Secure Face Recognition Using Fully Homomorphic Encryption and Convolutional Neural Networks

Tao Liu
School of Information Engineering, Jiangsu Open University, Nanjing, Jiangsu 210036, China
Email: liut1980@hotmail.com

Keywords: homomorphic encryption, face recognition, feature vector, convolutional neural network

Received: June 14, 2024

As a unique physiological characteristic, facial information is considered privacy information. This paper combines fully homomorphic encryption technology with a convolutional neural network (CNN) algorithm to develop a face recognition system. The CNN used for extracting facial features was a conventional structure of an CNN, consisting of input layer, convolutional layer, pooling layer, and output layer. The only difference is that during training, triplet samples are used. The differences in convolutional features between the triplet samples were directly utilized as the loss function to train the algorithm. The trained CNN used the convolutional features as facial features. The facial feature vector was encrypted using fully homomorphic encryption technology. Then, the ciphertext was directly used for matching operations to achieve face recognition under encrypted conditions. Finally, simulation experiments were carried out. The simulation experiment used facial data from the Public Figures Face Database. The experiment tested the impact of encryption parameters on encryption effectiveness, as well as the matching performance and security of the face recognition system. The results showed that a high polynomial modulus combined with a low ciphertext coefficient modulus in the encryption parameters led to a decline in both recognition accuracy and efficiency of face recognition. When the ciphertext coefficient modulus was 256 and the polynomial modulus was 1,024, the performance of the system based on fully homomorphic encryption was optimal, achieving a recognition accuracy of 97.7% and an efficiency of 3.34 faces per second. When the matching threshold was 0.8, the recognition accuracy of the system under full-homomorphic encryption was the highest (98.7%). Under the same matching threshold, the recognition efficiency of the system under fully homomorphic encryption was higher than that of the traditional encryption (3.21 per second). In the face of third-party attacks, the face recognition system with fully homomorphic encryption realized the recognition and matching of face feature vectors without exposing plaintext.

Povzetek: Raziskava združuje popolnoma homomorfno šifriranje s konvolucijsko nevronsko mrežo (CNN) za varno prepoznavanje obrazov.

1 Introduction

With the rapid development of science and technology, along with the popularization of artificial intelligence, people's identity authentication methods in daily life are becoming more and more diverse. Facial recognition technology is a form of authentication technology using biological characteristics [1]. The face, as a biological feature, has uniqueness and is not easy to change. Moreover, the biometric characteristics are closely linked to the corresponding individual, making facial features a reliable basis for identity verification. Due to its reliance on the uniqueness and strong association of facial features, facial recognition technology has been implemented in various fields that require identity confirmation, such as security monitoring, identity authentication, and mobile payments [2]. However, the face, as a biological characteristic, is also considered private information. When using facial recognition technology, users must submit their facial images to the platform, which means they are transferring their private data [3]. If the facial data stored on the platform is leaked, not only will the user's privacy be at risk, but the leaked facial data could also be exploited for identity fraud. it is essential to implement encryption measures. The relevant research is shown in Table 1. This paper provides a brief introduction to the fully homomorphic encryption technology and the convolutional neural network (CNN) algorithm used for extracting facial feature vectors, and combined them to implement a face recognition system based on fully homomorphic encryption. By encrypting the facial feature vectors using fully homomorphic encryption technology, matching operations were directly performed using ciphertexts, achieving face recognition under encrypted conditions. Subsequently, simulation experiments were conducted.

Table 1: Relevant research.

Author	Research content	Research results
Ghosh et al. [4]	They proposed a decision fusion method based on fuzzy belief factor weighted evidence	The effectiveness of this method was verified through experiments.

	theory to realize face recognition.	
Bi et al. [5]	They proposed a new multi-objective genetic programming algorithm for feature learning in face recognition.	The algorithm achieved better face recognition performance.
Zhou et al. [6]	They proposed a chaotic glowworm swarm optimization algorithm based on cloud model for face recognition.	The excellence of this method was verified through simulations.

2 Fully homomorphic encryption technology

The two ciphertexts generated by homomorphic encryption technology, when using the same public and private keys, can also be decrypted with those same keys. The decrypted plaintext represents the result of the operation performed on the plaintexts corresponding to the two ciphertexts [7]. Homomorphic encryption can be divided into addition homomorphism and multiplication homomorphism based on the operational modes between ciphertexts. If both addition and multiplication operations can be performed on the ciphertexts of an encryption algorithm, then that algorithm is classified as a fully homomorphic encryption algorithm.

