data Management & Biometrics Group, University of Twente, Enschede, The Netherlands

[2]EEMCS Faculty, Services and CyberSecurity Group, University of Twente, Enschede, The Netherlands

[3]Computer Science Department, Safety-Security-Interaction Group, University of Oldenburg, Oldenburg, Germany

[4]Department of Information Security and Communication Technology, Norwegian University of Science and Technology, Gjøvik, Norway

Correspondence

Amina Bassit, EEMCS Faculty, Data Management & Biometrics Group and Services and CyberSecurity Group, University of Twente, Drienerlolaan 5, 7522 NB, Enschede, The Netherlands.
Email: a.bassit@utwente.nl

Funding information

H2020 Marie Skłodowska-Curie Actions, Grant/Award Number: 860315

Abstract

Bloom filters (BFs) and homomorphic encryption (HE) are prominent techniques used to design biometric template protection (BTP) schemes that aim to protect sensitive biometric information during storage and biometric comparison. However, the pros and cons of BF- and HE-based BTPs are not well studied in literature. We investigate the strengths and weaknesses of these two approaches since both seem promising from a theoretical viewpoint. Our key insight is to extend our theoretical investigation to cover the practical case of iris recognition on the ground that iris (1) benefits from the alignment-free property of BFs and (2) induces huge computational burdens when implemented in the HE-encrypted domain. BF-based BTPs can be implemented to be either fast with high recognition accuracy while missing the important privacy property of 'unlinkability', or to be fast with unlinkability-property while missing the high accuracy. HE-based BTPs, on the other hand, are highly secure, achieve good accuracy, and meet the unlinkability-property, but they are much slower than BF-based approaches. As a synthesis, we propose a hybrid BTP scheme that combines the good properties of BFs and HE, ensuring unlinkability and high recognition accuracy, while being about seven times faster than the traditional HE-based approach.

1 | INTRODUCTION

A biometric template is a compact representation of a physiological or a behavioural biometric characteristic such as face, iris, voice, etc. The biometric characteristic itself is not a secret as, in human-to-human interaction, humans recognise each other from their actual characteristics. However, in a human-to-machine interaction, a biometric template becomes a numerical equivalent of the human characteristic understandable by a machine. Thus, a biometric template reflects the identity of an individual that allows him/her to be recognized by the system. Given the fact that systems are subject to various types of security threats, a biometric template must be well protected.

References [2, 3] define *biometric template protection* (BTP) schemes as the branch of biometrics that tackles the problem of persevering biometric templates while maintaining the recognition performance. There exist different approaches to design BTP schemes that try to satisfy the privacy requirements of the international standard ISO/IEC 24,745 [4]: irreversibility, unlinkability, and confidentiality. Among those approaches, *Bloom filter* (BF)-based BTPs, process the template in a transformed domain, while *homomorphic encryption* (HE)-based BTPs, process the template in an encrypted domain. Both approaches have common and exclusive interesting properties that deal with the BTP challenges and the tradeoffs. Several surveys investigate either Bloom filters [5, 6]

This paper is an extension of [1] published at BIOSIG 2021.

or HE [7–9] and their applications in general. However, none of them focuses on examining these two approaches from a biometrics point of view.

This paper investigates the theoretical differences between BF-based BTP schemes and HE-based BTP schemes and then compares both approaches experimentally using iris recognition as a study case. From a theoretical standpoint, we analyse the state of the art of both approaches by examining their core functionalities and how they are utilised in the design of BTP schemes. Then, based on the stated results of prior works, we compare their advantages and disadvantages with respect to different levels: fulfilment of the privacy requirements of ISO/IEC 24,745, application usability, and protected template flexibility.

Subsequently, we continue our investigation by experimentally comparing BF-based BTPs and HE-based BTPs approaches for the study case of iris recognition. Iris is a particularly interesting modality to look at in this comparative study. The comparison of the reference template against a probe in iris recognition [11] necessitates to calculate the Hamming distance between the reference template and n circular shiftings of the probe to the left and another n shiftings to the right. The dissimilarity score for such a reference-probe comparison is the minimum Hamming distance among those calculated distances. These circular shiftings are used to achieve the orientation invariance for iris recognition that comes from the iris rotation angle, which may vary at each capturing phase.

Although the computation of Hamming distance under encryption is relatively efficient since it requires only one homomorphic multiplication per row; the circular shiftings in the encrypted domain become homomorphic rotations over ciphertexts and additions among plaintext slots which are known to be computationally expensive operations. Despite the fact that HE-based BTPs are unlinkable by design and preserve the biometric performance of the underlying biometric comparator (in this case, shifted Hamming distances), implementing this in the encrypted domain yields a slow recognition system; since it has to perform $2n + 1$ homomorphic rotations per probe sample's row on the top of computing the Hamming distances for each of them. Thus, iris recognition could benefit from HE-based BTPs in achieving unlinkability and persevering biometric accuracy but will suffer from its computational cost.

On the other side of the spectrum, the use of BF-based BTPs for iris recognition cancels out the need for those circular shiftings since the invariant property of BFs makes them alignment-free with respect to iris features insertion. Additionally, BF-based BTPs could be used in an application-specific key setting or a user-specific key setting. The application-specific key setting requires the system to use the same key for all subjects to generate their templates within the same application. In contrast, the user-specific key setting requires the system to use a different key for each subject to generate its template. From a security perspective, knowing that the resulting templates are in the transformed domain if the same key is used to generate templates of different subjects, then when the system is compromised, the templates of other subjects are also compromised. In this case, requiring a different key per subject makes it difficult for an attacker to compromise all protected templates at once. Furthermore, as we will demonstrate in our experiments, it turns out that when BF-based BTPs are used in the user-specific key setting, the biometric performance improves significantly. It even outperforms the baseline comparator, the shifted Hamming distances. However, security-wise, BF-based BTPs are prone to linkability and reversibility, unlike HE-based BTPs that are immune to them by design. Hence, iris recognition could benefit from BF-based BTPs in improving its biometric performance at the cost of its security.

Those intertwined advantages and disadvantages of both approaches (see Table 1) in the case study of iris recognition motivate us to propose a hybrid BTP scheme where the BF and HE complement each other, and thus the new proposed scheme benefits from their best properties. Our proposed hybrid BTP scheme is in the user-specific key setting and thus requires two keys. A key for the subject to transform its iris-code into a BF representation and another key for the system to learn the recognition outcome. In this hybrid BTP, BF is used as a representation of the iris-code that leads to an accurate biometric comparison (an EER of 0.17% over the IITD iris database [10]) while HE brings

TABLE 1 Summary of our comparative study results for the case of iris recognition

BTP schemes	Unlinkability	EER (%)[a]	Runtime (ms)[b]		
HE BTP	✓	0.44	333.96	885.01	1276.06
First BF BTP app-spec key[c]	✗	0.44	0.725		
First BF BTP user-spec key[d]	✗	0.17	0.725		
Second BF BTP app-spec key[c]	✓	0.53–0.66[e]	0.905		
Second BF BTP user-spec key[d]	✓	0.44–0.61[e]	0.905		
Hybrid BTP user-spec key (this work)	✓	0.17	104.35	155.15	171.70

[a]Performance over the IITD iris database [10].

[b]HE and Hybrid BTPs are measured over 128, 192, and 256 bits security levels, respectively.

[c]Application-specific key.

[d]User-specific key.

[e]Range resulting from running four experiments where in each of the random permutations were chosen.

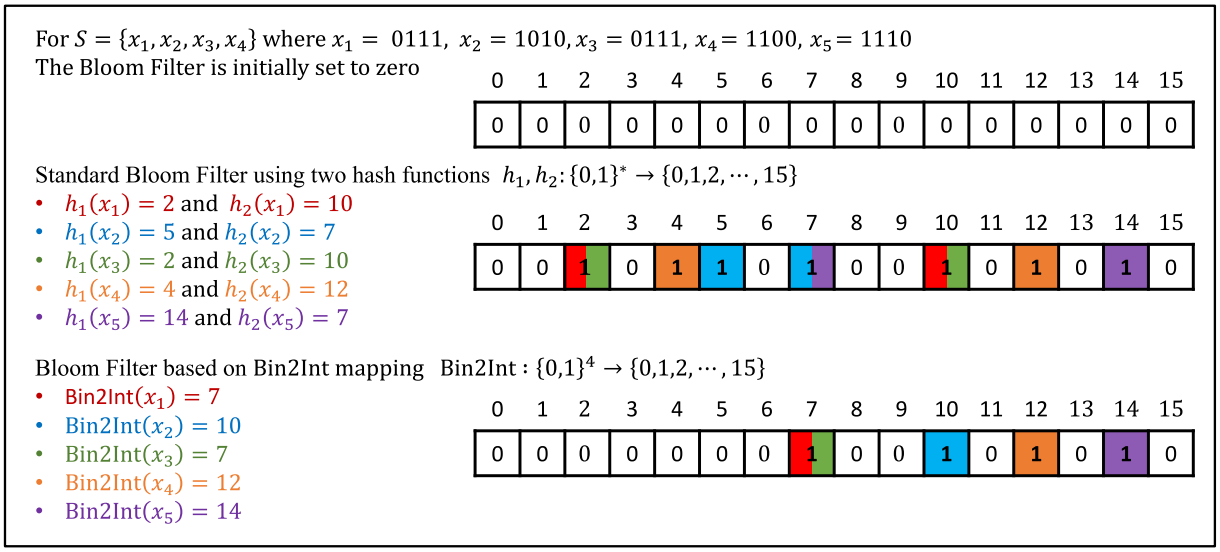


FIGURE 1 Toy example illustrating the difference between a standard BF that uses two hash functions and the BF used in BTP schemes that uses only a single binary-to-integer function. Observe that in the latter the number of ones corresponds to the number of distinct elements in the set S (three elements) while in the standard BF the number of ones (seven ones) exceeds the number of distinct elements

confidentiality, unlinkability, and irreversibility to the system. The results of our evaluation, that is a C++ implementation, showed that the hybrid BTP runs in 104.35 ms (resp. 155.15 and 171.70 ms) for 128 bits security level[1] (resp. 192 and 256 bits) that is three times (resp. five times and seven times) faster than HE-based BTP (that runs in 333.96 , 885.01, 1276.06 ms for 128, 192 and 256 bits respectively) and is three orders of magnitude slower than BF-based BTP. In summary our contributions are as follows:

1. We investigate the differences between BF-based BTP schemes and HE-based BTP schemes from a theoretical perspective.
2. We conduct an in-depth experimental examination of both approaches in the special case of iris recognition upon which we notice that both approaches can be combined to yield a hybrid BTP scheme benefiting from the best properties of both approaches combined, optimising the accuracy-efficiency-security tradeoff.
3. We propose a formulation for this hybrid BTP, evaluate its performance, and compare the three approaches, BF-based BTPs, HE-based BTPs, and hybrid BTP.

