Volume-09 Issue 05, May-2025 ISSN: 2456-9348
Impact Factor: 8.232

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

UNTRUSTED CLOUD OUTSOURCING FOR LDA-BASED FACE RECOGNITION

G.Bhargavi,
Y.Vijaya Tulasi,
G.Bhavya Sree,
B. Tech Students, Dept. of Computer Science and Engineering,
R.V.R & J.C College of Engineering, Chowdavaram, Guntur, Andhra Pradesh, India

Dr. B. Vara Prasad Rao
Professor, Dept. of Computer Science and Engineering,
R.V.R & J.C College of Engineering, Chowdavaram, Guntur, Andhra Pradesh, India

ABSTRACT
In practical applications, face recognition has been widely used in public security and attendance systems. Many clients find it extremely challenging to use the linear discriminant analysis (LDA) method on their resource-constrained devices, such as laptop computers and cellphones, despite it being one of the most important algorithms in the field of face recognition. Contracting out calculations gives clients a viable way to complete demanding tasks with less processing power. In this work, we develop a protocol for outsourcing LDA-based face recognition to an untrusted cloud that can assist the client in concurrently performing matrix multiplication (MM), matrix inversion (MI), and eigenvalue decomposition (ED). The suggested outsourcing methodology can conceal the client's sensitive information from the cloud. More significantly, the client can use probability one to confirm whether the outsourcing results are accurate, making it impossible for the server to mislead the client. Furthermore, the suggested protocol significantly reduces the client's processing complexity, allowing the client to efficiently finish the LDA algorithm. Lastly, we put the technique into practice and provide a thorough assessment. The experimental findings show that the client achieves significant computing savings and that the suggested protocol's face recognition accuracy is nearly the same as that of the original LDA algorithm.

Keywords:
Secure outsourcing, face recognition, cloud computing, linear discriminant analysis, privacy protection.

1.INTRODUCTION

Face recognition has been used in many domains over the past 20 years, including identity authentication [3], public security [2], and attendance systems [1], thanks to the rapid advancement of deep learning and machine learning. We can maintain social security by apprehending criminals through the use of face recognition technology. However, because face recognition algorithms involve a lot of matrix operations, they frequently have high computational costs. An image is stored as a matrix, as is well known. Due to their limited processing power, many small terminals find it challenging to finish the entire face recognition process. The Internet has changed significantly in recent years due to the ongoing advancements in cloud computing. Cloud computing can lower computing costs and assist devices with limited resources in completing computing tasks [4]. A type of cloud computing service called Function-as-a-Service (FaaS) enables users with limited processing power to perform intricate computations [5]. One approach to implementing a serverless architecture is to build an application that adheres to FaaS. Because FaaS can be turned off and save computational costs when not in use, it is widely used in on-demand services [6]. Consequently, a lot of small terminals would rather outsource the gathered face images to the cloud, where the cloud server would perform the face recognition function. However, if the original data that has to be calculated is sent straight to the cloud, outsourcing computing will present a number of security threats and difficulties. Certain rogue servers have the potential to gather personal data and engage in illicit activities [7], [8]. The revelation of private information is the first obstacle. The client could suffer unimaginable costs as a result of the confidential information being revealed. As a result, the client should make sure that the outputs linked to the original data are securely shielded from the server in addition to encrypting the original data. Verifiability presents the second difficulty. As is well known, third parties who provide outsourcing services may not be entirely reliable and may produce inaccurate computing results in an effort to conserve computing resources. Furthermore, some computation errors could still occur even if the third party is completely reliable. Given the aforementioned circumstances, a verification algorithm is preferred in order to allow the customer to verify that the outputs of the outsourcing process are accurate. Efficiency is the third difficulty. In comparison to the original computation task, the client must make sure that outsourcing computation can significantly lower computation expenses and save computing. Otherwise, carrying out calculation for a limited terminal by outsourcing is pointless. Consequently, a trustworthy

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

protocol needs to be quick, safe, and verifiable. Up till now, numerous secure outsourcing procedures have been put forth. A secure verifiable outsourcing technique based on a co-occurrence matrix for feature extraction was proposed by Ren et al. in [9]. The entire operation was carried out on a single cloud server. In order to compress high-order Big Data employing garbled circuits, Feng et al. [10] suggested an outsourcing approach of orthogonal tensor singular value decomposition (SVD), which can be widely used in network security as well as network forensics. Fu et al. [11] resolved security issues by outsourcing non-negative matrix factorization to a hostile cloud. Paillier homomorphism is used in the suggested plan to safeguard photos. An effective protocol for outsourcing local binary patterns (LBP) was proposed by Xia et al. [12]. The server can directly extract encrypted LBP features for the application via this protocol. In their innovative paradigm for outsourcing location-based services to a semi-honest cloud, Zhu et al. [13] carefully considered query privacy, identity privacy, and the verifiability of the outsourcing outcomes. A safe searchable protocol that allows the cloud to search through encrypted data was provided by He et al. [14]. Sparse matrices were utilized by Zhao et al. [15] to safeguard the inputs and outputs in their secure and verifiable computation methodology. However, the aforementioned protocols are limited to addressing particular issues; they cannot be utilized to address other issues, including facial recognition computation outsourcing. Numerous traditional facial recognition methods exist, including the LBP algorithm [17], the LDA algorithm [18], the Eigenface algorithm [16], and others. These algorithms make it simple to finish facial recognition applications and improve our quality of life. The LDA algorithm is widely used in machine learning and can be used to reduce image dimension and perform image categorization. Improved LDA-based techniques are presented in [20], [21] to solve the issue of inadequate feature extraction, and the between-class scatter matrix is redesigned. The LDA technique requires the client to compute matrix inversion (MI), matrix multiplication (MM), and eigenvalue decomposition (ED). These steps have an $O(n^3)$computing cost. Some terminal devices find it extremely challenging to perform the appropriate computations for large-scale face photos, which could result in face recognition failures. As a result, using an outsourcing mechanism to implement the LDA-based facial recognition algorithm in practice is crucial and significant.

Presently, a few face recognition outsourcing procedures have been put forth to lessen the clients' computational burdens. Lei et al. suggested two outsourcing procedures for matrix inversion and matrix determinant in [22] and [23]. To conceal the images' secret information, these two techniques use matrix alterations to the original matrices. Zhou et al. presented two ED and SVD outsourcing schemes in [24] and used them with the principle component analysis (PCA) methodology, the first ED and SVD outsourcing protocol. However, by counting the greatest common divisor of the encrypted data, the malicious server can obtain the eigenvalues and eigenvectors. When Zhang et al. [25] outsourced PCA-based face recognition, they primarily presented two outsourcing protocols: eigenvalue decomposition and matrix multiplication. To safeguard the private information, the client must perform three encryptions and decryptions during the three interactions between the client and the cloud required by these protocols. As far as we are aware, no outsourced algorithm for LDA-based facial recognition has been put forth as of yet.

Our Input. In this research, we suggest an LDA-based face recognition non-interactive outsourcing technique. The following are the primary contributions:

1) To finish the outsourcing process, the client only needs to do one encryption and decryption. It is possible to lower the computing complexity from $O(n^3)$ to $O(n^2)$. The suggested protocol can save the client's computation time and lower communication expenses. The protocol can outsource three different types of matrix computations using a single encryption, significantly reducing client-cloud server contacts when compared to earlier ones.

2) The server also has private access to the original inputs and the actual outsourcing outcomes. In particular, the server multiplies auxiliary matrices to determine the inputs, which are unknown. Our protocol's outsourcing outcomes are shown to be computationally identical to a random vector and matrix, indicating that the outputs are likewise highly secure.

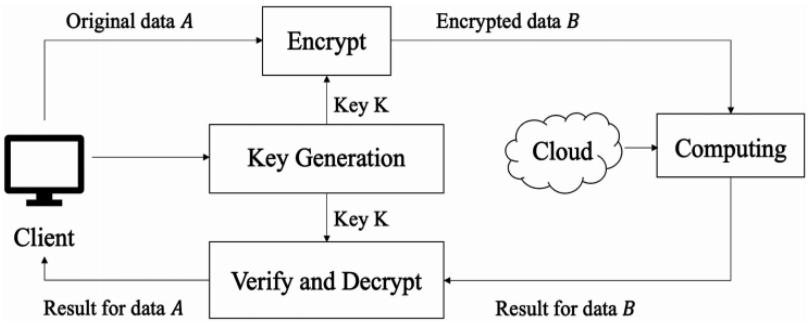

Fig. 1. Secure outsourcing computation model.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

3)The client can effectively verify the outcomes of the outsourcing. The client can use the verification algorithm to confirm that the findings are accurate, and there is an $1-\frac{1}{c_n^k}$ chance that the verification will be successful. There is a non-negligible chance that the wrong results will pass the verification process since the likelihood gets infinitely close to 1 as n and k increase. This is how the remainder of our paper is structured. The system model and framework are presented first in Section 2, followed by an introduction to LDA-based face recognition and a few eigenvalue decomposition theorems. The suggested LDA-based algorithm outsourcing protocol is shown in Section 3. In Section 4, we demonstrate the efficiency, security, and verifiability of the suggested outsourcing protocol. Section 5 presents the outcomes of the experiment. Finally, Section 6 brings the paper to a close.