The fully homomorphic encryption algorithm used in this paper is as follows.

① The public and private keys are built:

$$\begin{cases} (sk, pk) = (s, ([-(as+e)]_q, a)) \\ R = \dfrac{Z[x]}{x^n+1} \\ s \in R_2 \\ a \in R_q \\ e \in \chi \end{cases}, (1)$$

where sk is the private key, pk is the public key, R is a polynomial ring function, R_2 is modular 2 polynomial residual class ring, R_q is modular q polynomial residual class ring, s is the value chosen from R_2, a is a value chosen from R_q, e is the value chosen from the error probability distribution χ of R, and χ stands for a Gaussian distribution [8].

② The public key is used to encrypt the plaintext:

$$\begin{cases} ct = (\lfloor \Delta m + p_0 u + e_1 \rceil_q, \lfloor p_1 u + e_2 \rceil_q) \\ p_0 = pk[0] \\ p_1 = pk[1] \\ \Delta = \left\lfloor \dfrac{q}{t} \right\rfloor \end{cases}, (2)$$

where ct is the ciphertext, m is the plaintext, p_0 and p_1 are the child public keys, t is the coefficient for reducing the plaintext polynomial [9], u, e_1, and e_2 are values chosen from χ.

③ The decryption formula is:

$$\begin{cases} m = \left\lfloor \left\lfloor \dfrac{t[c_0 + c_1 \cdot sk]_q}{q} \right\rfloor \right\rceil_t \\ c_0 = ct[0] \\ c_1 = ct[1] \end{cases}, (3)$$

where c_0 and c_1 are the values of the ciphertext after modular calculation.

3 Face feature vector extraction algorithm

With the strong one-to-one correspondence between faces and individuals, facial recognition technology can be applied to a variety of fields, including security monitoring and identity authentication [10]. In this paper, CNN is used to extract Euclidean spatial feature vectors from faces. The overall steps of extraction are image input — forward calculations in CNN — the output of the facial feature vector (dimension is set according to specific requirements).

Before using a CNN for face feature vector extraction, it is essential to train the CNN. In the traditional training method, once the sample is input into the CNN, the output is compared to the annotated result of the sample, and the CNN parameters are adjusted in reverse based on the discrepancy. However, this paper aims to use the CNN specifically for extracting the feature vector of a face, which poses a challenge since directly labeling the feature vector of a face image is difficult. Therefore, this paper uses triplet data [11] to train the CNN and adjust the loss function. The specific training steps are as follows.

① The face image is preprocessed, including reducing image noise, removing the face-independent background, and aligning the face image.

② The triplet sample of the face image is constructed, which contains two face pictures of the same person and one face picture of another person.

③ The triplet samples are input into the CNN for forward calculation. The convolutional layer uses convolution kernels to extract the image convolution features, and the pooling layer compresses the convolutional features. After calculating the complex convolution layer and pooling layer, the face feature vector of each sample is output in the fully connected layer.

④ The ternary loss function is calculated by the face feature vector obtained from the triplet sample calculation. The formula is:

$$loss = \sum_{i}^{N}\left[\left\|f(x_i^a)-f(x_i^p)\right\|_2^2-\left\|f(x_i^a)-f(x_i^n)\right\|_2^2+\alpha\right]_+ , \quad (4)$$

where $f()$ is the function that converts a face image into a feature vector, i.e., the CNN in this paper, x_i^a and x_i^p are two face images of the same person in the i-th triplet, x_i^n is the face image of another person in the i-th triplet, and α is the margin of the feature vector between the group of x_i^a, x_i^p and the group of x_i^a, x_i^n.

⑤ According to the error calculated by the ternary loss function [12], the weight parameters in the CNN are reversely adjusted. Then, it returns to step ③ for repetition until the ternary loss function converges stably to the preset threshold.

After the CNN is trained by the triplet, the feature vector of the face image can be obtained after the face image is input to the CNN for forward calculation.

4 Security protection of face recognition system based on fully homomorphic encryption

When using face recognition technology, users must first upload their facial feature information to the technical platform, or the platform may already possess the user's facial data. This facial information is considered private, and there is a risk of data leakage when the platform retains such sensitive information. Although traditional encryption methods can secure stored facial feature data, they still require decryption of the plaintext during the matching and recognition processes [13]. To further enhance the security of facial feature information, this paper proposes the use of a fully homomorphic encryption algorithm. This algorithm allows the face recognition system to perform calculations directly on the ciphertext of the facial feature data, enabling matching and recognition without the need for decryption. The specific process is as follows.

