De-Identifying Facial Images Using Singular Value Decomposition and Projections on Hyperspheres

Panteleimon Chriskos
Olga Zoidi
Anastasios Tefas
Ioannis Pitas

AIIA Lab
Aristotle University of Thessaloniki
Greece
Introduction

• Two novel methods are analyzed that successfully hinder automatic face recognition.

• The methods:
  – maintain enough visual data so that viewers can identify the person or persons in a scene.
  – maintain the image quality so that the end product can be considered acceptable for everyday use.
  – de-identify faces using
    a) the Singular Value Decomposition Method (SVD)
    b) and utilizing projections on hyperspheres.
Singular Value Decomposition

- Singular value decomposition (SVD) is a matrix factorization method.

- It approximates a matrix $A \in \mathbb{R}^{n \times p}$ with the product of three matrices:

  $U \in \mathbb{R}^{n \times n}$  \quad $S \in \mathbb{R}^{n \times p}$  \quad $V \in \mathbb{R}^{p \times p}$

- These matrices must abide to the following conditions
  - matrix $S$ must be diagonal and
  - matrices $U$ and $V$ must be orthogonal
Singular Value Decomposition

• Generally the SVD theorem states that any matrix

\[ A \in \mathbb{R}^{n \times p} \]

can be written as

\[ A = U \times S \times V^T \]

• SVD is the workhorse of this person de-identification method. This method will be referenced as SVD-DID.
The proposed person de-identification method utilizes SVD.

The goal is to manipulate facial images in order to reduce facial identification by software agents.

This method alters the values in matrices $S$, $U$ and $V$ produced by the factorization of the input matrix $A$ (in our case a facial image).
Person De-identification based on SVD

In order to reduce the correct identification rate, the method follows the steps mentioned below:

- **Step 1: SVD Coefficient Zeroing (SVD-CZ)**
  The coefficients (singular values) of matrix $S$ with the largest values are reduced to zero.

- **Step 2: SVD Coefficient Averaging (SVD-CA)**
  The values of matrices $U$ and $V$ are blurred using an averaging filter.

- **Step 3: SVD Modified Sobel Filtering (SVD-MSF)**
  Matrices $U$ and $V$ are sharpened using a modified Sobel filter.
Step 1: SVD Coefficient Zeroing (SVD-CZ)

- Matrix $S$ is transformed into matrix $S_{CZ}$ by setting the first $N$ singular values to zero (0).

- The idea is to remove the most discriminative visual information in the image which lies in the coefficients (singular values) with the largest values.

- By removing the first coefficients, we actually are removing the majority of information that a software agent would use to successfully identify a subject.
Step 1: SVD Coefficient Zeroing (SVD-CZ)

- Applying this first step, the new image can be calculated through the following formula:

\[ A = U \times S_{\text{CZ}} \times V^T \]

- Resulting image (first frame of first individual from the first recording)

- Output image for \( N=1 \).

- Results:
  - Darkening
  - Visual Artifacts
Step 2: SVD Coefficient Averaging (SVD-CA)

- Matrices $\mathbf{U}$ and $\mathbf{V}$ are transformed into $\mathbf{U}_{\text{averaged}}$ and $\mathbf{V}_{\text{averaged}}$ by applying a circular averaging filter of radius $r$ which is a $2r + 1 \times 2r + 1$ filter containing zeroes and the value $1/n^2$ where $n$ is the number of cells that are contained in the circle of the filter.

- The idea is to blend the eigenvectors in the initial matrices in order to hinder classifiers trained with clean versions of facial images from identifying a certain individual.
Step 2: SVD Coefficient Averaging (SVD-CA)

• Matrices $U_{CA}$ and $V_{CA}$ are computed as follows

$$
U_{CA} = \frac{\alpha \cdot U_{\text{averaged}} + U}{1 + \alpha} \\
V_{CA} = \frac{\alpha \cdot V_{\text{averaged}} + V}{1 + \alpha}
$$

where $\alpha$ is a parameter preferably in the range $[0.5, 1.0]$

• Output image for
  - $V_{\text{averaged}}$, $U_{\text{averaged}}$
  - $r = 10$, $\alpha = 0.5$