2. MODELS AND DEFINITIONS

This section initially goes over the outsourcing computation framework and system model before introducing the LDA-based face recognition algorithm and a few eigenvalue decomposition theorems.

2.1 System Model and Framework

2.1.1. System Model : We imagine that the client needs to perform certain complex processes, but due to limited local processing power, he finds it extremely challenging to finish the work. Outsourcing data to the cloud is one possible solution for the client with restricted capabilities. The (sensitive) data is configured as A, as seen in Fig. 1. The client encrypts data A using a private key K to obtain encrypted data B since he expects the cloud to not receive any private information pertaining to A. Next, the client sends data B to the cloud which performs the corresponding calculations on B. The client verifies the result of data B after the cloud server sends it back. If yes, the client decrypts the encrypted result to obtain the computation result based on original data A; otherwise, the result is rejected.

2.1.2 Threat Models : The client may encounter numerous security risks when outsourcing compute, the majority of them stem from hostile assaults on the cloud server. Three different types of threat models are generally present in the outsourcing industry [26].

Honest but lazy model. In this concept, the cloud server will faithfully complete each step that the client requests, but it might return arbitrary outcomes as the actual ones that should be saved. computing power.

Sincere yet Inquisitive Model. In this architecture, the cloud server will honestly complete each step and provide the client with the outcomes. On the other hand, the cloud server might examine data and try to gather some personal data.

A malicious model. In addition to sending results to the client at random, the cloud server also wants to gather certain private information by examining the client's data. The malevolent model is undoubtedly the most potent one, and that is what this research examines.

2.1.3 Design Goals : The following design objectives must be met by the suggested outsourcing protocol.

Accuracy. If both the client and the cloud server follow the outsourcing protocol exactly, the client can eventually obtain the actual results by decrypting the encrypted results.

Privacy. The cloud cannot access the client's actual data or the actual computational output from the ciphertext and outsourcing outcomes when the protocol is in operation.

Verifiability. Any inaccurate findings cannot pass the verification procedure, which the client uses after getting the results to confirm whether they are true with a high likelihood.

Effectiveness. Outsourcing computation can significantly reduce clients computational overheads as compared to doing calculations directly. Otherwise, there is no point in outsourcing these compute duties to a cloud server.

2.1.4 Structure: To accomplish the aforementioned design objectives, the outsourcing protocol often incorporates the following algorithms.

Generation of Keys (1^k). The technique creates a secret key K upon receiving a security parameter k. This key is then used to both encrypt the original data and decrypt the output from the cloud server.

Encryption (A, K). The client obtains the ciphertext data B, which is then transmitted to the cloud server, after encrypting the raw data A using the secret key K.

Calculation (B). The calculations on Verification(α) are carried out by the cloud server. To confirm the computation results, the client uses the verification algorithm. The client accepts the findings if they pass the verification algorithm; if not, they reject them.

Decryption (α, K). In order to decode the confirmed results α and obtain the final results , the client uses secret key K.

2.2 LDA-Based Face Recognition Algorithm

2.2.1 An Overview of the LDA Algorithm

The LDA algorithm has been widely used in feature extraction, face recognition, and picture reduction.

It is frequently referred to as the traditional Fisher linear discriminant analysis approach. We provide a brief introduction to the processes of the LDA-based face recognition algorithm in accordance with [18], [19], [20], and [21]. Assume that there

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

are some samples of M classes and that there are N samples in total. With i ϵ [1,N] and j ϵ [1, M], we assume that the set of samples in each class is X_i and that the number of samples in each class is N_j. Additionally, we assume that every sample is X ={ x_1, x_2,..., x_N}. Following projection, m_j is the average value for each class, and Y_i is the collection of samples in the class. We can perform the calculation as follows using the data given.

Calculating the mean vectors average value:

$$m = \frac{1}{N}\sum_{i=1}^{N} x_i = \frac{1}{N}\sum_{j=1}^{M} N_j m_j \quad (1)$$

The within-class scatter matrix can be calculated:

$$S_w = \sum_{i=1}^{M}\sum_{x\in X_i}(x-m_i)(x-m_i)^T \quad (2)$$

The between-class scatter matrix can be computed:

$$S_b = \sum_{i=1}^{M} N_i\,(m_i-m)(m_i-m)^T \quad (3)$$

Calculate the matrix S :

$$S = S_w^{-1}S_b \quad (4)$$

The projection matrix W_{opt} is created by computing the eigenvalue σ_i and the accompanying eigenvector w_i of matrix S, choosing the eigenvectors that correspond to the L(L ≤ M-1) maximal eigenvalues, and then ensuring that it satisfies the Fisher decision criteria:

$$W_{opt} = \arg\max \frac{\left|w^T S_b w\right|}{\left|w^T S_w w\right|} \quad (5)$$

Lastly, the projection matrix is used to display the test samples. A distance measure classifier can be chosen to identify the face once the photos have been classed.

2.2.2 The Need to Outsource the LDA Algorithm: The client must preprocess the input image for the face recognition algorithm, performing operations such as geometric normalization and histogram equalization. The client must next convert the matrix of two-dimensional images into a column vector. Nevertheless, during the conversion, the dimension might increase significantly. For instance, if a 100 ×100 face image is transformed, the dimension of the within-class scatter matrix Sw is 10000 × 10000, whereas the column vector has a dimension of 100×100 = 10000. It has been shown in [22], [24], and [25] that using such a large-scale matrix for MI, MM, and ED on local devices can be quite time-consuming. As a result, the client with limited computing capacity finds it challenging to perform the calculation locally. We provide an LDA algorithm outsourcing mechanism to address the issue of LDA-based face recognition. The matrices S_w and S_b serve as the initial inputs. The cloud then determines the eigenvalues and eigenvectors of the matrix $S_w^{-1}S_b$, significantly lowering the client's local computation overhead.

2.3 Eigenvalue Decomposition :

Eigenvalues and eigenvectors of matrices are introduced before eigenvalue decomposition. We suppose that a non-zero vector x and a real number λ can make Eq. (6) hold for a given real matrix A ϵ $R^{n\times n}$. Then we call x and λ are the eigenvector and eigenvalue of matrix A, separately.

$$Ax = \lambda x \quad (6)$$

Eigenvalue decomposition breaks down a matrix into a product of its eigenvalues and eigenvectors, which takes the form of

$$AX = X\Lambda \quad (7)$$

where each column of matrix Xϵ $R^{n\times n}$ is the eigenvector of matrix A, and Λϵ $R^{n\times n}$ is a diagonal matrix with all of its diagonal elements being the eigenvalues of matrix A [27]. There are numerous uses for eigenvalues and eigenvectors in machine learning, including face recognition, latent semantic analysis, and data compression. We present two theorems concerning the eigenvalues and eigenvectors of matrices in accordance with [25] and [27].

Theorem 1. [27] states that if Gϵ $R^{n\times n}$ is an upper or lower triangular matrix and $G_1, G_2, \ldots, G_n$ are its diagonal elements, then $G_1, G_2, \ldots, G_n$ is its eigenvalue.

