



**HAL**  
open science

# Privacy Preserving Group Membership Verification and Identification

Marzieh Gheisari, Teddy Furon, Laurent Amsaleg

► **To cite this version:**

Marzieh Gheisari, Teddy Furon, Laurent Amsaleg. Privacy Preserving Group Membership Verification and Identification. CVPR 2019 - IEEE Conference on Computer Vision and Pattern Recognition, Jun 2019, Long Beach, United States. pp.1-9. hal-02107442

**HAL Id: hal-02107442**

**<https://hal.science/hal-02107442>**

Submitted on 24 Apr 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Privacy Preserving Group Membership Verification and Identification

Marzieh Gheisari, Teddy Furon, Laurent Amsaleg  
Univ Rennes, Inria, CNRS, IRISA, France

{marzieh.gheisari-khorasgani, teddy.furon}@inria.fr, laurent.amsaleg@irisa.fr

## Abstract

When convoking privacy, group membership verification checks if a biometric trait corresponds to one member of a group without revealing the identity of that member. Similarly, group membership identification states which group the individual belongs to, without knowing his/her identity. A recent contribution provides privacy and security for group membership protocols through the joint use of two mechanisms: quantizing biometric templates into discrete embeddings, and aggregating several templates into one group representation.

This paper significantly improves that contribution because it jointly learns how to embed and aggregate instead of imposing fixed and hard coded rules. This is demonstrated by exposing the mathematical underpinnings of the learning stage before showing the improvements through an extensive series of experiments targeting face recognition. Overall, experiments show that learning yields an excellent trade-off between security / privacy and the verification / identification performances.

## 1. Introduction

The verification that an item, a device or an individual is a member of a group is a natural task which forms the basis of many applications granting or refusing access to sensitive resources (buildings, wifi, payment, conveyor units, ...). Group membership can be implemented through a two-phase process where an *identification* is first performed, revealing the identity of the individual under scrutiny, followed by a *verification* phase where it is checked whether or not the identified individual is indeed a member of the claimed group. That implementation breaks privacy: there is no *fundamental* reason to identify the individual before running the verification step. It is easier, but not truly needed. It is fundamental to distinguish the members of the group from the non-members, but it does not require to

distinguish members from one another.

The same comments hold for group *identification*. Such a system manages multiple groups, separating *e.g.* individuals according to the team they work in. The goal is then to identify the precise group a member belongs to, without proceeding first to the identification of the individual.

Privacy preserving protocols exist, for the most part in the area of signal processing in the encrypted domain [19, 4, 13]. These protocols, however, are costly, making them hard to use in practice.

In contrast, some other approaches propose privacy preserving mechanisms that are cheaper to use because they solely rely on pure signal processing techniques, excluding all encryption mechanisms. They tend to be less secure, of course, but they put in the way of malicious users so many obstacles that breaking privacy or counterfeiting identities becomes discouragingly complicated and almost impossible. At their core, these approaches use for example sketching or quantization techniques that prohibit reconstructing identities with sufficient precision [2, 3]. Other techniques embed real vectors into another representational space with the purpose of security and privacy [12, 11].

Recently, Gheisari *et al.* proposed a privacy preserving group membership verification protocol quantizing identities' biometric templates into discrete embeddings and aggregating multiple embeddings into a group representation [5]. That scheme has several desirable properties: It is pure signal processing and linear algebra, hence it is cheap to run; quantization and aggregation fully succeed to make reconstruction difficult and impede identification; it is demonstrated to allow trading-off the strength of its security against group verification error rates.

That work, however, is fully deterministic in the sense that it sticks to a set of hard coded rules that drive the way templates are embedded, how they are grouped and then aggregated into group representations. Although well justified and sound, these rules govern the behavior of two independent procedures, one for embedding, the other for aggregating. The main contribution of this paper shows that jointly considering the embedding and aggregation stages results in better performances, *i.e.* a better membership verification

without damaging security.

This paper is therefore revisiting the core mechanisms proposed by Gheisari *et al.* in [5], by *jointly* optimizing the embedding and aggregation stages. It is structured as follows. Section 2 defines the context of this work and surveys existing approaches. Then, Section 3 gives a formal problem statement and the details of proposed method are introduced in Section 4. Section 5 presents the experimental protocol and results.

## 2. Context and Related Work

### 2.1. Context

Without any loss of generality, group membership verification needs first to acquire *templates* of items (passive PUF), devices (active PUF), or individuals (biometric trait) and to enroll them into a data structure stored in a server. Then, at verification time, that data structure is queried by a client with a new template and the access is granted or refused. Security assesses that the data structure is adequately protected so that a honest but curious server cannot reconstruct the templates. Privacy requires that verification should proceed without disclosing the identity.

The nature of templates can vary from one application to the other. For example, the templates encode information related to fingerprints, iris, or faces of individuals, or to Physically Unclonable Functions (PUF) like speckle patterns captured from laser-illuminated transparent plastic.

It is worth highlighting two fundamental properties of the templates. The template used at verification time is a noisy version of the one acquired at enrollment time. Lighting conditions, blood pressure, aging, worn-outs, transient physical conditions are possible factors that might cause variations at acquisition time. The verification protocol must be able to absorb such variations and cope with the continuous nature of the templates. However, it is very unlikely that a noisy version of the template corresponding to one group member gets similar enough to the enrolled template of any other group member. The first property is therefore in relation with the continuous and distinguishable nature of the templates. The second property is in relation with the statistical independence of the enrolled templates.

The traditional operational scenario considers a server which runs the group membership verification. The server receives queries from clients. A client acquires a new template and then queries the server. Clients are trusted. The server is honest but curious: It may try to reconstruct the enrolled templates or spy on the queries. The design intends to prevent the server from reconstructing the private template from the system while correctly determining whether or not a user is a member of the claimed group (group verification) or identifying the group of membership (group identification).