2 | BACKGROUND

In this section, we discuss Bloom filters and HE as technologies we are investigating in the context of biometric recognition. We also provide the privacy requirements recommended by ISO/IEC 24,745 [4].

2.1 | Bloom filter

A standard Bloom filter (BF) is an efficient data structure that is used to verify whether an element belongs to a set or not. Let us denote $S = \{x_1, \ldots, x_n\}$ where $x_i \in \{0,1\}^*$[2] a set of n elements to-be-represented. A BF consists of an m bits array initially set to zero. The filter uses k independent hash functions $h_1, \ldots, h_k$, where h_i: $\{0,1\}^* \rightarrow \{0, 1, \ldots, m-1\}$, are assumed to be uniformly random. To insert an element $x \in S$ in the BF, the bit at index $h_i(x)$ is set to one for all $1 \leq i \leq k$. To verify whether an element y belongs to S, for all $i \in [1, k]$ the bit at index $h_i(y)$ must be activated.[3] Hence, if at least one index is not activated then with certainty y does not belong to S otherwise y probably belongs to S since the indexes could have been activated by some elements of S distinct from y.

Figure 1 presents an example of generation of a standard BF and a BF used in BTP schemes that represent the same set S; the colours indicate how each element of S was inserted in the BF. This shows the effect of using a single hash function versus k independent hash functions. The use of binary-to-integer as a single hash to compute the BF makes it straightforward to learn the number of distinct elements in S as well as the elements themselves. Reference [12] provides an extensive study on the selection of optimal parameters (k, n and m) of a BF and reference [13] provides an online tool to estimate them and observe parameters variation.

BF is used in biometrics not only for being a space-efficient data structure but also for its invariant property with respect to element insertion since the BF of a set of elements S is identical to the BF of any permutation of S. This property is important for disposing of the inconvenient features alignment, and thus

[1]A security level of n bits indicates the number of operations an attacker should perform to break the cryptographic scheme that is $2n$ operations.

[2]The set $\{0,1\}^*$ refers to the binary set of arbitrary length.

[3]BF is activated at index j means it is set to one at index j.

to allow an alignment-free technique. The BFs used in biometrics differ from the standard ones in the number of hash functions. They use a single hash function that is binary-to-integer, and the verification of element membership. Instead they calculate the weighed Hamming distance between the BFs of two sets. BFs are close if the distance is small and thus their corresponding sets are likely to overlap.

2.2 | Homomorphic encryption

HE allows computation over encrypted data without decryption; $[\![x]\!] * [\![y]\!] = [\![x \circ y]\!]$ where $[\![\cdot]\!]$ represents an encryption, $*$ operation in an encrypted domain and $\circ$ operation in a plaintext domain. The operations $*$ and $\circ$ can be either an addition, a multiplication or both; depending on the HE scheme type. There are three types of HE schemes: partially HE (PHE), somewhat HE (SWHE) and fully HE (FHE). PHE schemes (e.g. Paillier [14], ElGamal [15]) support only one operation unlimited number of times with a plaintext space that is either a binary or an integer. SWHE schemes (e.g. BGN [16]) support a limited[4] number of operations, usually a limited number of multiplications and an arbitrary number of additions, and operate also on a binary or integer plaintext space. FHE schemes (e.g. BFV [17, 18], BGV [19], and CKKS [20]) support an unlimited number of both operations and are fundamentally based on Gentry's construction [21] that enables refreshing ciphertexts to prevent them from reaching the allowed limit in each operation, and thus they remain decryptable. Unlike the classical PHEs and SWHEs, that have a limited choice of the plaintext, the state-of-the-art FHEs support binary (e.g. BFV), integers (e.g. BGV), real numbers and complex numbers (e.g. CKKS). Moreover, they offer a new style of operations, called *single-instruction multiple-data* (SIMD), that significantly contributes to speeding up FHEs. For instance, they allow encryption of a vector of plaintexts, packing of a vector of ciphertexts into a single ciphertext, and permutations within the same ciphertext and automorphisms of a ciphertext. Although the practical improvements on accelerating FHE schemes are considerable, it is still an active area of research.

HE offers flexibility in processing encrypted data; however, it comes with a significant cost that impacts the storage as well as the runtime. The HE ciphertexts have a large size which implies that the biometric encrypted templates have a large size as well. The biometric recognition performed in the plaintext domain is significantly faster than the biometric recognition performed in the encrypted domain since they require several multiplications which are resource demanding operations under HE. The impact that HE has on the memory space and the runtime is undesirable in biometric recognition systems that try to minimise both to meet the usability requirement. However, this optimization should not be at the expense of their security.

[4]SWHE schemes produce noisy ciphertexts where the noise grows along with each homomorphic operation until it reaches its limit. Subsequently, the resulted ciphertext can no longer be decryptable.

2.3 | Privacy requirements of ISO/IEC 24745

The international standard ISO/IEC 24745 [4] establishes requirements and guidelines on how the biometric information should be protected throughout its entire lifecycle: storage, transfer and processing. The standard highlights the importance of binding a biometric reference with the corresponding subject identity as well as the privacy protection of the subjects' biometric information during the processing. In this work, we focus on the ISO/IEC 24745 privacy requirements: *Irreversibility:* for a fixed pre-defined usage (such as recognition), the raw biometric data must be transformed into an irreversible representation that precisely fits the task of the pre-defined usage. *Unlinkability:* there must be no relationship between the stored biometric templates neither across applications nor databases. *Confidentiality:* the biometric template must be preserved and not exposed to unauthorized parties trying to gain unauthorized accesses.

3 | THEORETICAL COMPARISON

3.1 | Bloom filter-based BTP schemes

Cancellable biometric systems [22–24], that apply non-invertible transformations to preserve the biometric template, suffer from significant degradation in their recognition performance due to the use of non-invertible transformations (such as cryptographic hash functions) that hurt the biometric accuracy. BF-based BTP schemes overcome this drawback by taking advantage from the invariant property of BFs to conceal a distorted version of the raw biometric sample in a BF-based template and thus achieve diffusion of the statistical properties of biometric features while maintaining their distinctiveness.

3.1.1 | First category BF-based BTPs

Rathgeb et al. [25] introduced the first BF-based BTP scheme (which we call the *first category BF-based BTP scheme* and illustrate in Figure 2), as a form of cancellable biometric system that preserves the recognition performance by circumventing the feature alignment problem during the comparison process. This is achieved since BFs are invariant with respect to the insertion of elements as the BF of a set of elements S is identical to the BF of any permutation of S. This first category of BF-based BTPs was tested on irises [25–28], faces [29] and fingerprints [30] to demonstrate the diversity of this approach with respect to the biometric modalities as long as they can be expressed as binary feature vectors.

The early security assessment of the first category of BF-based BTPs was studied by authors in [31] who confirmed the irreversibility of their templates but questioned their unlinkability. In particular, the authors showed that for $T_1 = \left\{ \mathrm{BF}_{B_i}^{M}(K_1) \right\}_1^k$ and $T_2 = \left\{ \mathrm{BF}_{B_i}^{M}(K_2) \right\}_1^k$ two BF-based