Theorem 2. [25]: Let Aϵ $R^{n\times n}$ represent a real matrix, and the associated eigenvectors and eigenvalues of A are

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

$\lambda_1, \lambda_2, \ldots \lambda_n$ along with $x_1, x_2, \ldots, x_n$. The eigenvalues of $G \epsilon R^{n \times n}$, which can be either an upper or lower triangular matrix, are the elements on its major diagonal, and the associated eigenvectors are $g_1, g_2, \ldots, g_n$. Allowing $O \epsilon R^{n \times n}$ to be a zero matrix, then mark $V=\begin{pmatrix} A & O \\ O & G \end{pmatrix}$ the eigenvalues of V are $\epsilon_1 = \lambda_1, \epsilon_2 = \lambda_2, \ldots, \epsilon_n = \lambda_n, \epsilon_{n+1} = G_1, \epsilon_{n+2} = G_2, \ldots, \epsilon_{2n} = G_n$, and eigenvectors of V are $v_1=\begin{pmatrix} x_1 \\ o \end{pmatrix}$, $v_2=\begin{pmatrix} x_2 \\ o \end{pmatrix}$, …, $v_n=\begin{pmatrix} x_n \\ o \end{pmatrix}$, $v_{n+1}=\begin{pmatrix} o \\ g_1 \end{pmatrix}$, $v_{n+2}=\begin{pmatrix} o \\ g_2 \end{pmatrix}$, …, $v_{2n}=\begin{pmatrix} o \\ g_n \end{pmatrix}$ where $o \epsilon R^{n \times 1}$ is a zero vector.

2.4 The indistinguishability of computation

This subsection will introduce the concept of computational indistinguishability [28]. Define $R \epsilon R^{n \times n}$ as a random matrix whose elements in the jth column are drawn from a uniform distribution in the interval $[-R_j, R_j]$, $\forall j \epsilon [1,n]$ If there is a small function m that ensures the following Eq. true, we claim that matrices Q and R are computationally indistinguishable for each probabilistic polynomial time distinguisher D:

$$\forall i, j, |Pr[D(q_{i,j}) = 1] - Pr[D(r_{i,j}) = 1]| < \mu \quad (8)$$

where the elements in matrices Q and R ith row and jth column are denoted by the symbols $q_{i,j}$ and $r_{i,j}$ respectively. Distinguisher D outputs one or zero if the input is identified as a random matrix that is uniformly distributed between $[-R_j, R_j]$ The elements of matrix Q and matrix R cannot be distinguished by a malevolent adversary, as demonstrated by Definition 1. In other words, the malevolent enemy cannot obtain any useful knowledge from matrix Q.

3 LDA-BASED FACIAL RECOGNITION THROUGH NON-INTERACTIVE,SECURE OUTSOURCING

In this part, we suggest an outsourcing technique for LDA-based facial recognition. Using the suggested outsourcing protocol, the client enters an encrypted matrix and obtains the original matrix's eigenvalues and eigenvectors.

3.1 Non-Interactive LDA-Based Outsourcing Algorithm

In an interactive outsourcing protocol, the client transmits the ciphertext of S_w after first encrypting the matrix $S_w \epsilon R^{n \times n}$ He then obtains S_w^{-1} after decrypting the results that the cloud returned. Second, the client decrypts the outsourcing results and obtains $S = S_w^{-1} S_b$ after submitting the ciphertexts of marriages S_w^{-1} and $S_b \epsilon R^{n \times n}$. Third, the server performs eigenvalue decomposition (ED) to obtain the eigenvalues and eigenvectors of $\bar{S}$ after the client encrypts S to obtain $\bar{S}$. The client finally decrypts the results and obtains the $S_w^{-1} S_b$ eigenvalues and eigenvectors.

Algorithm 1. Elementary Matrix Generation

Input: The number $2n$.
Output: Elementary matrices $P_i, i = 1, 2, \ldots, 2n$.
1: Set $\alpha_1 = 2n$ and $Y = \{1, 2, \ldots, 2n - 1\}$.
2: **for** $i = 1 : 2n$ **do**
3: Let P_i be an identity matrix and p_i be a random number in the interval $(-2^p, 2^p)$, where p is a positive constant.
4: Set $\beta_i = i$, and if β_i is in set Y, delete β_i from Y.
5: **if** $i > 1$ **then**
6: Set α_i to be an integral number in set Y.
7: **end if**
8: Delete α_i from set Y.
9: Add β_i to set Y if β_i is deleted at 4.
10: Let the element in the β_ith row and the α_ith column of matrix P_i be p_i.
11: **end for**

The client performs three encryptions and decryptions as well as three server interactions throughout the outsourcing process, requiring extremely high computational and communication expenses. This section presents a non-interactive outsourcing protocol of an LDA-based method to reduce the client's communication and computing overloads. The comprehensive procedure of the suggested outsourcing protocol is depicted in Fig. 2.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

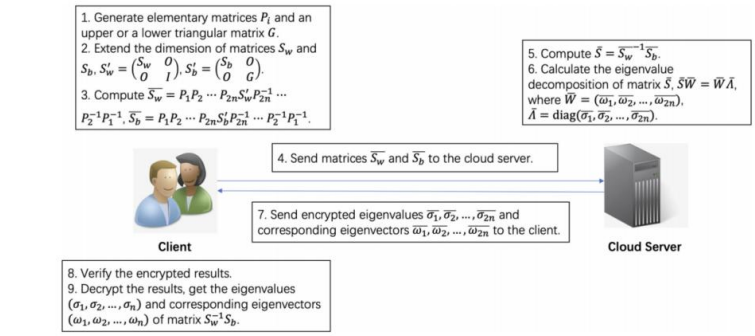

Fig. 2. Specific flow chart of the proposed outsourcing protocol.

3.1.1 Generation of Keys

In order to obtain the elementary matrices $P_i \in R^{2n\times 2n}$, the client uses Algorithm 1.
Algorithm 1 allows us to obtain the structure of matrix P_i

$$P_i = \begin{pmatrix} 1 & & & & & \\ & 1 & & & & \\ & & \ddots & & & \\ & & & \ddots & & \\ & & & & \ddots & \\ & p_i & \cdots & \cdots & 1 & \\ & & & & & 1 \end{pmatrix}, i = 1,2,3,\ldots.2n \qquad (9)$$

This structure states that the elements of row β_i and column α_i are p_i, the diagonal members of P_i are all 1, and the remaining elements are all 0.
The client can easily obtain the inverse of the matrix P_i because it is an elementary matrix.

$$P_i^{-1} = \begin{pmatrix} 1 & & & & & \\ & 1 & & & & \\ & & \ddots & & & \\ & & & \ddots & & \\ & & & & \ddots & \\ & -p_i & \cdots & \cdots & 1 & \\ & & & & & 1 \end{pmatrix}, i = 1,2,3,\ldots.2n \qquad (10)$$

Since p is a positive constant and p_i is a random variable selected from the interval $(-2^p, 2^p)$, its probability density function may be found in the following equation.

$$f_{P_i}(p_i) = \begin{cases} \frac{1}{2^{p+1}} & -2^p < p_i < 2^p \\ 0 & otherwise \end{cases} \qquad (11)$$

3.1.2 Encryption

First, the client expands the $S_w \in R^{n\times n}$ matrix's dimension. The client calculates $S'_w = \begin{pmatrix} S_w & O \\ O & I \end{pmatrix}$ in accordance with Theorem 2 in Section 3, where $O \in R^{n\times n}$ and $I \in R^{n\times n}$ are respectively, a zero matrix and an identity matrix. The client then creates 2n elementary matrices $P_i \in R^{2n\times 2n}$ in accordance with Algorithm 1 and computes:

$$\overline{S_w} = P_1P_2 \ldots . P_{2n}\begin{pmatrix} S_w & O \\ O & I \end{pmatrix}P_{2n}^{-1} \ldots P_2^{-1}P_1^{-1} \qquad (12)$$

The client then expands the $S_b \in R^{n\times n}$ matrix's dimension. The client receives $S'_b = \begin{pmatrix} S_b & O \\ O & G \end{pmatrix}$ in accordance with Theorem 2, where matrix $G \in R^{n\times n}$ is a triangular matrix that the client generates at random and has the entries G1, G2,..., Gn on its diagonal. Next, the client computes using the same basic matrices P_i

$$\overline{S_b} = P_1P_2 \ldots . P_{2n}\begin{pmatrix} S_b & O \\ O & G \end{pmatrix}P_{2n}^{-1} \ldots P_2^{-1}P_1^{-1} \qquad (13)$$

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

Lastly, $S_w \in R^{2n\times 2n}$ and $S_b \in R^{2n\times 2n}$ matrices are sent by the client.

3.1.3 Calculation

The cloud computes after obtaining the matrices $\overline{S_w}$ and $\overline{S_b}$.

The eigenvalues $\bar{\sigma}\iota$ and eigenvectors $\overline{w_\iota}$ of matrix $\bar{S}$ are then sent back after $\bar{S} = \overline{S_w^{-1}}\overline{S_b}$ and eigenvalue decomposition of matrix $\bar{S}$.