• Results:
  - Smoothing
  - Brightening of certain areas
Step 3: SVD Modified Sobel Filtering (SVD-MSF)

- Matrices $U_{CA}$ and $V_{CA}$ are transformed into $V_{MSF}$ and $U_{MSF}$ by applying a modified Sobel filter $G$

$$G = \begin{bmatrix}
    d & 2d & d \\
    0 & 0 & 0 \\
    -d & -2d & -d
\end{bmatrix}, d = [0.2, 0.8]$$

- The idea is to blend the eigenvectors even further while removing part of the blurring introduced by the averaging filter in the previous step.
Step 3: SVD Modified Sobel Filtering (SVD-MSF)

• Matrices $U_{\text{final}}$ and $V_{\text{final}}$ are computed as follows

$$U_{\text{final}} = \frac{\alpha \ast U_{\text{MSF}} + U}{1 + \alpha} \quad V_{\text{final}} = \frac{\alpha \ast V_{\text{MSF}} + V}{1 + \alpha}$$

where $\alpha$ is a parameter preferably in the range $[0.5, 1.0]$.

• Output image for
  - $d = 0.2, \alpha = 0.5$
  - $d = 1.0, \alpha = 0.5$

• Results:
  - Prominent edges
  - Visual artifacts
Putting it all together SVD-DID

- The output image $P$ is computed as

$$P = U_{\text{final}} \times S_{\text{CZ}} \times V_{\text{final}}^T$$

- To counterbalance the darkening effect imposed by this method, luminosity is added to the output image.

- Output image for

$N = 1, r = 10, d = 0.5$

- $\alpha = 0.5$
- $\alpha = 0.8$
De-identification based on Projections

• The proposed face de-identification method utilizes projections on hyperspheres.

• The goal is to manipulate facial images in order to reduce facial identification by software agents.

• Two methods were developed:
  - Projection De-Identification on Origin (PDID-O)
  - Projection De-Identification on Mean Image (PDID-M)
Projections on Hyperspheres

• A hypersphere $S^{n-1}$ centered at the origin is defined as:

$$S^{n-1} = \left\{ x \in \mathbb{R}^n : \|x\| = R \right\}$$

where $x$ is a point in the $n$-dimensional space, $\|x\|$ is the Euclidean norm, and $R$ is the radius of the hypersphere.

• The projection of a point $x \in \mathbb{R}^n$ onto $S^{n-1}$ is given by:

$$P_{S^{n-1}} (x) = \frac{R}{\|x\|} x$$

with the hypersphere centered at the origin.
The de-identified image $I_{DID}$ is defined as:

$$I_{DID} = \frac{1}{2} \left( \frac{R}{\|I\|} I + \bar{I} \right)$$

where $I$ is the original facial image, $R$ is the hypersphere radius and $\bar{I}$ is the average facial image of the dataset calculated as:

$$\bar{I} = \frac{1}{N_{im}} \sum_{i=1}^{N_{im}} I_i$$

where $N_{im}$ is the total number of facial images.
The de-identified image $I_{DID}$ is defined as:

$$I_{DID} = \left( \frac{R(I - \overline{I})}{\|I - \overline{I}\|} + \overline{I} \right)$$

where $\frac{R(I - \overline{I})}{\|I - \overline{I}\|}$ is the projection of the original facial image onto the hypersphere of radius $R$ centered at $\overline{I}$. 
Selection of Radius $R$

• By choosing a small value for radius $R$ the facial images are projected near the hypersphere center and further from their original positions.

• By choosing a large value for $R$ the images are projected close to their original positions.

• Therefore, it is expected that for large values of $R$ the identification error rates will be lower compared to the error rates for smaller values of $R$.

• But which values of $R$ are considered large/small?
Automatic Selection of Radius \( R \)

- Automatic selection of radius \( R \) is achieved through the Support Vector Data Description (SVDD) method.

