De-Identifying Facial Images Using Singular Value Decomposition and Projections

P. Chriskos, O. Zoidi, A. Tefas and I. Pitas Department of Informatics
Aristotle University of Thessloniki
Thessaloniki 54124 Greece
chriskos@csd.auth.gr, {ozoidi,tefas,pitas}@aiia.csd.auth.gr

***Abstract*—In this article, a number of methods are analyzed that manipulate images in a manner that hinders face recognition by automatic recognition algorithms. The purpose of these methods, is to partly degrade image quality, so that humans can identify the person or persons in a scene, while common classification algorithms fail to do so. The approach used to achieve this involves the use of singular value decomposition (SVD) and projections on hyperspheres. From experiments it can be concluded that, these methods reduce the percentage of correct classification rate by over 90% . In addition, the final image is not degraded beyond recognition by humans.**

I. Introduction

With the increasing amount of visual media that is shared, viewed and stored on-line, it is incontestable that privacy is a main concern for all users. The free access that is granted to all this visual information may carry many dangers concerning the privacy of the creators and of the subjects in these media. Face recognition algorithms are able to identify faces in videos and images without much effort, thus violating the privacy of the subjects. Malicious users can use video sharing sites and social media to collect information about specific individuals and groups fast and effortlessly. Moreover, the wide use of video surveillance in public places, in conjunction with face identification software, is a major threat of privacy, since, all persons can be identified regardless of suspicion level. Other examples of contributors to the problem include Google Street View and EverySpace among others, whose attempt to provide services which include visual data inevitably invade our everyday privacy, although not intentionally. As such, the necessity arises to develop methods that protect the subject's privacy, while maintaining a level of quality. This quality is not only limited to the visual quality of the final product, but the viewer must also be able to recognize the number of individuals in a scene, possibly even the individuals themselves and what actions are taking place in the image or video frame.

With this in mind, suppose a malicious user has trained a classifier in order to recognize images of targeted individuals or groups in a set of images available online. New images that are modified by a certain method, will not be recognized by the trained classifier, tackling the attempt of a malicious user searching new images of his targets and rendering further activities of the targets safe.

Most face de-identification methods attempt to deceive automatic face recognition methods by also hindering identification by human viewers. These methods aim to destroy the majority, if not all, of the data concerning the depicted individual. A method developed by the authors in [1] and [2], de-identify persons in videos by de-identifying not only the face area but the person as a whole. Ad-hoc solutions facial images de-identification [8] include the use of simple methods. Such Methods are applying a mask on parts of the face. Black bars are used in order to cover the eyes, while T-shaped masks cover both the eyes and the nose other mask shapes can also be used such as elliptical or circular masks that ususally cover the entire face area. Other masks reveal only the mouth and, finally, a black mask can be applied to the entire face, destroying all visual information of the facial image [8]. Additional simple methods include methods that blur the face area using low-pass filters [8], methods that add random noise with a predetermined distribution, methods that use the negative image and methods that swap facial areas, such as eyes, nose, mouth, between images that belong to different individuals [11]. Finally, simple methods also exist that subsample an image leading to pixelation, or that threshold the pixel values [8]. Moreover, more advanced methods exist that implement the k-anonymity model [9] [10], so that all of the de-identified images indiscriminately relate to at least k elements of the initial image set. In [3] a multi-factor farame-work is introduced that unifies linear, bilinear and quadratic models and an algorithm is also used that allows a better estimation of the parameters used in the algorithm. Other methods explloit characteristics of identification methods such as eigenface-based algorithms, k-anonymity models and PCA or LDA face recognition methods in order to defeat them [12]. Another method replaces faces in an image with 3D morphable models [4]. Finally, another method exists that reduces the number of eigenvectors used in constructing the final images from basis vectors [13].

In this article two methods are descibed that aim to reduce the percentage of positive face identification of common recognition algorithms, while retaining enough visual information to characterize the end product as visually acceptable. These images can then be used in context where circulation of images is unrestrained through various networks and the Internet. These images can be shared social media, on profile pictures, picture sharing sites and others. Another application can be in videos, where coupled with face detection software this method can de-identify the faces in each video frame thus rendering the video content safe for distribution through networks or sharing them on video sharing sites.

The proposed methods utilize in one case the singular value decomposition method (SVD), manipulating the values of the coefficients, in order to alter the initial image, and in the second case projections on hyperspheres. The purpose in both cases is to enable human viewers to identify the individual pictured, while hindering common identification methods from achieving a high identification rate.

The article is organized as follows. Section II, provides a description of the two methods. Section III, the results of the two methods are analyzed. Finally, the conclusions are drawn in Section V.

II. Facial Image De-Identification Methods

In this section the two approaches are analyzed. The first approach utilizes the Singular Value Decomposition method to de-identify facial images, while the second one uses projections on hypersphere to achieve de-identification.

Following is the description of the de-identification method based on Singular Value Decomposition (SVD) which is refered to as SVD-DID. This method consist of a series of steps that alter the initial image's decomposition matrices to achieve de-identification that are described below in more detail.

A. Person de-identification based on SVD

The workhorse of the proposed method is the Singular Value Decomposition (SVD) method applied on facial images. The SVD, [5] [6] [7] factorizes the input matrix (in our case a facial image) $\mathbf{A}$ as a product of three matrices: the singular values matrix $\mathbf{S}$ and the eigenvectors matrices $\mathbf{U}$ and $\mathbf{V}$. In more detail, Singular value decomposition (SVD) is a matrix factorization method that approximates a matrix $\mathbf{A} \in \Re^{n\times p}$ with the product of three matrices $\mathbf{U} \in \Re^{n\times n}, \mathbf{S} \in \Re^{n\times p}$ and $\mathbf{V} \in \Re^{p\times p}$. The SVD theorem, states that any real matrix $\mathbf{A} \in \Re^{n\times p}$ can be decomposed uniquely as

$$\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T \tag{1}$$

Matrices $\mathbf{U}$ and $\mathbf{V}$ are orthogonal. The eigenvectors of $\mathbf{A}\mathbf{A}^T$ make up the columns of matrix $\mathbf{U}$ and the eigenvectors of $\mathbf{A}^T\mathbf{A}$ consist the columns of matrix $\mathbf{V}$. Matrix $\mathbf{S}$ is a diagonal matrix with the same dimensions as the input matrix $\mathbf{A}$. The singular values in $\mathbf{S}$ are the square roots of the matrix $\mathbf{A}\mathbf{A}^T$ or $\mathbf{A}^T\mathbf{A}$ eigenvalues.