## 2.2. Related Work

### 2.2.1 Cryptography

There exist cryptographic protocols setting a key management system to provide the members of a group with anonymous authentication [14]. There is no enrollment of biometric or PUF templates since membership is equivalent to holding one of the valid keys. As for biometry or PUF applications, our scenario is different from authentication, identification and secret binding. These applications secure the templates at the server and/or the client sides, but ultimately reveal the identity of the user/object.

Signal processing in the encrypted domain can provide a solution to group membership verification. At enrollment time, each template is quantized and protected with homomorphic encryption. The query is protected in the same manner at the verification stage. This allows to compute distances between the query and the templates in the encrypted domain [19]. There exist protocols for comparing encrypted results to a threshold [4, 13]. These encrypted comparisons are sent back to the clients which decrypt and check whether there is at least one positive. Security and privacy are as high as the security of the cryptographic primitives. Homomorphic encryption, however, is costly, both in terms of memory footprint (space) and CPU consumption (time).

### 2.2.2 Sketching techniques

Group membership is linked to the well-known Bloom filter that is used to test whether an element is a member of a set. A Bloom filter hashes and blends elements into one array of bits. When used in the context of privacy, it was demonstrated that a server using Bloom filters cannot infer any information on one specific entry [2]. However, a Bloom filter can not be used as is in our application, but it needs two adaptations. First, a Bloom Filter deals with discrete objects whereas we consider continuous high dimensional vectors, the templates. We first need to design a quantizer hashing the continuous templates into discrete objects. That quantizer must absorb the noise, *i.e.* the difference between the enrolled and the fresh template. Second, at verification time, the hash of the query cannot be sent in the clear for privacy reasons [3]. For instance, Beck and Kerschbaum protect the query with partially homomorphic encryption since there is no need to protect the filter at the server side [1].

### 2.2.3 Aggregating, Embedding

Aggregating signals into one representation is a very common mechanism in computer vision. Approaches like BoW (Bag of Words) [17], VLAD [8], Fisher vectors [10], aggregate some local descriptors extracted from one image into one global description. However, these approaches are de-

signed to facilitate the identification of similar elements in images and have no security or privacy capabilities.

Another recent aggregation method better fits with the security and privacy requirements that we need. In [7], Iscen *et al.* use the *group testing* paradigm to design a strategy for packing a random set of image descriptors into a unique high-dimensional vector. One salient property of that strategy is that the similarity between images can be determined by solely comparing these (few) aggregated vectors to the description of the query, without the need of the original (and numerous) raw image descriptors. This saves space (memory footprint of the database) and time (complexity at query time). These gains are the main motivation of [7].

As for the embedding, our scenario is similar to a privacy-preserving identification mechanism based on sparsifying transform [12, 11]. It obtains a sparse ternary embedding preserving locality information while ensuring privacy of the data users and security of the templates.

Indeed, paper [5] uses the aggregation methods designed in [7] combined with the sparsifying transform [12, 11]. Our design resorts to this latter mechanism for inheriting its privacy-preserving properties as well, yet we replace the former ad-hoc hard coded rules of [7] by a machine learning approach.

### 2.2.4 Face recognition

Aggregating templates into one group representation also exists in face recognition, but privacy and security are not a requirement in the design of this primitive. They are usually taken into account with an extra layer (see Sect. 2.2.1). In a common approach, multiple face captures of the same person are combined to gain robustness against poses, expression, and quality variations [21]. In our scenario, the group is composed of unique faces of different persons.

Paper [20] computes a single compact descriptor for the faces of celebrities appearing in the same picture. The query is a small set of face descriptors and their system returns photos where these celebrities appear altogether. This paper consists in jointly learning the face description and the aggregation mechanism. The authors report “*a minimal loss of discriminability up to two faces per image, and degrades slowly after that*”.

Our paper deals with the aggregation for a given on-the-shelf face descriptor. On one hand, our setup is easier because the query is a single face. On the other hand, our group typically comprises more than two faces, and each is possibly captured according to very different conditions.

## 3. Problem Formulation

This section details the way the group membership problem is formulated and also defines the metrics used to evaluate performances.

### 3.1. Notations

The set of individuals is denoted as  $[N] := \{1, \dots, N\}$ , partitioned into  $M$  groups:  $[N] = \bigcup_{g \in [M]} \mathcal{S}_g$ . The size of a group  $\mathcal{S}$  is denoted by  $|\mathcal{S}|$ . In the experiments of Sect. 5, all the groups share the same size denoted as  $m$ .

The templates to be enrolled are the vectors  $\{\mathbf{x}_1, \dots, \mathbf{x}_N\} \subset \mathbb{R}^d$ . They are stored column wise in  $d \times N$  matrix  $\mathbf{X}$  according to the partition over the groups:  $\mathbf{X} = [\mathbf{X}_1, \dots, \mathbf{X}_M]$ , where  $\mathbf{X}_g$  stores the templates belonging to  $\mathcal{S}_g, \forall g \in [M]$ . The output of the enrollment is a  $\ell \times M$  matrix  $\mathbf{R} = [\mathbf{r}_1, \dots, \mathbf{r}_M]$  composed of the representations of the  $M$  groups. It is imposed that the group representations are quantized and sparse:  $\mathbf{r}_g \in \mathcal{A}^\ell$  with  $\mathcal{A} := \{-1, 0, 1\}$  and  $\|\mathbf{r}_g\|_0 \leq S < \ell, \forall g \in [M]$ . Moreover, the analysis is restricted to the case where  $\ell \leq d$ .

The template for which the membership has to be verified is a query vector  $\mathbf{y} \in \mathbb{R}^d$  that is cast onto  $\mathcal{A}^\ell$  thanks to an embedding before being compared to the group representations.

