IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 11 August 2022, accepted 30 September 2022, date of publication 13 October 2022, date of current version 20 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3214227

RESEARCH ARTICLE

Accurate and Privacy-Preserving Person Localization Using Federated-Learning and the Camera Surveillance Systems of Public Places

MAHMOUD NABIL[1], AHMED SHERIF[2], (Senior Member, IEEE), MOHAMED MAHMOUD[3], (Senior Member, IEEE), WALEED ALSMARY[4], (Senior Member, IEEE), AND MAAZEN ALSABAAN[5]

[1]Department of Electrical and Computer Engineering, North Carolina A&T University, Greensboro, NC 27411, USA
[2]School of Computing Sciences and Computer Engineering, The University of Southern Mississippi, Hattiesburg, MS 39406, USA
[3]Electrical and Computer Engineering Department, Tennessee Tech University, Cookeville, TN 38505, USA
[4]Computer Engineering Department, College of Computer and Information Systems, Umm Al-Qura University, Mecca 24382, Saudi Arabia
[5]Computer Engineering Department, College of Computer and Information Sciences, King Saud University, Riyadh 11451, Saudi Arabia

Corresponding author: Maazen Alsabaan (malsabaan@ksu.edu.sa)

This work was supported by the National Plan of Science, Technology and Innovation (MAARIFAH), King Abdulaziz City for Science and Technology, Saudi Arabia, under Award 15-INF3775-02.

ABSTRACT In this paper we propose an accurate and privacy-preserving scheme that enables a law enforcement agency to locate persons of interest using the camera surveillance systems of public places. Comparing to the existing schemes that measure the Euclidean distance to locate persons using their embedding vectors storing facial features, we use a more accurate approach by training a machine learning model. Moreover, to avoid leaking sensitive information by sharing the images of the public places' visitors to train the model, we use a federated learning technique to compute the model in a privacy-preserving way. The model is designed in such a way that makes executing it over encrypted data efficient. Specifically, the model is executed by three parties as follows. Each public place computes an embedding vector for each visitor's image and inputs it to a neural network and encrypts the output using a modified inner product encryption scheme and sends the ciphertext to a cloud server. The law enforcement agency does the same steps on the images of persons of interest. Finally, the server uses these ciphertexts to evaluate the last layer of the model by computing the inner product of the two vectors over encrypted data. The cryptosystem enables the server to compute the inner product of two vectors using their ciphertexts without being able to learn the vectors. We have modified an encryption cryptosystem that is designed for a single public place and a single law enforcement agency to make it more efficient in our application that has multiple public places. To evaluate our scheme, we have conducted extensive experiments and the results confirm that our model is accurate in locating persons of interest with low communication and computation overhead. A formal proof and analysis are used to demonstrate the ability of our scheme to preserve privacy.

INDEX TERMS Privacy-preservation, security, person localization, inner product over encrypted data, surveillance systems.

I. INTRODUCTION

Person localization is important for several applications, such as locating persons of interest by a law enforcement agency. Face recognition is the most popular technology for person identification because it is non-invasive and does not need the cooperation of the persons comparing to other identification technology such as iris and fingerprint [1]. Recently, deep-learning-based face recognition approaches, such as [2], [3], [4], [5], [6], [7], [8], [9], [10], have been used. In these approaches, machine learning models are trained on a massive amount of data, which allows them to learn the

The associate editor coordinating the review of this manuscript and approving it for publication was Jerry Chun-Wei Lin.

characteristics of expression, illumination, and angle. These approaches first localize the area of the face in the input image and then determine the locations of the landmarks in the face to produce an embedding vector which represents the face's landmark points of the brows, mouth, eyes, jawline, and nose. Finally, for person identification, the distances between an embedding vector and a database with embedding vectors of known persons are measured (e.g., using the Euclidean distance) and two vectors are deemed for the same person if the distance is below a pre-defined threshold [11], [12], [13]. However, this simple approach does not give accurate results especially when the images are taken by different cameras. In this paper, we conduct experiments to evaluate the accuracy of using the Euclidean distance to match the embedding vectors of images taken by different cameras. Our results indicate that this approach is not accurate and it is hard to find one threshold that can give good results for the images of all sources.

Moreover, the closed-circuit television (CCTV) surveillance systems are currently used in almost all public places. Using face recognition technology with the CCTV systems of public places to localize persons of interest is a low-cost and effective approach to fight crime by locating wanted or suspected individuals. However, the use of face recognition technology with the CCTV systems raises serious privacy concerns [14], [15] because the system can be misused to monitor people's daily activities by collecting information on the locations visited by them. The recent privacy breaches in several systems made the public worried about their privacy. Examples of these breaches include exposing the personal information (such as face photos, addresses, age, etc.) of millions of people collected by major Chinese surveillance service providers [16], [17], and charging Facebook $550 million for collecting facial data without authorization [18]. Due to privacy concerns, some legislators proposed bills to ensure that the existing systems preserve the privacy of the people [19], [20], [21].

In this paper, we investigate an efficient and accurate person localization scheme with privacy preservation using federated learning and the camera surveillance systems of public places. With the proposed scheme, a law enforcement entity can locate persons of interest using the surveillance cameras of the public places without revealing the images of the visitors or the persons of interest to preserve privacy. Unlike most of the existing techniques that measure the distance between two embedding vectors and compare the result to a predefined threshold to determine whether the vectors are for the same person, we use a more accurate technique by training a machine learning model using federated learning where the inputs of the model include two embedding vectors and output is either zero or one to indicate whether the two vectors are for the same person or not. The idea is that, instead of using a simple threshold to determine whether two vectors are for the same person, a machine learning model can make accurate decisions because it can learn the characteristics of the embedding vectors of the same persons.

To train the model, we have created a dataset where each sample has two embedding vectors and a label which is one in case that the two vectors are for the same person and zero otherwise. Then, to preserve privacy, we investigate an efficient cryptosystem to enable a cloud server to evaluate the model over encrypted data and report the visited locations to the law enforcement agency without being able to access the images or the embedding vectors of the visitors and the persons of interest. The architecture of the machine learning model is determined in such a way that makes executing it over encrypted data efficient. Specifically, the model is executed by three parties as follows. Each public place computes an embedding vector for each visitor's image and inputs it to a neural network (a part of the model) and encrypts the output using a modified inner product encryption scheme and sends the ciphertext to a cloud server. The law enforcement agency does the same steps on the images of persons of interest. Finally, the server computes the inner product of the two vectors over encrypted data and executes the last layer in the model to determine whether the two vectors are for the same person without learning the images or the embedding vectors of the persons of interest or the visitors of the public places.

We have modified the cryptosystem in [22] that computes the inner product of two vectors using their ciphertexts. This cryptosystem is designed to run by two parties (a single public place and a single law enforcement agency in our application) using a pairwise key, so we have modified it to be more efficient in our application that has multiple public places, where each public place and the law enforcement agency uses only one key for encryption. Six datasets are used to evaluate our proposal. The results demonstrate that our model exhibits more localization accuracy comparing to the use of the Euclidean distance in case of several camera surveillance systems with different image quality. Our evaluations also demonstrate that the overhead of our scheme in terms of computation/communication overhead is acceptable. A formal proof and analysis are used to demonstrate the ability of our scheme to preserve privacy.

To the best of our knowledge, this is the first work that uses a combination of a deep learning model, federated learning, and efficient cryptosystem to evaluate the model using encrypted data to create an accurate and privacy-preserving person localization for multiple camera surveillance systems. Specifically, this paper makes the following contributions:

- Most of the existing techniques depend on a simple approach that measures the distance between two embedding vectors and compares the result to a threshold value to determine whether the vectors are for the same person. In this paper, we use a more accurate approach that is suitable for several camera surveillance systems using a pre-trained machine learning model.
- We evaluate our model over encrypted data to preserve the privacy of the visitors of the public places and the persons of interest. To do that efficiently, the model is executed by there parties where layers of the model are

executed using plaintext data at the public places and law enforcement agency and only the last layer is executed over encrypted data by the server. We have also modified the inner product cryptosystem in [22] to make it more efficient in our application.

- We use federated learning to train our model on the data of different camera surveillance systems with varying image quality without revealing the data to preserve privacy.
- To evaluate the privacy preservation capability of our scheme, we use a formal proof and analysis, and to evaluate the accuracy and the communication/computation overhead of our scheme, we have conducted extensive experiments.

We organize the rest of this paper as follows. The related works are discussed in Section II. The system models including the network and threat models and the important requirements that should be achieved by our scheme are discussed in Section III. Section IV explains our scheme. The results of the evaluations are discussed in Section VI. Finally, Section VII draws the conclusions.

II. RELATED WORK

Content-based image retrieval and face-recognition based authentication schemes are the closest research works to this paper. In this section, we first explain these schemes, and then discuss the research gap and our motivations.