The proposed person de-identification method utilizes the SVD to manipulate facial images in order to reduce facial identification by software agents. This method alters the values in the matrices produced by the decomposition.

In order to reduce the correct identification rate, the following steps are followed. First, the coefficients (singular values) of matrix $\mathbf{S}$ with the largest values are reduced to zero. Next, the matrices $\mathbf{U}$ and $\mathbf{V}$ are blurred using an averaging filter. Finally, the same matrices are sharpened using a modified Sobel filter. The logic behind this course of action, is described below.

Fig. 1. Left: Original Frame , Right: Result for SVD-CZ with $N = 1$

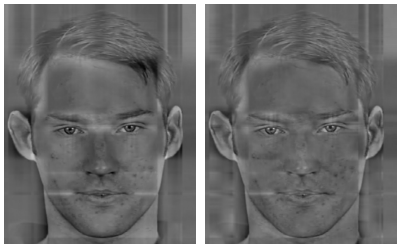

Fig. 2. Left: Result for SVD-CZ with $N = 2$, Right: Result for SVD-CZ with $N = 4$

1) SVD Coefficient Zeroing (SVD-CZ): The most discriminative visual information in an image lies in the coefficients (singular values) with the largest values. Therefore, in the first step, the idea is to remove this information contained in the first coefficients, in the form of pixel luminosity. Since we are removing the first N coefficients, we are actually removing those coefficients that contain the majority of information that a face recognition algorithm would use to successfully identify a subject. This is achieved by setting the first N singular values in $\mathbf{S}$ to zero. Equivalently, we remove the first N primary coefficients used in recomposing the final image. This process produces a new $\mathbf{S}$ matrix referred to as $\mathbf{S}_{CZ}$.

By setting the N largest singular values to zero, the final image tends to darken with respect to the input image. In order to preserve adequate visual data for easy face identification by human viewers, we increase the luminosity of all pixels in the end of the process, by adding a fixed value to the pixels of the output image. This darkening effect is due to the fact that the largest coefficients in matrix $\mathbf{S}$ are reduced to zero. These values are subsequently used in the calculation of the output image through matrix multiplication. Since matrix multiplication involves summing of coefficients some of which are set to zero instead of having their initial positive values, the result is smaller in numerical value. As a result, the output image is darker.

The effect of SVD coefficients zeroing can be viewed in Figures 1 and 2, where the darkening effect was reduced by adding luminosity 100 in each pixel of the final images.

2) SVD Coefficient Averaging (SVD-CA): As we have previously discussed, the method goal is to allow human viewers to recognize with relative ease the subject in an image and, at the same time, fool automatic classifiers trying to identify specific individuals. This difficulty will arise from the fact that these classifiers where trained with clean versions of the images and ,subsequently, will falsely identify the manipulated images. To achieve this, the entries of the eigenvectors in

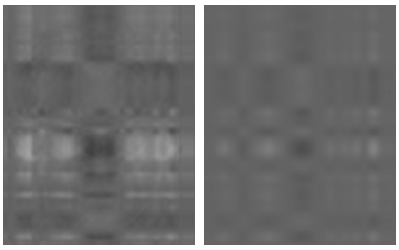

Fig. 3. Left: Result for SVD-CA with r = 4 Right: Result for SVD-CA with r=10

Fig. 4. Left: Result for SVD-CA with $r = 10$, Right: Result for SVD-CA with $r = 20$

matrices $\mathbf{U}$ and $\mathbf{V}$ are mixed by a blurring filter. The averaging filter employed is the $m \times m$ circular averaging filter, with $m = 2r + 1$, where r is the radius of the circular filter. By applying the averaging filter to matrices $\mathbf{U}$ and $\mathbf{V}$, we obtain the matrices $\mathbf{U}_{averaged}$ and $\mathbf{V}_{averaged}$. Recomposing the image solely from the averaged matrices, leads to poor visual quality, as portrayed in Figure 3. From Figure 3 we notice that the output images are degraded beyond recognition. In order to counterbalance this effect, only a percentage of the values from the new matrices is used. The final matrices $\mathbf{U}_{CA}$ and $\mathbf{V}_{CA}$ utilized to calculate the output image are given by the following equations:

$$\mathbf{U}_{CA} = \frac{\alpha * \mathbf{U}_{averaged} + \mathbf{U}}{1 + \alpha} \tag{2}$$

and

$$\mathbf{V}_{CA} = \frac{\alpha * \mathbf{V}_{averaged} + \mathbf{V}}{1 + \alpha}, \tag{3}$$

where the parameter α adjusts the equilibrium between visual quality and privacy protection. Similarly to the previous method, this step also introduces a darkening effect in the resulting image. This effect is adjusted as in the first step. The visual result of equations (2), (3) is displayed in Figure 4, with added luminosity 100.

3) SVD Modified Sobel Filtering (SVD-MSF): The final step utilizes a modified Sobel filter in order to manipulate matrices $\mathbf{U}_{CA}$ and $\mathbf{V}_{CA}$. Sobel filtering is generally used for edge detection in images. Edge detection is used to remove part of the previous blurring, while mixing the coefficient values even further.This modified filter, contains values different from the classic Sobel filter. More specifically, the filter $\mathbf{G}$ used is a 3×3 matrix of the form:

$$\mathbf{G} = \begin{bmatrix} d & 2d & d \\ 0 & 0 & 0 \\ -d & -2d & -d \end{bmatrix} \tag{4}$$

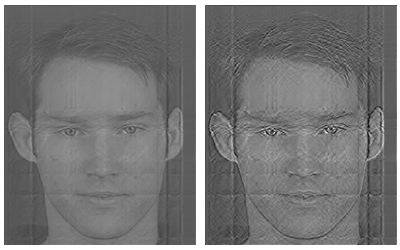

Fig. 5. Left: Result for SVD-MSF $d = 0.2$, Right: Result for SVD-MSF $d = 1.0$

Fig. 6. Left: Initial Image A, Right: Initial Image B

where the parameter d was empirically determined to be in the range $[0.2, 0.8]$. Edge detection when applied to matrices $\mathbf{U}_{CA}$ and $\mathbf{V}_{CA}$ results in matrices $\mathbf{U}_{final}$ and $\mathbf{V}_{final}$. Similar to the SVD-CA step, only a percentage of the resulting matrix is used in computing the output image according to (2), (3). The output of this individual step is show in Figure 5. After applying the above steps, the output image $\mathbf{P}$ is calculated, through the matrices $\mathbf{U}_{final}$, $\mathbf{S}_{CZ}$ and $\mathbf{V}_{final}$ using the formula:

$$\mathbf{P} = \mathbf{U}_{final}\mathbf{S}_{CZ}\mathbf{V}_{final}^{T} \tag{5}$$

In the rest if the article, this series of steps will be referred to as the SVD-DID method.