3.1.4 Verification

The client must confirm whether the results are accurate after getting them. Given that matrix-vector multiplication has a computational complexity of $O(n^2)$, we designed Algorithm 2 to reduce that complexity while maintaining correctness.

3.1.5 Decryption

The following equation allows the client to retrieve the matrix $S_w^{-1}S_b$ eigenvectors:

$$w = P_{2n}^{-1} \ldots P_2^{-1}P_1^{-1}\bar{w} \qquad (14)$$

The client then determines the eigenvalues $\bar{\sigma}\iota$. The elements on a triangular matrix G's diagonal are its eigenvalues, as demonstrated by Theorem 1.
Client simply needs to eliminate the eigenvalues G1, G2,..., Gn of the triangular matrix G, knowing its diagonal elements. The remaining eigenvalues $\bar{\sigma}\iota$ are the eigenvalues of matrix $S_w^{-1}S_b$

Algorithm 2. Verification Algorithm

Input: The unchecked results $\overline{W} = (\overline{w_1}, \overline{w_2}, \ldots, \overline{w_{2n}})$ and $\overline{\Lambda} = diag(\overline{\sigma_1}, \overline{\sigma_2}, \ldots, \overline{\sigma_{2n}})$.
Output: Accept the results or not.
1: **for** $i = 1 : k$ **do**
2: The client randomly chooses $\overline{w_i}$ and $\overline{\sigma_i}$ in $\overline{W}$ and $\overline{\Lambda}$.
3: The client computes $U = \overline{S_b}\overline{w_i} - \overline{\sigma_i}\overline{S_w}\overline{w_i}$.
4: **if** $U \neq (0, 0, \ldots, 0)^T$ **then**
5: Reject $\overline{W}$ and $\overline{\Lambda}$, and return them to cloud.
6: **end if**
7: **end for**
8: Take $\overline{W}$ and $\overline{\Lambda}$ as the correct results.

2) An overview of the suggested protocol

The following is a summary of the suggested protocol.
Generation of key. The client creates simple matrices. P_i a triangle matrix, either upper or lower, and an identity matrix.
Encryption. Matrix S_w and S_b are encrypted by the client and sent to the cloud as follows.

$$\overline{S_w} = P_1P_2 \ldots . P_{2n}\begin{pmatrix} S_w & O \\ O & I \end{pmatrix} P_{2n}^{-1} \ldots P_2^{-1}P_1^{-1}$$

$$\overline{S_b} = P_1P_2 \ldots . P_{2n}\begin{pmatrix} S_b & O \\ O & G \end{pmatrix} P_{2n}^{-1} \ldots P_2^{-1}P_1^{-1}$$

Calculation. After calculating matrix, $\bar{S}$ is $\overline{S_w^{-1}}, \bar{S} = \overline{S_w^{-1}}, \overline{S_b}$ and ED, the cloud returns the encrypted eigenvalues and eigenvectors confirmation.
Verification. The client does the computations after selecting k pairs of eigenvalues and eigenvectors at random. The client either accepts or rejects the responses based on whether they pass the verification algorithm.
Decryption. The client obtains the eigenvectors using (14) and the eigenvalues using Theorem 2.

3) The Outsourcing Protocol's Correctness

The client encrypts S_w and S_b matrices and transmits them to the cloud server in the manner previously mentioned. The cloud first calculates $\overline{S_w}$ inverse. The following is how $\overline{S_w}$ is displayed using Equation (12):

$$\overline{S_w}^{-1} = P_1P_2 \ldots . P_{2n}\begin{pmatrix} {S_w}^{-1} & O \\ O & I \end{pmatrix} P_{2n}^{-1} \ldots P_2^{-1}P_1^{-1} \qquad (15)$$

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

The cloud then calculates the multiplication of matrices $\overline{S_b}$ and $\overline{S_w}^{-1}$ The precise procedure is as follows:

$$\bar{S} = \overline{S_w^{-1} S_b}$$
$$= P_1 P_2 \ldots . P_{2n} \begin{pmatrix} {S_w}^{-1} & O \\ O & I \end{pmatrix} P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1} . P_1 P_2 \ldots . P_{2n} \begin{pmatrix} S_b & O \\ O & G \end{pmatrix} P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1}$$

$$= P_1 P_2 \ldots . P_{2n} \begin{pmatrix} {S_w}^{-1} S_b & O \\ O & G \end{pmatrix} P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1} \quad (16)$$

Ultimately, the cloud calculates matrix S's ED and outputs the findings.
Following receipt of the findings, the client computes Eq. (6) by selecting a random subset of eigenvalue and eigenvector pairings. The accuracy and efficiency of verification can be guaranteed by the suggested verification algorithm.
The results are then decrypted by the client. The following expression can be obtained using the relationships between matrices, eigenvalue, and eigenvector that were discussed in Section 2.3.

$$\bar{S}\overline{w} = \overline{\sigma w} \quad (17)$$

As stated in (16) and (17):

$$P_1 P_2 \ldots . P_{2n} \begin{pmatrix} {S_w}^{-1} S_b & O \\ O & G \end{pmatrix} P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1} \overline{w} = \overline{\sigma w} \quad (18)$$

Consequently, the following equation is obtained:

$$\begin{pmatrix} {S_w}^{-1} S_b & O \\ O & G \end{pmatrix} P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1} \overline{w} = P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1} \overline{\sigma w}$$
$$= \bar{\sigma} P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1} \overline{w} \quad (19)$$

Let $w = P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1} \overline{w}$ we get

$$\begin{pmatrix} {S_w}^{-1} S_b & O \\ O & G \end{pmatrix} w = \bar{\sigma} w \quad (20)$$

Theorem 2 states that the eigenvalues of matrices ${S_w}^{-1} S_b$ and G make up the eigenvalues $\bar{\sigma}$ of matrix $\begin{pmatrix} {S_w}^{-1} S_b & O \\ O & G \end{pmatrix}$. The formula $w = P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1} \overline{w}$ yields the eigenvectors w.

4 THE PROPOSED OUTSOURCING PROTOCOL ANALYSIS

In this section, we examine the suggested outsourcing protocol and demonstrate its security, verifiability, and efficiency. Specifically, we demonstrate that the inputs and outputs can be computationally distinguished from a random matrix and vector, and more significantly, the client can verify the errors with a non-negligible probability. Additionally, we demonstrate that the client's computational overhead can be reduced from O(n^3) to O(n^2).

4.1 Analysis of Security

Input Privacy. S_w and S_b are the inputs for the suggested protocol. The client encrypts them in the manner described in Section 3:

$$\overline{S_w} = P_1 P_2 \ldots . P_{2n} \begin{pmatrix} S_w & O \\ O & I \end{pmatrix} P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1}$$
$$\overline{S_b} = P_1 P_2 \ldots . P_{2n} \begin{pmatrix} S_b & O \\ O & G \end{pmatrix} P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1}$$

To hide a matrix's secret information, the client multiplies it by primitive matrices. In specifics, the matrix $Q \in R^{2n \times 2n}$ can be concealed in the manner shown below:

$$\overline{Q} = P_1 P_2 \ldots P_{2n} Q P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1} \quad (21)$$

As seen in Section 3, $P_i \in R^{2n\, x\, 2n} (i = 1, 2, \ldots, 2n)$ are all elementary matrices. Assuming r is a positive constant, we assume that the elements in matrix Q are situated in the interval $(-2^r, 2^r)$,
We must break matrix $\bar{Q}$ down into the following two components in order to demonstrate its privacy:

$$\hat{Q} = P_1 P_2 \ldots P_{2n} Q \quad (22)$$
$$\bar{Q} = \hat{Q} P_{2n}^{-1} \ldots P_2^{-1} P_1^{-1} \quad (23)$$

Zhou et al. demonstrated in [24] that multiplying a matrix by an elementary matrix on the left corresponds to the matrix's row transformation, and multiplying a matrix by an elementary matrix on the right corresponds to the matrix's column transformation.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

Each element in matrix $\hat{Q}$, represented by $\hat{q}_{i,j}$, can therefore be computed as follows following the transformation of (22):

$$\hat{q}_{i,j} = q_{i,j} + p_i Q_{i,j}, \qquad (24) \qquad (24)$$

where pi is a random variable that is randomly selected from the interval $(-2^p, 2^p)$, as detailed in Section 3.1, and $Q_{i,j} = q_{i\prime,j}(i' \in [1,2n],\ i \neq i')$, $q_{i,j}$, and $q_{i',j}$ are two distinct elements in the matrix Q. The theoretical maximum of $\{Q_{i,j} | j = 1,2,\dots,2n\}, \forall i \in [1,2n]$. is then defined as Z_i.
As a result, $\hat{q}_{i,j}$ falls between $(-2^r - 2^p . Z_i, 2^r + 2^p . Z_i)$.
As a result, we may now formulate a thesis concerning the computational inability to differentiate between matrix Q and a random matrix in which each row element is drawn from a uniform distribution.