A. CONTENT-BASED IMAGE RETRIEVAL

In image retrieval application, large image datasets are outsourced to a cloud server, and an image is sent to the server to search for similar images and return them. To ensure efficiency, especially, in case of large image datasets, instead of sending an image, the features of the query image is sent to the server which matches them to the features of the stored images, and then the server returns the images with close features. Image retrieval approaches are needed in many applications. Examples to these applications include medical diagnosis [23] and searching for similar clothes online [24].

Since revealing the images or their features to the cloud server may raise privacy concerns in some applications, various privacy-preserving image retrieval schemes have been proposed [11], [12], [25], [26]. In these schemes, the cloud server stores encrypted images' features, and it receives a query containing the features of an image of interest. Then, it searches the stored images to find the closest image to the query, i.e., the image that has close features to the queried image, without being able to learn neither the stored images nor the queried one or even their features.

In [11], a privacy-preserving hierarchical image retrieval system, called CASHEIRS, is proposed. CASHEIRS aims to address two main issues. The first issue is the low image retrieval accuracy and long time needed to search all stored images. The second issue is the privacy concerns raised when the images have sensitive information. For efficient search, CASHEIRS develops a hierarchical index tree which allows search over subsets of categories rather than whole set by clustering the images stored by the server. To improve the image retrieval accuracy, CASHEIRS uses Convolutional Neural Network (CNN) to extract the features of the images. To preserve privacy, the features of the images are encrypted while the server can measure the similarity score of the features, without decrypting the ciphertexts or learning the features.

The proposed scheme in [25] considers two privacy threats, including a cloud server that aims to infer sensitive information about the images, and a dishonest query user who illegally distributes the images he retrieved from the server. To protect against the first threat, the feature vectors are encrypted by the kNN searchable encryption algorithm. For the second threat, a watermark-based protocol is develop to deter distributing images. The cloud server uses this protocol to embed a unique watermark into each encrypted image retrieved by the user. The watermark can be extracted and the user is traced when an illegal copy of the image is found.

In [12], a large-scale content-based image retrieval scheme is proposed. Two different layers are used to preserve privacy. The first layer uses hash values for queries to hide the features because hash functions are one way. The server returns the hash values of all possible candidates and the user selects the best match for his query. In the second layer, the user deletes some bits in a hash value to make it computationally difficult for the server to learn the interest of the user. The paper also introduces the concept of tunable privacy, where the privacy protection level can be adjusted by dividing a feature vector into subsets and indexing every subset with a hash value which is associated with an inverted index list.

The existing content-base image retrieval schemes retrieve images based on the similarity of their visual features, but to improve the retrieval accuracy, interactive mechanism, namely relevance feedback, is integrated with these schemes to retrieve images based on both visual features and semantic concepts. The research work in [26] proposes a privacy-preserving relevance-feedback image retrieval scheme. The scheme has three main stages including private query, private feedback and local retrieval. The initial query with a privacy controllable feature vector is conducted in the private query stage, and the private feedback introduces confusing classes that adhere to the K-anonymity in the creation of the feedback image set to preserve privacy. Finally, in the local retrieval, images are ranked at the user side.

B. FACE RECOGNITION-BASED AUTHENTICATION

The use of face recognition in biometrics-based authentication is an interesting approach because it is non-invasive and does not need the cooperation of users for taking their face images compared to other biometrics-based approaches that use iris and fingerprint. In the literature, several privacy-preserving face recognition-based authentication schemes have been proposed [13], [27], [28], [29].

In [27], an efficient and privacy-preserving face representation scheme that can be used for authentication by IoT devices is proposed. The scheme can satisfy the resource limits of the IoT devices. Bloom filter is used to ensure the privacy of the scheme. The idea is that the face data is stored in a Bloom filter which can be analyzed to do classification operations. To presreve privacy, no raw face data are stored by distrusted servers which store only Bloom filter representations.

In [28], a privacy-preserving identity authentication scheme based on the face recognition technology is proposed. CNN model is used to extract the facial features from face images. To preserve privacy, the feature vectors are encrypted using a nearest neighbor approach and to match images and identify persons, the cosine similarity is computed over the encrypted vectors. Moreover, for high authentication efficiency, the paper adopts edge computing where some operations are transferred from the cloud to the edge nodes.

In [29], an efficient privacy-preserving face identification system is proposed. For indexing and retrieval of faces, a hash generation scheme based on a Product Quantisation is developed for computing hash codes from faces and creating hash look-up table. Fully homomorphic encryption (FHE) is used to encrypt the face templates to preserve privacy. For authentication, face hashes are used for fast retrieval, i.e., returning a short list of candidates. Two main approaches are used to ensure efficiency. First, the use of look-up table does not need a one-to-many search, but the search results are obtained directly. In the second approach, FHE-based comparisons are executed for a small fraction of facial references.

In [13], an efficient privacy-preserving person re-identification scheme is proposed. To extract the persons' features, convolutional neural network (CNN) and kernels based supervised hashing (KSH) are used. To calculate the similarity of the images' features by the cloud server, a secret sharing based Hamming distance computation protocol is developed. Moreover, to allow users to validate the correctness of the matching results, a dual Merkle hash trees based verification is developed.

C. RESEARCH GAP AND MOTIVATIONS

Most of the existing papers in the literature measure the distance between two embedding vectors and use a threshold to decide whether the vectors are for the same image. However, this approach may not give good results when the data is large and obtained from different camera sources with different image quality and resolutions. Our experiments confirm this and the results are consistent with other works such as [30], [31], [32] which confirm that Euclidean distance metric is not preferable for high dimensional data mining applications. To address this issue, we propose a more accurate approach for multiple cameras surveillance system using a pre-trained machine learning model. The model is designed in such a way that makes its evaluation over encrypted data to preserve privacy efficient. We also train the model using federated learning to avoid sharing the images of the visitors and the persons of interest, and thus preserve privacy.

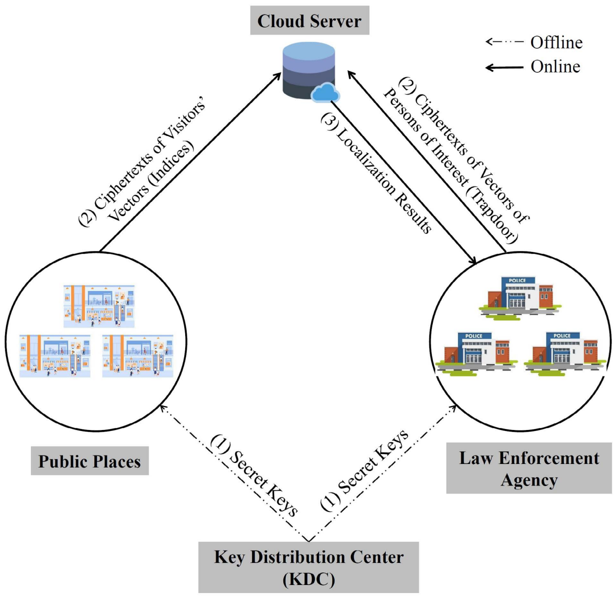
Cloud Server
Offline
Online
(2) Ciphertexts of Visitors' Vectors (Indices)
(3) Localization Results
(2) Ciphertexts of Vectors of Persons of Interest (Trapdoor)
Public Places
Law Enforcement Agency
(1) Secret Keys
(1) Secret Keys
Key Distribution Center (KDC)

FIGURE 1. The network model considered in this paper.

Moreover, most of the existing encryption schemes that are used in the literature to match the images' vectors over encrypted data to preserve privacy are designed for one data source (i.e., single pubic place), and thus, it is inefficient to use them in multi-data-source system (i.e., multiple pubic places). To address this issue, we have modified a cryptosystem [22] that does inner product operations over encrypted data to be suitable for the case of multi-data-source system where vectors can be encrypted by multiple public places and each place uses a different key and the other vector is encrypted by a single entity (law enforcement agency).

To the best of our knowledge, this is the first work that uses a combination of a deep learning model, federated learning, and efficient cryptosystem to evaluate the model using encrypted data to create an accurate and privacy-preserving person localization for multi-camera surveillance system.

III. SYSTEM MODELS AND DESIGN GOALS

In this section, we first discuss the network and threat models considered in this paper, and then, we discuss the design goals that should be realized in our scheme.

A. NETWORK MODEL

Figure 1 depicts the network model considered in this paper. It can be seen that the model has three main parties including an offline key distribution system (KDC), a law enforcement agency, public places, and a cloud server. The role of each party and the communication between them are explained as follows.

- **Offline Key Distribution Center (KDC)**: The KDC distributes the secret keys needed to execute our scheme to the different parties in the system. This process is offline in the sense that once the KDC distributes the keys, it is not involved in the execution of the scheme.