- SVDD is a method for defining the minimum bounding sphere that encompasses most of or all of the training vectors and as such it gives an estimate of the radius \( R \) that should be used.
• SVDD solves the following optimization problem:

\[
\min_{R, \xi, u} \quad R^2 + c \sum_{i}^{N} \xi_i
\]

s.t. \quad \|x_i - u\|_2^2 \leq R^2 + \xi_i

\xi_i \geq 0, \quad i = 1, 2, \ldots, N

where \( R \) is the radius, \( u \) is the center of the sphere, \( x_i \) are the facial image representations, \( \xi_i \) are slack variables, and \( c \) is a parameter that describes the importance of the error.
The optimization problem can be solved by finding the saddle point of the following Lagrangian:

\[
\mathcal{L}(R, \xi_i, u, \alpha, \beta) = R^2 + c \sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} \beta_i \xi_i - \sum_{i=1}^{N} a_i \left( R^2 + \xi_i - \|x_i - u\|^2 \right).
\]

and its optimality conditions:

\[
\frac{\partial \mathcal{L}}{\partial u} = 0 \Rightarrow \sum_{i=1}^{N} a_i u = \sum_{i=1}^{N} a_i x_i, \quad \frac{\partial \mathcal{L}}{\partial R} = 0 \Rightarrow \sum_{i=1}^{N} a_i = 1
\]

\[
\frac{\partial \mathcal{L}}{\partial \xi_i} = 0 \Rightarrow a_i = c - \beta_i
\]
• By using the Karush-Kuhn-Tucker theorem, the problem is formulated in its dual form as:

\[
\max_{\alpha} \sum_{j=1}^{N} a_i x_i^T x_i - \sum_{i=1}^{N} \sum_{j=1}^{N} a_i a_j x_i^T x_j,
\]

under the condition \(0 \leq \alpha_i \leq c\) and \(\sum a_i = 1\).

• Finally, the radius \(R\) is calculated as:

\[
R^2 = \left\{ \min ||x_i - u||^2_2, x_i \text{ is a support vector or } a_i > 0 \right\}
\]
Experimental Setup

- The effectiveness of the Projection-DID method was tested on two facial image datasets:
  - XM2VTS database having 388 train samples, 256 test samples and 128721 dimensions, and
  - Extended Yale B database having 1209 train samples, 1205 test samples and 1200 dimensions

- Facial image representation for face recognition was performed through:
  - pixel value vectorization
  - Linear Discriminant analysis (LDA)

- Three classifiers were used in the process:
  - the K-Nearest Neighbour Classifier (KNN)
  - the Nearest Centroid Classifier (NC) and
  - the Naive Bayes Classifier (NBC).
Experimental Procedure

• The de-identified image representation error and quality preservation were measured with the mean Peak SNR through the mean Mean Square Error (mmSE) metric:

\[ m\text{MSE} = \frac{1}{N_{im}} \sum_{i=1}^{N_{im}} \left[ \frac{1}{np} \sum_{j=1}^{np} (I_i - \hat{I}_j)^2 \right] \]

where \( np \) is the image pixel number, \( I \) is the de-identified facial image and \( \hat{I} \) is the original facial image. The mean PSNR (mPSNR) is calculated through:

\[ m\text{PSNR} = 20 \log_{10} (\text{MAX}_I) - 10 \log_{10} (m\text{MSE}) \]

where \( \text{MAX}_I \) is the maximum pixel value of the image.
SVD-DID Results for each step

• Step 1: SVD Coefficient Zeroing (SVD-CZ)
  ▪ Applying only the first step to de-identify a given image the result is of poor quality with visual artifacts by zeroing only a couple of the first singular values.

  ▪ The maximum error rates are 93.21% for the KNN classifier with parameters $N = 8, lum = +100$.

  ▪ The mPSNR ranging from 13.15 ($N = 1$) to 12.45 ($N = 8$).
SVD-DID Results for each step

• Step 2: SVD Coefficient Averaging (SVD-CA)
  - This individual step gives acceptable error rates and mMSE.
  - Increasing radius $r$ generally leads to higher face recognition error rates, while mPSNR does not fluctuate greatly.