Having analyzed the SVD-DID method and its individual steps we will now take a look at an extension of this method.

B. Extending the SVD-DID Method by Analyzing the Decomposition Matrices

In an attempt to increase the error rate of the classifiers each matrix resulting from the Singular Value Decomposition was more closely examined.

1) Matrix $\mathbf{S}$*:* Matrix $\mathbf{S}$ does not project any properties that could be used in order to potentially increase the error rate. The $\mathbf{S}$ matrices from each subjects image, did not differentiate much between subjects. It is possible to simply swap $\mathbf{S}$ matrices between images when recomposing the image without any effect on the ability of the classifiers to correctly identify a subject. The initial images are displayed in Figure 6 and the images with swapped $\mathbf{S}$ matrices can be seen in Figure 7. As it can be seen only a few visual artifacts have been introduced that do not hinder correct identification.

2) Matrices $\mathbf{U}$ *and* $\mathbf{V}$*:* Matrices $\mathbf{U}$ and $\mathbf{V}$ contain the majority of information that is used to recompose the final image. In contrast with matrix $\mathbf{S}$, they cannot be switched between images, since they introduce too many visual artifacts and greatly degrade the visual quality of the image as can be seen in Figure 8. In order to find characteristics of these

Fig. 7. Left: Image A with matrix $\mathbf{S}$ from B, Right: Image B with matrix $\mathbf{S}$ from A

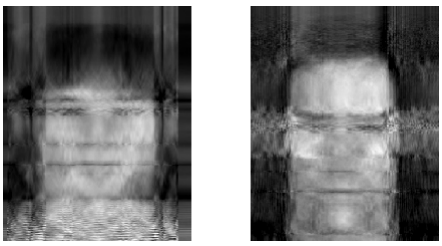

Fig. 8. Left: Image A with matrix $\mathbf{U}$ from B, Right: Image A with matrix $\mathbf{V}$ from B

matrices that could be used to increase error rates for both classifiers, the statistical properties of these two matrices where examined. The mean of the matrices generally displayed a random distribution that could not be utilized to increase classifier error rates. The same did not apply when the standard deviation (STD) was examined. STD was calculated using the following equation:

$$STD = \left(\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x}) \right)^{\frac{1}{2}}, \tag{6}$$

where $\bar{x}$ is the mean value given by:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i, \tag{7}$$

n is the number of values and x_i is each distinct value in each eigenvector. The STD of each eigenvector was calculated and was stored in a vertex containing the STD's of each eigenvector in the matrix. The STD of this resulting vertex, was to an extend related to the subject portrayed in each image and could subsequently be altered to misguide classifiers from correctly identifying the subject portrayed in each image.

The above observation hints that the standard deviation is a value that plays a crucial role in producing the output image. As such, adjusting the STD is a way to further increase the effectiveness of the SVD-DID method.

3) Adjusting the Standard Deviation: In order to adjust the standard deviation, equation 6 must be taken into account. From the three parameters used to calculate the STD, n the number of values cannot be adjusted and altering $\bar{x}$ requires adjusting the final parameter x_i which is the values of each eigenvector. In order to adjust the STD, the difference $x_i - \bar{x}$ must be altered. This was accomplished by adding a portion of the overall mean $\bar{x}'$ of the eigenvector matrix $\mathbf{E}$ to each value of the matrix using the following equation:

$$\mathbf{E}' = \gamma * E + (1 - \gamma) * \bar{x}', \tag{8}$$

where $\mathbf{E}'$ is the resulting eigenvector matrix, γ is a parameter in the range $[0, 1]$ which adjust the portion of the overall mean that is added to the values of each eigenvector. In this method, the output image is computed as:

$$\mathbf{P} = \mathbf{E}' \mathbf{S}_{CZ} \mathbf{V}_{final}^{T} \tag{9}$$

in which case matrix $\mathbf{U}_{final}$ has been replaced by $\mathbf{E}'$ which is computed through the formula:

$$\mathbf{E}' = \gamma * U + (1 - \gamma) * \bar{x}. \tag{10}$$

By applying the above process, the STD has been altered which succeeds in increasing the error rate of the classifiers in certain cases, as is discussed in a following section.

Having analyzed the SVD-DID method and its individual steps we will now take a look at the second approach which involved projecting the initial images on hyperspheres. The Projection-DID method is analyzed in more detail below.

C. Projections Used for De-Identiffication

Each image occupies a position in the n-dimensional space, where the dimensionality n of the image is equal to the number of pixels. Intuitively it is expected that images depicting the same individual with the same pose are bound to lie close together in space forming local clusters, while images depicting different individuals are bound to lie farther apart.

The general idea is to bring images of different individuals closer together in order to prevent classifiers from correctly identifying a subject in an image and at the same time, preserve enough information from the first image so that human viewers can identify the depicted individual. One way to achive this is to project the images on a hypersphere with radius R centered at some origin. This projection is excpected to distort the images such that, the new architecture of the data does not allow trained classifiers from discerning between the individuals. In order to achieve this the hypersphere must firstly be defined.

A hypersphere [14] [15] is a generalization of the ordinary circle in 1 dimension and the ordinary sphere in 2 dimensions to dimensions $n \geq 3$. A can be defined as the set of points in the n-dimensional space, which are at distance R from a center pointhypersphere S^{n-1} centered at some origin as:

$$S^{n-1} = \{\mathbf{x} \in \mathbb{R}^n : ||\mathbf{x}|| = R\}. \tag{11}$$

where $\mathbf{x}$ is a point in the n-dimensional space. The projection of a point $\mathbf{x} \in \mathbb{R}^n$ onto S^{n-1} is given by the following equation [17]:

$$P_{S^{n-1}}(\mathbf{x}) = \frac{R}{||\mathbf{x}||}\mathbf{x}, \tag{12}$$

where $P_{S^{n-1}}(\mathbf{x})$ denotes the projection of point $\mathbf{x}$ onto the hypersphere S^{n-1}.