- **Law Enforcement Agency**: This agency is the entity that has images for persons of interest and it needs to know the locations visited by these persons without knowing the images or the embedding vectors of the public places' visitors. For each image, it uses machine learning models to compute the embedding vector of the facial features of the person of interest, and then inputs the vector to a neural network (a part of the machine learning model we propose in this paper) and encrypts the output of the network with an inner product encryption cryptosystem and sends the ciphertext (called trapdoor) to the cloud server. The details of the neural networks and the cryptosystem are discussed in section IV
- **Public Places**: Examples for public places include grocery stores, banks, gymnastics centers, gas stations, etc. Surveillance cameras are installed at the public places. The cameras take pictures of the visitors and use machine learning models to compute embedding vectors containing the visitors' facial features. Then, each public place passes each visitor's vector to a neural network (a part of the machine learning model we propose in this paper) and encrypts the output of the network with an inner product encryption cryptosystem and sends the ciphertext (called index) to the cloud server. The details of the neural networks and the cryptosystem are discussed in section IV.
- **Cloud Server**: The cloud server is an independent entity that is managed and operated by a third party. Using the indices and trapdoors sent by the public places and the law enforcement agency, the server computes the output of our model to learn the locations visited by the persons of interest and communicate this information to the law enforcement agency without knowing the images or the feature vectors of the visitors or the persons of interest.

B. THREAT MODEL

The attackers can be external eavesdroppers or internal entities such as the cloud server, the law enforcement agency, and the public places. The attackers can eavesdrop on all communications in the system. The paper focuses on the honest-but-curious threat model, where attackers do not want to disrupt the system, but they want to infer sensitive information including the images and the embedding vectors of the visitors and the persons of interest.

C. DESIGN GOALS

We aim to achieve the following important requirements in our scheme.

1) Privacy Preservation. Our scheme should enable the law enforcement agency to locate persons of interest while preserving the privacy of the public places' visitors, i.e., attackers should not be able to identify the visitors by revealing their images or feature vectors.

2) Accurate Localization. The accuracy of the person localization should be high under the setting of different public places' camera surveillance systems. To do that, instead of using a simple approach that measures the distance between two embedding vectors and uses a threshold to determine whether the two vectors are for the same person, we train a machine learning model that can better learn the characteristics of the embedding vectors of the same persons to make accurate decisions.

3) Scalability and Efficiency. The system is scalable in the sense that it has many public places and visitors, so our scheme should be efficient in the communication and computation overhead and the server should able to compute the output of the model using the indices of the visitors and the trapdoors of the persons of interest fast. To achieve this requirement, we modify an inner product encryption cryptosystem that requires a pairwise shared key between each public place and the law enforcement agency (single-single setting) so that the law enforcement agency uses only one key and computes only one trapdoor for each person of interest while this trapdoor can be matched to the indices computed with different keys by the public places.

IV. PROPOSED SYSTEM

In a typical face recognition system, the facial features within an image are encoded as a set of real-numbers called an embedding vector. The embedding vector of a person of interest is compared against a set of candidate embedding vectors, and a hit is reported if the distance between two vectors (e.g., using the Euclidean distance) is below a pre-define threshold value. This paper, instead, uses a deep-learning model to decide whether two input vectors are for the same person. In the case of multiple camera surveillance systems, we will demonstrate that this approach is more accurate compared to conventional techniques. Three main stages are executed by our scheme, called *generation of embedding vectors*, *training of a similarity check model* and *encryption and localization*.

In the first stage, law enforcement agency and public places encode the facial features of the image of each person of interest and visitor into an embedding vectors and then encrypt these vectors. A machine learning model is used to locate the face in the input image, and then another model is used to estimate the locations of the face landmarks. A face detection model, in specific, can be used to locate image areas containing faces. Because of its superior results in similar tasks, we use a pre-trained Convolutional Neural Network (CNN) sliding window model, called Dlib [33]. Dlib takes into account cases where a person's face might change depending on his/her posture and emotion. After the face detection, a face landmark localization is used to locate the important features of the face. A set of 68 landmark points on the human face are used to define these features. The points include the mouth, right and left eyes, right and left eyebrows, jawline, and nose. An embedding vector is generated using the 68 landmark points to better quantify face features. The landmark points' coordinates are used to generate a 128-d real-valued number embedding vector that encodes the input face. The details of this stage will be discussed further in subsection IV-A.

In the second stage, a federated deep-learning model is computed to decide whether two embedding vectors are for the same person instead of depending on threshold-based distance metrics. To train the model, a dataset that resembles the images from various public places and a law enforcement agency is created. Each row in the dataset contains two embedding vectors and a label, where one vector is from the dataset of the public places and the other vector is from the dataset of the law enforcement agency. The label indicates whether the two vectors are for different persons (i.e., belong to the negative class) or for the same person (i.e., belong to the positive class). The model is designed in such a way that makes it efficient to be evaluated by the cloud server using encrypted vectors. To do that, each of the two input vectors goes through a distinct set of learning layers. This part of the model is executed by the law enforcement agency and public places. Then, the output vectors of these layers are multiplied by the cloud server (over encrypted data) to compute the classification of the model, i.e., whether the two vectors are for the same person. This stage is demonstrated in detail in subsection IV-B.

Lastly, in the third stage, we investigate an efficient cryptosystem to enable the cloud server to determine the locations visited by a person of interest by executing the last layer in the model over the encrypted data without inferring the vectors or the images of the persons of interest and the visitors of the public places. This stage will be explained in subsection IV-C.

A. GENERATION OF EMBEDDING VECTORS

This subsection demonstrates the generation of the embedding vectors from the images of persons' faces. To generate embedding vectors, three stages are required including *face detection*, *face landmark localization*, and *embedding vector computation*. The details of these stages are as follow.

Face Detection: This step locates the areas of human faces in an image [34]. Viola-Jones' face detection system is developed for low-cost cameras [35]. The system is an object detection framework that integrates concepts such as Haar-like features, integral images, and cascade classifiers to provide fast and accurate object detection system. Recently, more accurate and low-cost solutions were developed. Deep learning approaches such as [33], [36], [37], [38], [39] are regarded to have the best detection performance. As a result, we use Dlib [33], the pre-trained CNN sliding window model, because of its high performance. The input of the model is a window taken from the image of interest and the output indicates whether there is a face in the window or not. The model is trained on the iBUG 300-Faces-In-the-Wild landmark dataset [40], and it has three down-sampling layers, four convolution layers, and a feed-forward layer.

Landmarks Localization: The features of a person's face may change depending on his or her posture, lighting conditions, and facial expression. Consequently, even images of the same person can result in low similarity score under different conditions. Thus, face landmark localization is adopted to extract a variation-independent set of features that can boost the similarity score under different conditions. A set of 68 landmark points that define mouth, right and left eyebrows, nose, right and left eyes, and jawline are used as variation-independent features. Table 1 gives the number of the landmark points and its belonging to the facial regions. Our approach relies on the Dlib [33] face detection and the face alignment technique proposed in [41]. The Dlib is trained on a dataset of facial landmarks of different persons where each landmark is defined with (x, y) coordinates.

TABLE 1. Facial landmarks and their positions.

Feature	Range
Chin	[1-17]
Left brow	[18-22]
Right brow	[23-27]
Nasal bridge	[28-31]
Nose tip	[32-36]
Left eye	[37-42]
Right eye	[43-48]
Top lip	[49-55] & [61-65]
Bottom lip	[56-60] & [66-68]

The intensity of the pixels at each landmark is used to train a set of t tree-based regression predictor functions $\{r_0, r_1, \ldots, r_{t-1}\}$. For an input image, each predictor function estimates the position of the facial landmarks' position. The gradient boosting tree algorithm [40] is used to train the predictor functions on the iBUG 300-Faces-In-the-Wild landmark dataset.

Computation of Embedding Vector: This stage converts the detected landmark positions into an embedding vector that expresses the facial features. A deep learning model is used to generate the embedding vector where the input is three images called *triplets*. Two images of the triplets are for the same person but under different conditions. These two images are referred as the *anchor*, and the *positive*. The third image is called the *negative* input where a different random face is used.