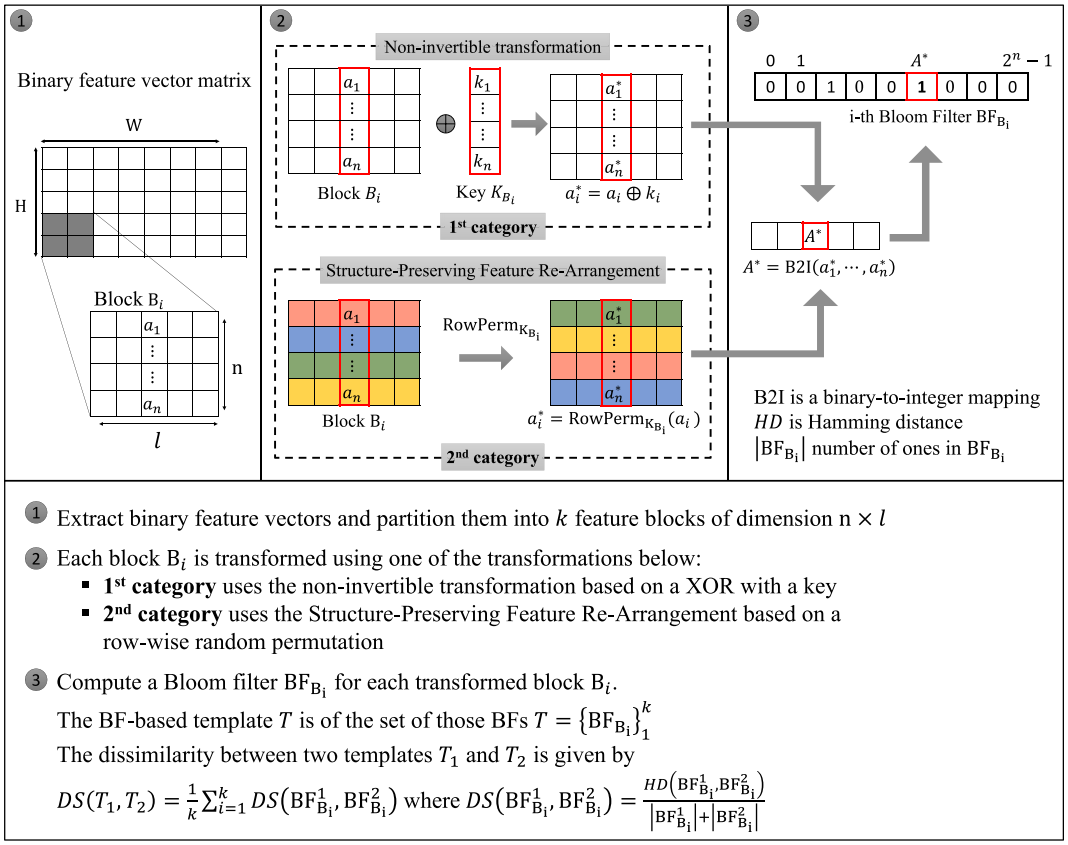


FIGURE 2 Overview of the first category and the second category BF-based BTP schemes (illustration from [1]). Steps 1 and 3 are common to both categories. In Step 2, the first category (resp. second category) of each block is transformed via a XOR with a key (resp. row-wise random permutation). Note that the key[5] should be different from an application to another to avoid cross-matching over databases. The original scheme [25] uses the same key for all blocks while authors in [32], who assessed its security, proposed to use a different key per block, as depicted in this figure

templates generated from the same iris code M using different keys $K_1 \neq K_2$ are determined to conceal the same iris code with a probability of 96% assuming that the biometric samples are uniformly random. Later, authors in [32] extended the unlinkability analysis and considered the non-uniformity of biometric samples inherited from the acquisition noise to determine whether $\tilde{T}_1 = \left\{BF_{B_i}^{M_1}(K_1^i)\right\}_1^k$ and $\tilde{T}_2 = \{BF_{B_i}^{M_2}(K_2^i)\}_1^k$, with different iris codes and different keys, are from the same iris. Their attack is a brute force over the possible keys K per block that saves the key with the lowest dissimilarity score. In other terms, for each block B_i it searches for

$$K = \underset{\hat{K}\in[0,2^n-1]}{\text{argmin}} \; DS\left(BF_{B_i}^{M_1}(K_1^i), BF_{B_i}^{M_2}\left(K_2^i \oplus \hat{K}\right)\right)$$

where $BF_{B_i}^{M_2}\left(K_2^i \oplus \hat{K}\right)$ is computed only from $BF_{B_i}^{M_2}(K_2^i)$ and $\hat{K}$ by activating the BF at index $j \oplus \hat{K}$ if and only if BF at index j is activated. Hence, the distribution of the dissimilarity scores of the original BF-based templates $DS\left(BF_{B_i}^{M_1}(K_1^i), BF_{B_i}^{M_2}(K_2^i)\right)$ and the distribution of the attacked templates $DS(BF_{B_i}^{M_1}(K_1^i), BF_{B_i}^{M_2}(K_2^i \oplus K))$, where the key K has been chosen from the lowest dissimilarity score, overlap and have a slightly similar error rate. Then, authors in [32] analysed the irreversibility of a first category BF-based template without key $K = 0$ and proposed two attacks that try to reconstruct an approximation of the unprotected template only by extracting some partial information from the protected template. The first attack consists of reconstructing a block by replacing all its columns with the same column computed from averaging the activated indexes of the BF of the protected template. The second attack requires a training set of the form (M_{ID}, T_{ID}) where T_{ID} is the protected template concealing the iris code M_{ID}. The attack consists of reconstructing the iris code of a protected template from the test set by replacing each block with the block corresponding to the nearest BF belonging to the protected templates of the training set. This attack assumes that $K = 0$ which implies that it does not take into account neither the variability of the key among different subjects nor the effect of the key for the same subject. As reported by the authors, the experimental results of both attacks are ineffective.

[5]This is slightly different from [1], since for the same application we are considering two cases: same key for all users and different key per user.

3.1.2 | Second category BF-based BTPs

In order to address the linkability vulnerability of the first category BF-based BTPs, authors in [33] proposed a technique called *structure-preserving feature re-arrangement* to replace the XOR with the key before computing the BF, and thus the *second category BF-based BTP scheme* that we illustrate in Figure 2. This technique permutes the rows of a feature block according to a keyed random permutation to diffuse the statistical properties of a biometric feature vector and at the same time to preserve the biometric performance. Later, [34] uses the same technique with a minor addition, that is, after a row-wise permutation there is a circular shift within each column. However, this circular shifting does not contribute to the dissipation of the biometric information but rather might lead to some accuracy loss since different columns after shifting might result in the same column.

Gomez et al. [35] studied the unlinkability of any BTP scheme from an information theory perspective and proposed a linkability evaluation procedure (Section 5 in [35]). This procedure helps to assess whether two protected templates of a given BTP scheme are concealing the same or different biometric instances. This is determined only by observing the score that resulted from the BTP's comparison measure and comparing it with the prior mated score distribution and the prior unmated score distribution. The same work defined three degrees of unlinkability that are: *fully unlinkable*, *semi unlinkable*, and *fully linkable* templates. Authors in [35] tested their framework analysis on a HE-based BTP that uses Euclidean distance and reported that it is fully unlinkable while the BF-based BTP in [33] lies between fully unlinkable and semi unlinkable. Note that this procedure works only if the comparison score is known, however, for an HE-base BTPs this score can be hidden [36] and only the comparison outcome is revealed. Hence, this procedure studied the unlinkability of the underlying unprotected template instead of the one protected by HE.

3.2 | Homomorphic encryption-based BTP schemes

HE has been the centrepiece of many privacy-preserving schemes, in particular biometric recognition in the encrypted domain [37–39] as it allows processing of encrypted templates without decryption. The use of an IND-CPA[6] secure HE scheme guarantees unlinkability, irreversibility and confidentiality under the constraint of the hardness of the underlying mathematical problem. Unlike classical BTP schemes, HE-based BTPs provide template protection even for a remote biometric recognition since an encrypted template can be sent over an unprotected public channel as only the party holding the private key is able to decrypt, and thus the importance of key management in the design of HE-based BTPs. Hence, HE allows a distributed comparison between the client and the server where only the party with the disclosure right is entitled to learn the recognition outcome. Therefore, in this survey, we classify HE-based BTPs according to their key management approach: either a single key HE[7], where the template is encrypted with the public key of one of the parties and is decryptable with its private key, or threshold HE where the template is encrypted using a joint public key between the client and the server and is decryptable using their both partial private keys.

3.2.1 | Single key HE-based BTPs

The choice of a suitable HE scheme for designing a HE-based BTP scheme depends on the comparison measure that produces either a similarity score or a dissimilarity score. Some comparison measures (such as Hamming distance) can be efficiently implemented under encryption using only a PHE scheme while others that consume more multiplications (such as cosine similarity) can benefit from SIMD operations of a SWHE scheme or a FHE scheme to improve their efficiency under encryption. The design of a HE-based BTP scheme also depends on the recognition protocol architecture, the parties involved (such as client, authentication server and database server, where both the latter servers are sometimes combined as a single server), which party has the right to learn the recognition outcome based on which key management is handled.

For applications such as access control, the client is entitled to learn the recognition outcome. For instance, schemes such as [40–42] encrypt the reference template with the client's public key and stores the encrypted reference template on the server's database who computes the comparison measure under encryption and sends the final score encrypted to the client. While in other applications such as remote authentication to a service, the authentication server is entitled to learn the recognition outcome. For example, schemes such as [43–47] differentiate between an authentication server and a database server with the assumption that both do not collude. In these schemes, the reference template is encrypted with the authentication server's public key and stored on the database server. This time the database server performs the comparison under encryption and sends the encrypted final score to the authentication server. In both cases, the party, entitled to learn the recognition outcome, decrypts the encrypted final score and then compares it with the system's threshold; if the score exceeds the threshold then the party counts it as a match otherwise a no match. Hence, the comparison is not fully in the encrypted domain as the comparison with the threshold is performed after the decryption and the entitled party learns

[6]Indistinguishability under Chosen Plaintext Attack ensures that the encryption of the same plaintext twice yields two different ciphertexts. This property contributes to the dynamism of the protected template.

[7]Here, single key means that there is one single private key (decryption key) that is retained by one single party unlike in threshold HE where the private key is divided between more than one single party or in multi-key HE where each party holds its own private key.

more than what it needs to learn, the final score and the recognition outcome.

