Design on Face Recognition System with Privacy Preservation Based on Homomorphic Encryption

Yatao Yang1,2 · Qilin Zhang1,2 · Wenbin Gao2 · Chenghao Fan2 · Qinyuan Shu2 · Hang Yun2

Accepted: 20 October 2021 / Published online: 5 November 2021
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract
Face recognition is playing an increasingly important role in present society, and suffers from the privacy leakage in plaintext. Therefore, a recognition system based on homomorphic encryption that supports privacy preservation is designed and implemented in this paper. This system uses the CKKS algorithm in the SEAL library, latest homomorphic encryption achievement, to encrypt the normalized face feature vectors, and uses the FaceNet neural network to learn on the image’s ciphertext to achieve face classification. Finally, face recognition in ciphertext is accomplished. After been tested, the whole process of extracting feature vectors and encrypting a face image takes only about 1.712s in the developed system. The average time to compare a group of images in ciphertext is about 2.06s, and a group of images can be effectively recognized within 30 degrees of face bias, with a recognition accuracy of 96.71%. Compared to the face recognition scheme based on the Advanced Encryption Standard encryption algorithm in ciphertext proposed by Wang et al. in 2019, our scheme improves the recognition accuracy by 4.21%. Compared to the image recognition scheme based on the Elliptical encryption algorithm in ciphertext proposed by Kumar S et al. in 2018, the total spent time in our system is decreased by 76.2%. Therefore, our scheme has better operational efficiency and practical value while ensuring the users’ personal privacy. Compared to the face recognition systems in plaintext presented in recent years, our scheme has almost the same level on recognition accuracy and time efficiency.

Keywords Face recognition · Privacy preservation · Homomorphic encryption · FaceNet · SEAL

1 Introduction

As an important part of artificial intelligence, face recognition is widely used in hotel check-in and mobile payment, and has become an indispensable way of identity authentication in modern life. At the same time, with the hidden danger of face information leakage,
there is an increasing demand for face recognition schemes and systems that support privacy preservation.

For a long time, the ciphertext is irregular and difficult to operate, which limits the application of traditional encryption algorithms. To address this problem, the concept of homomorphic encryption was proposed by Rivest et al. [1] in 1978. This concept means that a certain operation in ciphertext space, such as multiplication and addition, can be mapped to the plaintext space with the same operation. A fully homomorphic encryption (FHE) scheme based on the ideal lattice was proposed by Gentry [2] in 2009, which has attracted great attention from cryptographers all over the world. The FHE scheme based on integers and polynomials was proposed by Smart et al. [3] in 2010, which shortened the key and ciphertext size. The DGHV scheme based on the FHE adopting the integer ring was proposed by Gentry et al. [4] in 2010.

The DeepID2 model was proposed by Sun et al.[5] In 2014, which can simultaneously train two types of data: face verification and face Classification. In 2015, Schroff et al [6] proposed the FaceNet neural network and designed a unified solution framework for face recognition: recognition, verification, search and other issues can be implemented in the feature space, which has achieved good performance.

For the traditional face recognition systems [7–11], the recognition processes were totally performed in the plaintext state, which provided potential attackers to steal and tamper user’s data, and traditional face recognition system cannot preserve user privacy any more. Therefore, the traditional face recognition system is not suitable for the rapid development of cloud technology today. In order to solve this problem, we combine homomorphic encryption with face recognition in this paper, which can not only provide security for data, but also provide privacy preservation for users.

In this paper, a face recognition system in ciphertext based on SEAL library is designed and implemented. This system has better application prospects than the system in plaintext. The main contributions of this paper are summarized as follows:

(1) A face recognition scheme based on SEAL library in ciphertext is proposed for the first time. Firstly, the CKKS algorithm in the SEAL library, latest homomorphic encryption result, is used to encrypt the normalized face feature vector. Secondly, the FaceNet neural network is used to learn on the ciphertext of the image to achieve face classification. Finally, the face recognition in the ciphertext is realized, and the privacy leakage of the user’s face information can be effectively avoided.

(2) A face recognition system based on SEAL library in ciphertext is implemented and tested. The whole process of extracting feature vectors and encrypting a face image takes only 1.712s on average. The average time spent by comparing a group of images in ciphertext is about 2.06s, and a group of images can be effectively recognized within 30 degrees of face bias. The recognition accuracy can reach 96.71%. While ensuring the personal privacy and security of users, this scheme has better operational efficiency and practical value compared to the face recognition systems [12–21] on recognition accuracy and time efficiency.