The loss function used in the training has two components. The first component aims to reduce the difference between the vectors generated for the *positive* and the *anchor* inputs. The second component aims to increase the difference between the vectors generated for the *negative* and *anchor* input. The triplet loss function is defined as follows:

$$\mathcal{L}_\theta(Pos, An, Neg) = max(\|f_\theta(Pos) - f_\theta(An)\|^2 - \|f_\theta(Neg) - f_\theta(An)\|^2 + \gamma, 0)$$

where Pos, Neg, and An define the *positive*, *negative*, and *anchor* inputs, respectively. The value γ is a small margin between *negative* and *positive* inputs. It should be noted that

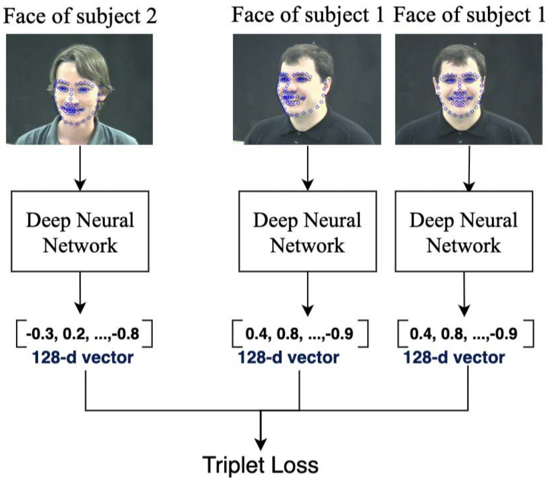

FIGURE 2. **The embedding vector is constructed by a deep learning model that takes three inputs called triplets. The triplets provide positive and anchor inputs for a person's face, as well as negative input for a second person face. The model generates a 128d vector for each positive face by employing a triplet loss function.**

θ defines the optimal parameters of the deep neural network. The cost over a batch of M training triplets can be used to describe the training optimization as follows:

$$\mathcal{J} = \min_{\theta} \sum_{j=1}^{M} \mathcal{L}_{\theta}(Pos^j, An^j, Neg^j)$$

The architecture of a network relies on the implementation of the ResNet34 network as discussed in [42]. However, less number of layers and filters is used to speed up the training. A dataset collected from a variety of sources is used for training [33]. The dataset size is approximately three million images. The performance of this network outperforms the existing image recognition approaches in accuracy [33].

B. TRAINING OF A SIMILARITY CHECK MODEL

Using a standard distance metrics (e.g., Euclidean distance) to measure the similarity between embedding vectors is not often a preferable choice because of the low performance in many practical applications [30]. This is because they need to compute a threshold for the maximum distance allowed between two embedding vectors to be for the same person. In addition, standard distance metrics are excessively reliant on the image source from which the vectors are computed.

As discussed in our network model, various kinds of surveillance cameras may be deployed in public spaces, with cameras of varying models, resolutions, and images quality. Thus, *an optimal distance threshold for a dataset generated by one camera might not be the optimal threshold for another camera*. To demonstrate this claim, we have used six publicly available datasets to conduct experiments that identify the optimal threshold of each dataset. The datasets include IRIS Dataset [43], Head Pose Image Dataset (HPID) [44], the Extended Yale Face Dataset B (EYaleB) [45], FEI Face Database [46], Georgia Tech Face Dataset (GTech) [44], and Yale Face Dataset [47]. More details about the datasets will be discussed in section VI.

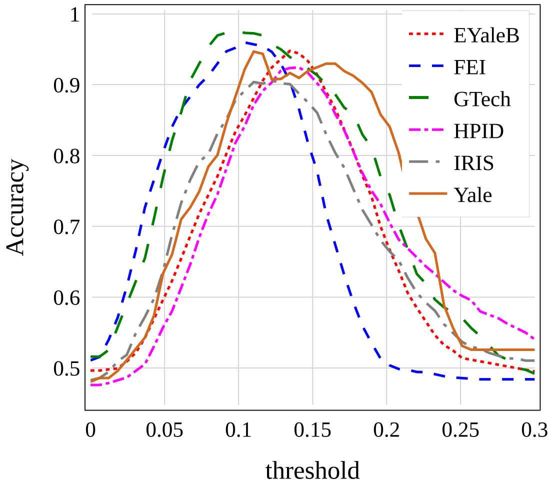

FIGURE 3. **The accuracy versus the threshold value.**

Figure 3 gives the relation between the accuracy in terms of the percentage of properly identified images and the threshold value for the six datasets. It can be seen that FEI's optimal threshold is around 0.1 which means that if the distance between two vectors is less 0.1, they are deemed for the same person. When the threshold value is below 0.1, the accuracy starts to degrade as the Euclidean distance between the two vectors of the same person exceeds this low threshold, and thus, they are deemed for different persons. A degradation in the accuracy also occurs when the threshold value is set greater than 0.1. This is because a person's image is mistakenly deemed to belong to other persons because their Euclidean distances are less than this large threshold value. The same conclusions can be drawn from the results of the other datasets, but with different optimal threshold values. Consequently, *It is not possible to find a threshold value that provides optimal performance for all datasets*. To address this issue, we train a deep learning model that can accurately determine the embedding vectors of the same person instead of depending on threshold-based distance metrics. We design the model in such a way that requires an efficient cryptosystem to evaluate it using encrypted data to preserve privacy, as will be explained later.

The design of our deep learning model is shown in Figure 4. The model is executed by three parties as follows. The public place evaluates the first set of layers of the model using the embedding vector of each visitor, while the law enforcement agency evaluates the second set of layers using the embedding vector of each person of interest. The outputs of these two sets of layers are encrypted using the cryptosystem that will be discussed in subsection IV-C, and then the ciphertexts are sent to the cloud server. Finally, the server evaluates the last layer in the model by computing the inner product of the two vectors using their ciphertexts and executes

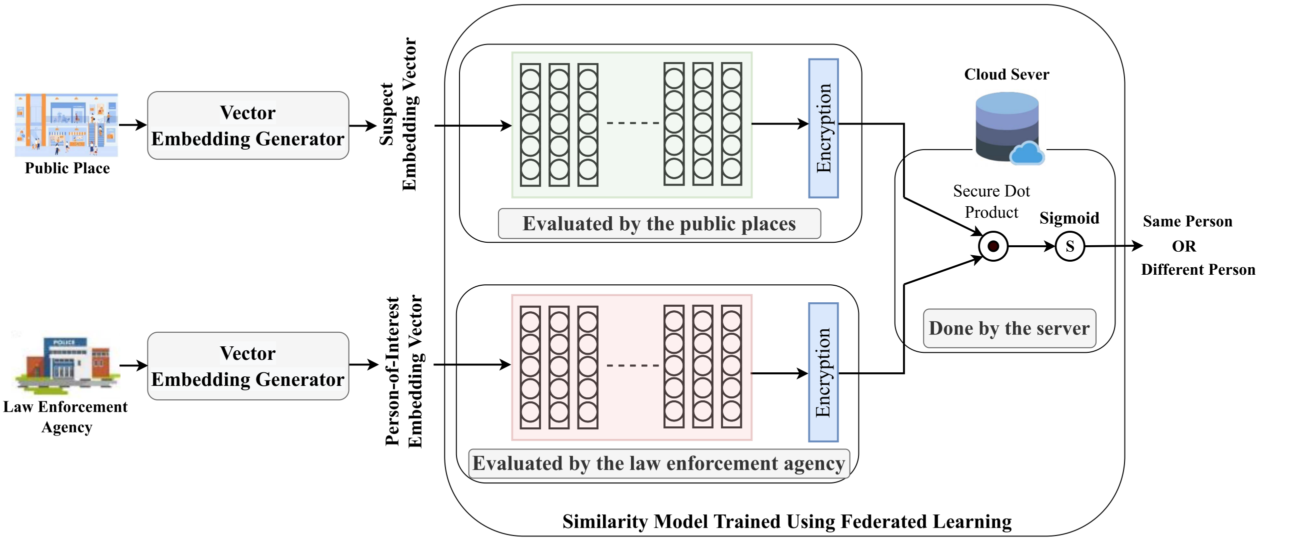
Public Place
Vector Embedding Generator
Suspect Embedding Vector
Law Enforcement Agency
Vector Embedding Generator
Person-of-Interest Embedding Vector
Encryption
Evaluated by the public places
Encryption
Evaluated by the law enforcement agency
Cloud Sever
Secure Dot Product
Sigmoid
S
Same Person OR Different Person
Done by the server
Similarity Model Trained Using Federated Learning

FIGURE 4. An overview for the deep learning model proposed in this paper.

Algorithm 1: `FederatedAveraging`. The Training Batch Size Is B. The Learning Rate Is η. [48]

```
Server()
    Randomly initialize weights $w_0$;
    for each communication round $t = 1, 2, \ldots$ do
        P ← Select random set of participants;
        for each participant $p_k \in P$ do
            $w^k_{t+1} \leftarrow$ ParticipantUpdate(p,$w_t$);
        $w_{t+1} \leftarrow \sum_k \frac{|p_k|}{|P|} w^k_{t+1}$;
return;
ParticipantUpdate(p, w);
    for each training epoch $e = 1, 2, \ldots, E$ do
        for each batch b of size B in client p data do
            $w \leftarrow= w - \eta \nabla \ell(w; b)$
return w to the server;
```

a sigmoid activation function over the output to classify the vectors either for the same person or not. The key reason for this design is that most of the computations can be done in the plaintext domain and it needs an efficient cryptosystem to enable the server to execute the last layer and learn the visited locations by the persons of interest without being able to obtain the images or the embedding vectors of the persons of interest or the public places' visitors. The details of the cryptosystem and the secure inner product evaluation are discussed in subsection IV-C.