In some schemes, such as [39, 48], the reference template is encrypted with the client's public key although the authentication server is the entitled party. For the comparison measure, [48] uses the support vector machine (SVM) classifier while [39] uses the squared Euclidean distance (SED). During the enrolment of a given individual, in [48], the classifier is trained on several biometric samples of that individual and the encrypted reference template is formed by encrypting the classifier's parameters using the client's public key while in [39] the encrypted reference template is simply the encrypted feature vector.

During comparison, in [48], the client sends an encrypted freshly extracted feature vector to the authentication server who multiples them feature-wise with the encrypted reference template and a random value in order to blind[8] the individual products. Subsequently, the server sends these blinded products to the client who decrypts and adds them and then sends back the result to the server so that it cancels out the blinding to learn the final score based on which it makes its decision. Similarly, in [39] the server computes a blinded SED under encryption, sends the encrypted blinded final score to the client who decrypts it and sends it back. Then, the server removes the blinding from the blinded final score and performs the comparison with the threshold. Again in these cases the final score is revealed to the server and thus the comparison with the threshold is performed outside the encrypted domain.

3.2.2 | Threshold HE-based BTPs

The encryption of the reference template with the authentication server's public key, even if the encrypted reference template is stored on the database server, is unsafe since in case the authentication server intercepts the communication between the client and the database server or illegally obtains the encrypted reference template, the authentication server is able to decrypt the encrypted reference template and learns the clear template that is supposed to be protected. HE-based BTP schemes such as those in [36, 38] use a threshold variant of HE to encrypt the reference template in order to address the above mentioned limitation introduced by the use of a single key HE scheme. Hence, a threshold HE encrypted template cannot be decrypted by neither the client nor the server on their own but instead both of them need to participate in the decryption process, and thus a better control of the biometric data flow from both parties.

In general, the exposure of the final score, whether to the client or to the server, leaks the closeness between a freshly processed biometric data (probe) and the static previously processed biometric data (reference template) as well as the quality of a user's biometric modality. Taking advantage from HE that allows processing under encryption, [36] shows that the final score can be hidden. Moreover, [36] performs the comparison with the threshold under encryption and then reveals only the recognition outcome, *match* or *no match*, at the moment of decryption.

3.3 | BF-based BTPs versus HE-based BTPs

Both approaches present the pros and cons and differently satisfy the tradeoff efficiency-security which makes the binary decision difficult between these approaches. Table 2 summarises and compares BF-based BTP schemes and HE-based BTP schemes with respect to the privacy requirements of ISO/IEC 24,745, supported modalities and their nature, biometric recognition protocol, template's characteristics and performance of the overall BTP. Note that *malleability* indicates whether the protected template can be inconspicuously altered. A BF-based template can be modified by flipping activated/deactivated bits while the HE-based template can be modified by injecting ciphertexts to the encrypted template since HE is malleable by nature. Therefore, a verification mechanism needs to be applied along with BTP schemes to check the validity of the protected template and monitor the correctness of comparison operations.

4 | STUDY CASE OF IRIS RECOGNITION

4.1 | Biometric performance

For the study case of iris recognition, we evaluate the biometric performance of those BTP approaches on the IITD iris database (Version 1.0) [10] that comprises 224 different subjects and 5 samples of each eye per subject; we considered the left eye only. The iris-codes are of dimension 20×512 and were extracted using the Log-Gabor (LG) [49] feature extraction algorithm from the Iris Toolkit [50] that is available in [51]. In the following experiments, we perform 2240 mated[9] comparisons and 124,880 non-mated[10] comparisons. We implement those experiments in Python 3.9 for which the source code is publicly available.[11]

The biometric performance of HE-based BTPs is the same as the performance of the unprotected baseline system since HE performs the computations in the encrypted domain in the same way they are calculated in the plaintext domain. In this case, HE wraps the biometric data with encryption that is concealed by the key pair, the encryption, and decryption keys. This prevents the HE key pair from influencing the biometric performance since the encryption key is cancelled out by

[8]When a party is not in the possession of the decryption key and wants to protect its plaintext, the blinding technique is used to hide its encrypted plaintext via an addition or multiplication by a random value known to this party only.

[9]Samples coming from the same subject.

[10]Samples coming from different subjects.

[11]https://github.com/aminabassit/hybridBTP.

TABLE 2 Comparison table showing the advantages and disadvantages of each approach

BTP approaches	BF-based BTP		HE-based BTP	
Categories	**First category**	**Second category**	**Single key HE**	**Threshold HE**
Schemes	[25–30]	[33, 34]	[39–43, 47, 48]	[36, 38]
Irreversibility	✓	✓	✓	✓
Unlinkability	✗[a]	✓[b]	✓	✓
Confidentiality	✓	✓	✓	✓
Supported modalities	All	All	All	All
Supported features	Binary and integer	Binary and integer	Binary, integer and float	Binary, integer and float
Feature alignment	Not needed[c]	Needed[c,d]	Needed	Needed
Comparison	Centralised	Centralised	Centralised and distributed	Centralised and distributed
Malleability	Malleable	Malleable	Malleable	Malleable
Final score exposure	Exposed	Exposed	Can be hidden	Can be hidden
Template dynamism	Static[e]	Static[e]	Refreshable and randomisable	Refreshable and randomisable
Template size	Linear in number feature blocks and BF size	Linear in number feature blocks and BF size		
Runtime efficiency	Fast	Fast	Practical to slow[f]	Practical to slow[f]
Recognition accuracy	No accuracy loss	No accuracy loss	No accuracy loss	No accuracy loss

[a]Shown by [31, 32].
[b]Ref. [35] reports that it is slightly linkable.
[c]However, it compares BFs generated from the same block of features.
[d]For faces, it assumes pre-aligned images.
[e]Once it is generated, it cannot be refreshed.
[f]Depends on HE scheme security level.

the decryption key, leaving the comparison outcome exactly the same as in the unprotected system. Thus, the performance of HE-based BTP is independent of the chosen HE key pair. In contrast with HE-based BTPs, both BF-based BTPs require the use of a key to project the biometric sample onto the transformed domain upon which their templates are compared. This makes their biometric performance dependent on the key since the distance is measured without removing it.

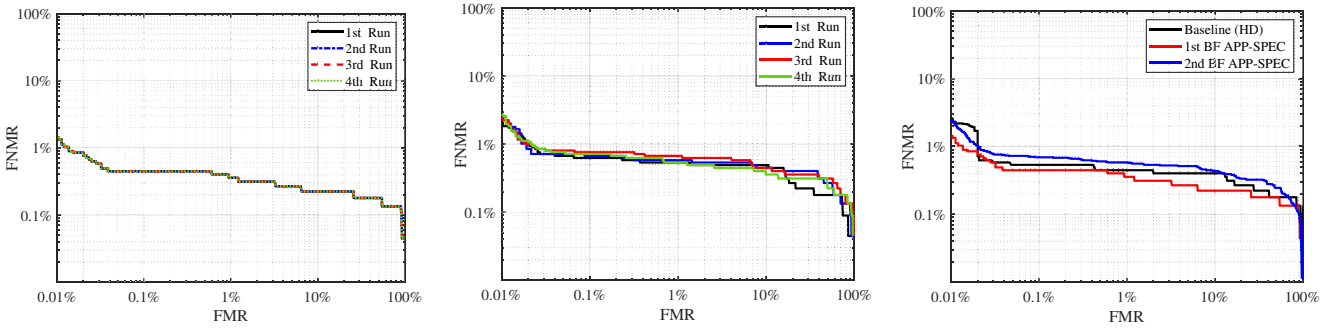
The unlinkability of HE-based BTPs is guaranteed by the probabilistic nature of the HE schemes, not by the use of the key, since fresh randomness is used at each encryption, even for the same plaintext, producing completely different ciphertexts that are unlinkable. Contrary to BF-based BTPs, the key must be different from one application to another to satisfy the unlinkability requirement. Given two BF-based templates from different applications, one cannot guess whether they belong to the same subject or not, thus preventing cross-matching databases. Unlike the HE approach, even if the same key is used, the resulting protected templates are indistinguishable, thus unlinkable.

However, within the same application, it is not clear from previous studies of both approaches [25, 33, 52] whether the key should be different from one subject to another. In this section, we measure the biometric performance of both approaches and distinguish between two cases that we call *application-specific key* and *user-specific key* only when considering the first and second categories' BF-based BTPs since technically their performances are related to the key. The application-specific key case is given when the same key is used to generate the BF-based templates for all subjects of this application, while the user-specific key case is given when a different key is used per subject. For the HE-based BTPs, we exclude them from these cases[12] since, in the case of iris recognition, the performance of HE-based BTPs is equal to the baseline unprotected system's performance.

4.1.1 | Application-specific key setting

Figure 4 shows the performance of the baseline unprotected iris recognition system that is the minimum Hamming distances between the iris-code reference template IC_R and eight circular shiftings to the right and eight others to the left of the iris-code probe IC_P. This is given by Equation (1) where $|\cdot|$ counts the number of ones, N is the length of the iris-code and $Shift_k$ indicates circular shifting to the right (resp. left) of the rows when k is positive (resp. negative).

[12]Those two cases could be discussed for the HE-based BTPs from a design perspective, and both of them can be achieved with the use of special HE schemes (e.g., Multi-Key FHE); however, this is out of the scope of this work.



(a) 1st BF-based BTP DETs where a different key is used per run (EER of 0.44%).

(b) 2nd BF-based BTP DETs where a different permutation is used per run (EERs between 0.53% and 0.66%).