① The face recognition system generates a public-private key pair.

② In the registration stage, the client uses a CNN to extract the feature vector from the face image and performs fully homomorphic encryption on it. The ciphertext is uploaded to the server together with the user identifier (UID).

③ In the matching stage, the user inputs his face image again, uses the CNN to calculate the feature vector of the face image, and encrypts the face feature vector using the full-homomorphic encryption algorithm. Then, the user uploads the UID and face feature vector ciphertext to the server for matching.

④ In the matching process, the uploaded UID is first used to retrieve the stored face feature vector ciphertext corresponding to the UID from the server database. Then, the uploaded feature vector ciphertext is compared with the stored feature vector ciphertext. In this paper, cosine similarity [14] is used for comparison. The formula is:

$$\cos(c_X, c_Y) = \frac{\sum_{i=1}^{M} c_{X_i} \cdot c_{Y_i}}{\sqrt{\sum_{i=1}^{M} c_{X_i}^2} \cdot \sqrt{\sum_{i=1}^{M} c_{Y_i}^2}} , \quad (5)$$

where c_{X_i} and c_{Y_i} are the uploaded feature vector ciphertext and the stored feature vector ciphertext, c_{X_i} and c_{Y_i} are the uploaded i-dimensional feature vector ciphertext and the stored i-dimensional feature vector ciphertext. The closer the value of $\cos(c_X, c_Y)$ is to 1, the closer the two feature vectors are to each other. When $\cos(c_X, c_Y)$ is less than the preset threshold, it can be determined that they are matched.

5 Simulation experiment

5.1 Experimental data

The face data required for the simulation experiment were from PubFig: Public Figures Face Database.

5.2 Experimental settings

The simulation experiment was carried out in the server of the laboratory. Three servers were set up during the experiment, with server 1 as the client, server 2 as the server, and server 3 as the third-party attack end. The configuration of the three servers was the same, all of them were Windows11 operating system, 16 G memory, and i7 processor.

Table 2: CNN parameter settings.

Parameter	**Setting**	**Parameter**	**Setting**
Input layer	200 × 200 pixels	Convolution layer 1	16 convolution kernels (3 ×3), a step size of 2, and the sigmoid activation function

Convolution layer 2	32 convolution kernels (3 ×3), a step size of 2, and the sigmoid activation function	Pooling layer 1	Mean pooling [15], a 3 ×3 pooling box, a step size of 2
Convolution layer 3	64 convolution kernels (3 ×3), a step size of 2, and the sigmoid activation function	Pooling layer 2	Mean pooling, a 3 ×3 pooling box, a step size of 2
Output layer	128 nodes	Number of trainings	500

The relevant parameters of the CNN that were obtained through orthogonal experiments and were used to calculate the face feature vector are shown in Table 2. First, 60% of the face dataset was used as the training set to train the CNN, and 40% was used as the test set for the subsequent face recognition system based on fully homomorphic encryption.

First, the face images in the test set were classified. The face images of the same person were classified as one class, and a unique UID was generated for that person. After that, the trained CNN was used to calculate the face feature vector in the test set, and the dimension of the feature vector was set to 128. Then, the full-homomorphic encryption algorithm described above was employed to encrypt the face feature vector in the test set. A random face feature vector ciphertext under each UID was stored in server 2.

(1) The influence of different encryption parameters on the face recognition system with fully homomorphic encryption

The polynomial modulus is an important parameter to ensure the security of the encryption algorithm, and it must be set. The modulus of ciphertext coefficient is a large integer, and the larger it is, the higher noise upper limit it can provide for the encryption algorithm. The plaintext modulus is any positive integer, which determines the plaintext size that the encryption algorithm can encrypt at a time. In order to test the impact of encryption parameters on the face recognition system, this paper set the polynomial modulus to 1,024, 2,048, 4,096, 8,192, and 16,384, and the ciphertext coefficient modulus to 128, 192, and 256. The plaintext modulus is determined by the plaintext size. The recognition accuracy and efficiency of the face recognition system under different polynomial modulus and ciphertext coefficient modulus were tested. The above encryption parameters were commonly used fixed parameters in fully homomorphic encryption algorithms.