The structure of this paper is as follows. In the first and second section, we introduce the research background, development status of homomorphic encryption and face recognition. In the third section, we briefly introduce the homomorphic encryption algorithm and the FaceNet architecture, the data set and development tools used in the test are introduced at the same time. In the fourth section, we test all aspects of the
A neural network architecture for face recognition was proposed by Sun et al. [7] in 2015, which is called DeepID3. This architecture is based on two network architectures of VGG net and GoogleNet. The image in the gallery was converted into a sketch style by Wan et al. [22] in 2017, which reduced the amount of calculation when comparing with the original image. In 2015, the FaceNet architecture was proposed by Schroff et al. [6] of Google. Since then, the development of face recognition has entered a new stage. FaceNet is a very versatile system, which can be used in Face Verification, Classification and Clustering. FaceNet selects a vector space with a dimension of 128 and uses the loss function of triplets’ Maximum Boundary Neighbor Classification (LMNN) to train the neural network. FaceNet learns to map the face information in images to Euclidean space through convolutional neural network, and uses the Euclidean distance of feature points in the images to determine the similarity between images. The FaceNet algorithm uses two data training sets, called CASIA-WebFace and VGGFace2, also known as 20180408-102900 and 20180402-114759. FaceNet is tested on the widely used standard facial image data set, and achieves high recognition accuracy with excellent performance [23].

In 2015, the Learning with error (LWE) problem was reduced to a variant of the Approximate Greatest Common Divisor (AGCD) problem by Cheon et al. [24]. On this basis, a new AGCD-based FHE scheme was proposed, and has achieved better performance than any previous FHE schemes based on AGCD. In 2016, Jaschke and Armknecht [25] pointed out that a rational number can be approximately expressed as an integer by iteratively multiplying with a power of 2. In 2017, a more efficient method of representing fixed-point decimals was proposed by Dowlin et al. [26]. In 2017, an approximate FHE scheme was constructed by Cheon et al. [27], which uses rescaling technique to change the ciphertext modulus growth from exponential growth to linear growth while ensuring calculation accuracy, and improves the computational efficiency of the algorithm by using batch processing technique. In 2012, the BFV homomorphic scheme was proposed by Fan et al [28], based on which Microsoft disclosed Simple Encrypted Arithmetic Library (SEAL). In 2016, a Residue Number Systems (RNS) variant of the BFV scheme was proposed by Bajard et al. [29]. SEAL 2.3.0 was released by Microsoft in 2017. In 2018, a new technology for updating low-level ciphertext based on bootstrapping was proposed by Cheon et al. [30]. The application adopting homomorphic encryption algorithms has also begun to rise in various fields [31-32]. In 2018, Melchor et al. [33] compared the performance of HElib, SEAL and FV-NFLlib, and found that the performance of the SEAL library is better than HElib and FV-NFLlib in all aspects, so the SEAL library is selected as the development tool for this solution.
3 Preliminary Knowledge

3.1 Fully Homomorphic Encryption

Homomorphic encryption requires that the ciphertext data can be directly calculated without decryption. The same operation can be mapped to the plaintext space. Fully homomorphic encryption is a homomorphic encryption algorithm whose addition operation and multiplication operation works on ciphertext. For any operation \( f \) and plaintext \( m \), the following formula is satisfied when the algorithm is additive homomorphism and multiplication homomorphism.

\[
  f(Enc(m)) = Enc(f(m))
\]  

In the case of fully homomorphic encryption, the operations such as addition, subtraction, multiplication and division, polynomial evaluation, exponents, logarithms, trigonometric functions can be performed on the ciphertext. A fully homomorphic encryption scheme usually includes KeyGen, Enc, Dec and Evaluate these four algorithms.

The SEAL 3.0 released by Microsoft in 2018 supports the CKKS scheme proposed by Cheon et al. [27], and can realize the approximate calculation of ciphertext in the real number domain. Since the BFV scheme cannot directly encode, encrypt and operate on floating numbers, the floating numbers should be turned into integer numbers by multiplying these numbers by a magnification factor before operating on them. However, the introduction of the magnification results in a rapid accumulation and expansion of the ciphertext. The CounTeR(CRT) scheme can be used to solve this problem, but greatly increases the complexity of the ciphertext calculation. Compared to the BFV scheme, the CKKS scheme can directly encode, encrypt and operate on double-precision floating-point real numbers and complex numbers. SEAL 3.2 also can support .NET development, which made it easier for .NET developers to develop homomorphic encryption applications. The latest version of SEAL is SEAL3.4.