(c) Comparison between the baseline (HE-based BTP), 1st BF-based BTP, and 2nd BF-based BTP.

FIGURE 3 Biometric performance of the first BF-based BTP and the second BF-based BTP on the IITD iris database in the application-specific key setting. The first and second BF DET curves, in (c), were generated from averaging the FMRs and FNMRs of their corresponding experiments plot in (a, b)

$$\text{ShiftedHD}(\text{IC}_\text{R}, \text{IC}_\text{P}) = \min_{k\in[-8,8]} \frac{|\text{IC}_\text{R} \oplus \text{Shift}_k(\text{IC}_\text{P})|}{N} \quad (1)$$

For both BF-based BTPs, we split the iris-codes into 32 blocks of dimension 10×32 each, as in [33]. In Figure 3a,b, we measure the performance of both BF-based BTPs with respect to different keys in the application-specific key setting. This is calculated by the normalized weighted Hamming distance between the BFs of blocks with the same index; the formula is given in Equation (2). We observe, in Figure 3a, that the first BF DETs are invariant as the keys differ, the DETs this figure overlay, which means that the first BF-based BTPs are independent of the key. While in Figure 3b, we notice that the permutation affects the performance of the second BF BTP since the three DETs overlap but never overlay as well as each permutation yields a different EER, although the shape of the DET is similar to the shape of the baseline system Figure 4.

$$\text{WHD}(\text{BF}_K(\text{IC}_\text{R}), \text{BF}_K(\text{IC}_\text{P})) = \frac{1}{b} \sum_{i\in[1,b]} \frac{|\text{BF}_i^\text{R} \oplus \text{BF}_i^\text{P}|}{|\text{BF}_i^\text{R}| + |\text{BF}_i^\text{P}|} \quad (2)$$

Figure 3c compares the performance of the three BTPs. We observe that the first BF-based BTP performs as good as the HE-based BTP (the unprotected baseline system), while the second BF-based BTP performs slightly less well than the others.

4.1.2 | User-specific key setting

In Figure 5a,b, we measure the performance of the first and second BF-based BTPs in the user-specific key setting where each subject generates its template using a different key. Hence, this will only impact the non-mated comparisons since they will be comparing two different BFs coming from different subjects and are generated with different keys, which yields higher non-mated scores. The mated comparisons are not affected by this setting since these experiments are exactly the same as in the previous experiments in Section 4.1.1 where the mated comparisons compared the BFs coming from the same subject and are generated using the same key. Surprisingly, the DETs in the user-specific key setting (Figure 5a,b) outperform the DETs in the application-specific key setting (Figure 3a,b). Moreover, in this user-specific key setting, we surprisingly observe a noticeable improvement of the biometric performance in the first BF BTP (achieves an EER of 0.17%) that surpasses even the baseline system (EER of 0.44%), as shown in Figure 5c where the red solid line DET curve is under all other DETs without overlapping with the others. This has motivated us to propose a hybrid BTP that we discuss in Section 5.

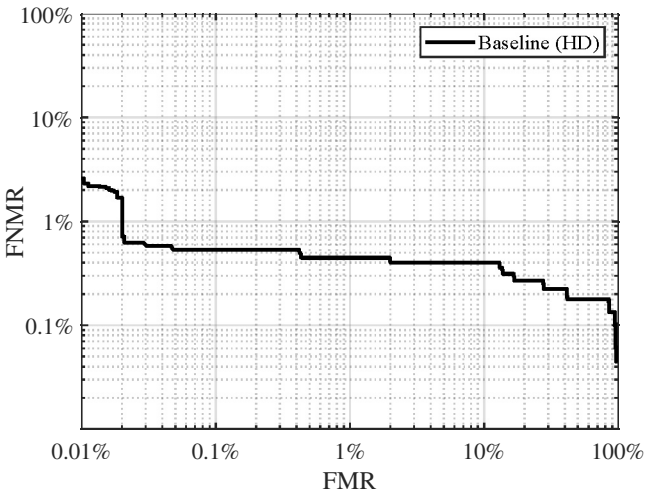
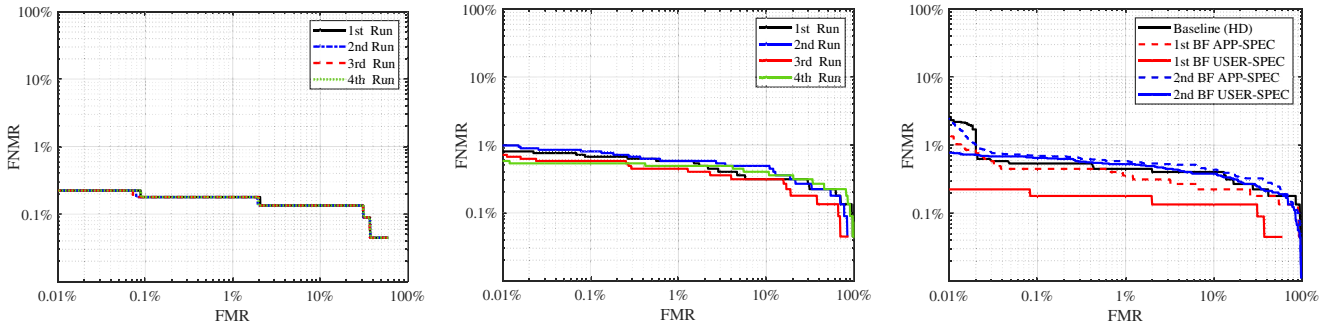


FIGURE 4 DET curve of the baseline biometric unprotected system calculated as shown in Equation (1) which also corresponds to the DET of the HE-based BTP tested over the IITD iris database. It achieves an EER of 0.44%

4.2 | Runtime performance

In order to fairly compare the runtime of the first BF-based BTP, second BF-based BTP and HE-based BTP in this iris



(a) 1st BF-based BTP DETs where a different key is used per run (EER of 0.17%).

(b) 2nd BF-based BTP DETs where a different permutation is used per run (EER between 0.44% and 0.61%).

(c) Comparison between the baseline (HE-based BTP), 1st BF-based BTP, 2nd BF-based BTP in both settings.

FIGURE 5 Biometric performance of the first BF-based BTP and the second BF-based BTP on the IITD iris database in the user-specific key setting. The first and second BF DET curves, in (c), were generated from averaging the FMRs and FNMRs of their corresponding experimental plots in (a, b) and Figure 3a,b

recognition study case, we implemented the three BTPs in C++ using the PALISADE library [53] for HE schemes and OpenMP [54] for parallelisation. We used a Linux Ubuntu 20.04.3 LTS machine run on a 64-bit computer Intel(R) Core i7-10,750H CPU with four cores (eight logical processors) rated at 2.60 GHz and 16 GB of memory. The source code of these experiments is publicly available.[13]

Figures 6 and 7 show the runtime of 1000 comparisons using first BF-based BTP, second BF-based BTP, and HE-based BTP over the IITD iris database. The runtime measured for the first BF-based and the second BF-based BTPs comprises the generation of the probe's BF-based template and the normalized weighted Hamming distance (Equation (2)). For the first BF-based BTP, we record a runtime of 0.725 ms, which is 1.2 times faster than the second BF-based BTP for which we record 0.905 ms. Both the first BF-based template and the second BF-based template are of the same size, which is 256 KB.

For HE-based BTP, we used the BFVrns scheme [17, 18, 55–57] under three security levels (128, 192, and 256 bits) to encrypt the iris-codes. The resulting encrypted iris-code is a vector of nC ciphertexts each packing nR rows of the iris-code (512 × nR plaintext bits). The values of nC and nR vary as the ring dimension varies; these values are given in Table 3.

For computing shifted Hamming distance (Equation (1)) under encryption, we rotate each ciphertext of the encrypted probe $[\![\mathrm{IC_P}]\!]$ by $k \in [-8, 8]$ positions,[14] then we use the resulting $\mathrm{Rot}_k([\![\mathrm{IC_P}]\!])$ together with the reference template ciphertext $[\![\mathrm{IC_R}]\!]$ to compute the XOR under encryption that becomes as in Equation (3), where $[\![.]\!]$ denotes the BFVrns scheme. The average runtime we recorded for a security level of 128 bits is 333.96 ms, 192 bits is 885.01 ms, and 256 bits is 1276.06 ms. For the 128 bits security level, the reference template size is 584 KB, while for both the 192 and 256 bits, the size is 1 MB.