(2) The matching performance test of the face recognition system based on fully homomorphic encryption

During the matching test, the test set was input into the client of server 1, the client performed feature vector calculation and fully homomorphic encryption on the face image of the test set and then transfers the ciphertext and the UID corresponding to the ciphertext to server 2 to match the ciphertext with the stored feature vector ciphertext of the same UID. The matching degree was measured by cosine similarity. When the cosine similarity was higher than the preset threshold, it was determined that they were matched. The threshold values were set to 0.5, 0.6, 0.7, 0.8, 0.9, and 1.0 respectively, and the accuracy of face recognition under different thresholds was tested.

Moreover, the face recognition system without encryption and the system adopting the traditional encryption algorithm were also tested. The former did not encrypt the feature vector extracted by the CNN and directly compared the cosine similarity between the feature vectors. The latter used the traditional advanced encryption standard (AES) encryption algorithm to encrypt the feature vector extracted by the CNN, and the cosine similarity was decrypted before comparing the feature vector. The accuracy of face recognition under the threshold values of 0.5, 0.6, 0.7, 0.8, 0.9 and 1.0 was also tested.

(3) Security test of the face recognition system based on fully homomorphic encryption

In order to test the security of the system based on fully homomorphic encryption, server 3 can directly operate the data on server 2, so as to simulate the scenario that a third-party attacker has successfully penetrated the server. In the registration phase, server 3 grabs the face feature data stored in server 2 database. In the matching stage, server 3 grabs the face feature data of server 2 when the two face features were compared. The face recognition system without encryption and the system adopting the traditional encryption algorithm also performed the same situational operation.

5.3 Experimental results

As shown in Table 3, with the increase of the number of polynomial modulus in the encryption parameters, the recognition efficiency of the face recognition system decreased, and the recognition accuracy also decreased. With the increase of the modulus of the ciphertext coefficient in the encryption parameters, the recognition efficiency of the face recognition system had no obvious change, but the recognition accuracy was significantly improved.

Table 3: Face recognition performance under different encryption parameters.

Modulus of ciphertext coefficient	**128**		**192**		**256**	
Polynomial modulus	Recognition accuracy /%	Recognition efficiency n/s	Recognition accuracy /%	Recognition efficiency n/s	Recognition accuracy /%	Recognition efficiency n/s
1,024	87.5	3.34	92.3	3.33	97.7	3.34
2,048	87.6	3.22	92.4	3.21	97.8	3.22
4,096	85.5	2.87	90.2	2.88	94.6	2.86
8,192	81.4	2.06	86.4	2.07	89.3	2.06
16,384	75.8	1.65	81.3	1.66	83.5	1.65

Different values were set for the cosine similarity threshold for judging the matching degree of face feature vectors. The recognition accuracy and average recognition efficiency of the three systems under different recognition thresholds are shown in Table 4. As can be seen from the table, with the increase of the threshold, the recognition accuracy of the three systems showed a trend of rising first and then decreasing. When the threshold was 0.8, the face recognition accuracy was the highest. Under the same threshold, the recognition accuracy of the three systems was not very different. In terms of recognition efficiency, there was no significant difference between the system without encryption and the system with fully homomorphic encryption, while the recognition efficiency of the traditional system was significantly lower.

Table 4: Recognition accuracy and average recognition efficiency of three face recognition systems under different recognition thresholds.

Threshold	**The face recognition system without encryption**		**The face recognition system with encryption**		**The face recognition system with fully homomorphic encryption**	
	Recognition accuracy /%	Recognition efficiency n/s	Recognition accuracy /%	Recognition efficiency n/s	Recognition accuracy /%	Recognition efficiency n/s
0.5	70.3	3.22	71.5	1.02	72.1	3.21
0.6	82.8		82.9		83.4	
0.7	90.1		90.3		90.2	
0.8	97.5		97.8		98.7	
0.9	91.8		92.3		92.5	
1.0	81.2		81.9		82.4	

Table 5: Results of face feature data captured from three face recognition systems at different phases by third-party attackers.

Algorithm	**Registration phase**	**Matching phase**
The face recognition system without encryption	The face feature vector plaintext is captured.	The face feature vector plaintext can be captured.
The traditional face recognition system with encryption	Only the face feature vector ciphertext can be captured	Only the face feature vector plaintext can be captured.
The face recognition system with fully homomorphic encryption	Only the face feature vector ciphertext can be captured.	Only the face feature vector ciphertext can be captured.