The CKKS scheme basically consists of those algorithms: key Generation, encryption, decryption, homomorphic addition and multiplication, and rescaling. For a positive integer \( q \), let \( R_q : = R/qR \) be the quotient ring of \( R \) modulo \( q \). Let \( \chi_s, \chi_r, \chi_e \) be distributions over \( R \) which output polynomials with small coefficients. These distributions, the initial modulus \( Q \), and the ring dimension \( n \) are predetermined before the key generation operation.

Key generation algorithm: Sample a secret polynomial \( s \leftarrow \chi_s \). Sample \( a, a' \) uniformly randomly from \( R^2 \), a public key \( pk = (b = -a \cdot s + e, a) \in R^2 \), and an evaluation key \( evk = (b' = -a' \cdot s + e', a) \in R^2 \).

Decryption algorithm: For a given ciphertext \( ct \in R^2_q \), output a message \( m' \leftarrow \langle ct, sk \rangle \mod q \).

Homomorphic addition operation: Given two ciphertexts \( ct \) and \( ct' \) in \( R^2_q \), output \( ct_{add} \leftarrow ct + ct' \in R^2_q \). The correctness holds as \( Dec(sk, ct_{add}) \approx Dec(sk, ct) \cdot Dec(sk', ct') \).

Homomorphic multiplication operation : Given two ciphertext \( ct = (c_0, c_1) \) and \( ct' = (c'_0, c'_1) \) in \( R^2_q \), compute \( (d_0, d_1, d_2) = (c_0 c'_0, c_0 c'_1 + c_1 c'_0, c_1 c'_1) \mod q \). Output \( ct_{mult} \leftarrow (d_0, d_1) + [P^{d_2} \cdot d_2 \cdot evk] \in R^2_q \).

Rescaling operation: Given a ciphertext \( ct \in R^2_q \) and a new modulus \( q' < q \), output a rescaled ciphertext \( ct_{rs} \leftarrow [(q'/q) \cdot ct] \in R^2_{q'} \).
3.2 FaceNet

In 2015, the FaceNet algorithm was proposed by Schroff et al. in Google. The algorithm can map image feature data to points in Euclidean space through learning, and the distance of the points directly corresponds to the similarity of the points.

Figure 1 illustrates the network structure of FaceNet. The front part is the same as CNN, followed by a feature normalization module to make its feature $||f(x)||^2 = 1$. Then, FaceNet proposes a new loss function Triplet Loss, which is the biggest feature of the algorithm. Through the Triplet Loss module, all image features will be mapped to a hypersphere, the feature distance between the same identities should be as small as possible, and the feature distance between different identities should be as large as possible. An example of Triplet Loss is as follows.

In Fig. 2, the left part of the figure is a representation of the distance between data features in the original space. Through learning, it can be ensured that the image of a specific person is closer to the image of the target object (positive) and far away from the image of the non-target object (negative). The optimization function adopted during learning is expressed as.

$$||x^a_i - x^p_i||_2^2 + \alpha < ||x^a_i - x^n_i||_2^2, \forall (f(x^a_i, x^p_i, x^n_i)) \in \tau$$

The loss function is expressed as follows.

$$L_{\text{min}} = \sum_i^N [||f(x^a_i) - f(x^p_i)||_2^2 - ||f(x^a_i) - f(x^n_i)||_2^2 + \alpha]_+$$

4 System Design

4.1 Overall Design

This system is developed based on python3.6, and completes the learning and training of FaceNet algorithm by calling various python libraries. The python libraries and versions required for this system are listed in Table 1.
The overall structure of this scheme is divided into the client and the server on the cloud. The client is responsible for the extraction of the face feature data, and then this data is encrypted and sent to the server on the cloud. After the server obtains the data, it calculates result in ciphertext and then returns the result in ciphertext to the client. During the whole process, both the data and the results are encoded in ciphertext. So this scheme achieve the purpose of protecting face feature privacy and security.