Theorem 3.[25]Set $R \in R^{2n\,x\,2n}$ to be a random matrix with $\hat{Q}$is defined as (22). Its elements in row I are drawn from a uniform distribution in the range of $(-2^{\,p} . Z_i,\ 2^p . Z_i), \forall_i \in [1,2n]$. Matrix $\hat{Q}$ and matrix R are said to be computationally identical.

Since multiplying a matrix by an elementary matrix on the left is equivalent to changing its row, and multiplying a matrix by an elementary matrix on the right is equivalent to changing its column, as was previously mentioned, (22) and (23) execute row and column operations on a matrix, respectively. Then comes Theorem 3. Regarding the computational indistinguishability of matrix $\bar{Q}$ and a random matrix with column components drawn from a uniform distribution, we can put forth Theorem 4.
We assume that matrix $\hat{Q}$ entries are valued. where t is a positive constant, and between $(-2^t, 2^t)$.
Each member in matrix $\bar{Q}$, represented by $\bar{q}_{i,j}$, can be computed as follows, per equation (23):

$$\bar{q}_{i,j} = \hat{q}_{i,j} - p_i \hat{Q}_{i,j}, \qquad (25)$$

where $\hat{Q}_{i,j} = \hat{q}_{i,j'}\ (j' \in [1,2n],\ j \neq j')$, $\hat{q}_{i,j}$ and $\hat{q}_{i,j'}$ are two distinct elements in matrix $\hat{Q}$. The theoretical maximum of $\{Q_{i,j} \mid i = 1,2,\dots,2n\}, \forall j \in [1,2n]$ { should be set to $Z^l{}_i$. As a result, $\bar{q}_{i,j}$ falls between $(-2^t - 2^p . Z^l{}_i,\ 2^t + 2^p . Z^l{}_i)$.
Theorem 4. It states that each element in column j of a random matrix $R \in R^{2n\,x\,2n}$ is drawn from a uniform distribution in the range $(-2^p . Z^l{}_i\ 2^p . Z^l{}_i), \forall j \in [1,2n]$ and $\bar{Q}$ is defined as (23). We assert that there is no computational difference between matrices $\bar{Q}$ and R.
Proof. As stated in Definition 1, we must show that any $\bar{q}_{i;j}$ and $r_{i,j}(\forall i,j \in [1,2n])$ are computationally identical for matrices $\bar{Q}$ and R in order to verify Theorem 4. In particular, we must demonstrate that it is not feasible
'Unless there is a low probability, any probabilistic polynomial time distinguisher D can be used to separate from $r_{i,j'}\ \forall i,j \in [1,2n]$. Therefore, we must determine the precise likelihood of successful verification and demonstrate that it is infinitely close to 1.

We have shown that the values obtained from matrices $\bar{Q}$ and R fall between $(-2^t - 2^p . Z'_i, 2^t + 2^p . Z'_i)$ and $(-2^p . Z'_i, 2^p . Z'_i)$.

Therefore, the optimal approach for distinguisher D given a sampled $x = \bar{q}_{i,j}$ is to return $b \leftarrow \{0, 1\}$ with one chance if $-2^p . Z'_i < x < 2^p . Z'_i$ and the same chance if $x \leq -2^p . Z'_i$ or $x \geq 2^p . Z'_i$. The success probability of the distinguisher can therefore be calculated as follows for $x = \bar{q}_{i,j}$;

$$\Pr\left[D\left(\bar{q}_{i,j}\right)\right] = 1$$

$$= \frac{1}{2}\Pr\left[-2^p \cdot Z'_i < \bar{q}_{i,j} < 2^p \cdot Z'_i\right] + \Pr\left[\bar{q}_{i,j} \leq -2^p \cdot Z'_i\right] + \Pr\left[\bar{q}_{i,j} \geq 2^p \cdot Z'_i\right]$$

$$= \frac{1}{2}\left(1 - \Pr\left[\bar{q}_{i,j} \leq -2^p \cdot Z'_i\right] - \Pr\left[\bar{q}_{i,j} \geq 2^p \cdot Z'_i\right]\right) + \Pr\left[\bar{q}_{i,j} \leq -2^p \cdot Z'_i\right] + \Pr\left[\bar{q}_{i,j} \geq 2^p \cdot Z'_i\right]$$

$$= \frac{1}{2} + \frac{1}{2}\Pr\left[\bar{q}_{i,j} \leq -2^p \cdot Z'_i\right] + \frac{1}{2}\Pr\left[\bar{q}_{i,j} \geq 2^p \cdot Z'_i\right]$$

Where

$$\Pr\left[\bar{q_{i,j}} \geq 2p.Z'i\right] = \Pr[\widehat{q_{i,j}} + p_i\widehat{Q_{i,j}} \geq 2^p.Z'i]$$

$$\leq \Pr[2^t + |p_i|z'_i \geq 2^p . z'_i]$$

$$= \Pr\left[|p_i| \geq 2^p - \frac{2^t}{z'_i}\right]$$

$$= \frac{2^{t-p}}{z'_i}$$

Similarly, we learn that $\Pr\left[\bar{q}_{i,j} \leq 2^p . Z'_i\right] \leq \frac{2^{t-q}}{Z'_i}$.
Therefore, the following equation displays the successful probability for distinguisher D when $x = \bar{q}_{i,j}$

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

We obtain that $\Pr[D(r_{i,j} = 1)] = \frac{1}{2}$, if $x = r_{i,j}$.
Using (8), the following equation can be obtained for any $i, j \in [1,2n]$,
$\Pr\left[D\left(\overline{q}_{i,j} = 1\right)\right] - \Pr[D\left(r_{i,j} = 1\right)] \leq \frac{2^{t-p}}{Z_i'}$. (27)
Observe that $Z_i' \geq 2^t$. Consequently, we get The function
$\mu(p) = \frac{2^{t-p}}{Z_i'} \leq \frac{2^{t-p}}{2^t} = \frac{1}{2^p}$ (28)

is insignificant. Thus, the proof is complete.

$0 < \Pr\left[D\left(\overline{q}_{i,j} = 1\right)\right] \leq \frac{1}{2} + \frac{2^{t-p}}{Z_i'}$ (26)
We obtain that $\Pr[D(r_{i,j} = 1)] = \frac{1}{2}$, if $x = r_{i,j}$.
Using (8), the following equation can be obtained for any $i, j \in [1,2n]$,
$\Pr\left[D\left(\overline{q}_{i,j} = 1\right)\right] - \Pr[D\left(r_{i,j} = 1\right)] \leq \frac{2^{t-p}}{Z_i'}$. (27)
Observe that $Z_i' \geq 2^t$. Consequently, we get The function
$\mu(p) = \frac{2^{t-p}}{Z_i'} \leq \frac{2^{t-p}}{2^t} = \frac{1}{2^p}$ (28)

is insignificant. Thus, the proof is complete.

Theorems 3 and 4 state that matrices $\bar{S}_w$ and $\bar{S}_b$ can be computed to be identical to a random matrix R.
As a result, we can state that the suggested protocol's inputs S_w and S_b are safe and secure.

Output confidentiality. The cloud computes the matrix $\bar{S}$ eigenvalue decomposition and returns the results. The following is the procedure for decrypting eigenvectors:

$$w = P_{2n}^{-1} \cdots P_2^{-1} P_1^{-1} \overline{w} \quad (29)$$

where the encrypted and decrypted eigenvectors are denoted by the letters $\overline{w}$ and w, respectively. In other words:

$$\overline{w} = P_1 P_2 \cdots P_{2n} w \quad (30)$$

Assume that the values of vector w elements fall between $(-2^b, 2^b)$, where b is a constant and b >0. Consequently, each element in vector w can be represented by w_i in accordance with :

$$\overline{w}_i = w_i + p_i w_i , \quad (31)$$

where In vector $W_i = w_j$ or $\overline{w}_j$ and $w_i, w_j (j \in [1,2n])$. are two distinct elements. Make Z00 I the theoretical . As a result, $\overline{w}_i$ falls between $(-2^b - 2^p . Z_i , 2^b + 2^p . Z_i)$. As far as the computational indistinguishability of vector w and a random vector with elements sampled from a uniform distribution is concerned, we derive a theorem.