To train the model, federate learning algorithm denoted as `FederatedAveraging` [48] is employed. The idea is that each public place and the law enforcement agency creates a local dataset where each row in the dataset contains two embedding vectors and a label which indicates whether the two vectors are for the same person (i.e., belong to the positive class) or for different person (i.e., belong to negative class). Once the local datasets are created, the public places and the law enforcement agency participate in the training of the model by first training local models on their local datasets and then sharing their models' updates with the cloud server as illustrated in Algorithm 1.

For every communication round t, the global model w_t computed by the server is downloaded by each participant. Then, each participant p computes the weights updates on their local dataset p_k using the current version of the global model and sends the ephemeral and focused updates to the server. The cloud server combines the updates of the different participants by averaging them to create a more accurate global model w_{t+1}. Weighted averaging is used by the server to compute the aggregated model weights. Note that, the initial global model w_0 is either selected randomly or by pre-trained model on a public dataset. Federated learning has been proved to be more secure with using approaches like privacy-preserving data aggregation [49] and differential privacy [50].

There are two main advantages that can be achieved by using federated learning. First, it can preserve privacy because the participants do not need to reveal their sensitive data. It can also achieve efficiency because the participants share only the updates of the local models whose size is much smaller than the size of the dataset.

C. ENCRYPTION AND LOCALIZATION

This subsection explains

V. PRIVACY AND SECURITY ANALYSIS

Our scheme can achieve the following prepositions.

Preposition 1: The cloud server can learn the locations visited by a person-of interest without being able to identify the visitors or the persons of interest.

Proof: We use the following notations to prove this preposition.

History. We denote the group of encrypted vectors resulted from the evaluations of the neural networks by the public places on the embedding vectors of the visitors' images as a set of indices $\mathcal{I}_P = \{I_1, I_2 \ldots, I_k\}$ which correspond to the vectors $W = \{w_1, \ldots, w_k\}$. In additions, we denote the group of encrypted vectors resulted from the evaluations of the law enforcement agency's neural networks on the embedding vectors of the person-of-interest images as trapdoors $T_{LEA} = \{T_1, T_2, \ldots, T_j\}$, which correspond to $\mathcal{V} = \{v_1, \ldots, v_j\}$. The history is denoted as $Hst = \{\mathcal{I}_P,\ T_{LEA}\}$.

Trace. It represents the information the cloud server can deduce by analyzing the history Hst, denoted as $Tra(Hst)$, where $Tra(Hst)$ is defined over all the trapdoors, i.e., $Tra(Hst) = \{Tra(T_1), \ldots, Tra(T_l)\}$. The search pattern is an example for traces.

View. It represents the observation of the cloud server, which is represented by the encrypted history and its trace, denoted by $View(\mathcal{I}_P, T_{LEA}, Tra(Hst))$.

A simulator $\mathcal{SIM}$ wants to compute a false view $View'$ that is indistinguishable from the true view $View$ using the following steps.

1) $\mathcal{SIM}$ executes the oracle $SystemSetup()$ to get a secret key $\mathcal{SK}'$.
2) $\mathcal{SIM}$ computes a set of visitors' embedding vectors $W' = \{w'_1, \ldots, w'_k\}$ such that $|w_i| = |w'_i|$, $1 \leq i \leq k$, where W' is a random copy of W.
3) $\mathcal{SIM}$ computes a set of embedding vectors for random images $\mathcal{V}' = \{v'_1, \ldots, v'_j\}$ such that $|v_i| = |v'_i|$, $1 \leq i \leq j$. Note that $\mathcal{V}'$ is a random copy of $\mathcal{V}$.
4) $\mathcal{SIM}$ computes an index $\mathcal{I}'_P$ and trapdoor T'_{LEA} using $\mathcal{SK}'$, $\mathcal{V}'$, and T'_{LEA}.

Based on the above construction, our scheme can achieve adaptive distinguishability if for any $\mathcal{SIM}$ with a history $Hst' = \{\mathcal{I}'_i,\ I_T{}'\}$ and trace $Tra(Hst')$ similar to $Tra(Hst)$ such that an adversary cannot distinguish between the two views $View(\mathcal{I}_P, T_{LEA}, Tra(Hst))$ and $View'(\mathcal{I}'_P, T'_{LEA}, Tra(Hst'))$.

Preposition 2: The embedding vectors of outsourced visitors' indices and trapdoors of persons of interest cannot be obtained by adversaries.

Proof: In our scheme, an inner product encryption cryptosystem is used to encrypt the resulted vector after inputting the embedding vector of a visitor or a person of interest to a neural network. Without knowledge of the secret keys, decrypting the indices and trapdoors is impossible. Because each pubic place uses a unique key, the indices of a public place cannot be decrypted with the secret keys of the other places. We conclude that our scheme is secure in the *known ciphertext model*, where attackers cannot obtain the secret keys or the plaintext vectors using the trapdoors and indices.

Preposition 3: The indices of same persons are not linkable under the known-ciphertext model.

Proof: Our cryptosystem uses random numbers in the encryption process to ensure that the ciphertexts of the same embedding vectors look different and are unlinkable. Specifically, for each visitor's embedding vector, the public place generates a random number by picking up a random element $\alpha \leftarrow \mathbb{Z}_q$ and uses it to compute the index, and the law enforcement agency uses $\beta \leftarrow \mathbb{Z}_q$ in the computation of the trapdoor, so when an index or trapdoor is computed for the same image's embedding vector, it looks different. This feature is important to prevent linking the indices of the same person who visits different public places. Tracing the locations of a person for a long time may lead to the identification of the person from the visited locations.

Preposition 4: Each public place cannot decrypt the ciphertexts of other places because a shared key is not used, i.e., each public place has a unique secret key.

Proof: If a public place can decrypt the ciphertexts of other places, it can track the locations visited by the visitors because the plaintext embedding vectors of the same person are close, and then it can identify the persons from the visited locations. In our scheme, the indices computed by one public place cannot be decrypted by other places because all public places do not use the same secret key, but each place uses a unique key. In spite of using different keys by the public places to compute the indices, the cloud server is still able to evaluate the machine learning model by computing the inner product of the indices and the trapdoors computed by the law enforcement agency. Moreover, a public place cannot use its secret key to compute the secret keys of the other public places because the key has $\left(\mathbf{N_1}^{-1}\mathbf{B}', \mathbf{N_2}^{-1}\mathbf{B}''\right)$ and $\mathbf{B}' + \mathbf{B}'' = \mathbf{B}^{-1}$ and thus the public place cannot know the master key $\mathbf{N_2}^{-1}$, $\mathbf{N_1}^{-1}$, and $\mathbf{B}^{-1}$. It cannot also know the random matrices $\mathbf{B}'$ and $\mathbf{B}''$ that are used to compute the other public places' keys.

Preposition 5: The cloud server should not be able to match a large number of indices and trapdoors to avoid leaking side information

Proof: The cloud server should be able to match indices and trapdoors to find the locations visited by persons of interest without being able to identify the persons. However, if the cloud server has a large amount of data collected over a long period of time, it may use the data to infer statistical and side information such as collecting a large number of locations visited by an anonymous person of interest. To prevent the cloud server from collecting side information, the keys of the involved parties should change frequently, e.g., every month, to make sure that the ciphertexts sent after updating the keys cannot be matched to the old ciphertexts because they are encrypted with different keys.

VI. EXPERIMENTAL RESULTS

In this section, the performance of the proposed scheme is evaluated using the following metrics: (1) computation and communication overhead, and (2) localization accuracy.

TABLE 2. The computation times of the main operations used by our scheme.

Operation	Average Time
Bilinear pairing (T_B)	6.6981 $msec$
Exponentiation (T_E)	1.196 $msec$
Multiplication (T_M)	0.005 $msec$
Addition (T_A)	0.0021 $msec$

A. EVALUATIONS OF THE CRYPTOSYSTEM

Our scheme is implemented using Python programming language and a machine with Intel 8 Cores i7-8665U CPU 1.90GHz processor and 16 GB RAM. This subsection discusses the communication and computation overhead of our scheme.

1) COMPUTATION OVERHEAD

The computation overhead is measured by the times needed to encrypt a vector by the law enforcement agency and public places, evaluate the model using encrypted data by the cloud, and compute the keys by the KDC. Table 2 gives the computation times of the main operations used by our scheme, where T_B, T_E, T_M and T_A stands for the times required for computing one bilinear pairing, exponentiation, multiplication, and addition, respectively. These operations are used in our scheme to compute keys, encrypt vectors, and measure the similarity of two vectors using their trapdoors and indices.