When the user uses it, the data training set is first learned by the FaceNet algorithm. The face information in the image is extracted by the FaceNet algorithm and normalized. The plaintext information obtained after these operations is put into the CKKS encryption scheme of the SEAL library for encryption. The system uses the order-preserving feature of the homomorphic encryption to compare the distance between the ciphertexts of the images and obtain the recognition results. The flow chart of the recognition algorithm is illustrated in Fig. 3 (Tables 2, 3 and 4).

4.2 FaceNet Data Training Set

There are two main types of FaceNet data training sets: 20180408-102900 and 20180402-114759. The basic information of the two data training sets is as follows.

There are 494,414 images of 10,575 persons in the CASIA-WebFace dataset. VGGFace2 is another large-scale face recognition data set, which contains a total of 9131 facial data. According to the test, 20180402-114759 derived from VGGFace2 has high recognition accuracy. So the system proposed in this paper is developed based on the 20180402-114759 data training set.

4.3 Selection of Training Data Set

Two major data training sets for the FaceNet algorithm are 20180402-114759 and 20180408-102900. We tested both the two data training sets in order to select the better one as the training data of this system.

The judgment value indicates the deviation degree of facial feature values from different images. After a lot of repeated training and testing, we set 0.7 as the threshold to judge whether two images are corresponding to the same person. If the judgement value is greater than 0.7, we consider that the two images correspond to different persons. Otherwise, they correspond to the same person. If the judgment value of two images of the same person is smaller and the judgment value of different persons’s images is larger, the recognition effect will be better and the data training set will meet our requirements.

According to the test results, the 20180402-114759 data training set can meet our requirements, and it is chosen as the training data set to be used in our system.

5 Testing and Analysis

As two very important indicators of the face recognition system, the efficiency and reliability of the system are tested in this section. The information of this test platform is as follows. The computer model is Lenovo 310S-14IKB, the CPU is Intel i5 7200U 2.4G, and the running memory is 4G. The operating system of the computer is Windows 10 Home, and the PyCharm based on python3.6 is used to run the system code.
5.1 Recognition Accuracy Test

5.1.1 Accurate Face Recognition Test

Firstly, we select the most frequently used LFW test data set in the FaceNet algorithm. Images of 40 persons were randomly selected for face recognition and comparison, and we conducted a total of 820 tests and conducted comparative tests based on different thresholds. The test results are shown in Table 5.
It can be seen from Table 5 that the recognition accuracy reaches the maximum when the threshold is 0.7, and the maximum recognition accuracy is 96.71% on the LFW test set. Therefore, we choose 0.7 as the judgment threshold. When the judgment value is not large than 0.7, we judge that the images correspond to the same person, otherwise the opposite.

Secondly, we take into account that there may be problems with irregular and non-standard image collection in actual situations. So the irregular and non-standard image collection is also taken into consideration in the test. Table 6 shows the test results.
Table 4  Test results of data training set 20180402-114759(6-10)

| Judgment value | 6     | 7  | 8     | 9     | 10    |
|----------------|-------|----|-------|-------|-------|
| 1              | 1.437864923 | 1.010112231 | 1.49955164 | 1.317951593 | 2.045026626 |
| 2              | 2.317579966 | 1.341110062 | 1.859040663 | 1.546797734 | 1.756720476 |
| 3              | 2.04752096  | 0.876740254 | 1.924247625 | 1.366852299 | 1.273279664 |
| 4              | 1.907548419 | 1.232143936 | 1.631334838 | 0.989025719 | 2.026135348 |
| 5              | 1.56298845  | 0.757405971 | 1.944442916 | 0.97836309  | 1.942069001 |
| 6              | 0.690749359 | 1.70332697  | 1.778362157 | 1.867852366 | 1.974373368 |
| 7              | 1.70332697  | 0.241359484 | 1.371275318 | 0.741099997 | 1.740024466 |
| 8              | 1.778362157 | 1.371275318 | 0.137172137 | 1.473936187 | 2.092824951 |
| 9              | 1.867852366 | 0.741099997 | 1.473936187 | 0.282951663 | 1.868859959 |
| 10             | 1.974373368 | 1.740024466 | 2.092824951 | 1.868859959 | 0.152850843 |