### 3.2. Embedding

Function  $e : \mathbb{R}^d \rightarrow \mathcal{A}^\ell$  maps a vector to a sequence of  $\ell$  discrete symbols. We intentionally choose the sparsifying transform coding described in [12, 11] for its security and privacy good properties. It projects  $\mathbf{x} \in \mathbb{R}^d$  on the column vectors of  $\mathbf{W} \in \mathbb{R}^{d \times \ell}$ . The output alphabet  $\mathcal{A} = \{-1, 0, +1\}$  is imposed by quantizing the components of  $\mathbf{W}^T \mathbf{x}$ . The  $\ell - S$  components having the lowest amplitude are set to 0. The  $S$  remaining ones are quantized to +1 or -1 according to their sign.

$$\begin{aligned} e : \mathbb{R}^d &\rightarrow \mathcal{A}^\ell \\ \mathbf{x} &\mapsto e(\mathbf{x}) = \mathsf{T}_S(\mathbf{W}^T \mathbf{x}). \end{aligned} \quad (1)$$

### 3.3. Performances Metrics

Paper [5] defines metrics to assess (i) the ability of the protocol to correctly perform the verification task, and (ii) the security and privacy.

#### 3.3.1 Verification Performance

The first metrics is seeded by considering the following two hypotheses:

- $\mathcal{H}_1$ : The query is related to one of the vectors of group  $g$ . For instance, it is a noisy version of vector  $j$ ,  $\mathbf{y} = \mathbf{x}_j + \mathbf{n}$ , with  $\mathbf{n}$  to be a noise vector and  $j \in \mathcal{S}_g$ .
- $\mathcal{H}_0$ : The query is not related to any vector in the group.

The group membership test decides which hypothesis is deemed true by comparing  $e(\mathbf{y})$  to  $\mathbf{r}_g$ . This is done by first computing a score function  $c$  and thresholding its result:  $t := [c(e(\mathbf{y}), \mathbf{r}_g) > \tau]$ .

The probabilities of false negative,  $p_{fn}(\tau) := \mathbb{P}(t = 0|\mathcal{H}_1)$ , and false positive,  $p_{fp}(\tau) := \mathbb{P}(t = 1|\mathcal{H}_0)$  are summarized by the AUC (Area Under the ROC Curve). Paper [5] also considers  $p_{fn}(\tau)$  for  $\tau$  s.t.  $p_{fp}(\tau) = \epsilon$ , a required false positive level, 0.05 for example.

### 3.3.2 Security and Privacy Performance

A curious server can reconstruct the query template  $\mathbf{y}$  from its embedding:  $\hat{\mathbf{y}} := \text{rec}(e(\mathbf{y}))$ . This endangers privacy of the querying user. The mean squared error assesses how accurate is this reconstruction:

$$\text{MSE}_P = d^{-1} \mathbb{E}(\|\mathbf{Y} - \text{rec}(e(\mathbf{Y}))\|^2), \quad (2)$$

assuming that  $\mathbf{Y}$  is a white gaussian vector in  $\mathbb{R}^d$ .

As for the security of the enrolled templates, a curious server can only reconstruct a single vector  $\hat{\mathbf{x}}$  from the aggregated representation, and this vector serves as an estimation of any template in the group:

$$\text{MSE}_S = (dN)^{-1} \sum_{g=1}^M \sum_{i=1}^{|\mathcal{S}_g|} \mathbb{E}(\|\mathbf{x}_i - \hat{\mathbf{x}}_g\|^2). \quad (3)$$

## 4. Proposed Method

### 4.1. Variants of the Protocol

Our group membership protocol is based on embedding and aggregation functions. Whereas the embedding function  $e$  is fixed, we can have two constructions: The aggregation of embeddings or the embedding of the aggregation.

**EoA** aggregates raw templates into one vector of  $\mathbb{R}^d$ , and then embeds this vector. Then, the group representative vector  $\mathbf{r}_g$  is computed as:

$$\mathbf{r}_g = e(\mathbf{a}_{\text{EoA}}(\mathbf{X}_g)), \quad (4)$$

where  $\mathbf{X}_g$  is the  $d \times |\mathcal{S}_g|$  matrix storing the templates of the  $g$ -th group.

**AoE** first embeds each template according to (1) before aggregating them into  $\mathbf{r}_g$ :

$$\mathbf{r}_g = \mathbf{a}_{\text{AoE}}(\{e(\mathbf{x}_i)\}_{i \in \mathcal{S}_g}). \quad (5)$$

This work aims at learning the aggregated vectors and the embeddings *jointly*. For both construction, this is done by minimizing an objective function summing a cost for embedding  $C^E$  and a cost for aggregating  $C^A$ .

For AoE (first embed, then aggregate), denote  $\mathbf{E} \in \mathcal{A}^{\ell \times N}$  the matrix storing the embeddings of the enrolled templates. Like for  $\mathbf{X}$ , we write  $\mathbf{E} := [\mathbf{E}_1, \dots, \mathbf{E}_M]$  with  $\mathbf{E}_g$  the matrix gathering the embeddings of the templates of

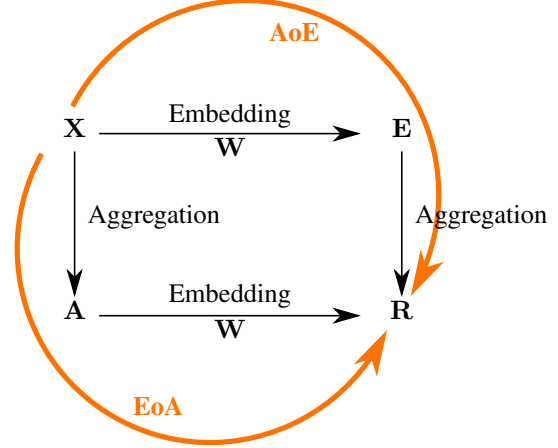


Figure 1: Overview

group  $\mathcal{S}_g$ . For EoA (first aggregate, then embed), denote  $\mathbf{A} := [\mathbf{a}_1, \dots, \mathbf{a}_M] \in \mathbb{R}^{d \times M}$  the matrix gathering the aggregations of the templates enrolled in a group.