The last layer of the neural network that is encrypted by the public places and the law enforcement agency is composed of 16 group elements. To compute the key of the law enforcement agency, 4096 multiplication operations are needed for a vector size of 16 elements and 3840 addition operations, i.e., $4096 * T_M + 3840 * T_A$, which takes around $20.48 + 8.1 = 28.58$ ms using the measurements given in Table 2. The key of each public place requires 4096 multiplication operations and 4096 addition operations, i.e., $4096 * T_M + 4096\ T_A$, which takes $20.48 + 8.6 = 29.08$ ms. Also, as shown in Figure 5, our scheme can reduce the number of keys that need to be computed in the system from $2n$ in the cryptosystem [22] to $n + 1$ in our scheme, where n is the number of public places. This is because the cryptosystem [22] is designed for single public place and single law enforcement agency setting, where the law enforcement agency needs to share a unique key with each public place, while our scheme is designed for multiple public places and single law enforcement agency setting, where each public place and the law enforcement agency use only one key.

To encrypt the vector of a person-of-interest by the law enforcement agency or the vector of a visitor by each public place, 33 exponentiation operations, 480 addition operations, and 544 multiplication operations are needed, i.e., $33 * T_E + 480 * T_A + 544 * T_M$, which takes $39.47+1+2.72 = 43.2$ ms. To compute the inner product of two vectors using indices and trapdoors, the cloud server needs 33 bilinear pairing and 16 multiplication operations, i.e., $33 * T_B + 16\ T_M$, which takes $221.04+0.08 = 221.12$ ms. Also, as shown in Figure 6, our scheme can reduce the number of encryption operations that are needed for each vector of a person-of-interest from n in the cryptosystem [22] to only one. This is because in [22], the law enforcement agency needs to encrypt each vector n times with the n keys shared with the public places, while in our scheme, the law enforcement agency has only one key that is used to do only one encryption operation.

Based on the results given above, we can conclude that the computation times are in the order of $msecs$. This proves that our scheme is efficient, practical, and scalable. The scalability is important in our application because the public places may be visited by a large number of persons, and thus they need to do many encryptions and the cloud server needs to do a lot of localization operations.

2) COMMUNICATION OVERHEAD

The communication overhead is measured by the size of the messages sent to the cloud server and also the number of keys that are distributed to the system's parties. Since the last layer of the neural network at the public places and the law enforcement agency is composed of 16 elements, as given in Section VI-B2, using asymmetric pairing curve (BN256) of size 256 bits where the size of a group element is 32 Bytes, the size of each encrypted vector (index or trapdoor) is 33× 32 Bytes (1.056 KB). Also, as shown in Figure 6, our scheme reduces the number of encryptions that are needed for each vector of a person-of-interest from n in the cryptosystem [22] to only one. For the key size, it has two matrices with 16×16 elements in $\mathbb{Z}_q$. The key size is $2 \times 16 \times 16 \times 16 = 8$ KB, where the size of each element in $\mathbb{Z}_q$ is 16 Bytes. Figure 5 indicates that our scheme can reduce the number of keys that need to be distributed in the system from $2n$ in the cryptosystem [22] to $n + 1$ in our scheme, where n is the number of public places.

The results given above indicate that the communication overhead of our scheme is acceptable and the existing communication protocols can transmit the encrypted vectors in short time. We can conclude that our scheme is efficient and practical.

B. LOCALIZATION ACCURACY

1) METRICS AND DATASETS

a: PERFORMANCE METRICS:

Four metrics are used to evaluate the accuracy of the localization, including the false acceptance rate (FA), localization rate (LR), and highest difference (HD), and the accuracy. FA measures the percentage of visitors that are incorrectly classified as persons of interest. LR calculates the percentage of persons of interest that are correctly localized. HD calculates the difference between LR and FA. The accuracy calculates the ratio of number of correct predictions to the total number of predictions. These metrics are defined as follows.

$$LR = \frac{TP}{TP + FP}, \quad FA = \frac{FP}{TN + FP}, \quad HD = LR - FA$$

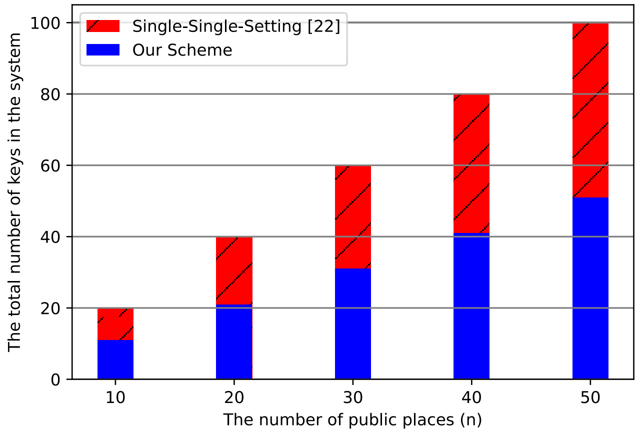

FIGURE 5. **The total number of keys in the system with using the cryptosystem in [22] (single public place and single law enforcement agency setting) and our scheme (multiple public places and single law enforcement agency setting).**

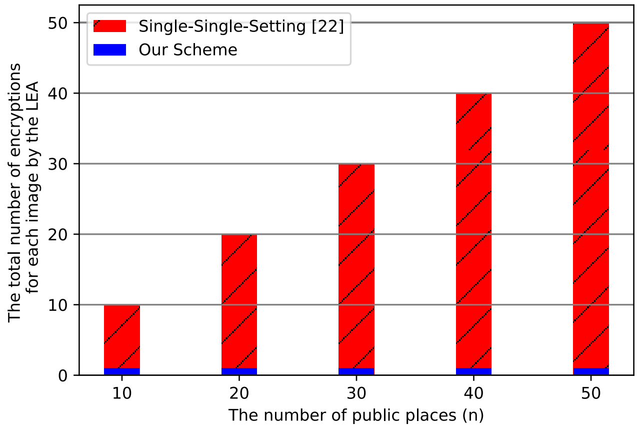

FIGURE 6. **The total number of encryptions needed for each image of a person-of-interest using the cryptosystem in [22] (single public place and single law enforcement agency setting) and our scheme (multiple public places and single law enforcement agency setting).**

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP, FP, and TN stand for true positive, false positive, and true negative.

b: DATASETS USED AND PREPROCESSING

The datasets used in our experiments to evaluate our machine learning model include IRIS Dataset [43], Head Pose Image Dataset (HPID) [44], Georgia Tech Face Dataset [44], Yale Face Dataset [47], FEI Face Database [46], and the Extended Yale Face Dataset B (EYaleB) [45]. Each dataset is processed and assumed to belong to one public place. The subjects in each dataset were divided into two groups. The first group in dataset i is selected randomly and denoted as X^i_{POI}, and it represents the images of the persons of interest. The second set of images, denoted as X_{PP}, represents the images of the visitors of the public places. The two groups have an equal number of images.

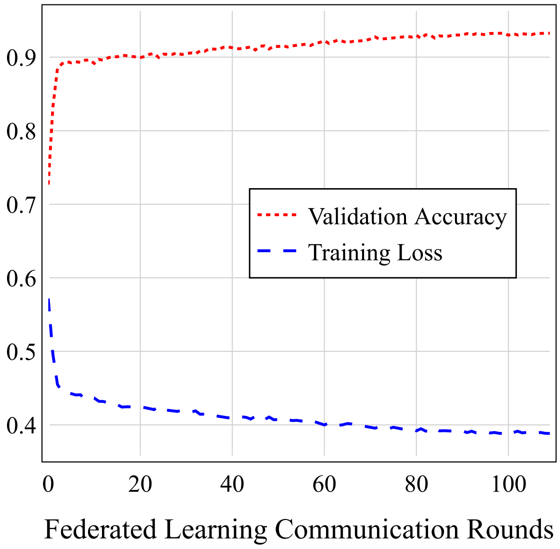

FIGURE 7. **The communication rounds required by the federated learning versus training loss and accuracy.**

2) RESULTS AND DISCUSSION

Python Dlib [33] *face_recognition* library is used to generate the embedding vectors of each dataset. Each embedding vector is normalized to the unit norm using l2-normalization, allowing the Euclidean distance to be determined from the dot product of any two embedding vectors. As shown in Fig. 4, the input of our model is two embedding vectors and a feed forward architecture is used. We use 5-fold cross validation to find the optimal values for the model's hyper-parameters, such as the type of activation functions and the number of layers, and the top performing model is selected. We found that the top performing model has eight hidden layers with the following dimensions [128, 128, 64, 64, 32, 32, 16, 16], hyperbolic tangent activation function, and Adam optimizer.