Theorem 5. It states that $r \in R^{2n \times 1}$ is a random vector whose elements in row i are drawn from a uniform distribution between $(-2^p . Z''_i , 2^p . Z''_i)$ The definition of $\overline{w}$ and $\overline{w}$ is. Then, we can state that there is no computational difference between vectors w and r.

proof. Since the evidence is comparable to that of Theorem 4, it is not included here.
Theorem 5 states that vector $\overline{w}$ is computationally identical to a random vector r, thus the cloud server is unable to obtain any useful information about vector $\overline{w}$. Vector $\overline{w}$ is hence well protected. We employ a random triangular matrix G and set V = $\begin{pmatrix} S_w^{-1} S_b & O \\ O & G \end{pmatrix}$ to safeguard the eigenvalues s. The eigenvalues of matrices $S_w^{-1} S_b$ and G make up the eigenvalues of matrix V, as stated in Section 2.3. Because the cloud doesn't know regarding the eigenvalues of matrix G, it is unable to retrieve the eigenvalues of matrix $S_w^{-1} S_b$, which are obscured by the matrix G eigenvalues. The cloud finds it harder and harder to decode the eigenvalues as the image size grows since the dimension of matrix G grows along with the number of eigenvalues. Nonetheless, the client is familiar with matrix G components and can easily retrieve the eigenvalues. Consequently, the matrix's eigenvalues and eigenvectors.The $S_w^{-1} S_b$ are safe and secure.

4.2 Verification of Outsourcing Results : We demonstrate in this paragraph that the suggested outsourcing protocol has strong anti-cheating capabilities [22], meaning that the client has a non-negligible chance of checking for errors.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

Theorem 6: In the suggested protocol, there is a very small chance that the adversary will deceive the client into accepting incorrect outsourced results.

Proof . This theorem can be proved in two steps. We must first demonstrate that any accurate findings may effectively navigate our verification process. If the findings are accurate, $\overline{S_b}\overline{w_i}$ equals, $\overline{\sigma_i}\overline{S_b}\overline{w_i}$ as (32), illustrates. As a result, U is always a zero vector, and this verification procedure may correctly process any accurate results. (32)

$$U = \overline{S_b}\overline{w_i} - \overline{\sigma_i}\overline{S_b}\overline{w_i} \quad (32)$$

Second, we demonstrate that there is a non-negligible chance that any wrong results will fail the verification algorithm. Let Prob be the likelihood that the verification will be successful. The probability can be calculated using the permutation and combination formula by

$$prob = 1 - \frac{1}{C_n^k} \quad (33)$$

where n is the dimension of the matrix $S_w{}^{-1}S_b$ and k is the number of eigenvalues and eigenvectors that must be chosen in the verification process.

We now examine why (33) is true. The client will choose k from n eigenvalues and eigenvectors, as demonstrated in Section 3, meaning that there is a $1/C_n^k$ chance that the cloud would deceive the client. For instance, the probability is $1 - 5 \times 10^{-5}$ when n = 200, k = 2, whereas it is 0.99 when n = 100, k = 1. As a result, the probability will increase as N and k do. Our study shows that for n = 1000, k = 5, Prob is nearly equal to 1, meaning that if 5 pairs of eigenvalues and eigenvectors are randomly selected, it is nearly difficult for the cloud to defraud the client. Therefore, it is impossible for errors to pass our verification procedure if there are any in the returned outsourcing results. Theorem 6's proof states that there is a maximum chance of $1/C_n^k$ for incorrect results to pass the verification procedure. It goes without saying that a higher k can improve the performance of our verification procedure, but it may also result in higher client computing expenses. Consequently, k affects both computational efficiency and the likelihood of verification.

4.3 Analysis of Efficiency

The suggested protocol consists of five algorithms. We examine the client-side and cloud-side overheads independently in this subsection.

Overhead on the client side. The client must use the four algorithms generation of keys, encryption, verification, and decryption discussed in Section 3. During the Key Generation process, the client must produce an upper or lower triangular matrix in addition to a number of elementary matrices. The computational complexity is clearly O(n). The encryption algorithm has an $O(n^2)$ computing complexity and involves the client expanding the matrix's dimension before multiplying the original matrix by auxiliary matrices. The verification algorithm's highest computational complexity, $O(n^2)$, is matrix-vector multiplication.

TABLE 1
Comparison of Computing and Communication Cost

Protocol	Matrix Operations	Non-interactive	Client-side Overhead	Cloud-side Overhead	Input Privacy	Output Privacy	Verifiability
Zhou *et al.* [24]	ED	No	$O(n^2)$	$O(n^3)$	No	No	Yes
Zhang *et al.* [25]	MM, ED	No	$O(n^2)$	$O(n^3)$	Yes	Yes	Yes
Our protocol	MI, MM, ED	Yes	$O(n^2)$	$O(n^3)$	Yes	Yes	Yes

and $O(n^2)$.is also the computational complexity. In summary, $O(n^2)$. is the client-side overhead.

Overhead from the cloud. The cloud server simply needs to execute the computation algorithm, as opposed to the client. The cloud performs eigenvalue decomposition (ED), matrix multiplication (MM), and matrix inversion (MI) in this approach. As is well known, each of these stages has an $O(n^3)$. computational complexity. As a result, $O(n^3)$. is the cloud-side above.

The facial recognition outsourcing procedures are compared in Table 1. Although our suggested protocol can outsource more types of matrix operations and have less instances of communication overhead, the client-side and cloud-side computational overheads in [24], [25], and our proposed protocol are all $O(n^2)$. and $O(n^3)$., respectively. While [24] can only finish the ED operation and [25] can only finish the MM and ED operations, the suggested outsourcing protocol can finish the MI, MM, and ED operations simultaneously.

More significantly, in the proposed protocol, which is a non-interactive protocol, the client only needs one encryption and one decryption to carry out the outsourcing algorithm, whereas in [24] and [25], the client requires three encryptions and three decryptions, resulting in significant communication and computational overheads for both the client and the cloud server. The client can accomplish LDA-based facial recognition more rapidly and precisely with the aid of the suggested procedure.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

As a result, the client can reduce its communication overheads and computational complexity by utilizing the suggested outsourcing protocol.

5 EXPERIMENTS

In the preceding section, we shown hypothetically that the suggested protocol can significantly lower the client's computational and communication expenses. The efficiency and performance analyses of face recognition will be illustrated in this part using the next two trials.

MATLAB 2016a simulates the client in the experiments using a computer with 8 GB of RAM and an Intel Core i5 rated at 1.8 GHz, while a MacBook Pro laptop with an Intel Core i5 running with 4 cores rated at 1.4 GHz and 16 GB of RAM simulates the cloud server.

5.1 Evaluation of the Proposed Protocol's Efficiency

We independently define $t_{original}$ and t_{client} as the times that the client runs the outsourcing protocol and the original LDA-based facial recognition algorithm in order to demonstrate the effectiveness of the suggested outsourcing protocol in this experiment. Based on the aforementioned criteria, the performance improvement is characterized as $\frac{t_{original}}{t_{client}}$. In most cases, a performance increase greater than one indicates that the client can save computer resources; if not, there is no reason for the client to choose outsourcing.

In Algorithm 1, p is a positive constant, and p_i is a random number selected from the interval $(-2^p, 2^p)$, p =10 is the value we set for this experiment. As seen in Section 4.2, the verification probability increases with parameter k; nevertheless, a greater k will result in higher client computing overheads. The successful probability is infinitely near to 1 when k = 10. As a result, we can raise the client's computational overheads without using a higher k. The trials for k = 5 and k = 10 will be simulated in the experiments that follow. Zhou et al. and Zhang et al. create random matrices in [24] and [25] to evaluate the effectiveness of their suggested techniques. For an even comparison, Additionally, we produce random matrices with various sizes.

The performance benefit of the suggested outsourcing approach is mostly seen in Tables 2 and 3. Table 2 indicates that the client can attain a minimum performance gain of 3.72. The client can obtain additional performance with the dimension expansion. The client can attain a performance improvement of 26.87 when the matrix's dimension is 5000. The client will require extra time for verification in Table 3 as a result of the increase in k.

A performance boost of 25.89 can be obtained by the client when the matrix's dimension is 5000.