Matrices  $\mathbf{E}$  and  $\mathbf{A}$  will be defined through optimization problems detailed below. For the embedding, function  $e$  is still prototyped according to Sect. 3.2. Papers [12, 11] show that privacy and security stem from the sparsifying transform. Only its matrix  $\mathbf{W}$  is learned.

For EoA, Fig. 1 shows that it starts from  $\mathbf{X}$  to create  $\mathbf{A}$  before outputting  $\mathbf{R}$  using matrix  $\mathbf{W}$ . This defines the optimization problem:

$$\min_{\mathbf{A}, \mathbf{W}, \mathbf{R}} \gamma C_{\text{EoA}}^A(\mathbf{X}, \mathbf{A}) + C_{\text{EoA}}^E(\mathbf{A}, \mathbf{R}, \mathbf{W}), \quad (6)$$

where  $\gamma$  is the penalty parameter,  $\mathbf{A} \in \mathbb{R}^{d \times M}$ ,  $\mathbf{W} \in \mathbb{R}^{d \times \ell}$ , and  $\mathbf{R} \in \mathcal{A}^{\ell \times M}$ .

On the other hand, Fig. 1 shows that for AoE, we start from  $\mathbf{X}$  to create  $\mathbf{E}$  using matrix  $\mathbf{W}$  before outputting the group representations  $\mathbf{R}$ . This defines the optimization problem:

$$\min_{\mathbf{E}, \mathbf{W}, \mathbf{R}} C_{\text{AoE}}^E(\mathbf{X}, \mathbf{W}, \mathbf{E}) + \xi C_{\text{AoE}}^A(\mathbf{E}, \mathbf{R}), \quad (7)$$

with  $\mathbf{E} \in \mathcal{A}^{\ell \times M}$ ,  $\mathbf{W} \in \mathbb{R}^{d \times \ell}$ , and  $\mathbf{R} \in \mathcal{A}^{\ell \times M}$ .

Under both constructions, the optimization is joint because the embedding and the aggregating costs share a common variable ( $\mathbf{A}$  or  $\mathbf{E}$ ). What follows define the costs and solve the optimization problems.

### 4.2. AoE: Aggregation of Embeddings

This scheme first embeds and then aggregates by solving (7). The cost for embedding is defined as

$$C_{\text{AoE}}^E(\mathbf{X}, \mathbf{W}, \mathbf{E}) := \sum_{i=1}^N \|\mathbf{e}_i - \mathbf{W}^\top \mathbf{x}_i\|_2^2, \quad (8)$$

$$= \|\mathbf{E} - \mathbf{W}^\top \mathbf{X}\|_F^2. \quad (9)$$



This term represents the fidelity penalty of an embedding  $\mathbf{e}$  w.r.t. a template  $\mathbf{x}$  in the transformed domain.

For each group of embedded templates, the aggregated vector should satisfy some properties as well:

- For each group the overall distance between group members and the aggregated vector is minimized.
- Aggregated vector should be represented as a sparse ternary code.

The cost of aggregation is then defined as

$$C_{\text{AoE}}^A(\mathbf{E}, \mathbf{R}) := \sum_{g=1}^M \|\mathbf{E}_g - \mathbf{r}_g \mathbf{1}_{|S_g|}^\top\|_F^2. \quad (10)$$

We add the following constraints:

$$\mathbf{W}^\top \mathbf{W} = \mathbf{I}_\ell, \quad (11)$$

$$\|\mathbf{e}_i\|_0 \leq S, \quad \forall i \in [N] \quad (12)$$

$$\|\mathbf{r}_g\|_0 \leq S \quad \forall g \in [M]. \quad (13)$$

The constraint makes sure that the representative  $\mathbf{r}$  is sparse, ternary, and diverse.

We propose to optimize problem (7) iteratively by alternating updates of one parameter while fixing the remaining ones. Each step minimizes the total cost function lower bounded by 0, insuring a convergence to a local minimum.

**W-Step.** We fix  $\mathbf{E}$  and  $\mathbf{R}$  and update  $\mathbf{W}$  by solving:

$$\begin{aligned} \min_{\mathbf{W}} \quad & \|\mathbf{E} - \mathbf{W}^\top \mathbf{X}\|_F^2 \\ \text{s.t.} \quad & \mathbf{W}^\top \mathbf{W} = \mathbf{I}_\ell \end{aligned} \quad (14)$$

This problem is a least square Procruste problem with orthogonality constraint. By setting  $\mathbf{S} := \mathbf{X}\mathbf{E}^\top$ , [15] shows that  $\mathbf{W} = \mathbf{U}\mathbf{V}^\top$ , where  $\mathbf{U}$  contains the eigenvectors corresponding to the  $\ell$  ( $\ell < d$ ) largest eigenvalues of  $\mathbf{S}\mathbf{S}^\top$  and  $\mathbf{V}$  contains the eigenvectors of  $\mathbf{S}^\top \mathbf{S}$ .