Table 5  Recognition accuracy for different thresholds

| Threshold | Recognition accuracy(%) |
|-----------|-------------------------|
| 0.66      | 93.78                   |
| 0.67      | 95.73                   |
| 0.68      | 96.34                   |
| 0.69      | 96.59                   |
| 0.7       | 96.71                   |
| 0.71      | 96.59                   |
| 0.72      | 96.46                   |
| 0.73      | 96.22                   |
| 0.74      | 95.85                   |
| 0.75      | 95.12                   |

Table 6  Face recognition accuracy in different situations

| User | A     | B     | C     | D     | E     | F     | G     | H     | I     |
|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1    | 0.2794| 0.4178| 1.0716| 0.2297| 0.3413| 0.5031| 1.2482| 0.6686| Error |
| 2    | 0.3641| 0.6216| 1.3264| 0.6919| 0.318 | 0.3152| 0.9428| 0.4971| 0.7834 |
| 3    | 0.3919| 0.656 | Error | 0.4064| 0.1716| 0.3902| 0.5569| 0.5589| 0.6997 |

Note: A: Head left for 30 degrees; B: Head left for 60 degrees; C: Head left for 90 degrees; D: Head up for 30 degrees; E: Head down for 30 degrees; F: Wearing glasses; G: Wearing mask; H: Covering one third of face; I: Covering half of face.

It can be seen from Table 6 that our system works effectively when the head deflection is less than 90 degrees and the face shielding area is small.
5.1.2 Multi-person’s Image Recognition Test

Reliability testing is not only to determine whether two images are corresponding to the same person, but also to accurately distinguish the feature values of different person’s images. We used 20 images of 10 persons and tested two different images of each person. For any two images, the threshold of the judgment value is 0.7. When the judgment value is less than or equal to 0.7, we consider the two images corresponding to the same person, otherwise the opposite. The test results are shown in Tables 7, 8.

As shown in these two tables, the average judgment value of the same face is 0.38513367, and the average judgement value of the different faces is 1.56347265. Therefore, the judgment value of images from the same person is within 0.7, and the judgment value of images from different persons is greater than 0.7 or even greater than 2. Thus, this scheme performs high recognition accuracy in the multi-person situation.

| Table 7 | Test results of multi-persons recognition |
|---------|-------------------------------------------|
| Judgment value | 1       | 2       | 3       | 4       | 5       |
| 1        | 0.191590444 | 2.085974566 | 1.665879482 | 0.955805684 | 1.108747767 |
| 2        | 2.085974566 | 0.668983675 | 1.865972053 | 1.963615144 | 1.632532211 |
| 3        | 1.665879482 | 1.865972053 | 0.664539289 | 1.418245682 | 1.331221488 |
| 4        | 0.955805684 | 1.963615144 | 1.418245682 | 0.501566885 | 1.124485028 |
| 5        | 1.108747767 | 1.632532211 | 1.331221488 | 1.124485028 | 0.319572919 |
| 6        | 1.437864923 | 2.317579966 | 2.04752096  | 1.907548419 | 1.56298845 |
| 7        | 1.01012231 | 1.341110062 | 0.876740254 | 1.232143936 | 0.757405971 |
| 8        | 1.49955164 | 1.859040663 | 1.924247625 | 1.631334838 | 1.944442916 |
| 9        | 1.317951593 | 1.546797734 | 1.366852299 | 0.989025719 | 0.97836309 |
| 10       | 2.045026626 | 1.756720476 | 1.273279664 | 2.026153548 | 1.942069001 |

| Table 8 | Test results of data training set 20180402-114759 |
|---------|-----------------------------------------------|
| Judgment value | 6       | 7       | 8       | 9       | 10      |
| 1        | 1.437864923 | 1.010112231 | 1.49955164 | 1.317951593 | 2.045026626 |
| 2        | 2.317579966 | 1.341110062 | 1.859040663 | 1.546797734 | 1.756720476 |
| 3        | 2.04752096  | 0.876740254 | 1.924247625 | 1.366852299 | 1.273279664 |
| 4        | 1.907548419 | 1.232143936 | 1.631334838 | 0.989025719 | 2.026135348 |
| 5        | 1.56298845 | 0.757405971 | 1.944442916 | 0.97836309 | 1.942069001 |
| 6        | 0.690749359 | 1.70332697  | 1.778362157 | 1.867852366 | 1.974373368 |
| 7        | 1.70332697 | 0.241359484 | 1.371275318 | 0.741099997 | 1.740024466 |
| 8        | 1.778362157 | 1.371275318 | 0.137172137 | 1.473936187 | 2.092824951 |
| 9        | 1.867852366 | 0.741099997 | 1.473936187 | 0.282951663 | 1.86859959 |
| 10       | 1.974373368 | 1.740024466 | 2.092824951 | 1.86859959 | 0.152850843 |
5.2 Efficiency Test

In the efficiency test, we randomly selected two groups of images, each including 20 sets of images. Each set is comprised of two different images of the same person.