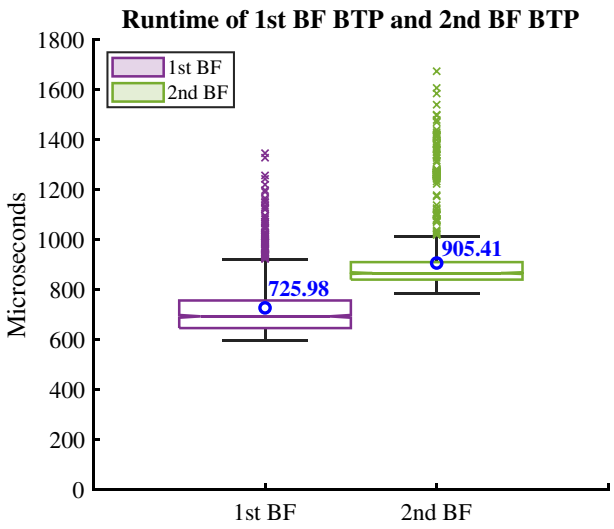


FIGURE 6 Verification runtime of first BF-based BTP and second BF-based BTP in microseconds (μs). The blue circles depict the average runtime

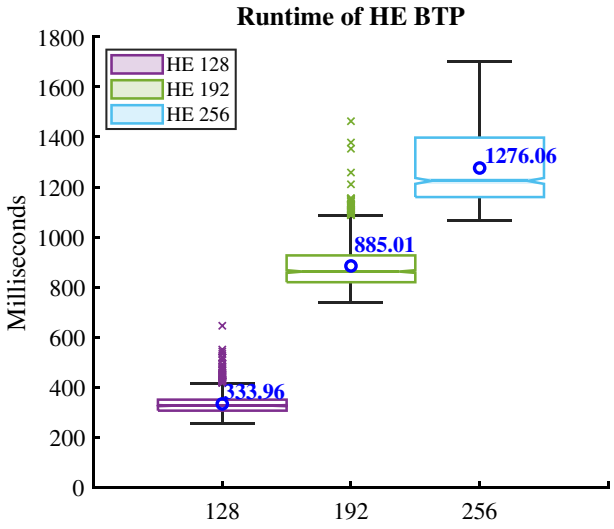


FIGURE 7 Verification runtime of HE-based BTP in milliseconds (ms). The blue circles depict the average runtime

[13] https://github.com/aminabassit/hybridBTP.

[14] Note that here, thanks to the packing technique, we rotate nR rows at once.

$$[\![\mathrm{IC_R}]\!] \oplus \mathrm{Rot}_k([\![\mathrm{IC_P}]\!]) := [\![\mathrm{IC_R}]\!] + \mathrm{Rot}_k([\![\mathrm{IC_P}]\!]) - 2 \cdot [\![\mathrm{IC_R}]\!] \cdot \mathrm{Rot}_k([\![\mathrm{IC_P}]\!]) \quad (3)$$

5 | HYBRID BTP SCHEME

On the one hand, our results in Section 4.1.2 show that the biometric performance of the first BF BTP in the user-specific setting surpasses the performance of both the second BF BTP and HE-based BTP. However, the first BF BTP is linkable as shown in previous works [32, 33]. On the other hand, although HE BTP preserves the biometric performance of the baseline system, HE BTP has the slowest runtime among those BTPs, as mentioned in Section 4.2 but guarantees the unlinkability by design. The second BF BTP is as fast as the first BF BTP and unlinkable in contrast to the first BF; however, its biometric performance is sensitive to the chosen permutation. It performs the least among the BTPs compared in this case study of iris recognition.

TABLE 3 nR and nC values depending on the chosen parameters for BFVrns implemented in PALISADE used in HE-based BTP

Security level	128 bits	192 bits	256 bits
Error distribution (σ)	3.2	3.2	3.2
CRT moduli sizes	30 bits	30 bits	30 bits
Plaintext modulus (p)	65,537	65,537	65,537
$\log_2(p)$	16	16	16
Ciphertext modulus (q)	1.23769×10^{27}	1.23747×10^{27}	1.32724×10^{36}
$\log_2(q)$	89.9997	89.9994	119.998
Ring dimension (n)	4096	8192	16,384
Packed rows per ciphertext (nR)	8	16	20
Ciphertexts (nC)	3	2	1
Rows packed in the last ciphertext	4	4	0
Reference template size	584 KB	1MB	1MB

5.1 | First category BF-HE hybrid BTP scheme

In order to address the linkability issue of the first BF BTP and compensate the slow runtime of HE based BTP for iris recognition that stems from the ciphertext rotations and the addition among plaintext slots, we merge both approaches by using the first BF as an efficient representation of the iris-codes that leads to an accurate biometric recognition and allows to speed up the homomorphic computations, since the shiftings are no longer needed. In addition, the HE layer reinforces the security of the first BF as well as preserves its biometric performance. Hence, the resulting hybrid BTP combines the best properties of both approaches. In this section, we formulate how a hybrid template is formed as well as how the dissimilarity scores are computed in such a BTP.

Figure 8 shows an overview of our proposed hybrid BTP that combines the first BF in the user-specific key setting, and the HE approaches. The architecture of this hybrid BTP follows is that of the semi-honest model. It comprises three parties: a user, a database server, and an authentication server, assuming no collusion between both servers. The user is the biometric data owner and has a key K_{U_i} used to generate the first BF-based template; this key must be kept secret on the user side. The database server is responsible for storing the encrypted first BF-based templates as well as computing the weighted Hamming distance under encryption. The authentication server is responsible for granting accesses and has a public-private key pair (pk_{AS}, sk_{AS}) corresponding to an HE scheme. In the enrolment phase, a user U_i receives pk_{AS} from the authentication server then generates his/her first BF-based template with K_{U_i}, encrypts, then sends the encrypted reference template $[\![1\mathrm{stBF^R}]\!]$ to the database server who stores it in its database for a later retrieval. In the verification phase, the user generates and encrypts the template corresponding to his/her probe and sends $[\![1\mathrm{stBF^P}]\!]$ to the database server who fetches the encrypted reference template $[\![1\mathrm{stBF^R}]\!]$ and compares it with the encrypted probe

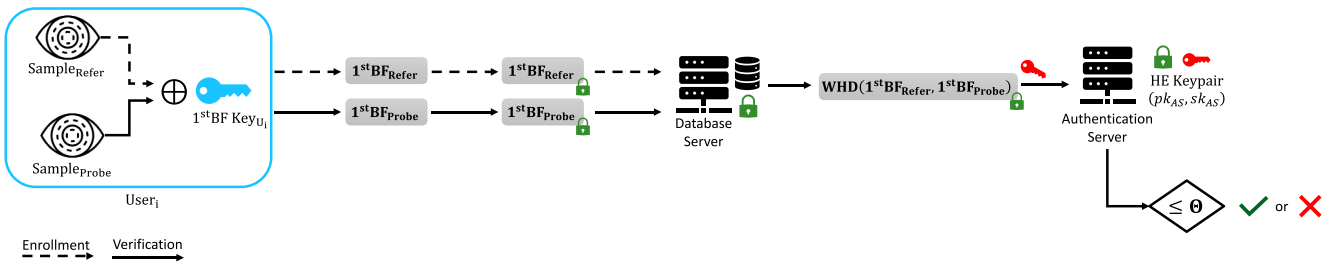


FIGURE 8 Overview of the hybrid BTP scheme. The blue rounded box depicts the region where the first BF-BTP in the user-specific key setting is applied. The blue key is specific to the user while the key pair is specific to the authentication server, green lock and red key for public and private keys respectively. The green lock on the grey boxes means encryption

by computing the weighted Hamming distance Equation (4) under encryption.

$$\mathrm{WHD}\left(\left[\!\left[1\mathrm{stBF}^{\mathrm{R}}\right]\!\right],\left[\!\left[1\mathrm{stBF}^{\mathrm{P}}\right]\!\right]\right) = \frac{1}{b}\sum_{i\in[1,b]}\frac{\mathrm{WHD}_{\mathrm{Num}}\left(\left[\!\left[1\mathrm{stBF}_i^{\mathrm{R}}\right]\!\right],\left[\!\left[1\mathrm{stBF}_i^{\mathrm{P}}\right]\!\right]\right)}{\mathrm{WHD}_{\mathrm{Denom}}\left(\left[\!\left[1\mathrm{stBF}_i^{\mathrm{R}}\right]\!\right],\left[\!\left[1\mathrm{stBF}_i^{\mathrm{P}}\right]\!\right]\right)} \quad (4)$$

Since it is a fraction, the numerator $[\![\mathrm{WHD}_{\mathrm{Num}}]\!]$ (Equation (5)) and denominator $[\![\mathrm{WHD}_{\mathrm{Denom}}]\!]$ (Equation (6)) are calculated separately.

$$\mathrm{WHD}_{\mathrm{Num}}\left(\left[\!\left[1\mathrm{stBF}_i^{\mathrm{R}}\right]\!\right],\left[\!\left[1\mathrm{stBF}_i^{\mathrm{P}}\right]\!\right]\right) = \sum_{j\in[1,\mathrm{len}_{\mathrm{BF}}]}\left(\left[\!\left[1\mathrm{stBF}_{i,j}^{\mathrm{R}}\right]\!\right] + \left[\!\left[1\mathrm{stBF}_{i,j}^{\mathrm{P}}\right]\!\right] - 2\cdot\left[\!\left[1\mathrm{stBF}_{i,j}^{\mathrm{R}}\right]\!\right]\cdot\left[\!\left[1\mathrm{stBF}_{i,j}^{\mathrm{P}}\right]\!\right]\right) \quad (5)$$

$$\mathrm{WHD}_{\mathrm{Denom}}\left(\left[\!\left[1\mathrm{stBF}_i^{\mathrm{R}}\right]\!\right],\left[\!\left[1\mathrm{stBF}_i^{\mathrm{P}}\right]\!\right]\right) = \sum_{j\in[1,\mathrm{len}_{\mathrm{BF}}]}\left(\left[\!\left[1\mathrm{stBF}_{i,j}^{\mathrm{R}}\right]\!\right]\right) + \sum_{j\in[1,\mathrm{len}_{\mathrm{BF}}]}\left(\left[\!\left[1\mathrm{stBF}_{i,j}^{\mathrm{P}}\right]\!\right]\right) \quad (6)$$

Once the authentication server receives the encrypted numerator and encrypted denominator from the database server, it decrypts them with its private key sk_{AS}, assembles their fraction, and divides it by b the number of Bloom filters to learn the dissimilarity score that is compared against a biometric threshold θ. If the score is below θ, the authentication server accepts; otherwise, it rejects.