**E-Step.**  $\mathbf{W}$  and  $\mathbf{R}$  being fixed, we can solve the problem for each  $\mathbf{E}_g$  independently:  $\forall g \in [M]$ ,

$$\begin{aligned} \min_{\mathbf{E}_g} \quad & \|\mathbf{E}_g - \mathbf{W}^\top \mathbf{X}_g\|_F^2 + \xi \|\mathbf{E}_g - \mathbf{r}_g \mathbf{1}_{|S_g|}^\top\|_F^2 \\ \text{s.t.} \quad & \mathbf{E}_g \in \mathcal{A}^{\ell \times |S_g|}, \|\mathbf{e}_i\|_0 \leq S, \quad \forall i \in S_g. \end{aligned} \quad (15)$$

According to [12], we first find the solution without considering the constraints and then apply ternarization function  $\mathsf{T}_S$  (1) to obtain sparse codes. Therefore  $\mathbf{E}_g$  is found as:

$$\mathbf{E}_g = \mathsf{T}_S(\mathbf{W}^\top \mathbf{X}_g + \xi \mathbf{r}_g \mathbf{1}_{|S_g|}^\top). \quad (16)$$

**R-Step.** Like for the **E**-step, updating each group representation  $\mathbf{r}_g$  is done independently, while fixing  $\mathbf{W}$  and  $\mathbf{E}$ :

$$\begin{aligned} \min_{\mathbf{r}_g} \quad & \|\mathbf{E}_g - \mathbf{r}_g \mathbf{1}_{|S_g|}^\top\|_F^2 \\ \text{s.t.} \quad & \mathbf{r}_g \in \mathcal{A}^\ell, \quad \|\mathbf{r}_g\|_0 \leq S. \end{aligned} \quad (17)$$

Then the representative of  $g$ -th group is obtained as:

$$\mathbf{r}_g = \mathsf{T}_S(\mathbf{E}_g \mathbf{1}_{|S_g|}). \quad (18)$$

### 4.3. EoA: Embedding of Aggregation

We now consider the construction of (6) that first aggregates and then embeds. The cost of the aggregation is defined as:

$$C_{\text{EoA}}^A(\mathbf{X}, \mathbf{A}) := \sum_{g=1}^M \|\mathbf{X}_g^\top \mathbf{a}_g - \mathbf{1}_{|S_g|}\|_2^2 + \eta \|\mathbf{a}_g\|_2^2. \quad (19)$$

Minimizing this cost amounts to equalize the similarity between each members of the group and the aggregated vector  $\mathbf{a}$ . The cost for embedding is defined as previously:

$$C_{\text{EoA}}^E(\mathbf{A}, \mathbf{R}, \mathbf{W}) := \|\mathbf{R} - \mathbf{W}^\top \mathbf{A}\|_F^2. \quad (20)$$

As for the constraints:

$$\mathbf{W}^\top \mathbf{W} = \mathbf{I}_\ell, \quad (21)$$

$$\mathbf{r}_g \in \mathcal{A}^\ell, \|\mathbf{r}_g\|_0 \leq S, \quad \forall g \in [M]. \quad (22)$$

The optimization problem (6) with these costs and constraints is solved by iterating the following steps.

**W-Step.** Like for (14), updating  $\mathbf{W}$  while  $\mathbf{R}$ ,  $\mathbf{A}$  are fixed is a Procruste problem under orthogonality constraint:

$$\begin{aligned} \min_{\mathbf{W}} \quad & \|\mathbf{R} - \mathbf{W}^\top \mathbf{A}\|_F^2, \\ \text{s.t.} \quad & \mathbf{W}^\top \mathbf{W} = \mathbf{I}_\ell. \end{aligned} \quad (23)$$

Similar to (14), we define  $\mathbf{S} := \mathbf{A}\mathbf{R}^\top$ . The solution is found as  $\mathbf{W} = \mathbf{U}\mathbf{V}^\top$ , where  $\mathbf{U}$  contains the eigenvectors corresponding to the  $\ell$  largest eigenvalues of  $\mathbf{S}\mathbf{S}^\top$  and  $\mathbf{V}$  the eigenvectors of  $\mathbf{S}^\top \mathbf{S}$ .

**A-Step.** When fixing  $\mathbf{W}$  and  $\mathbf{R}$ , the aggregated vector for each group  $g \in [M]$  is found independently by minimizing:

$$\min_{\mathbf{a}_g} \|\mathbf{r}_g - \mathbf{W}^\top \mathbf{a}_g\|_2^2 + \gamma (\|\mathbf{X}_g^\top \mathbf{a}_g - \mathbf{1}_{|S_g|}\|_2^2 + \eta \|\mathbf{a}_g\|_2^2), \quad (24)$$

whose solution is

$$\mathbf{a}_g = (\mathbf{W}\mathbf{W}^\top + \gamma(\mathbf{X}_g \mathbf{X}_g^\top + \eta \mathbf{I}_d))^{-1} (\mathbf{W}\mathbf{r}_g + \gamma \mathbf{X}_g \mathbf{1}_{|S_g|}).$$

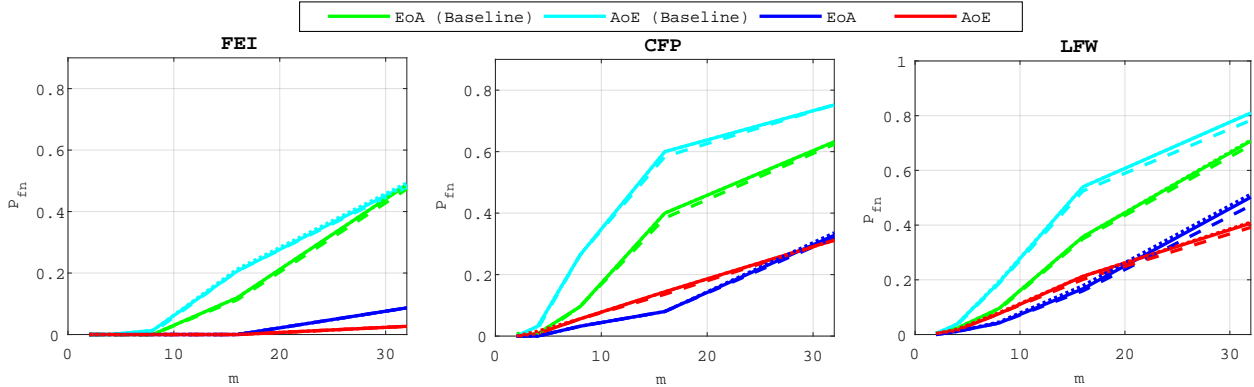


Figure 2: Performances comparison of our schemes vs. baseline for varying group size  $m$ .  $P_{fn}$  at  $P_{fp} = 0.05$  for group verification (solid), the first step of group identification (dashed), and  $P_\epsilon$  for the second step of group identification (dotted).