We sorted and analyzed the time consumed by processing 120 images, and got the time probability distribution for each image, which is shown in Fig. 4.

The time $t_1$ for processing one image is concentrated in 1.3s-1.8s, with an average value of 1.712s.

To test the time $t_2$ consumed by the encryption operation, we also select 20 feature values for encryption, and the result is shown in Fig. 5.

It can be calculated from Figure 5 that the average encryption time of each image feature value is about 0.086s.

To evaluate the ciphertext comparison operation time $t_3$, we select 20 groups of ciphertexts for comparison, and the result is shown in Fig. 6.
Through the above test, the total time tall for ciphertext image identification can be obtained as follows:

\[ t_{all} = t_1 + t_2 + t_3 \]  \hspace{1cm} (4)

In the above formula, tall, t1, t2, and t3 are the total time, processing time, encryption time, and comparison time, respectively. The total time is about 2.06s. The processing time for two images is 1.712s, and accounting for 83.1% of the total time. The processing time is closely related to computer performance. If a computer with more powerful performance is used, the comparison time will be further reduced.

5.3 Performance Comparison

In order to further illustrate the comprehensive performance of this recognition scheme, we compared the latest relevant research results in the field of face recognition in ciphertext in the past 5 years, and obtained the results shown in Table 9.

As we can see from Table 9, scheme [18] and [21] are slightly higher than our scheme in recognition accuracy. Such as to the scheme [21], our scheme is 0.22% lower than it in recognition accuracy. However scheme [21] did not provide privacy preservation. To the scheme [18], the authors propose two new schemes for the secure computation in face recognition. First scheme relies face image database on the third-party server, which is easier to reveal user personal privacy information. At the same time, the overall scheme is even more complex. The second scheme computes and returns the results to the client by using the special protocol, which also adds some additional steps to complicate the scheme as a whole. Scheme [18] uses the Paillier algorithm to perform partial homomorphic encryption operations, this algorithm can only perform homomorphic encryption on the addition computing, but not perform the real number multiplication operation. Although literature [18] provides privacy preservation solution,
Table 9  Comparison of face recognition performance in ciphertext

| Scheme       | Encryption Algorithm | Processing time | Encryption time | Contrast time | Total time | Recognition accuracy | Privacy preservation |
|--------------|----------------------|-----------------|----------------|---------------|------------|----------------------|----------------------|
| [16]         | AES                  | –               | –              | –             | –          | 92.46%               | N                    |
| [17]         | Random unitary transform | –             | –              | 0.00164       | –          | 95.99%               | N                    |
| [18]         | Paillier             | –               | 0.102          | 0.00045       | –          | 99.27%               | Y                    |
| [19]         | RSA                  | –               | –              | –             | –          | 95.00%               | N                    |
| [20]         | Catalano-Fiore       | 0.62336         | –              | –             | 1.07       | 99%                 | N                    |
| [21]         | Elliptical           | –               | –              | –             | 8.67       | 96.89%               | N                    |
| This scheme  | CKKS                 | 1.712           | 0.086          | 0.264         | 2.06       | 96.67%               | Y                    |
it has inherent limitations in ciphertext operations when recognizing faces. The CKKS encryption scheme used in our solution can not only provide calculation operation for real number, but also provide the fully homomorphic encryption on multiplication and addition computing. Our scheme does not require any other specific third-party or any other complex protocol, which makes this scheme easier to deploy and expand.

From the Table 9, the encryption time of this scheme is 15.7% higher than the encryption time in [18]. The total time of this scheme is 76.2% higher than the total time of literature [21], and the recognition accuracy of this scheme is 4.21, 0.68, 1.67% higher than that of literature [16], [17], and [19], respectively.

In order to explain the advantages in our scheme, We also compared some recent achievements on face recognition in plaintext [12–15]. The maximum performance difference between those achievements and our scheme in recognition accuracy is only 2.3%. In other words, considering the security and the privacy preservation of the user face data, this difference is acceptable and negligible.