Server 3 was used as a third-party attacker to capture face feature data from the three face recognition systems in the registration and matching stages, and the results are shown in Table 5. It can be seen that for the face recognition system without encryption, both in the registration stage and the matching stage, the face feature vector in the plaintext state was captured. For the traditional system, only the face feature vector ciphertext was captured in the registration stage, and the face feature vector plaintext was captured in the matching stage. For the system with fully homomorphic encryption, no matter in the registration stage or the matching stage, it only captured the face feature vector in the ciphertext state.

6 Discussion

In the digital era, facial recognition technology has been widely used in various fields such as unlocking smartphones, payment verification, and security monitoring due to its convenience and efficiency. However, with the popularization of this technology, users' concerns about personal information security and privacy protection are increasing. Traditional encryption methods are prone to leakage risks because they require decryption when processing encrypted data. To overcome this challenge, fully homomorphic encryption technology has emerged. This technology allows various calculations to be performed directly on encrypted data without decryption, enabling separation of data processing rights and data ownership.

In the facial recognition algorithm proposed in this paper, CNN was used to extract features from facial images and then compare them with feature data in the

database. Since the purpose of CNN is to extract feature vectors from facial images, the traditional one-to-one labeling method for samples is not suitable for facial feature vectors. Therefore, this article used a triplet sample form to train the CNN. The triplet consisted of two facial images of the same person (with different angles) and one facial image of another person. Then, the CNN extracted features from each of the three samples and used their corresponding face feature vectors to calculate a triplet loss function to adjust the parameters in the CNN.

In the subsequent simulation experiments, the impact of encryption parameters on encryption effectiveness, matching performance, and security of the face recognition system were tested. As shown in the previous text, increasing the polynomial modulus in the encryption parameters decreased both efficiency and accuracy of the face recognition algorithm. On the other hand, increasing ciphertext coefficient modulus enhanced recognition precision without significantly affecting efficiency. This is because polynomial modulus is a major factor influencing encryption schemes; as it increases, the ciphertext size also increases, thereby enhancing security but reducing computational efficiency. Increasing ciphertext coefficient modulus means there is greater noise budget during encryption, allowing for more homomorphic computations to avoid significant damage caused by zeroing out noise budget during homomorphic calculations.

With the increase of matching threshold, the recognition accuracy of all three facial recognition systems showed a trend of initially rising and then declining. When the threshold was set at 0.8, the facial recognition accuracy was highest. Under the same matching threshold, there was little difference in recognition accuracy among the three facial recognition systems. There was not much difference in recognition efficiency between unencrypted and fully homomorphic encryption systems, while traditional system had the lowest efficiency. The reasons were analyzed. When the matching threshold was small, part of the unmatched face were misjudged as matched, reducing the recognition accuracy, and when the matching threshold was high, the matched faces with slight calculation differences were misjudged as unmatched, reducing the recognition accuracy. Although some of the three algorithms encrypted the data, their essence was to use the cosine similarity between the face feature vectors to determine whether the images are matched, so the recognition accuracy was insignificantly different. In terms of recognition efficiency, the face recognition system without encryption directly computed the feature vector, the the algorithm with fully homomorphic encryption could directly calculate the ciphertext of the feature vector, so the recognition efficiency between them was similar. The traditional face recognition system could not directly calculate the ciphertext. It decrypted the ciphertext before calculating, so the recognition efficiency was lower.

Homomorphic encryption-based facial recognition algorithm ensures that no plaintext facial feature information is revealed during the entire registration and matching phase. The reason is that during the registration phase, encrypted data is uploaded and matching can be directly performed using the ciphertext without decryption. Throughout the entire process, plaintext will not appear.

The limitation of this study lies in the use of a single facial database for simulation experiments, so the future research direction is expanding the facial databases to improve the generalization of face recognition algorithms.

7 Conclusions

This paper combined fully homomorphic encryption technology with a CNN algorithm to develop a face recognition system based on fully homomorphic encryption. The face feature vector was encrypted using fully homomorphic encryption, and then matching operations were performed directly using the ciphertext to realize face recognition under encrypted conditions. Simulation experiments were conducted. It was observed that increasing the polynomial modulus in the encryption parameters decreased both the recognition efficiency and accuracy of the system. Conversely, increasing the ciphertext coefficient modulus improved the recognition accuracy without significantly affecting the recognition efficiency. With the increase of matching threshold, the recognition accuracy of the three systems showed a trend of first rising and then decreasing. The highest face recognition accuracy was achieved when the threshold was set at 0.8. Under the same matching threshold, there was little difference in the recognition accuracy among the three systems. The recognition efficiency of the system without encryption and with fully homomorphic encryption was slightly different, while the recognition efficiency of the traditional system was the least. The plaintext of the facial feature information did not appear during the registration and recognition stages of the face recognition algorithm employing fully homomorphic encryption.