We separately show the effectiveness of the suggested protocol against that of the earlier outsourcing protocols and the original LDA-based algorithms using Figs. 3, 4, and 5.

Figure 3 leads us to the conclusion that, in comparison to the original algorithm, the suggested protocol can significantly reduce the computational cost and save the client more time. Next, as illustrated in Fig. 4, we contrast the cloud-side overhead with that of various outsourcing protocols. We discover that Zhang et al. [25] and our suggested approach spend more

TABLE 2
Experiment Results of the Outsourcing Protocol When $k = 5$

No.	Dimension(n)	$t_{original}$(sec)	t_{client}(sec)	t_{cloud}(sec)	$t_{original}/t_{client}$
1	500	1.3951	0.3747	0.4468	3.72
2	1000	7.7163	1.1980	2.7853	6.44
3	1500	27.2441	2.7087	8.2763	10.05
4	2000	58.9292	4.6446	17.6197	12.69
5	2500	108.9184	6.7036	39.6343	16.25
6	3000	183.4635	9.2592	68.6919	19.81
7	3500	289.1965	13.0359	111.1952	22.18
8	4000	419.4783	17.4855	167.9263	23.99
9	4500	607.6743	23.8556	250.2278	25.47
10	5000	853.9909	31.7768	362.3223	26.87

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

TABLE 3
Experiment Results of the Outsourcing Protocol When $k = 10$

No.	Dimension(n)	$t_{original}$(sec)	t_{client}(sec)	t_{cloud}(sec)	$t_{original}/t_{client}$
1	500	1.3503	0.3925	0.4323	3.44
2	1000	7.6304	1.2563	2.8711	6.07
3	1500	26.8636	2.8768	8.6601	9.34
4	2000	58.3099	4.8931	17.6027	12.12
5	2500	108.6013	7.0523	39.7892	15.40
6	3000	183.2340	9.6089	67.9081	19.07
7	3500	289.2011	13.5359	112.2060	21.37
8	4000	419.6159	18.2012	165.0293	23.05
9	4500	606.9187	24.7083	255.9527	24.56
10	5000	853.1093	32.9499	358.3951	25.89

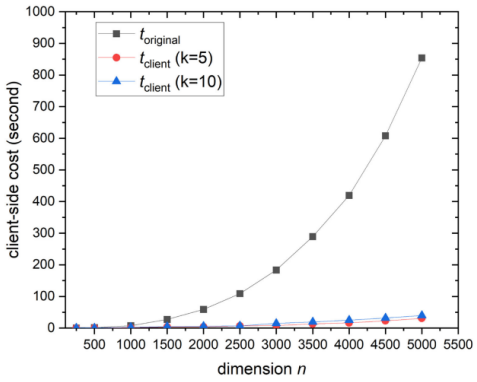

Fig. 3. Client-side time comparison for different k.

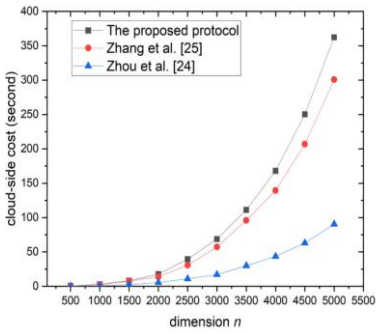

Fig. 4. Comparison of cloud-side time among the outsourcing protocols.

time for cloud-side overhead than Zhou et al. [24], as these two protocols expanded the matrix's dimension and could better safeguard private data. In addition, our suggested protocol can simultaneously outsource three different types of matrix computations, whereas the other two can only outsource one type; as a result, it takes a little longer to execute at the cloud's side than the other two. Lastly, we compare the performance gain with other outsourcing protocols in Fig. 5, and it is clear that the suggested outsourcing protocol can acquire more performance improvements over those in Zhou et al. [24] and Zhang et al. [25], and they all contract out three processes: eigenvalue decomposition, matrix multiplication, and matrix inversion. More significantly, the client can save more money on communication because the suggested protocol is non-interactive.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

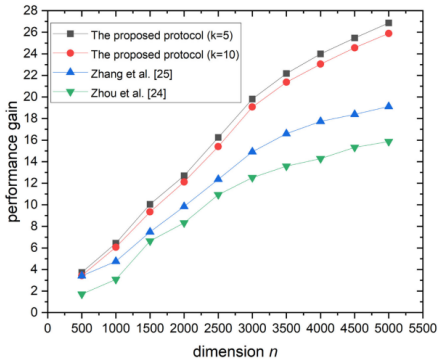

Fig. 5. Comparison of performance gain among the outsourcing protocols.

Fig. 6. The image samples of ORL database.

5.2 Evaluation of Face Recognition Performance

We shall show in this experiment that the suggested protocol's face recognition performance is nearly identical to that of the original LDA technique. The experiment's precise steps are as follows.

5.2.1 Databases

We use the ORL Face Database [12], AR Face Dataset [29], and expanded YaleB Face Database [30] to demonstrate the effectiveness of the suggested outsourcing protocol.

There are 400 face photos of 40 people in the ORL Face Database, with 10 face photos for each individual. There are minor variations in the pictures in terms of face accessories, posture, and attitude. Every object sample has ten normalized grayscale pictures. Each image has a black background and measures 92 ×112. Each person's characteristics and facial expressions are altered, including whether they laugh or not, if they wear glasses, and whether their eyes are open or closed.

Fig. 6 displays a few image samples from the ORL Face Database.

Fig. 7. The image samples of AR face database.

There are over 3000 photos of 126 people in the AR Face Database. For the experimental database in this paper, we chose 100 individuals, each of whom had 26 photos. The 26 photos were gathered across two time periods. There are thirteen pictures in each period, three of which feature sunglasses, three of which have scarves, and the remaining seven of which show variations in light and expression. Fig. 7 shows some of the AR Face Database's pictures.The 21888 photos of 38 people that make up the expanded YaleB Face Database were gathered from 9 positions and 64 illumination variations. One of the subdatabases, which

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

includes 38 individuals and 64 photos each, is chosen for this paper. 64 different types of lighting alterations are gathered under the frontal posture. Fig. 8 displays a few image examples from the expanded Yale Face Database B.

5.2.2 Implementation of Experiment

We employ a two-stage approach in this experiment, which is an enhanced LDA-based face recognition technique [20]. First, the matrix Sw is made complete rank and the face image's dimension is decreased. The LDA method is then used to perform facial recognition and extract the image features.

We conducted three sets of experiments based on various databases. In particular, a subset of face photos are used as training samples, and the remaining images are used as test samples. Additionally, the same amount of samples are chosen for each individual. Since there is no overlap between the two groups, the training samples do not contain the test samples. To get rid of selection, each experiment was run 20 times, and the final recognition accuracy is determined by averaging the recognition accuracy.

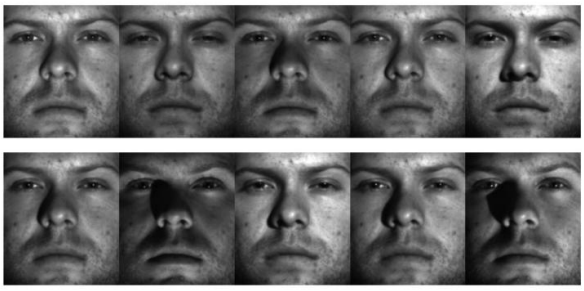

Fig. 8. The image samples of extended YaleB face database.

TABLE 4
Comparison of Accuracy Between the Outsourcing Protocol and the Original One

Databases	Image Number	Accuracy (LDA-based)	Accuracy (Outsourcing)
ORL Database	400	94.50%	94.50%
AR Database	2600	79.36%	79.15%
extended YaleB	21888	96.80%	96.58%

Next, we use the facial recognition algorithm with the suggested outsourcing protocol. We use the suggested technique to obtain the recognition accuracy and carry out the experiment as previously described.

5.2.3 Results

The accuracy of the original LDA-based face recognition is tested first, and then the suggested outsourcing technique is used to assess the accuracy. Table 4 displays the test results.

Table 4 indicates that the face recognition accuracy of the suggested outsourcing protocol is almost identical to that of the original LDA algorithm. This indicates that the features taken from the suggested protocol are accurate, and the suggested protocol can also save the client a significant amount of money.

We can therefore make two deductions from the tests mentioned above. First, compared to earlier protocols, the suggested one can achieve greater performance gains. Furthermore, when applied to LDA-based face recognition, the suggested protocol's recognition accuracy is almost identical to that of the original algorithm, indicating that it is practical and effective.