**R-Step.** Projection matrix  $\mathbf{W}$  and the group aggregations  $\mathbf{A}$  are fixed. The group representatives are obtained by applying sparse ternarization function on the projected aggregated vectors:  $\mathbf{R} = \mathbf{T}_S(\mathbf{W}^\top \mathbf{A})$ .

## 5. Experiments

We fully implemented the group membership protocol that is described in [5]. Experimenting with this implementation gives the baseline performances. Note that the experimental part of [5] only deals with synthetic data. Our implementation could reproduce these results. However, the sequel presents comparisons on real data.

### 5.1. Experimental Setup

We evaluate the performances of our scheme with face recognition. Face images are coming from LFW [6], CFP [16] and FEI [18] databases. Face descriptors are obtained from a pre-trained network based on VGG-Face architecture [9]. The output vector of the penultimate layer (*i.e.* before the final classifier layer) is PCA reduced to a lower dimension ( $d = 1,024$  for LFW and CFP,  $d = 256$  for FEI database), and then  $L_2$ -normalized. The result is the template  $\mathbf{x} \in \mathbb{R}^d$ . The values of  $\ell$ ,  $S$ ,  $\xi$ ,  $\gamma$  and  $\eta$  are set empirically as  $0.9d$ ,  $0.7\ell$ ,  $1$ ,  $10^4$  and  $1$  respectively. Also, not all individuals from these databases are enrolled.

**LFW.** Labeled Faces in the Wild contains 13,233 images of faces collected from the web. We used cropped LFW images. The enrollment set consists of  $N = 1,680$  individuals with at least two images in the LFW database. One random template of each individual is enrolled in the system, playing the role of  $\mathbf{x}_i$ . The other templates are used for queries. These are partitioned into two subsets: templates that are correlated with  $\mathbf{x}_i$  with a similarity bigger than 0.95 form the “easy queries” set ; the remaining templates with a similarity bigger than 0.9 form the “hard queries” set. The re-

maining individuals not enrolled in the system ( $5,749 - N$ ) play the role of impostors (hypothesis  $\mathcal{H}_0$ ).

**CFP.** The Celebrities in Frontal-Profile (CFP) database is composed of 500 subjects with 10 frontal and 4 profile images for each subject in a wild setting. We only use the frontal images. The impostor set is a random selection of 100 individuals. One random template of the remaining individuals is enrolled in the system. Like the setting described for LFW, we have two subsets of queries.

**FEI.** We use frontal and pre-aligned images of the Brazilian FEI database. There are 200 subjects with two frontal images (one with a neutral expression and the other with a smiling facial expression). The database is created by random sampling 150 individuals. For each identity, one random image is enrolled while the other is used as query. The remaining individuals are considered as impostors.

At the enrollment phase, all groups have exactly the same number of members:  $|\mathcal{S}_g| = m, \forall g \in [M]$ . Individuals are randomly assigned to a group.

The performances of the system are gauged with error probabilities evaluated by Monte Carlo estimator over the testing set. Since  $N$  is not so large, the confidence interval at 95% is  $\approx \pm N^{-1/2} = \pm 2.5\%$ , which prevents us from estimating small probabilities. Therefore, *we put our system under stress by selecting a hard setup*. First, note that LFW and CFP are difficult datasets due to the ‘in the wild’ variations (poses, illuminations, expressions and occlusions). They do not reflect the application of accessing a building (as mentioned in the Introduction) where the capture environment is more under control and the individuals collaborate. Second, not only the dimension of the templates have been reduced but also the length of the embeddings and the group representation ( $\ell = 0.9d$ ) with a sparsity of  $S = 0.7\ell$  (unless stated otherwise). Probabilities of errors are then big

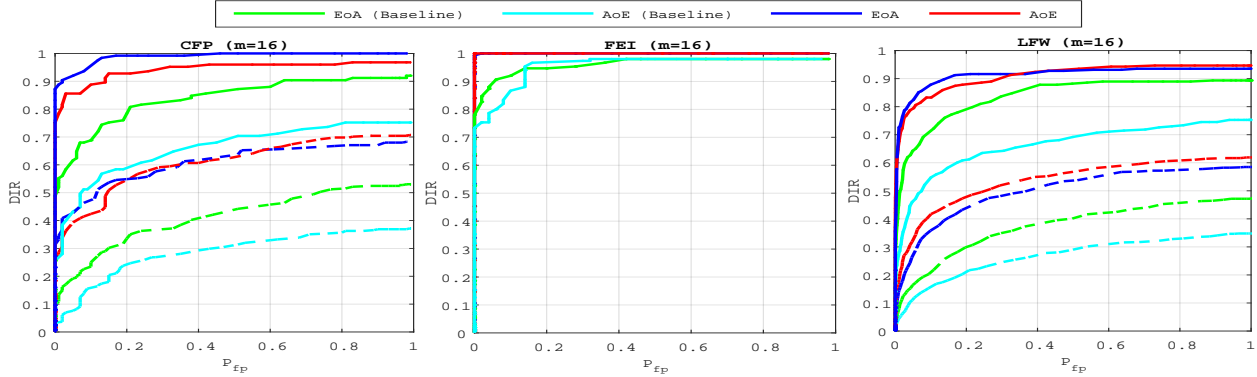


Figure 3: The Detection and Identification Rate ( $DIR$ ) vs.  $P_{fp}$  for group identification. Performances for hard queries are plotted in dashed lines.

but measurable with accuracy. We believe that this protocol makes sense to benchmark approaches.

Two applications scenarios are investigated: group verification and group identification.