At this point the value of radius R and the exact center must be addressed. Choosing a small value for radius R allows us to project the initial images close to the center, and subsequently close to each other. This means that images of different individuals will also be close to images from other individuals. Choosing a large value for R, it is possible to project the initial images farther from the center, closer to the

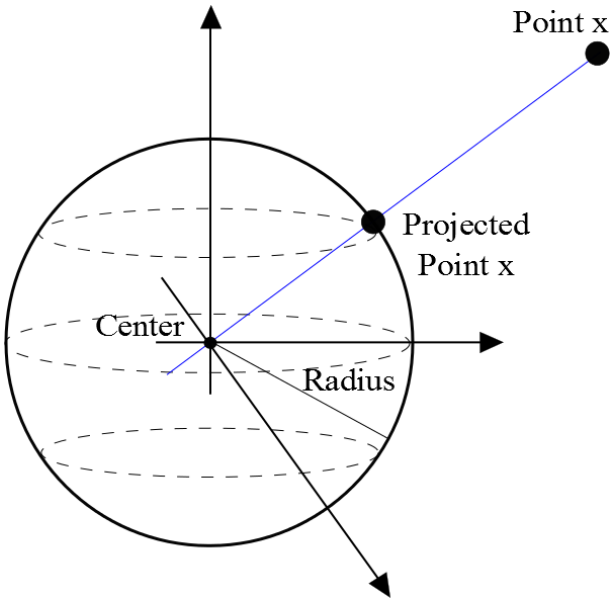

Fig. 9. Projection of point onto a sphere.

initial locations. It is suspected that for small values of R the error rates of the classifiers will be high, since the classifiers will be unable to discern between the images from different individuals and as a result will classify them falsely. The value of R will also be responsible for preserving the quality of the initial images. For small values of R image quality will suffer, while for large values of R the quality of the output images will be closer to that of the initial image. These observations can hint to the choice for the value of parameter R.It would be preferable though if radius R was calculated based on the images in each dataset. This can be achieved using the Support Vector Data Description (SVDD) method.

The Support Vector Data Description or SVDD [19] is a method for defining the minimum bounding sphere that encompasses most of or all of the training vectors $\mathbf{x}_i$ where $i = 1, 2, \ldots, N$ and N denotes the number of training vectors. This sphere S can be defined by a center u and a radius R, which can be computed by optimizing:

$$\min_{R,\xi,\mathbf{u}} \quad R^2 + c\sum_{i}^{N} \xi_i \tag{13}$$

$$s.t. \quad ||\mathbf{x}_i - \mathbf{u}||_2^2 \leq R^2 + \xi_i \tag{14}$$

$$\xi_i \geq 0, \quad i = 1, 2, \ldots, N \tag{15}$$

where ξ_i are the slack variables and c is a parameter that describes the importance of the error in the optimization problem.

Using the Karush-Kuhn-Tucker (KKT) theorem [18] the optimization problem mentioned above can be solved by finding the saddle point a Lagrangian. From the optimality conditions of the above problem, the center
$mathbfu$ is given by:

$$\mathbf{u} = \sum_{i=1}^{N} a_i \mathbf{x}_i \tag{16}$$

where a_i is a Lagrangian curve parameter. It can be proven that center $\mathbf{u}$ can be approximated by the mean of a given dataset and this is the rason why the mean image is used as a center in the PDID-M method below.

Finally, the optimization problem (13) can be formulated to its dual from:

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{N} a_i \mathbf{x}_i^T \mathbf{x}_i - \sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j \mathbf{x}_i^T \mathbf{x}_i, \tag{17}$$

under the condition $0 \leq a_i \leq c$ and $\sum_i a_i = 1$. After solving 17 radius R can be calculated as:

$$R^2 = \{\min ||\mathbf{x}_i - \mathbf{u}||_2^2, \mathbf{x}_i \text{ is a support vector or } a_i > 0\} \tag{18}$$

With the above approach it is possible to calculate a good estimate of radius R that will provide with the required distortion to de-identify the input facial images.

Two different projections where used in order to de-identify facial images. The first one is the average of the projection on the origin and the mean image. The formula used to calculate the de-identified version $\mathbf{x}_{DID}$ of an image $\mathbf{x}$ is the following:

$$\mathbf{x}_{DID} = \frac{1}{2}\left(\frac{R}{||\mathbf{x}||}\mathbf{x} + \bar{\mathbf{x}}\right). \tag{19}$$

where $\bar{\mathbf{x}}$ is the mean image, R denotes the radius of the hypersphere and $||\mathbf{x}||$ is the measure of image $\mathbf{x}$. This projection method will be referred to as Projection De-Deidentifiaction on Origin or PDID-O for short.

The second projection used was the projection with a hypersphere centered on the mean image. The mean image is computed using the following equation:

$$\bar{\mathbf{x}} = \frac{1}{N_{im}} \sum_{i=1}^{N_{im}} \mathbf{x}_i \tag{20}$$

where $\bar{\mathbf{x}}$ is the average image, N_{im} is the number of images in the given dataset and $\mathbf{x}_i$ is each individual image in the dataset. The de-identified image can be calculated using the following formula:

$$\mathbf{x}_{DID} = \left(\frac{R * (\mathbf{x} - \bar{\mathbf{x}})}{||\mathbf{x} - \bar{\mathbf{x}}||} + \bar{\mathbf{x}}\right). \tag{21}$$

and as above $\bar{\mathbf{x}}$ is the mean image, R denotes the radius and $||\mathbf{x}||$ is the measure of image $\mathbf{x}$. This projection method will be referred to as Projection De-Deidentifiaction on Mean Image or PDID-M for short.

In sections II-A and II-C the two facial image de-identification methods where analyzed. To recap the SVD-DID method alters the values in the decomposition matrices in order to achieve de-identification, while the Projection-DID method utilizes projections on hyperspheres to achieve the same goal.

III. Experimental Procedure and Results

Having described these ,methods we will move on to describing the experimental procedure and the databases used. There will also be a discussion of the visual results and the error rates the classifiers used display.

A. Database Description, Classifiers and Metric Used

Experiments to test the effectiveness of the SVD-DID and Projection-DID method where run on the XM2VTS [21] and the Extended Yale B [20] databases. From the XM2VTS database 16 individuals from the first recording where selected and used in the experimental process. The individuals face the camera on a neutral background. The frontal images where isolated and subsequently where cropped to the face area. Finally the images where converted to 8-bit grayscale images. This process resulted in a dataset with 388 train samples and 265 test samples from the 16 videos. Each sample of the above dataset has 128721 dimensions (401×321), with both train and test samples converted into vectors with dimensions 128721×1. The Extended Yale B database contains images from 38 individuals under different lighting conditions. Train and test sets contain 1209 and 1205 samples respectively. These sets where defined by randomly selecting half the images from each individual. Each image has 1200 dimensions (40×30) and was used in vector form with dimensions 1200×1. The train sets mentioned above where used to train classifiers and then the test data where used to measure the efficiency of the proposed method. The classifiers used in the process where the K-Nearest Neighbour Classifier (KNN) with 1 nearest neighbour and the Naive Bayes Classifier. In the case of the KNN classifier varying the number of nearest neighbours to 3 and 5 yielded similar results.