Our privacy-preserving federated learning model (i.e., with encrypted vectors, denoted as MD + Privacy) is compared to two baselines. The first baseline is a federated learning model without privacy (denoted as MD). In the federated learning, each dataset is partitioned into training and testing with ratio 5:1. The deep architecture of all the models is set to be the same. The second baseline uses the Euclidean distance metric for localization instead of a machine learning model, where each dataset is used to compute the threshold needed to locate the persons-of-interest. The results are given in Table 3.

The results indicate that our scheme performs better than the Euclidean distance approach because instead of using a threshold to decide the similarity of an index and a trapdoor, we use a machine learning which can learn the features of the embedding vectors of the same persons and thus make accurate decisions. Note that, as discussed earlier, it may be difficult to find a good threshold that can give good performance in case of images taken from different sources. Moreover, the results indicate that the privacy-preserving model (MD + Privacy) performs almost similar to the plaintext model (MD)

TABLE 3. The performance results of MD, MD+Privacy, and the Euclidean distance approach.

	Euclidean Distance						Federated Learning	
	EYaleB	FEI	GTech	HPID	IRIS	YALE	MD	MD+Privacy
LR	0.876	0.905	0.902	0.862	0.871	0.884	0.950	**0.945**
FA	0.082	0.079	0.070	0.082	0.070	0.083	0.040	**0.042**
HD	0.794	0.826	0.832	0.780	0.801	0.801	0.910	**0.903**
Accuracy	0.907	0.895	0.912	0.886	0.892	0.901	0.954	**0.944**

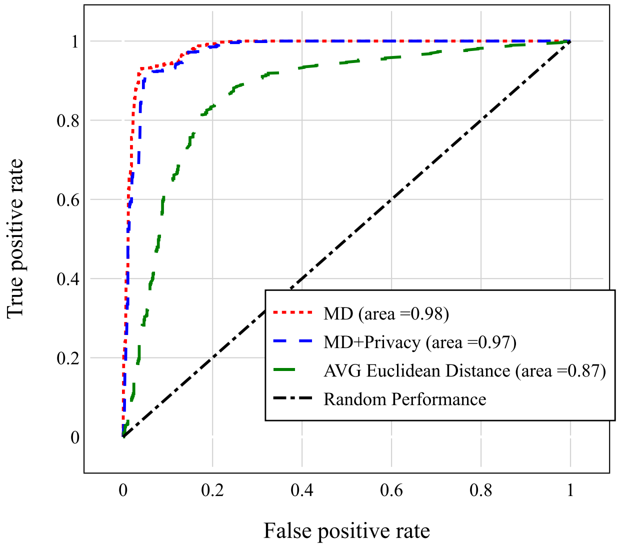

FIGURE 8. The receiver operating characteristic curves of MD, MD+Privacy, and the average result of the Euclidean distance.

in terms of overall performance. This indicates that executing the model over encrypted data using our cryptosystem does not degrade the accuracy of the model.

Figure 7 shows the number of communication rounds needed by the federated averaging algorithm to converge. The figure shows that our model starts to achieve over 90% of training accuracy after the first 20 communication rounds, and it takes around 80 rounds for the loss and the validation accuracy to converge. The given results indicate the efficient training of the developed neural network architecture using federated learning. Figure 8 shows the Receiver Operating Characteristics (ROC) and the Area Under Curve (AUC) for MD, MD+Privacy, and Euclidean distance approach. The black line indicates a random performance classifier. The given results of both MD and MD+Privacy indicate that the performance of our scheme with privacy preservation is comparable to that of the scheme without privacy preservation with almost no performance loss. In addition, both MD and MD+Privacy outperform the Euclidean distance approach.

VII. CONCLUSION

This paper proposes an accurate person localization scheme that enables a law enforcement agency to locate persons of interest with privacy preservation. Our scheme trains a machine learning model to decide whether two embedding vectors storing facial features are for the same persons. Using six publicly available datasets, our experimental results indicate that our approach is more accurate than the existing approaches that measure the Euclidean distance because an optimal decision threshold of a dataset might not be the optimal threshold for the other datasets. Our machine learning model is designed in such a way that makes executing it over encrypted data efficient. Most of the model's layers are executed using plaintext data by the public places and the law enforcement agency and only one layer is executed by the server over encrypted data using an inner product encryption scheme to preserve privacy. We have also modified an inner product encryption cryptosystem that is designed for a single public place to make it more efficient in our application that has multiple public places. Our experiments indicate that this modification can significantly reduce the number of keys in the system and the number of ciphertexts that are computed by the law enforcement agency. To prevent leaking sensitive information by sharing the images of the visitors to train the model, we use a federated learning training approach. Our experiments indicate that our scheme has high localization accuracy and the use of federated learning and executing a part of the model over encrypted data has a slight impact on the accuracy. The results of a formal proof and extensive analysis confirm that our scheme can preserve the privacy of the public places visitors.

REFERENCES

[1] C.-T. Hsieh, C.-C. Han, C.-H. Lee, and K.-C. Fan, "Person authentication using nearest feature line embedding transformation and biased discriminant analysis," in *Proc. Int. Carnahan Conf. Secur. Technol. (ICCST)*, Oct. 2017, pp. 1–5.

[2] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1701–1708.

[3] Z. Yang, S. Yu, W. Lou, and C. Liu, "P^2: Privacy-preserving communication and precise reward architecture for V2G networks in smart grid," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 697–706, May 2011.

[4] Y. Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2892–2900.

[5] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, vol. 2. Cambridge, MA, USA: MIT Press, 2014, pp. 1988–1996.

[6] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. Brit. Mach. Vis. Assoc.*, 2015, pp. 1–12.

[7] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "VGGFace2: A dataset for recognising faces across pose and age," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*. Los Alamitos, CA, USA: IEEE Computer Society, May 2018, pp. 67–74, doi: 10.1109/FG.2018.00020.

[8] R. Ranjan, V. M. Patel, and R. Chellappa, "HyperFace: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 1, pp. 121–135, Jan. 2019.

[9] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.

[10] L. Du and H. Hu, "Face recognition using simultaneous discriminative feature and adaptive weight learning based on group sparse representation," *IEEE Signal Process. Lett.*, vol. 26, no. 3, pp. 390–394, Mar. 2019.

[11] X. Li, Q. Xue, and M. C. Chuah, "CASHEIRS: Cloud assisted scalable hierarchical encrypted based image retrieval system," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, May 2017, pp. 1–9.

[12] L. Weng, L. Amsaleg, A. Morton, and S. Marchand-Maillet, "A privacy-preserving framework for large-scale content-based information retrieval," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 1, pp. 152–167, Jan. 2015.

[13] H. Cheng, H. Wang, X. Liu, Y. Fang, M. Wang, and X. Zhang, "Person re-identification over encrypted outsourced surveillance videos," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 3, pp. 1456–1473, Jun. 2021.

[14] E. M. Newton, L. Sweeney, and B. Malin, "Preserving privacy by de-identifying face images," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 2, pp. 232–243, Feb. 2005.

[15] K. W. Bowyer, "Face recognition technology: Security versus privacy," *IEEE Technol. Soc. Mag.*, vol. 23, no. 1, pp. 9–19, Jun. 2004.

[16] *Chinese Facial Recognition Company Exposes Data of Millions of People*. Accessed: Apr. 10, 2022. [Online]. Available: https://edgy.app/chinese-facial-recognition-company-data

[17] *What is, 'Skynet', China's Massive Video Surveillance Network*. Accessed: Apr. 10, 2022. [Online]. Available: https://www.abacusnews.com/who-what/skynet-chinas- massive-video-surveillance-network/article/2166938

[18] *Facebook Agrees to Pay 550 Million to Settle Privacy Class Action*. Accessed: Apr. 10, 2022. [Online]. Available: https://www.courthousenews.com/facebook-agrees-to-pay-550-million-to-settle-privacy-class-action/

[19] *San Francisco Becomes the First us City to Ban Facial Recognition by Government Agencies*. Accessed: Apr. 10, 2022. [Online]. Available: https://www.theverge.com/2019/5/14/18623013/sanfrancisco-facial-recognition-ban-vote-city-agencies

[20] *Oakland City Council Votes to Ban Government Use of Facial Recognition*. Accessed: Apr. 10, 2022. [Online]. Available: https://www.theverge.com/2019/7/17/20697821/oakland-facial-recogntiion- ban-vote-governement-california

[21] *Facial Recognition Technology Scrapped at King's Cross Site*. Accessed: Apr. 10, 2022. [Online]. Available: https://www.theguardian.com/technology/2019/sep/02/facial-recognition-technology-scrapped-at-kings-cross-development

[22] S. Kim, K. Lewi, A. Mandal, H. Montgomery, A. Roy, and D. J. Wu, "Function-hiding inner product encryption is practical," in *Security and Cryptography for Networks*, D. Catalano and R. De Prisco, Eds. Cham, Switzerland: Springer, 2018, pp. 544–562.