Note that this hybrid BTP can be implemented in a multi-server setting where a user can enrol to multiple servers using his/her same secret key K_{U_i} to generate the first BF templates and encrypts them under the servers' corresponding public keys. The resulting hybrid templates are unlinkable since the HE scheme is IND-CPA.

5.2 | Evaluation

For assessing the efficiency of our proposed hybrid BTP, we measure over three different security levels, the verification runtime that includes the probe encryption, the weighted Hamming distance under encryption as given in Equation (4), the decryption of the numerator (Equation (5)), and the decryption of the denominator (Equation (6)). To generate the encrypted reference template (resp. probe), we use the SIMD property of the BFVrns scheme to process more Bloom filters at once and thus boost the runtime of our proposed hybrid BTP. As the first BF yields 32 BFs of size 1024, we pack as many BFs as the ring dimension allows, where the ring dimension is also the packed plaintexts capacity. In other words, nB Bloom filters of size 1024 each are packed per ciphertext yielding an encrypted reference template of length nC ciphertexts, $n\mathrm{C} = 32/n\mathrm{B}$ where 32 is the number of blocks that is the number of Bloom filters. The values of nC and nB vary as the ring dimension varies; these values are given in Table 4. Note that this packing of BFs preserves the biometric performance since the comparison is performed Bloom filter wise and the ring dimension is a multiple of the BF's size. In other terms, the packed BFs are compared against each other as in the plaintext BFs are compared.

In order to calculate the weighted Hamming distance (Equation (4)) over ciphertexts packing nB BFs each, we need to extract under encryption the individual encrypted numerators and the individual encrypted denominators per BF. To achieve this, for each ciphertexts $[\![1\mathrm{stBF}^{\mathrm{R}}]\!]_i$ and

TABLE 4 nB and nC values depending on the chosen parameters for BFVrns implemented in PALISADE used in Hybrid BTP

Security level	128 bits	192 bits	256 bits
Error distribution (σ)	3.2	3.2	3.2
CRT moduli sizes	30 bits	30 bits	30 bits
Plaintext modulus (p)	65,537	65,537	65,537
$\log_2(p)$	16	16	16
Ciphertext modulus (q)	1.23769×10^{27}	1.23747×10^{27}	1.23747×10^{27}
$\log_2(q)$	89.9997	89.9994	89.9994
Ring dimension (n)	4096	8192	8192
Packed BF per ciphertext (nB)	4	8	8
Ciphertexts (nC)	8	4	4
BFs packed in the last ciphertext	0	0	0
Reference template size	1.5 MB	1.5 MB	1.5 MB

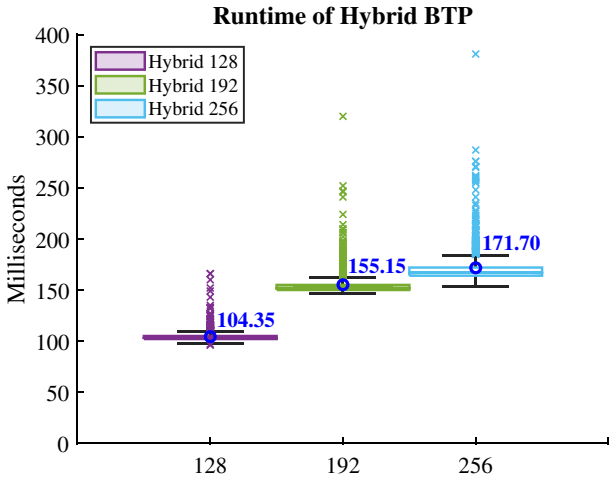


FIGURE 9 Verification runtime in milliseconds (ms) of the Hybrid BTP on three security levels. The blue circles depict the average runtime

TABLE 5 Comparison of our Hybrid BTP and [52] BTP

BTP schemes	Hybrid BTP			[52] BTP		
Dataset	IITD iris			IITD iris		
Implementation	C++			Python		
HE scheme	BFVrns			NTRU		
Security levels (bits)	128	192	256	128	192	256
EER (%)	0.17	0.17	0.17	-	-	-
155.15	0.4	0.4	0.4	0.45	0.36	0.39
FNMR (%)	0.17	0.17	0.17	18.7	2.10	1.83
Reference template encryption runtime (ms)	3	5.5	6.5	12.8×10^3	16.1×10^3	18.2×10^3
Probe encryption runtime (ms)	3	5.5	6.5	602	794	952
Verification runtime (ms)	104.35		171.70	643	837	1000

$[\![1\text{stBF}^{\text{P}}]\!]_i$ where $i \in [1, n\text{C}]$, we calculate the nB numerators at once and then add up batches of size equal to 1024, which is the size of one BF. Then, we extract the individual encrypted numerators and the individual encrypted denominators by rotating them to the indexes that are multiples of 1024 along the ring dimension. This packing technique allows us to significantly decrease the runtime and the hybrid template size since BFs are packed into a few ciphertexts to be processed at once.

Figure 9 shows the runtime performance of our proposed hybrid BTP on three security levels (126, 192, and 256 bits and the security parameters given in Table 4). It runs in 104.35 ms (resp. 155.15 and 171.70 ms) for 128 bit security level (resp. 192 and 256 bits) that is three times (resp. five times and seven times) faster than HE-based BTP (that runs in 333.96 , 885.01, 1276.06 ms for 128, 192 and 256 bits, respectively) and is three orders of magnitude slower than the first and second BF-based BTPs, as shown in Figures 6 and 7, respectively. The reference template size of the hybrid BTP equals 1.5 MB for the three security levels.

In Table 5, we compare our proposed hybrid BTP with the BTP presented in [52] since it is also an iris recognition scheme based on HE (NTRU scheme) and is evaluated on the same IITD iris database. In [52], the authors were the first to propose an early decision HE-based BTP scheme for iris recognition. Their technique is based on a reordering over the iris-code split into blocks. This reordering moves the significant bits of the iris-code to the beginning of the template to improve both the biometric performance and the runtime. Their technique consists of performing a block-wise comparison through which an early decision, early acceptance (i.e. a match) or early rejection (i.e. a non-match), can be made without parsing all blocks. Their system relies on this early decision technique to speed up the comparison in the encrypted domain; however, their biometric performance varies with the security levels. Since the block size gets large as the security level gets high, thus more bits are processed at once, leading to better biometric performance. In contrast to [52], our proposed hybrid BTP scheme has a stable biometric accuracy with respect to different security levels. For FMR of 0.4%, we achieve an FNMR of 0.17% that is 10 times more accurate than their best FNMR, which is 1.83% for a 256 bits security level. For the runtime, our proposed hybrid BTP runs 6 times (resp. 5 times) faster than [52] for 128 and 256 bits security levels (resp. 192).

6 | CONCLUSION

In this paper, we conducted a comparative study on the existing BF-based BTPs and HE-based BTPs by examining their advantages and disadvantages from a theoretical standpoint. We categorised them as: first BF-based BTPs [25], second BF-BTPs [33], and HE-based BTPs [39–43, 47, 48]. Then, we extended this comparison to iris recognition as a study case where we experimentally compared the biometric and runtime performances of the three BTP approaches on the same setting, dataset, and implementation language. Our results showed that in the user-specific key setting, the first BF-based BTP achieves the highest biometric accuracy among the three BTPs, however, it is the least secure since its linkability was shown by [31, 32]. Given the fact that the unlinkability requirement in HE-based BTPs is satisfied by the probabilistic nature of the underlying HE scheme and that HE preserves the biometric accuracy, we propose a hybrid BTP that combines the benefits of the first BF BTP with those of the HE BTP. The evaluation of the proposed scheme confirmed its biometric accuracy (an EER of 0.17% over the IITD iris database) and runtime efficiency (104.35 , 155.15 and 171.70 ms for 128, 192, and 256 bits security level, respectively).

ACKNOWLEDGEMENTS

This work was supported by the PriMa project that has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 860315. We would like to thank Christoph Busch and Marcel Grimmer for their inspiring lecture that motivated the research carried out. We also would

like to express our gratitude to Jascha Kolberg for providing us with the binary features of the IITD database.

CONFLICT OF INTEREST

The authors have no conflicts of interest to declare.

PERMISSION TO REPRODUCE MATERIALS FROM OTHER SOURCES

The authors have used the IIT Delhi Iris Database (Version 1.0) for which they were granted a license upon a request from the Indian Institute of Technology Delhi.

DATA AVAILABILITY STATEMENT

The IIT Delhi Iris database (Version 1.0) that supports the findings of this study is available from the Indian Institute of Technology–Delhi. Restrictions apply to the availability of these data, which were used under license for this study. The IIT Delhi Iris database (Version 1.0) is available from the Indian Institute of Technology–Delhi at http://www4.comp.polyu.edu.hk/~csajaykr/IITD/database_Iris.htm/ with the permission of the Indian Institute of Technology–Delhi.