References

[1] Li D, Huang L (2020). Reweighted sparse principal component analysis algorithm and its application in face recognition. *Journal of Computer Applications*, 40(3), pp. 717-722. https://doi.org/10.11772/j.issn.1001-9081.2019071270.

[2] Salim R J, Surantha N (2023). Masked face recognition by zeroing the masked region without model retraining. *International Journal of Innovative Computing, Information and Control*, 19(4), pp. 1087-1101.

[3] Huang F, Tang X, Li C, Ban D (2024). Cyclic style generative adversarial network for near infrared and visible light face recognition. *Applied Soft Computing*, 150, pp. 1-9.

[4] Ghosh M, Dey A, Kahali S (2024). A weighted fuzzy belief factor?based D-S evidence theory of sensor data fusion method and its application to face recognition. *Multimedia Tools and Applications*, 83(4), pp. 10637-10659.

[5] Bi Y, Xue B, Zhang M (2021). Multi-objective genetic programming for feature learning in face recognition. *Applied Soft Computing*, 103(4), pp. 1-14. https://doi.org/10.1016/j.asoc.2021.107152.

[6] Zhou G, Ouyang A, Xu Y (2020). Chaos glowworm swarm optimization algorithm based on cloud model for face recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 34(12), pp. 2056009.1-2056009.20. https://doi.org/10.1142/S0218001420560091.

[7] Ramaraj P (2021). A neural network in convolution with constant error carousel based long short term memory for better face recognition. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 12(2), pp. 2042-2052. https://doi.org/10.17762/turcomat.v12i2.1808

[8] Gao S, Wu R, Wang X, Liu J, Li Q, Tang X (2022). EFR-CSTP: Encryption for face recognition based on the chaos and semi-tensor product theory. *Information Science*, 621, pp. 766-781. https://doi.org/10.1016/j.ins.2022.11.121.

[9] Liu Y, Zhao F, Xu Y, Cao Y (2016). A novel face template protection algorithm based on the fusion of chaos theory and rsa encryption. *International Journal of Security & Its Applications*, 10(6), pp. 315-330. https://doi.org/10.14257/ijsia.2016.10.6.30.

[10] Antonijevic M, Strumberger I, Lazarevic S, Bacanin N, Mladenovic D, Jovanovic D (2022). Robust encrypted face recognition robot based on bit slicing and Fourier transform for cloud environments. *Journal of Electronic Imaging*, 31, pp. 061808-061808. https://doi.org/10.1117/1.JEI.31.6.061808.

[11] Liu J, Wang X, Chen B, Tu Z, Zhao K (2021). Outsourced secure face recognition based on ckks homomorphic encryption in cloud computing. *International Journal of Mobile Computing and Multimedia Communications*, 12(3), pp. 27-43. https://doi.org/10.4018/IJMCMC.2021070103.

[12] Masadeh S R, Zraqou J S, Alazab M (2018). A novel authentication and authorization model based on multiple encryption techniques for adopting secure e-learning system. *Journal of Theoretical and Applied Information Technology*, 96(6), pp. 1529-1537.

[13] Hahn V K, Sébastien Marcel (2023). Biometric template protection for neural-network-based face recognition systems: a survey of methods and evaluation techniques. *IEEE Transactions on Information Forensics and Security*, 18, pp. 639-666.

[14] Kang X B, Lin G F, Chen Y J, Zhao F, Zhang E, Jin C (2020). Robust and secure zero-watermarking algorithm for color images based on majority voting pattern and hyper-chaotic encryption. *Multimedia Tools and Applications*, 79(1/2), pp. 1169-1202. https://doi.org/10.1007/s11042-019-08191-y.

[15] Yu J, Yu X, Zhang L, Xie W (2023). A two-stage chaotic encryption algorithm for color face image based on circular diffusion. *Multimedia Tools and Applications*, 82(26), pp. 40009-40038. https://doi.org/10.1007/s11042-023-14804-4.