Obviously, our scheme has a better comprehensive performance.

### 6 Summary

In this paper, a ciphertext face recognition scheme and system based on SEAL library and FaceNet is proposed by combining homomorphic encryption and face recognition. Firstly, the normalized facial feature vector is encrypted using the CKKS algorithm in the SEAL library, Microsoft’s latest homomorphic encryption achievement. Secondly, the FaceNet neural network is used to perform machine learning on the ciphertext of images to realize face classification. Finally, face recognition in ciphertext is realized. Through the test, the average time to compare a group of images in ciphertext is about 2.06s. Our system works effectively when the head deflection is less than 90 degrees and the face shielding area is small. Compared with the plaintext face recognition system proposed in the past 5 years, this scheme not only has almost the same level of recognition accuracy, time efficiency and better practical value, but also provide privacy preservation to users. In practice, the ciphertext information can be stored in a cloud database, and cloud computing is used to overcome the problem that encryption and decryption cost too much time in ciphertext recognition. On the whole, this scheme and system have good reference value and application prospects in the fields of ciphertext face recognition, privacy preservation, and identity authentication.

**Data Availability**  Not applicable.

**Declarations**

**Funding** This work was supported by The State Cryptography Development Fund of Thirteen Five-year (MMJJ20170110).

**Conflict of interest** The authors declare that they have no Conflict of interest.
References

1. Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems[J]. *Communications of the ACM*, 21(2), 120–126.
2. Gentry, C. (2009). Fully homomorphic encryption using ideal lattices[C]. *Proceedings of the forty-first annual ACM symposium on Theory of computing*, 1, 169–178.
3. Smart, N. P., & Vercauteren, F. (2010). Fully homomorphic encryption with relatively small key and ciphertext sizes[C]. *420–443*. Springer, Berlin, Heidelberg: International Workshop on Public Key Cryptography.
4. Van Dijk, M., Gentry, C., & Halevi, S., et al. (2010). Fully homomorphic encryption over the integers[C]. 24-43. Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, Berlin, Heidelberg
5. Sun, Y., Chen, Y., & Wang, X., et al. (2014). Deep learning face representation by joint identification-verification[C]. *Advances in neural information processing systems.*, 1988–1996
6. Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering[C]. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 815–823.
7. Sun, Y., Wang, X., & Tang, X. (2015). Deeply learned face representations are sparse, selective, and robust[C]. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2892–2900.
8. Wen, Y., Zhang, K., Li, Z., et al. (2016). A discriminative feature learning approach for deep face recognition[C]. *European conference on computer vision*, Springer, 9911, 499–515.
9. Guo, Y., & Zhang, L. One-shot face recognition by promoting underrepresented classes[J]. arXiv preprint arXiv:1707.05574(2017).
10. Deng, J., Zhou, Y., & Zafeiriou, S. (2017). Marginal loss for deep face recognition[C]. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 60–68.
11. Chen, B., Deng, W., & Du, J. (2017). Noisy softmax: Improving the generalization ability of dcmn via postponing the early softmax saturation[C]. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5372–5381.
12. Zhang, X., Fang, Z., Wen, Y., et al. (2017). Range loss for deep face recognition with long-tailed training data[C]. *Proceedings of the IEEE International Conference on Computer Vision*, 5409–5418.
13. Liu, Y., Li, H., Wang, X. Rethinking feature discrimination and polymerization for large-scale recognition[J]. arXiv preprint arXiv:1710.00870(2017).
14. Wu, W., Kan, M., Liu, X., et al. (2017). Recursive spatial transformer (rest) for alignment-free face recognition[C]. *Proceedings of the IEEE International Conference on Computer Vision*, 3772–3780.
15. Liu, W., Wen, Y., Yu, Z., et al. (2017). Sphereface: Deep hypersphere embedding for face recognition[C]. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 212–220.
16. Wang, X., Xue, H., Liu, X., et al. (2019). A privacy-preserving edge computation-based face verification system for user authentication[J]. *IEEE Access*, 7, 14186–14197.
17. Wang, Y., & Nakachi, T. (2020). A Privacy-Preserving Learning Framework for Face Recognition in Edge and Cloud Networks[J]. *IEEE Access*, 8, 136056–136070.
18. Jin, X., Han, Q., & Li, X., et al. (2020). Efficient blind face recognition in the cloud[J]. *Multimedia Tools and Applications*, 1–18.
19. Troncoso-Pastoriza, J. R., González-Jiménez, D., & Pérez-González, F. (2013). Fully private noninteractive face verification[J]. *IEEE Transactions on Information Forensics and Security*, 8, 1101–1114.
20. Im, J. H., Jeon, S. Y., & Lee, M. K. (2020). Practical privacy-preserving face authentication for smartphones secure against malicious clients[J]. *IEEE Transactions on Information Forensics and Security*, 15, 2386–2401.
21. Kumar, S., Singh, S. K., Singh, A. K., et al. (2018). Privacy preserving security using biometrics in cloud computing[J]. *Multimedia Tools and Applications*, 77, 11017–11039.
22. Wan, W., & Lee, H.J. (2017). FaceNet Based Face Sketch Recognition[C], 432-436, 2017 International Conference on Computational Science and Computational Intelligence (CSCI 2017), IEEE, Las Vegas, NV, USA
23. William, I., Rachmawanto, E.H., & Santoso, H.A., et al. (2019). Face recognition using facenet (Survey, Performance Test, and Comparison)[C], 1-6, 2019 Fourth International Conference on Informatics and Computing (ICIC), IEEE, Semarang, Indonesia
24. Cheon, J.H, & Stehle, D. (2015). Fully homomophic encryption over the integers revisited[C], 513-536, Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, Berlin, Heidelberg(2015).
25. Jäschke, A., & Armknecht, F. (2016). Accelerating homomorphic computations on rational numbers[C], 405-423, International Conference on Applied Cryptography and Network Security, Springer, Cham