In order to calculate the difference between the initial and de-identified images and to measure the degradation of quality introduced by the two methods, the mean Mean Square Error (mMSE) metric was used. To calculate the mMSE the images must be in vector form $np \times 1$, where np is the number of pixels in each image. As such the formula that is used to calculate the mMSE is:

$$mMSE = \frac{1}{N_{im}} \sum_{i=1}^{N_{im}} \left[\frac{1}{np} \sum_{j=1}^{np} (\mathbf{x}_i(j) - \hat{\mathbf{x}}_i(j))^2 \right] \qquad (22)$$

where N_{im} is the total number of images, np is the number of image pixels, $\mathbf{x}_i$ is the i^{th} original image and finally $\hat{\mathbf{x}}_i$ is the i^{th} output image of the applied method. All calculations for the mMSE are done with the images having values in the range $[0, 1]$, after they where divided by 255.

These two datasets contain only a small number of individuals compared to the datasets that an attacker would use to identify a target. It is intuitively expected that if the two methods succeed in protecting privacy in these small datasets they will achieve even higher levels of privacy protection in large datasets.

B. Results for SVD-DID

In this section, we present and analyze the results from training and testing the efficiency of each of the steps described in Section II-A. The results are presented for each step with error percentages and the mean Mean Square Error (mMSE) for the test set of images, compared to the initial set.As mentioned above, the necessity to increase the luminosity of all pixels in the final image arises in order to counterbalance the darkening effect introduced by the algorithm steps. In the experiments, the values 0, 100 and 150 were used for reducing the darkening effect.

TABLE V
ERROR RATES FOR SVD-CA $r = 10$

Param. α	KNN (K=3)	Naive Bayes	mMSE
α = 0.5	52.08 %	68.30 %	0.0549
α = 0.8	53.21 %	83.02 %	0.0477
α = 1.0	59.25 %	86.79 %	0.0468

Following are the results for each step in theSVD-DID method.

1) Results for SVD-CZ: Experimental results of setting the N largest singular values to zero are depicted in Tables III-B1 and III-B1. It can be observed that, the increase of the number of zeroed singular values tends to increase the mMSE while, at the same time, the error rate is increased for both classifiers. Altering the number of nearest neighbors in the KNN classifier such as 1 and 5, yields the same results. These results are displayed for different number of zeroed coefficients and for different amounts of brightness added to the final image. Visual results can be seen in Figures 1 and 2. It can be easily seen from these figures that this method alone does not provide an acceptable output image, since too many visual artifacts are introduced that decrease the overall image quality, even by zeroing only a couple of the first singular values.

2) Results for SVD-CA: For the circular averaging filter, the error rates are displayed in Tables III and IV. The error rates where calculated in relation with the radius r of the circular filter and the amount of brightness that is applied for both databases. The mMSE in this case does not increase by increasing radius r. However, is shows a relevance to the added luminosity as well. On the other hand, error rates increase by increasing the radius value. Resulting images can be seen in Figure 4.

For this step, it was mentioned that only a percentage of the newly calculated matrices is used. By varying parameter α, we obtain the results in Table V. We conclude that parameter α affects the error rate of both classifiers. The parameters where $r = 10$, $\alpha = 0.8$. From this table it can be observed that by increasing parameter α the mMSE increases along with the error rate of the classifiers.

3) Results for SVD-MSF: Applying the modified Sobel filter to the matrices, we obtain the error rates displayed in Tables VI and VII. The results are related with parameter d and the added luminosity. By increasing the value of parameter d we obtain higher mMSE but, generally, the error rates remain unchanged. As before, parameter α was set to 0.8. Image results of the method are displayed in Figure 5 for parameters $d = 0.5$, $\alpha = 0.8$.

In this method, altering parameter α, leads to the error rates in Table IX. The error rates are for the parameter d value $d = 0.5$ and added luminosity 100. In this case, altering α leads to a decrease of the mMSE and varying error rates.

Summarizing the results for each phase independently, we observe that some of these phases either degrade image quality to a great extend, or provide insufficient privacy protection. By

TABLE I
ERROR RATES FOR NUMBER OF ZEROED COEFFICIENTS (XM2VTS)

Zeroed Coefficients	Luminosity +0			Luminosity +100			Luminosity +150		
	KNN	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**
1	69.34 %	69.34 %	0.1903	55.47 %	55.47 %	0.0484	52.45 %	52.45 %	0.0927
2	90.57 %	90.57 %	0.1959	72.45 %	72.45 %	0.0523	72.45 %	72.45 %	0.0959
4	90.57 %	90.57 %	0.2001	83.02 %	83.02 %	0.0552	78.49 %	78.49 %	0.0981
8	93.21 %	93.21 %	0.2023	93.21 %	93.21 %	0.0569	79.25 %	79.25 %	0.0996

TABLE II
ERROR RATES FOR NUMBER OF ZEROED COEFFICIENTS (YALEB)

Zeroed Coefficients	Luminosity +0			Luminosity +100			Luminosity +150		
	KNN	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**
1	92.53 %	96.43 %	11.373 e-4	79.25 %	92.53 %	5.5735 e-4	90.12 %	97.51 %	13.426 e-4
2	93.61 %	97.34 %	11.910 e-4	97.26 %	96.51 %	6.0820 e-4	93.94 %	97.51 %	13.921 e-4
4	95.60 %	97.34 %	12.187 e-4	97.34 %	96.51 %	6.3574 e-4	95.93 %	97.51 %	14.195 e-4
8	96.93 %	97.34 %	12.312 e-4	97.34 %	97.51 %	6.4748 e-4	97.26 %	97.51 %	14.309 e-4

TABLE III
ERROR RATES FOR CIRCULAR AVERAGING FILTER (XM2VTS)