**Group verification.** A user claims she/he belongs to group  $g$ . This claim is true under hypothesis  $\mathcal{H}_1$  and false under hypothesis  $\mathcal{H}_0$  (*i.e.* the user is an impostor). Her/his template  $\mathbf{y}$  is embedded, and  $(h(\mathbf{y}), g)$  is sent to the system, which compares  $h(\mathbf{y})$  to the group representation  $\mathbf{r}_g$ . The system accepts ( $t = 1$ ) or rejects ( $t = 0$ ) the claim. This is a two hypothesis test with two probabilities of errors:  $P_{fp} := \mathbb{P}(t = 1|\mathcal{H}_0)$  is the false positive rate and  $P_{fn} := \mathbb{P}(t = 0|\mathcal{H}_1)$  is the false negative rate. The figure of merit is  $P_{fn}$  when  $P_{fp} = 0.05$ .

**Group identification.** The scenario is an open set identification where the querying user is either enrolled or an impostor. The system has two steps. First, it decides whether or not this user is enrolled. This is verification as above, except that the group is unknown: The system computes  $\delta_j = \|h(\mathbf{y}) - \mathbf{r}_j\|, \forall j \in [M]$ . The system accepts ( $t = 1$ ) if the minimum of these  $M$  distances is below a given threshold  $\tau$ . The figure of merit is  $P_{fn}$  when  $P_{fp} = 0.05$ .

When  $t = 1$ , the system proceeds to the second step. The estimated group is given by  $\hat{g} = \arg \min_{j \in [M]} \delta_j$ . The figure of merit for this second step is  $P_\epsilon := \mathbb{P}(\hat{g} \neq g)$  or the Detection and Identification Rate  $DIR := (1 - P_\epsilon)(1 - P_{fn})$ .

## 5.2. Exp. #1: Comparison to the baseline

Figure 2 shows that our method brings improvement compared to the baseline, since the AoE and the EoA plots are way below the ones corresponding to the baseline. The high probabilities of false negatives for the baseline are caused by the great losses in information: AoE (Baseline) loses information from each template it embeds before the

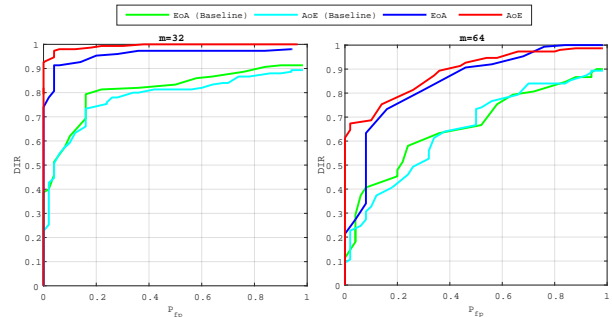


Figure 4: The Detection and Identification Rate ( $DIR$ ) vs.  $P_{fp}$  for group identification on FEI.

aggregation—the accumulated losses are therefore great; EoA (Baseline) has better performances since plain templates are first aggregated before running the embedding step which causes less information loss.

Our method does not suffer that much from this information loss: EoA and AoE have roughly similar performances, with much more acceptable  $P_{fn}$  values.

## 5.3. Exp. #2: Detection and Identification Rate

Figure 3 compares the  $DIR$  performances for group identification with  $m = 16$ . Our schemes have results close to perfection on the FEI dataset. Easy queries are correctly handled on CFP but not on the LFW dataset at this size of group. Hard queries are more difficult to cope with. This is explained by the poor correlations they have with their corresponding  $\mathbf{x}_i$ . That poor correlation, already existing on the original templates, before any embedding or aggregation, can only lower the performances of any membership identification scheme.

Figure 4 shows the impact of the size of group on  $DIR$ . Packing more templates into one group representation is detrimental even if the queries are well correlated with



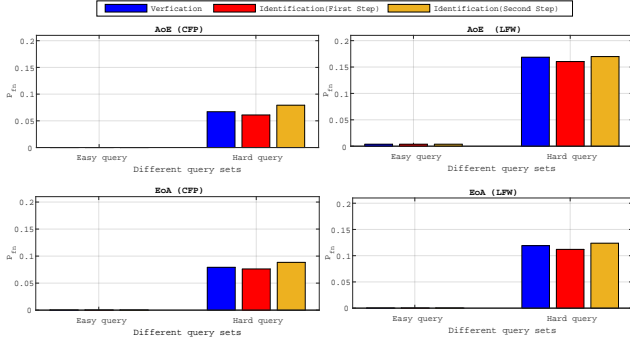


Figure 5: The impact of the similarity of the query with the enrolled template on group verification and identification.

their corresponding enrolled template. This suggests to split large groups into subgroups of size lower or equal to  $m = 32$ . This restricts privacy to  $m$ -anonymity as the server is now able to identify the subgroup a query belongs to.

#### 5.4. Exp. #3: Easy vs. Hard Queries

Figure 5 gives an additional perspective on the phenomenon highlighted above, that is, the genuine similarity between the query and the enrolled template is a key factor. Easy queries are very well handled whereas hard queries are more problematic. Put differently, the proposed method do not severely degrade the recognition power of the descriptors obtained through the VGG16 network. Descriptors poorly correlated already at the image level can only cause poor performance once embedded and aggregated. This is also shown in Fig. 7 which displays some enrolled and querying faces of the ‘in the wild’ datasets LFW and CFP. All the failed identification examples show a change of lighting, pose or expression, and / or occlusion. Yet, such changes do not automatically give a failure.

#### 5.5. Exp. #4: Security and Privacy

As for the security and privacy, the quantities (2) and (3) were measured as empirical average over the dataset.