26. Dowlin, N., Gilad-Bachrach, R., Laine, K., et al. (2017). Manual for using homomorphic encryption for bioinformatics[J]. Proceedings of the IEEE, 105, 552–567.

27. Cheon, J.H., Kim, A., & Kim, M., et al. (2017). Homomorphic encryption for arithmetic of approximate numbers[C], 409-437, International Conference on the Theory and Application of Cryptology and Information Security, Springer, Cham

28. Fan, J., & Vercauteren, F. (2012). Somewhat Practical Fully Homomorphic Encryption[J]. IACR Cryptol. ePrint Arch., 2012, 144.

29. Bajard, J.C., Eynard, J., Hasan, M.A., et al. (2016). A full RNS variant of FV like somewhat homomorphic encryption schemes[C], 423-442, International Conference on Selected Areas in Cryptography, Springer, Cham

30. Cheon, J.H., Han, K., & Kim, A., et al. (2018). Bootstrapping for approximate homomorphic encryption[C], 360-384, Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, Cham

31. Yatao, Y., Yang, Z., Qilin, Z., et al. (2020). Weighted electronic voting system with homomorphic encryption based on SEAL[J]. Chinese Journal of Computers, 043, 711–723.

32. Yatao, Y. A. N. G., & Yang, Z. H. A. O. (2021). Juannie ZHANG, Jierun HUANG, Yuan GAO, recent development of theory and application on homomorphic encryption[J]. Journal of Electronics and Information Technology, 43, 475–487.

33. Melchor, C.A., Kilijian, M.O., & Lefebvre, C., et al. (2018). A comparison of the homomorphic encryption libraries HElib, SEAL and FV-NFLlib[C], 425-442, International Conference on Security for Information Technology and Communications, Springer, Cham

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Yatao Yang born in 1978. Professor at Beijing Electronic Science and Technology Institute. Received his PhD degree from Beijing University of Posts and Telecommunications in 2009. His main research interest includes Information security, Cryptographic protocol and algorithm.
Qilin Zhang  Master student at Xidian University. His main research interest includes Information security, Cryptographic protocol and algorithm.

Wenbin Gao  born in 1989. He received the Ph.D degree in Information and communication engineering from Beijing Institute of Technology, Beijing, China, in 2018. Since 2018, he has been a lecturer with the Department of Electronics and Communication Engineering, Beijing Electronic Science and Technology Institute. His research interests include Information security, Cryptographic protocol and algorithm.

Chenghao Fan  His main research interest includes Information security, Cryptographic protocol and algorithm.
Qinyuan Shu  Her main research interest includes Information security, Cryptographic protocol and algorithm.

Hang Yun  His main research interest includes Information security, Cryptographic protocol and algorithm.