Filter Radius	Luminosity +0			Luminosity +100			Luminosity +150		
	KNN	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**
5	85.66 %	67.55 %	0.0815	49.43 %	83.02 %	0.0518	80.38 %	72.08 %	0.1524
10	86.04 %	69.06 %	0.0895	53.21 %	83.02 %	0.0477	80.38 %	72.08 %	0.1422
20	90.57 %	71.70 %	0.0935	50.06 %	86.79 %	0.0459	80.38 %	72.08 %	0.1375

TABLE IV
ERROR RATES FOR CIRCULAR AVERAGING FILTER (YALEB)

Filter Radius	Luminosity +0			Luminosity +100			Luminosity +150		
	KNN	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**
5	94.85 %	96.18 %	1.1050 e-4	90.54 %	97.51 %	1.4683 e-4	97.01 %	97.51 %	4.0146 e-4
10	94.61 %	96.43 %	1.2097 e-4	89.88 %	97.51 %	1.4204 e-4	96.93 %	97.51 %	3.8905 e-4
20	94.77 %	96.43 %	1.2632 e-4	89.88 %	97.51 %	1.3986 e-4	96.85 %	97.51 %	3.8310 e-4

TABLE VIII
ERROR RATES FOR SVD-MSF $d = 0.5$

Param. α	KNN (K=3)	Naive Bayes	mMSE
α = 0.5	52.08 %	68.30 %	0.0512
α = 0.8	50.56 %	86.79 %	0.0454
α = 1.0	55.47 %	90.57 %	0.0453

merging all these phases in one method we obtain the results shown in the following section.

4) Putting it all together for the SVD-DID method: The SVD-DID method as a whole includes the three steps described in the previous sections (II-A1,II-A2,II-A3). By applying these in the following order, i.e. SVD-CZ, SVD-CA and SVD-MSF, we derive this method that encompasses the advantages of all phases which are image quality and privacy protection. The defined parameters of this method can be altered to adjust the equilibrium between image quality and privacy protection, depending on the purpose of applying this method. The results for the full application of this method are displayed in Tables IX and X and Figures 10 and 11. The results in the tables are displayed in relation with parameter α, added luminosity and number of zeroed coefficients. Other visual results are displayed in Figure 12 for higher luminosity added to the image at 150. Figure 13 shows the result of applying a circular filter and a modified Sobel filter with inappropriate parameters.

From these results we observe that with the correct selection of parameter values, we can attain high levels of privacy, while maintaining acceptable image quality. Error rates for both classifiers are high for both databases with maximum values at 93.71 % for the XM2VTS database and 97.51 % for the YaleB database.

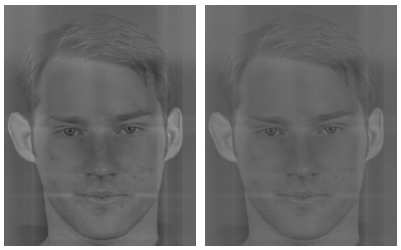

Fig. 10. Left: Result for SVD-DID for N=1, luminosity +100, α=0.5, Right: Result for SVD-DID for N=1, luminosity +100, α=0.8 (r = 10 and d = 0.5)

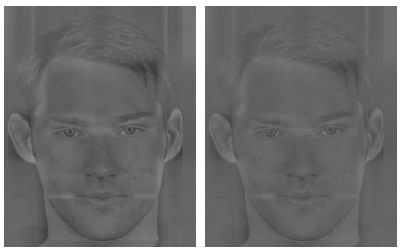

Fig. 11. Left: Result for SVD-DID for N=2, luminosity +100, α=0.5, Right: Result for SVD-DID for N=2, luminosity +100, α=0.8 (r = 10 and d = 0.5)

5) Results for SVD-DID with Standard Deviation Adjustment: The aim of applying the method mentioned in the previous section is to increase the effectiveness of the SVD-

TABLE VI
ERROR RATES FOR MODIFIED SOBEL FILTERING (XM2VTS)

Value of d	Luminosity +0			Luminosity +100			Luminosity +150		
	KNN	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**
0.2	90.57 %	67.17 %	0.0978	50.57 %	86.79 %	0.0447	84.53 %	72.08 %	0.1335
0.5	90.57 %	67.17 %	0.0988	50.57 %	86.79 %	0.0447	85.66 %	72.08 %	0.1342
1.0	69.43 %	67.17 %	0.1041	49.81 %	86.79 %	0.0509	85.66 %	72.08 %	0.1397

TABLE VII
ERROR RATES FOR MODIFIED SOBEL FILTERING (YALEB)

Value of d	Luminosity +0			Luminosity +100			Luminosity +150		
	KNN	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**
0.2	94.11 %	96.35 %	1.3880 e-4	89.38 %	97.51 %	1.4197 e-4	96.85 %	97.51 %	3.8252 e-4
0.5	95.19 %	96.10 %	1.6637 e-4	90.04 %	97.34 %	1.7403 e-4	96.93 %	97.51 %	4.1433 e-4
1.0	95.52 %	95.44 %	4.4246 e-4	90.04 %	97.01 %	4.4916 e-4	96.93 %	97.51 %	6.8898 e-4

TABLE IX
ERROR RATES FOR SVD-DID (XM2VTS)

Zeroed Coefficients	Luminosity +0						Luminosity +100					
	$\alpha = 0.5$			$\alpha = 0.8$			$\alpha = 0.5$			$\alpha = 0.8$		
	KNN	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**
1	90.57 %	97.36 %	0.1947	90.57 %	93.74 %	0.1971	76.60 %	93.21 %	0.0508	90.57 %	93.21 %	0.0527
2	90.57 %	97.36 %	0.1985	90.57 %	97.36 %	0.2000	90.57 %	93.21 %	0.0539	90.57 %	93.21 %	0.0551
4	93.21 %	97.36 %	0.2014	93.71 %	97.36 %	0.2022	93.21 %	93.21 %	0.0562	93.21 %	93.21 %	0.0569

TABLE X
ERROR RATES FOR SVD-DID (YALEB)

Zeroed Coefficients	Luminosity +0						Luminosity +100					
	$\alpha = 0.5$			$\alpha = 0.8$			$\alpha = 0.5$			$\alpha = 0.8$		
	KNN	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**	**KNN**	**NBC**	**mMSE**
1	93.53 %	97.34 %	2.5675 e-4	94.85 %	97.34 %	2.6036 e-4	97.01 %	97.18 %	1.2868 e-4	97.34 %	97.51 %	1.3220 e-4
2	95.85 %	97.34 %	2.6490 e-4	95.93 %	97.34 %	2.6653 e-4	97.34 %	97.51 %	1.3655 e-4	97.34 %	97.51 %	1.3814 e-4
4	96.68 %	97.34 %	2.6912 e-4	96.76 %	97.34 %	2.6972 e-4	97.34 %	97.51 %	1.4075 e-4	97.34 %	97.51 %	1.4131 e-4