<table>
<thead>
<tr>
<th>XM2VTS</th>
<th>Luminosity +100</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Filter Radius</td>
</tr>
<tr>
<td></td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>20</td>
</tr>
</tbody>
</table>
SVD-DID Results for each step

• Step 3: SVD Modified Sobel Filtering (SVD-MSF)
  ▪ The SVD-MSF step, gives the highest error rates from the other two individual steps.
  ▪ By increasing the value of parameter $d$ we obtain higher mPSNR but, generally the face recognition error rates remain unchanged.

<table>
<thead>
<tr>
<th>XM2VTS</th>
<th>Luminosity +100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value of $d$</td>
<td>KNN</td>
</tr>
<tr>
<td>0.2</td>
<td>50.57 %</td>
</tr>
<tr>
<td>0.5</td>
<td>50.57 %</td>
</tr>
<tr>
<td>1.0</td>
<td>49.81 %</td>
</tr>
</tbody>
</table>
Results for SVD-DID

- The overall result of applying the above steps, leads to high error rates and adequate image quality.

- The parameters can be altered to adjust the equilibrium between image quality and privacy protection depending on the use.

<table>
<thead>
<tr>
<th>XM2VTS</th>
<th>1</th>
<th>90.57 %</th>
<th>90.57 %</th>
<th>93.21 %</th>
<th>12.78</th>
</tr>
</thead>
<tbody>
<tr>
<td>KNN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NBC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mPSNR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$lum = +100, a = 0.8, r = 10, d = 0.5$
### Experimental Results for PDID-O

**TABLE XIV**

<table>
<thead>
<tr>
<th>Radius</th>
<th>KNN</th>
<th>NC</th>
<th>NBC</th>
<th>mPSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>93.21%</td>
<td>93.21%</td>
<td>97.36%</td>
<td>12.19</td>
</tr>
<tr>
<td>30</td>
<td>93.21%</td>
<td>93.21%</td>
<td>93.58%</td>
<td>13.17</td>
</tr>
<tr>
<td>50</td>
<td>90.57%</td>
<td>90.57%</td>
<td>93.58%</td>
<td>14.26</td>
</tr>
<tr>
<td>60</td>
<td>90.57%</td>
<td>90.57%</td>
<td>93.58%</td>
<td>14.86</td>
</tr>
<tr>
<td>67.4034</td>
<td>90.57%</td>
<td>90.57%</td>
<td>93.58%</td>
<td>15.32</td>
</tr>
<tr>
<td>70</td>
<td>90.57%</td>
<td>90.57%</td>
<td>93.58%</td>
<td>15.48</td>
</tr>
<tr>
<td>80</td>
<td>90.57%</td>
<td>90.57%</td>
<td>93.58%</td>
<td>16.15</td>
</tr>
<tr>
<td>100</td>
<td>49.06%</td>
<td>48.30%</td>
<td>61.89%</td>
<td>17.58</td>
</tr>
<tr>
<td>120</td>
<td>26.04%</td>
<td>26.04%</td>
<td>54.72%</td>
<td>19.15</td>
</tr>
</tbody>
</table>

**TABLE XV**

<table>
<thead>
<tr>
<th>Radius</th>
<th>KNN</th>
<th>NC</th>
<th>NBC</th>
<th>mPSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>94.94%</td>
<td>94.19%</td>
<td>92.94%</td>
<td>13.22</td>
</tr>
<tr>
<td>10</td>
<td>89.96%</td>
<td>79.92%</td>
<td>72.61%</td>
<td>15.41</td>
</tr>
<tr>
<td>15</td>
<td>60.83%</td>
<td>88.13%</td>
<td>82.57%</td>
<td>16.91</td>
</tr>
<tr>
<td>17.4241</td>
<td>48.30%</td>
<td>90.37%</td>
<td>86.14%</td>
<td>16.98</td>
</tr>
<tr>
<td>20</td>
<td>38.67%</td>
<td>91.95%</td>
<td>89.38%</td>
<td>16.50</td>
</tr>
</tbody>
</table>
Experimental Results for PDID-O