Knowing that the query has unit norm, the reconstruction mechanism yields a unit vector as follows:  $\hat{\mathbf{y}} = \mathbf{W}\mathbf{e}(\mathbf{y})/\|\mathbf{W}\mathbf{e}(\mathbf{y})\|$ . The quality of the reconstruction mainly depends on the sparsity factor  $S$ . When  $S$  is small, the template is reconstructed with few columns of  $\mathbf{W}$ . When  $S$  is big, more columns are used but the amplitude modulating each column is coarsely reconstructed. There might be two values of  $S$ , one small, one large, providing the same reconstruction MSE. However, these two values do not yield the same performances as shown in Fig. 6.

Reconstructing enrolled templates is even more difficult due to the aggregation (see (3)). Fig. 6 shows that our method has decreased the security a little, but overall the

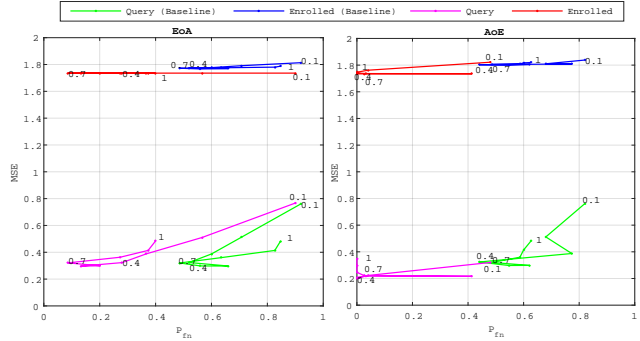


Figure 6: The impact of sparsity factor  $S$  on the trade-off between security and performances, on FEI with  $m = 32$  and  $S/d \in (0.1, 1)$ .



Figure 7: Examples of group identification on CFP(left) and LFW(right). Blue frames indicate enrolled samples, green / red frames successful / failed queries, respectively.

trade-off between security and performances is more interesting especially for AoE.

## 6. Conclusion

This paper proposes a framework for group membership verification and identification by jointly learning the embedding and the aggregation. Yet, this learning was not completely free. Some guidances were still imposed, especially the prototyping of the embedding based on a sparse ternary quantization. This is mainly for inheriting security and privacy properties of this lossy information processing [12, 11]. It is not clear whether an alternative approach does exist.

Our future work looks at increasing the length of the group representation in order to pack more templates into a group. Yet, this raises scalability issues appealing for faster but approximative learning.

## References

- [1] M. Beck and F. Kerschbaum. Approximate two-party privacy-preserving string matching with linear complexity. In *Proceedings of the IEEE International Congress on Big Data*, 2013. 2
- [2] G. Bianchi, L. Bracciale, and P. Loreti. "better than nothing" privacy with bloom filters: To what extent? In *Proceedings of the International Conference on Privacy in Statistical Databases*, 2012. 1, 2
- [3] D. Boneh, E. Kushilevitz, R. Ostrovsky, and W. E. Skeith. Public key encryption that allows pir queries. In *Proceedings of the International Cryptology Conference, Advances in Cryptology*, 2007. 1, 2
- [4] Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft. Privacy-preserving face recognition. In *Proceedings of the International Symposium on Privacy Enhancing Technologies*, 2009. 1, 2
- [5] M. Gheisari, T. Furon, L. Amsaleg, B. Razeghi, and S. Voloshynovskiy. Aggregation and embedding for group membership verification. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019. 1, 2, 3, 4, 6
- [6] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *Workshop on faces in 'Real-Life' Images: detection, alignment, and recognition*, 2008. 6
- [7] A. Iscen, T. Furon, V. Gripon, M. Rabbat, and H. Jégou. Memory vectors for similarity search in high-dimensional spaces. *IEEE Transactions on Big Data*, 2017. 3
- [8] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(9):1704–1716, 2012. 2
- [9] O. M. Parkhi, A. Vedaldi, A. Zisserman, et al. Deep face recognition. In *Proceedings of the British Machine Vision Conference*, 2015. 6
- [10] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, 2007. 2
- [11] B. Razeghi and S. Voloshynovskiy. Privacy-preserving outsourced media search using secure sparse ternary codes. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018. 1, 3, 4, 8
- [12] B. Razeghi, S. Voloshynovskiy, D. Kostadinov, and O. Taran. Privacy preserving identification using sparse approximation with ambiguization. In *Proceedings of the IEEE International Workshop on Information Forensics and Security*, 2017. 1, 3, 4, 5, 8
- [13] A.-R. Sadeghi, T. Schneider, and I. Wehrenberg. Efficient privacy-preserving face recognition. In *Proceedings of the International Conference on Information, Security and Cryptology*, 2010. 1, 2
- [14] S. Schechter, T. Parnell, and A. Hartemink. Anonymous authentication of membership in dynamic groups. In *Proceedings of the International Conference on Financial Cryptography*, 1999. 2
- [15] P. H. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 1966. 5
- [16] S. Sengupta, J.-C. Chen, C. Castillo, V. M. Patel, R. Chellappa, and D. W. Jacobs. Frontal to profile face verification in the wild. In *Proceeding of the IEEE Winter Conference on Applications of Computer Vision*, 2016. 6
- [17] J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, 2003. 2
- [18] C. E. Thomaz and G. A. Giraldi. A new ranking method for principal components analysis and its application to face image analysis. *Image and Vision Computing*, 28(6):902–913, 2010. 6
- [19] J. R. Troncoso-Pastoriza, D. González-Jiménez, and F. Pérez-González. Fully private noninteractive face verification. *IEEE Transactions on Information Forensics and Security*, 8(7):1101–1114, 2013. 1, 2
- [20] Y. Zhong, R. Arandjelović, and A. Zisserman. Compact deep aggregation for set retrieval. In *Proceedings of the European Conference on Computer Vision Workshops*, 2018. 3
- [21] Y. Zhong, R. Arandjelović, and A. Zisserman. Ghostvlad for set-based face recognition. In *Proceedings of the Asian Conference on Computer Vision*, 2018. 3