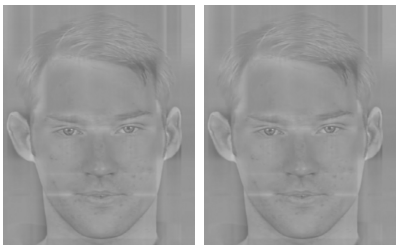

Fig. 12. Left: Result for SVD-DID for N=2, luminosity +150, α=0.5, Right: Result for SVD-DID for N=2, luminosity +150, α=0.8 ($r = 10$ and d = 0.5)

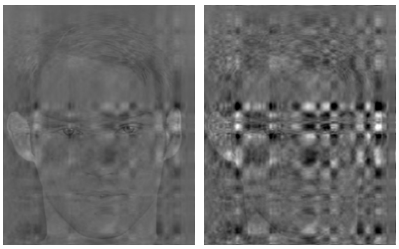

Fig. 13. Left: Result for SVD-DID for N=1, luminosity +100, α=0.8, $r = 10$ and d = 5 Right: Result for SVD-DID for N=1, luminosity +100, α=0.8, $r = 10$ and d = 10

DID method. The SVD-DID method with STD adjustment will be referred to as SVD-SDID.

The SVD-SDID method increases the error rates given by the KNN and NBC classifiers as is shown in Table XI where the SVD-DID parameters have the following values: $\alpha = 0.5, d = 0.5, r = 10, lum = +100$ and SVD-SDID parameter γ is $\gamma = 0.8$. These error rates are for applying Equation 8 to matrix $\mathbf{U}$ of the SVD. Applying to both $\mathbf{U}$ and $\mathbf{V}$ matrices gives poorer visual results without increasing the error rates. Applying solely to matrix $\mathbf{V}$ leads to similar results as when applied to matrix $\mathbf{U}$.

From the results displayed in Table XI it can been seen that applying the SVD-SDID method results in higher error rates for the classifiers used. In the case of the KNN classifier and the XM2VTS database, for $N = 0$ there is a 0.38% increase in error rate. When $N = 1$ the increase is greater reaching 10.95%. This is a major increase in error rate without any further image quality degradation, as can be visually confirmed by the images in Figure 14 and through the mMSE which displays only minor increases. In the case of the YaleB database, the results are shown in Table XII, the differences between error rates where not as notable with a 1.16% increase in error rate for $N = 0$ for the KNN classifier and for $N = 1$ a 0.33% increase for the same classifier. The error rates remained the same for all but one case for the NBC classifier with an increase of 0.33% for $N = 1$. For both databases the mMSE generally increased with a decrease only in the case of the XM2VTS database for $N = 0$.

6) The Effect of Parameter γ: Parameter γ is the new parameter introduced in the SVD-SDID method. In this section the effect of the parameter value will be examined concerning error rates and mMSE. As above the SVD-DID parameters will have the values: $N = 1\alpha = 0.5, d = 0.5, r = 10, lum = +100$ and for different values of γ the visual results are displayed in Figure 15 and Figure 16. Error rates for the two classifiers and the mMSE are displayed in Table XIII.

From Figures 15 and 16 not is visible that by decreasing parameter γ ever more visual information is lost as the value

TABLE XI
ERROR RATE COMPARISON FOR SVD-DID AND SVD-SDID (XM2VTS)

Zeroed Coefficients	KNN		NBC		mMSE	
	SVD-DID	SVD-SDID	SVD-DID	SVD-SDID	SVD-DID	SVD-SDID
0	52.83 %	53.21 %	68.30 %	75.09 %	0.0507	0.0459
1	76.60 %	87.55 %	93.21 %	93.21 %	0.0508	0.0520
2	90.57 %	90.57 %	93.21 %	93.21 %	0.0539	0.0546
4	93.21 %	93.21 %	93.21 %	93.21 %	0.0562	0.0566

TABLE XII
ERROR RATE COMPARISON FOR SVD-DID AND SVD-SDID (YALEB)

Zeroed Coefficients	KNN		NBC		mMSE	
	SVD-DID	SVD-SDID	SVD-DID	SVD-SDID	SVD-DID	SVD-SDID
0	89.21 %	90.37 %	97.51 %	97.51 %	1.5260 e-4	6.4498 e-4
1	97.01 %	97.34 %	97.18 %	97.51 %	1.2868 e-4	6.3664 e-4
2	97.34 %	97.34 %	97.51 %	97.51 %	1.3655 e-4	6.6453 e-4
4	97.34 %	97.34 %	97.51 %	97.51 %	1.4075 e-4	6.7919 e-4

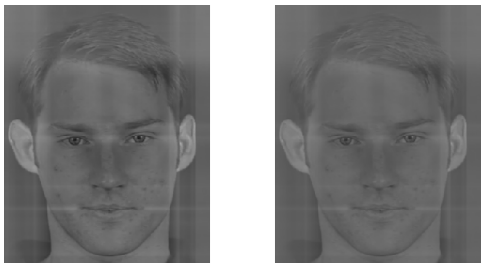

Fig. 14. Images de-identified using Left: the SVD-DID method, Right: the SVD-SDID method

TABLE XIII
ERROR RATES FOR SVD-SDID (XM2VTS)

Param. α	KNN (K=3)	Naive Bayes	mMSE
γ = 1.0	76.60 %	93.21 %	0.0508
γ = 0.8	87.55 %	93.21 %	0.0520
γ = 0.7	90.56 %	93.21 %	0.0526
γ = 0.5	90.56 %	93.21 %	0.0542
γ = 0.3	93.21 %	93.21 %	0.0559
γ = 0.2	93.21 %	93.21 %	0.0569

(a) (b) (c)

Fig. 15. SVD-SDID with (a) $\gamma = 1.0$, (b) $\gamma = 0.8$, (c) $\gamma = 0.7$

(a) (b) (c)

Fig. 16. SVD-SDID with (a) $\gamma = 0.5$, (b) $\gamma = 0.3$, (c) $\gamma = 0.2$

of STD, that differentiates subjects from one another, also decreases. The value of 0.8 is preferred since it provides a good increase in error rate and low image quality deterioration. From Table XIII the error rate increases as parameter γ decreases for the KNN classifier but levels off at $\gamma = 0.3$. The NBC classifier does not display any increase for the parameters selected. The mMSE displays a steady increase as parameter γ decreases, which is consistent with the increased error rates displayed by the classifiers.