ORCID

Amina Bassit https://orcid.org/0000-0002-1331-9702
Florian Hahn https://orcid.org/0000-0003-4049-5354
Raymond Veldhuis https://orcid.org/0000-0002-0381-5235
Andreas Peter https://orcid.org/0000-0003-2929-5001

REFERENCES

1. Bassit, A., et al.: Bloom filter vs homomorphic encryption: which approach protects the biometric data and satisfies ISO/IEC 24745?. Paper presented at the 2021 international conference of the biometrics special interest group (BIOSIG), pp. 1–6. IEEE (2021)
2. Jain, A.K., Nandakumar, K., Nagar, A.: Biometric template security. EURASIP J. Adv. Signal Process. (2008)
3. Sandhya, M., Prasad, M.V.: Biometric template protection: A systematic literature review of approaches and modalities. Biometr. Secur. Privacy (2017)
4. Secretary, I.C.: Information Technology – Security Techniques – Biometric Information Protection. Standard ISO/IEC 24745:2011, International Organization for Standardization. (2011). [Online]. https://www.iso.org/standard/52946.html
5. Broder, A., Mitzenmacher, M.: Network applications of Bloom filters: A survey. Internet Math. 1 (2004)
6. Geravand, S., Ahmadi, M.: Bloom filter applications in network security: A state-of-the-art survey. Comput. Netw. 57 (2013)
7. Martins, P., Sousa, L., Mariano, A.: A survey on fully homomorphic encryption: An engineering perspective. ACM Comput. Surv. 50 (2017)
8. Acar, A., et al.: A survey on homomorphic encryption schemes: Theory and implementation. ACM Comput. Surv. 51 (2018)
9. Wood, A., Najarian, K., Kahrobaei, D.: Homomorphic encryption for machine learning in medicine and bioinformatics. ACM Comput. Surv. 53 (2020)
10. Technology–Delhi, I. I. O.: IIT Delhi Iris Database (Version). (2007). [Online]. http://www4.comp.polyu.edu.hk/~csajaykr/IITD/Database_Iris.htm/
11. Daugman, J.: How iris recognition works. In: The Essential Guide to Image Processing, pp. 715–739. Elsevier (2009)
12. Kirsch, A., Mitzenmacher, M.: Less hashing, same performance: Building a better Bloom filter. Random Struct. Algorithms. 33 (2008)
13. Hurst, T.: Bloom filter calculator. (2009). https://hur.st/bloomfilter/
14. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. Paper presented at The International Conference On The Theory And Applications Of Cryptographic Techniques. Springer (1999)
15. ElGamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inf. Theor. 31 (1985)
16. Boneh, D., Goh, E.-J., Nissim, K.: Evaluating 2-DNF formulas on ciphertexts. In: The Theory Of Cryptography Conference. Springer (2005)
17. Brakerski, Z.: Fully homomorphic encryption without modulus switching from classical GAPSVP. In: Annual Cryptology Conference. Springer (2012)
18. Fan, J., Vercauteren, F.: Somewhat practical fully homomorphic encryption. IACR Cryptol. ePrint Arch. (2012)
19. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (Leveled) fully homomorphic encryption without bootstrapping. ACM Trans. Comput. Theory (TOCT). 6 (2014)
20. Cheon, J.H., et al.: Homomorphic encryption for arithmetic of approximate numbers. Paper presented at The International Conference on the Theory and Application of Cryptology And Information Security. Springer (2017)
21. Gentry, C., et al.: A Fully Homomorphic Encryption Scheme, vol. 20. Stanford University (2009)
22. Teoh, A.B., Kuan, Y.W., Lee, S.: Cancellable biometrics and annotations on biohash. Pattern Recogn. 41 (2008)
23. Sadhya, D., Singh, S.K.: Design of a cancelable biometric template protection scheme for fingerprints based on cryptographic hash functions. Multimedia Tools Appl. 77. (2018)
24. Kumar, N., et al.: Cancelable biometrics: A comprehensive survey. Artif. Intell. Rev. 53 (2020)
25. Rathgeb, C., Breitinger, F., Busch, C.: Alignment-free cancelable iris biometric templates based on adaptive Bloom filters. Paper presented at The International Conference on Biometrics (ICB). IEEE, (2013)
26. Rathgeb, C., et al.: On application of Bloom filters to iris biometrics. IET Biom. 3 (2014)
27. Rathgeb, C., Busch, C.: Cancelable multi-biometrics: Mixing iris-codes based on adaptive Bloom filters. Comput. Secur. 42 (2014)
28. Rathgeb, C., et al.: Towards bloom filter-based indexing of iris biometric data. Paper presented at The International Conference on Biometrics (ICB). IEEE, (2015)
29. Gomez-Barrero, M., et al.: Protected facial biometric templates based on local Gabor patterns and adaptive Bloom filters. Paper presented at The 22nd International Conference on Pattern Recognition. IEEE (2014)
30. Li, G., et al.: Towards generating protected fingerprint templates based on bloom filters. Paper presented at the 3rd International Workshop on Biometrics and Forensics (IWBF 2015). IEEE (2015)
31. Hermans, J., Mennink, B., Peeters, R.: When a Bloom filter is a doom filter: security assessment of a novel iris biometric template protection system. Paper presented at The International Conference of the Biometrics Special Interest Group (BIOSIG). IEEE, (2014)
32. Bringer, J., Morel, C., Rathgeb, C.: Security analysis and improvement of some biometric protected templates based on Bloom filters. Image Vis Comput. 58 (2017)
33. Gomez-Barrero, M., et al.: Unlinkable and irreversible biometric template protection based on Bloom filters. Informat. Sci. 370. (2016)
34. Martiri, E., et al.: Biometric template protection based on Bloom filters and honey templates. IET Biom. 6 (2017)
35. Gomez-Barrero, M., et al.: General framework to evaluate unlinkability in biometric template protection systems. IEEE Trans. Inf. Forensics Secur. 13 (2017)
36. Peeters, J., Peter, A., Veldhuis, R.N.: Fast and accurate likelihood ratio based biometric comparison in the encrypted domain. arXiv Preprint. arXiv:1705.09936 (2017)
37. Abidin, A., Mitrokotsa, A.: Security aspects of privacy-preserving biometric authentication based on ideal lattices and ring-LWE. Paper

presented at The IEEE International Workshop on Information FORENSICS AND security (WIFS). IEEE, (2014)
38. Karabat, C., et al.: Thrive: threshold homomorphic encryption based secure and privacy preserving biometric verification system. EURASIP J. Adv. Signal Process. (2015)
39. Im, J.-H., Jeon, S.-Y., Lee, M.-K.: Practical privacy-preserving face authentication for smartphones secure against malicious clients. IEEE Trans. Inf. Forensics Secur. 15 (2020)
40. Barni, M., et al.: A privacy-compliant fingerprint recognition system based on homomorphic encryption and fingercode templates. Paper presented at The Fourth IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS). IEEE, (2010)
41. Shahandashti, S.F., Safavi-Naini, R., Safa, N.A.: Reconciling user privacy and implicit authentication for mobile devices. Comput. Secur. 53 (2015)
42. Cheon, J.H., et al.: Ghostshell: Secure Biometric Authentication Using Integrity-Based Homomorphic Evaluations. IACR Cryptology ePrint Archive (2016)
43. Šeděnka, J., et al.: Secure outsourced biometric authentication with performance evaluation on smartphones. IEEE Trans. Inf. Forensics Secur. 10 (2014)
44. Gomez-Barrero, M., et al.: Implementation of fixed-length template protection based on homomorphic encryption with application to signature biometrics. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (2016)
45. Gomez-Barrero, M., Fierrez, J., Galbally, J.: Variable-length template protection based on homomorphic encryption with application to signature biometrics. Paper presented at The 4th International Conference on Biometrics and Forensics (IWBF). IEEE (2016)
46. Gomez-Barrero, M., et al.: Multi-biometric template protection based on homomorphic encryption. Pattern Recogn. 67 (2017)
47. Kolberg, J., et al.: Efficiency analysis of post-quantum-secure face template protection schemes based on homomorphic encryption. Paper presented at The International Conference of the Biometrics Special Interest Group (BIOSIG). IEEE, (2020)
48. Upmanyu, M., et al.: Blind authentication: A secure crypto-biometric verification protocol. IEEE Trans. Inf. Forensics Secur. 5 (2010)
49. Masek, L., et al.: Recognition of human iris patterns for biometric identification. Ph.D. dissertation (2003)
50. Rathgeb, C., et al.: Design decisions for an iris recognition SDK. In: Bowyer, K., Burge, M.J. (eds.) Handbook of Iris Recognition. Ser. Advances in Computer Vision and Pattern Recognition, 2nd ed. Springer (2016)
51. USIT – University of Salzburg Iris Toolkit. (2016). https://www.wavelab.at/sources/
52. Kolberg, J., et al.: Template protection based on homomorphic encryption: computationally efficient application to iris-biometric verification and identification. Paper presented at the IEEE international workshop on information forensics and security (WIFS). IEEE, (2019)
53. PALISADE Lattice Cryptography Library (Release 1.11.5). (2021). https://palisade-crypto.org/
54. Dagum, L., Menon, R.: OpenMP: an industry standard API for shared-memory programming. IEEE Comput. Sci. Eng. 5(1), (1998)
55. Halevi, S., Polyakov, Y., Shoup, V.: An improved RNS variant of the BFV homomorphic encryption scheme. Paper presented at The Cryptographers' Track at the RSA Conference, pp. 83–105. Springer (2019)
56. Bajard, J.-C., et al.: A full RNS variant of FV like somewhat homomorphic encryption schemes. Paper presented at The International Conference on Selected Areas in Cryptography, pp. 423–442. Springer (2016)
57. Al Badawi, A.Q.A., et al.: Implementation and performance evaluation of RNS variants of the BFV homomorphic encryption scheme. IEEE Trans. Emerg. Topics Comput. (2019)

How to cite this article: Bassit, A., et al.: Hybrid biometric template protection: Resolving the agony of choice between bloom filters and homomorphic encryption. IET Biome. 1–15 (2022). https://doi.org/10.1049/bme2.12075