- $R = 10$
- $R = 30$
- $R = 50$
- $R = 70$
- $R = 100$
- $R = 120$
### Experimental Results for PDID-M

#### TABLE XVI
**Error rates for PDID-M (XM2VTS)**

<table>
<thead>
<tr>
<th>Radius</th>
<th>KNN</th>
<th>NC</th>
<th>NBC</th>
<th>mPSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>96.23%</td>
<td>96.23%</td>
<td>96.23%</td>
<td>17.09</td>
</tr>
<tr>
<td>6</td>
<td>90.19%</td>
<td>94.72%</td>
<td>96.23%</td>
<td>17.44</td>
</tr>
<tr>
<td>8</td>
<td>90.19%</td>
<td>90.19%</td>
<td>90.19%</td>
<td>17.80</td>
</tr>
<tr>
<td>10</td>
<td>90.19%</td>
<td>90.19%</td>
<td>90.19%</td>
<td>18.18</td>
</tr>
<tr>
<td>12</td>
<td>66.04%</td>
<td>71.70%</td>
<td>90.19%</td>
<td>18.57</td>
</tr>
<tr>
<td>14</td>
<td>53.21%</td>
<td>53.58%</td>
<td>73.58%</td>
<td>18.98</td>
</tr>
</tbody>
</table>

#### TABLE XVII
**Error rates for PDID-M (Yale B)**

<table>
<thead>
<tr>
<th>Radius</th>
<th>KNN</th>
<th>NC</th>
<th>NBC</th>
<th>mPSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>96.76%</td>
<td>92.61%</td>
<td>88.13%</td>
<td>13.58</td>
</tr>
<tr>
<td>2</td>
<td>95.02%</td>
<td>89.21%</td>
<td>83.32%</td>
<td>14.81</td>
</tr>
<tr>
<td>3</td>
<td>88.71%</td>
<td>89.71%</td>
<td>83.15%</td>
<td>16.21</td>
</tr>
<tr>
<td>4</td>
<td>76.51%</td>
<td>89.96%</td>
<td>81.74%</td>
<td>17.82</td>
</tr>
<tr>
<td>5</td>
<td>66.14%</td>
<td>90.54%</td>
<td>81.41%</td>
<td>19.69</td>
</tr>
</tbody>
</table>
Experimental Results for PDID-M

$R = 4$  
$R = 6$  
$R = 8$  
$R = 10$  
$R = 12$  
$R = 14$
Method Comparison

• From the above discussion it can be concluded that both methods offer high face recognition error rates.

• In general, the SVD-DID methods display slightly higher face recognition error rates compared to the Projection-DID methods by one or two percent.

• Depending on whether privacy is a must or if quality is more important, correct selection of the parameters in each method can offer the desired outcome.
Method Comparison

SVD-DID error rate > 90%

SVD-SDID error rate > 90%

PDID-O error rate > 90%

PDID-M error rate > 90%
Conclusions SVD-DID

• The SVD-DID method, aims to limit the effectiveness of face identification methods, while retaining part of the initial visual quality.

• From the above results and discussion, it can be deducted that using the appropriate parameter values in each step, a high level of privacy can be attained.

• In the YaleB database, the highest error rate achieved was 97.51% and the highest face recognition error rate for the XM2VTS database was 97.36%.

• Despite the high face recognition error rate, the end product of these methods can be characterized as acceptable for everyday use.
Conclusions Projection-DID

- The developed Projection-DID methods aim to limit the effectiveness of face identification methods while retaining adequate visual quality.
- By applying these methods a high level of privacy can be attained.
- The highest identification error rates achieved were:
  - [PDID-O] 97.36% (XM2VTS) and 94.94% (Yale B) and
  - [PDID-M] 96.23% (XM2VTS) and 96.76% (Yale B).
- Despite the high error rates, the end product of these methods can be characterized as acceptable for everyday use, rendering the Projection-DID method successful in protecting privacy and providing a visually acceptable output.
Thank You.