[23] B. Fischer, A. Brosig, P. Welter, C. Grouls, R. W. Günther, and T. M. Deserno, "Content-based image retrieval applied to bone age assessment," *Proc. SPIE*, vol. 7624, Mar. 2010, Art. no. 762412.

[24] S. Liu, Z. Song, G. Liu, C. Xu, H. Lu, and S. Yan, "Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1335–1336.

[25] Z. Xia, X. Wang, L. Zhang, Z. Qin, X. Sun, and K. Ren, "A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 11, pp. 2594–2608, Nov. 2016.

[26] Y. Huang, J. Zhang, L. Pan, and Y. Xiang, "Privacy protection in interactive content based image retrieval," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 3, pp. 595–607, Jun. 2020.

[27] W. Xue, W. Hu, P. Gauranvaram, A. Seneviratne, and S. Jha, "An efficient privacy-preserving IoT system for face recognition," in *Proc. Workshop Emerg. Technol. Secur. IoT (ETSecIoT)*, Apr. 2020, pp. 7–11.

[28] X. Wang, H. Xue, X. Liu, and Q. Pei, "A privacy-preserving edge computation-based face verification system for user authentication," *IEEE Access*, vol. 7, pp. 14186–14197, 2019.

[29] D. Osorio-Roig, C. Rathgeb, P. Drozdowski, and C. Busch, "Stable hash generation for efficient privacy-preserving face identification," *IEEE Trans. Biometrics, Behav., Identity Sci.*, vol. 4, no. 3, pp. 333–348, Jul. 2021.

[30] C. Aggarwal, D. Keim, and A. Hinneburg, "On the surprising behavior of distance metrics in high dimensional spaces," in *Proc. Int. Conf. Database Theory*. Springer, 2001, pp. 420–434.

[31] S. Xia, Z. Xiong, Y. Luo, WeiXu, and G. Zhang, "Effectiveness of the Euclidean distance in high dimensional spaces," *Optik*, vol. 126, no. 24, pp. 5614–5619, Dec. 2015.

[32] C. R. Giannella, "Instability results for Euclidean distance, nearest neighbor search on high dimensional Gaussian data," *Inf. Process. Lett.*, vol. 169, Aug. 2021, Art. no. 106115.

[33] D. E. King, "Dlib-ml: A machine learning toolkit," *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, Jul. 2009.

[34] E. Hjelmås and B. K. Low, "Face detection: A survey," *Comput. Vis. Image Understand.*, vol. 83, no. 3, pp. 236–274, Sep. 2001.

[35] P. Viola and M. J. Jones, "Robust real-time object detection," *Int. J. Comput. Vis.*, vol. 4, nos. 34–47, p. 4, 2001.

[36] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "From facial parts responses to face detection: A deep learning approach," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3676–3684.

[37] X. Sun, P. Wu, and S. C. H. Hoi, "Face detection using deep learning: An improved faster RCNN approach," *Neurocomputing*, vol. 299, pp. 42–50, Jul. 2018.

[38] J. Mehta, E. Ramnani, and S. Singh, "Face detection and tagging using deep learning," in *Proc. Int. Conf. Comput., Commun., Signal Process. (ICCCSP)*, Feb. 2018, pp. 1–6.

[39] S. Yang, Y. Xiong, C. Change Loy, and X. Tang, "Face detection through scale-friendly deep convolutional networks," 2017, *arXiv:1706.02863*.

[40] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: Database and results," *Image Vis. Comput.*, vol. 47, pp. 3–18, Mar. 2016.

[41] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1867–1874.

[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[43] J. W. Davis. *OTCBVS Benchmark Dataset Collection*. Accessed: Jan. 10, 2022. [Online]. Available: http://vcipl-okstate.org/pbvs/bench/index.html

[44] L. Chen, H. Man, and A. V. Nefian, "Face recognition based on multi-class mapping of Fisher scores," *Pattern Recognit.*, vol. 38, no. 6, pp. 799–811, 2005.

[45] A. S. Georghiades, P. N. 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.

[46] N. Gourier, D. Hall, and J. L. Crowley, "Estimating face orientation from robust detection of salient facial features," in *Proc. ICPR Int. Workshop Vis. Observ. Deictic Gestures*, 2004, pp. 1–9.

[47] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[48] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artif. Intell. Statist.* 2017, pp. 1273–1282.

[49] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. Brendan McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for federated learning on user-held data," 2016, *arXiv:1611.04482*.

[50] S. Truex, L. Liu, K.-H. Chow, M. E. Gursoy, and W. Wei, "LDP-Fed: Federated learning with local differential privacy," in *Proc. 3rd ACM Int. Workshop Edge Syst., Anal. Netw.*, Apr. 2020, pp. 61–66.

MAHMOUD NABIL received the B.S. and M.S. degrees in computer engineering from Cairo University, Cairo, Egypt, in 2012 and 2016, respectively, and the Ph.D. degree from Tennessee Tech University, USA, in 2019. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, North Carolina A&T University, USA. His research interests include machine learning, cryptography and network security, smart grid and AMI networks, and vehicular ad hoc networks.

AHMED SHERIF (Senior Member, IEEE) received the M.Sc. degree in computer science and engineering from the Egypt-Japan University of Science and Technology (E-JUST), in 2014, and the Ph.D. degree in electrical and computer engineering from Tennessee Tech University, Cookeville, TN, USA, in August 2017. He is currently an Assistant Professor with the School of Computing Sciences and Computer Engineering, The University of Southern Mississippi (USM), USA. He is the author of numerous papers published in major IEEE conferences and journals, such as IEEE International Conference on Communications (IEEE ICC), IEEE Vehicular Technology Conference (IEEE VTC), the IEEE Transactions on Dependable and Secure Computing, and the IEEE Internet of Things Journal. His research interests include cybersecurity, security and privacy-preserving schemes in autonomous vehicles (AVs), vehicular ad hoc networks (VANETs), the Internet of Things (IoT) applications, and smart grid advanced metering infrastructure (AMI) networks. He served as a Reviewer for several journals and conferences, such as the IEEE Transactions on Vehicular Technology, the IEEE Internet of Things Journal, and the journal of *Peer-to-Peer Networking and Applications*.

MOHAMED MAHMOUD (Senior Member, IEEE) received the Ph.D. degree from the University of Waterloo, in April 2011. From May 2011 to May 2012, he worked as a Postdoctoral Fellow with the Broadband Communications Research Group, University of Waterloo. From August 2012 to July 2013, he was a Visiting Scholar at the University of Waterloo and a Postdoctoral Fellow at Ryerson University. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, Tennessee Tech University, USA. He is the author of more than 100 papers published in IEEE conferences and journals. His research interests include security and privacy-preserving schemes for smart grid communication networks, e-health, smart transportation, blockchain, and machine learning. He served as a technical program committee member for several IEEE conferences. He received the NSERC-PDF Award. He won the Best Paper Award from IEEE International Conference on Communications (ICC 2009), Dresden, Germany, in 2009. He serves as an Associate Editor for the IEEE Internet of Things (IoT) Journal, and the journal *Peer-to-Peer Networking and Applications* journal (Springer). He served as a Reviewer for several journals and conferences, such as IEEE Transactions on Vehicular Technology, IEEE Transactions on Parallel and Distributed Systems, and the *Peer-to-Peer Networking and Applications* journal.

WALEED ALSMARY (Senior Member, IEEE) received the B.Sc. degree (Hons.) in computer engineering from Umm Al-Qura University, Saudi Arabia, in 2005, the M.A.Sc. degree in electrical and computer engineering from the University of Waterloo, Canada, in 2010, and the Ph.D. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2015. During his Ph.D. degree, he was a Visiting Research Scholar with the Network Research Laboratory, UCLA, in 2014. He was a Fulbright Visiting Scholar with the CSAIL Laboratory, MIT, from 2016 to 2017. He subsequently joined the College of Computer and Information Systems, Umm Al-Qura University, as an Assistant Professor of computer engineering, where he currently holds an Associate Professor position. His current research interests include machine learning-based and privacy-preserving smart systems. His ‘‘Mobility Impact on the IEEE 802.11p’’ article is among the most cited *Ad Hoc Networks* journal articles list, during 2016–2018. He is also an Associate Editor of *Array* journal.

MAAZEN ALSABAAN received the B.S. degree in electrical engineering from King Saud University (KSU), Saudi Arabia, in 2004, and the M.A.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Waterloo, Canada, in 2007 and 2013, respectively. He is currently an Associate Professor with the Department of Computer Engineering, KSU. From 2015 to 2018, he was the Chairperson of the Department. He serves as a consultant for different agencies and has been awarded many grants from KSU and King Abdulaziz City for Science and Technology (KACST). His current research interests include wireless communications and networking, surveillance systems, vehicular networks, green communications, intelligent transportation systems, and cybersecurity.