In conclusion, we demonstrate that the protocol can maintain almost the same accuracy as employing LDA-based face recognition directly while achieving significant computational and communication cost savings. Therefore, using the suggested outsourcing technique to finish the LDA-based facial recognition is a superior option for a limited client.

6 CONCLUSION

In this paper, develop a protocol for outsourcing LDA-based facial recognition to an untrusted cloud. The suggested approach significantly decreases the client's local computational complexity and interaction time by requiring only one encryption and one

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

decryption to perform matrix inversion, matrix multiplication, and eigenvalue decomposition. Furthermore, it has been demonstrated that it is safe to conceal the input and output privacy by multiplying a sequence of simple matrices. Additionally, the suggested verification procedure has a non-negligible likelihood of checking for errors. Above all, our suggested protocol achieves more performance gains than with the earlier facial recognition outsourcing methodology. However, only LDA-based facial recognition is compatible with the suggested outsourcing protocol. Future studies will examine further face recognition and machine learning outsourcing protocols.

REFERENCES

[1] M. Arsenovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "Facetime – deep learning based face recognition attendance system," in Proc. IEEE Int. Symp. Intell. Syst. Inf., 2017, pp. 53–58.
[2] P. Hu, H. Ning, T. Qiu, H. Song, Y. Wang, and X. Yao, "Security and privacy preservation scheme of face identification and resolution framework using fog computing in Internet of Things," IEEE Internet Things J., vol. 4, no. 5, pp. 1143–1155, Oct. 2017.
[3] C. Behaine and J. Scharcanski, "Enhancing the performance of active shape models in face recognition applications," IEEE Trans. Instrum. Meas., vol. 61, no. 8, pp. 2330–2333, Aug. 2012.
[4] X. Li, Y. Zhu, J. Wang, Z. Liu, Y. Liu, and M. Zhang, "On the soundness and security of privacy-preserving SVM for outsourcing data classification," IEEE Trans. Dependable Secure Comput., vol. 15, no. 5, pp. 906–912, Sep./Oct. 2018.
[5] J. Scheuner and P. Leitner, "Function-as-a-service performance evaluation: A multivocal literature review," J. Syst. Softw., vol. 170, 2020, pp. 110708.
[6] O. Ascigil, A. Tasiopoulos, T. K. Phan, V. Sourlas, I. Psaras, and G. Pavlou, "Resource provisioning and allocation in function-as-a service edge-clouds," IEEE Trans. Services Comput., to be published, doi: 10.1109/TSC.2021.3052139.
[7] P. Anu and S. Vimala, "A survey on sniffing attacks on computer networks," in Proc. Int. Conf. Intell. Comput. Control, 2017, pp. 1–5.
[8] M. Agarwal, S. Biswas, and S. Nandi, "Discrete event system framework for fault diagnosis with measurement inconsistency: Case study of rogue dhcp attack," IEEE/CAA J. Automatica Sinica, vol. 6, no. 3, pp. 789–806, May 2019.
[9] Y. Ren, X. Zhang, G. Feng, Z. Qian, and F. Li, "How to extract image features based on co-occurrence matrix securely and efficiently in cloud computing," IEEE Trans. Cloud Comput., vol. 8, no. 1, pp. 207–219, Jan.–Mar. 2020.
[10] J. Feng, L. T. Yang, G. Dai, W. Wang, and D. Zou, "A secure highorder lanczos-based orthogonal tensor SVD for Big Data reduction in cloud environment," IEEE Trans. Big Data, vol. 5, no. 3, pp. 355–367, Sep. 2019.
[11] A. Fu, Z. Chen, Y. Mu, W. Susilo, Y. Sun, and J. Wu, "Cloud-based outsourcing for enabling privacy-preserving large-scale non-negative matrix factorization," IEEE Trans. Services Comput., vol. 15, no. 1, pp. 266–278, Jan./Feb. 2022.
[12] Z. Xia, L. Jiang, X. Ma, W. Yang, P. Ji, and N. N. Xiong, "A privacy-preserving outsourcing scheme for image local binary pattern in secure industrial Internet of Things," IEEE Trans. Ind. Informat., vol. 16, no. 1, pp. 629–638, Jan. 2020.
[13] X. Zhu, E. Ayday, and R. Vitenberg, "A privacy-preserving framework for outsourcing location-based services to the cloud," IEEE Trans. Dependable Secure Comput., vol. 18, no. 1, pp. 384–399, Jan./ Feb. 2021.
[14] K. He, J. Guo, J. Weng, J. Weng, J. K. Liu, and X. Yi, "Attributebased hybrid boolean keyword search over outsourced encrypted data," IEEE Trans. Dependable Secure Comput., vol. 17, no. 6, pp. 1207–1217, Nov./Dec. 2020.
[15] L. Zhao and L. Chen, "Sparse matrix masking-based non-interactive verifiable (outsourced) computation, revisited," IEEE Trans. Dependable Secure Comput., vol. 17, no. 6, pp. 1188–1206, Nov./ Dec. 2020.
[16] U. Devani, V. Nikam, and B. Meshram, "Super-fast parallel eigenface implementation on GPU for face recognition," in Proc. Int. Conf. Parallel Distrib. Grid Comput., 2014, pp. 130–136.
[17] B. Zhang, Y. Gao, S. Zhao, and J. Liu, "Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor," IEEE Trans. Image Process., vol. 19, no. 2, pp. 533–544, Feb. 2010.
[18] R. Martın-Clemente and V. Zarzoso, "LDA via l1-PCA of whitened data," IEEE Trans. Signal Process., vol. 68, pp. 225–240, 2020.
[19] M. Akbar et al., "An empirical study for PCA- and LDA-based feature reduction for gas identification," IEEE Sensors J., vol. 16, no. 14, pp. 5734–5746, Jul. 2016.
[20] J. Seng and K. Ang, "Big feature data analytics: Split and combine linear discriminant analysis (SC-LDA) for integration towards decision making analytics," IEEE Access, vol. 5, pp. 14 056–14 065, 2017.
[21] Q. Ye, J. Yang, F. Liu, C. Zhao, N. Ye, and T. Yin, "L1-norm distance linear discriminant analysis based on an effective iterative algorithm," IEEE Trans. Circuits Syst. Video Technol., vol. 28, no. 1, pp. 114–129, Jan. 2018.
[22] X. Lei, X. Liao, T. Huang, H. Li, and C. Hu, "Outsourcing large matrix inversion computation to a public cloud," IEEE Trans. Cloud Comput., vol. 1, no. 1, pp. 78–87, Jan.–Jun. 2013.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

[23] X. Lei, X. Liao, T. Huang, and H. Li, "Cloud computing service: The caseof large matrix determinant computation," IEEE Trans. Services Comput., vol. 8, no. 5, pp. 688–700, Sep./Oct. 2015. [24] L. Zhou and C. Li, "Outsourcing eigen-decomposition and singular value decomposition of large matrix to a public cloud," IEEE Access, vol. 4, pp. 869–879, 2016.
[25] Y. Zhang, X. Xiao, L. Yang, Y. Xiang, and S. Zhong, "Secure and efficient outsourcing of PCA-based face recognition," IEEE Trans. Inf. Forensics Security, vol. 15, pp. 1683–1695, 2020.
[26] L. Jiang, C. Xu, X. Wang, B. Luo, and H. Wang, "Secure outsourcing SIFT: Efficient and privacy-preserving image feature extraction in the encrypted domain," IEEE Trans. Dependable Secure Comput., vol. 17, no. 1, pp. 179–193, Jan./Feb. 2020.
[27] X. Luciani and L. Albera, "Joint eigenvalue decomposition of nondefective matrices based on the LU factorization with application to ICA," IEEE Trans. Signal Process., vol. 63, no. 17, pp. 4594–4608, Sep. 2015.
[28] S. Salinas, C. Luo, X. Chen, W. Liao, and P. Li, "Efficient secure outsourcing of large-scale sparse linear systems of equations," IEEE Trans. Big Data, vol. 4, no. 1, pp. 26–39, Mar. 2018.
[29] A. Martinez and A. Kak, "PCA versus LDA," IEEE Trans. Pattern Anal. Mach. Intell., vol. 23, no. 2, pp. 228–233, Feb. 2001.
[30] A. Georghiades, P. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," IEEE Trans. Pattern Anal. Mach. Intell., vol. 23, no. 6, pp. 643–660, Jun. 2001.