As can be concluded from the above discussion, this extension of the SVD-DID method increases error rates in several cases and also retains visual quality, thus making this method more effective in protecting the privacy of individuals while retaining an acceptable image quality.

Having fully analyzed the results give by the SVD-DID method and used the initial results to extend this method it is time to move on the results for the Projection-DID method which are presented in the following section.

C. Results for Projection-DID

1) Results for the PDID-O Method: This method uses formula 19 to de-identify the input images. The radius used for the PDID-O was calculated using the SVDD method. For the XM2VTS dataset the calculated radius was $R = 67.4034$ and for the Yale B dataset the value for radius R was calculated to be $R = 17.4241$.

In order to test the above radii in respect to error rates and visual quality, other values where also used in the experimental process. For the XM2VTS dataset Table XIV summarizes the results for different radii and classifiers. As it can be seen more values where selected near the calculated radius in order to assess the effectiveness of the calculated radius. Visual results can be seen in Figure 17 and Figure 18.

For the XM2VTS dataset the results are presented in Table XIV from which we can conclude that parameter R plays a large role in the error rates that are displayed by the error

rates, as well as the mMSE. As suspected increasing radius R reduces the error rates displayed by the classifiers. For a radius of 10 very high error rates are observed reaching 97.36% for the NBC classifier and with an mMSE of 0.06046. Increasing the radius leads to a decline of the mMSE while error rates remain almost the same for a radius $R = 30$ and slightly falling by about 3% for radii $R = 50$ and 70. For a radius with a value of $R = 100$ error rates fall sharply to 49.06% for the KNN classifier and for $R = 120$ the same error rate is 26.04%. The mMSE is also reduced from 0.06046 for $R = 10$, to 0.02829 for $R = 70$ and reaches 0.01216 for a radius $R = 120$.Focusing on the values near the calculated value of $R = 67.4034$ and more specifically from 50 to 80 it can be observed that although the mMSE varies, the error rates remain stable for all three classifiers. The error rate is 90.57% for the KNN and NC classifiers, while slightly higher for the NBC classifier at 93.58%, both being high enough to offer privacy protection. From the results in Table XIV we can conclude that the calculated radius R by the SVDD method is a really good choice for de-identifying facial images and retaining an acceptable level of quality for this dataset and the PDID-O method. From these results, we propose the value of 70 for radius R for the XM2VTS dataset since $R = 70$ provides high error rates and acceptable image quality. Finally it can be verified from the results that increasing radius R causes a decline in error rates for all classifiers also for the mMSE, as we approach the initial image by increasing the radius R of the hypersphere.

TABLE XIV
ERROR RATES FOR PDID-O (XM2VTS)

	Classifiers		mMSE
Radius	KNN	NBC	
10	93.21 %	97.36 %	0.06046
30	93.21 %	93.58 %	0.04818
50	90.57 %	93.58 %	0.03746
60	90.57 %	93.58 %	0.03268
67.4034	90.57 %	93.58 %	0.02939
70	90.57 %	93.58 %	0.02829
80	90.57 %	93.58 %	0.02428
100	49.06 %	61.89 %	0.01745
120	26.04 %	54.72 %	0.01216

TABLE XV
ERROR RATES FOR PDID-O (YALE B)

	Classifiers		mMSE
Radius	KNN	NBC	
5	94.94 %	92.94 %	0.04760
10	89.96 %	72.61 %	0.02878
15	60.83 %	82.57 %	0.02038
17.4241	48.30 %	86.14 %	0.02005
20	38.67 %	89.38 %	0.02239

Fig. 17. Results for PDID-O with Left: $R = 10$, Middle: $R = 30$, Right: $R = 50$

Fig. 18. Results for PDID-O with Left: $R = 70$, Middle: $R = 100$, Right: $R = 120$

For the Yale B dataset the radius R that was calculated using the SVDD method has the value $R = 17.4241$. For this R and radii in the same area, the error rates are shown in Table XV. As can be seen for a small radius $R = 10$, error rates for all classifiers are high. Increasing the radius leads to low error rates for the KNN classifier, while the NBC and NC classifiers display high error rates. This observation mean that the radius that is computed using the SVDD method is a good estimate of the radius that should be used in order to de-identify the images sufficiently. For the selected radii the mMSE displays at first a decline from $R = 10$ to $R = 17.4241$ and then increases. In this case the estimate by the SVDD method is not ideal and a smaller radius should be used to attain high de-identification rates. As such we propose a value of $R = 10$ for the Yale B dataset.

In both datasets apart from simply using the original images the LDA method was applied. The results gave varying error rates that where either slightly higher than the ones with the original images and some where lower. In the case of the XM2VTS dataset the images where resized to 40×30. In this case the radius R calculated with the SVDD method was $R = 0.9819$. For this radius the NBC and NC classifiers gave the same error rates at with the original images and the ones with LDA giving 96.23% and 93.21% respectively. The KNN classifier showed error rates at 93.21% for the initial images and 91.32% for the LDA. For the Yale B dataset and a radius of $R = 10$ the NC classifier displays the same error rates at 79.17%. In the case of the NBC classifier the error rate increases if LDA is used from 72.61% to 87.14%. Finally for the KNN classifier there is a drop from 89.96% to 79.50% which is still an acceptable de-identification rate.

2) Results for the PDID-M Method: This method projects the input image on a hypersphere centered on the mean image using formula 21. The radius calculated using the SVDD method did not provide adequate de-identification with the PDID-M method and the radii used here found empirically. For the XM2VTS dataset the radius proposed is $R = 10$ and for the Yale B dataset $R = 2$. This is a drawback of this method,

Fig. 19. Results for PDID-M with Left: $R = 4$, Middle: $R = 6$, Right: $R = 8$

Fig. 20. Results for PDID-M with Left: $R = 10$, Middle: $R = 12$, Right: $R = 14$

since the radii cannot be calculated automatically. Error rates for the XM2VTS dataset can be are displayed in Table XVI and visual results can be seen in Figure 19 and Figure 20. From the results in Table XVI it can be seen that the PDID-M method gives high error rates with lower mMSE compared to the PDID-O method. From a $R = 4$ with error rates at 96.23% for all classifiers a slight drop is displayed up to a radius of $R = 10$ for which value the error rates are 90.19% for the three classifiers used. Beyond this value the error rates drop sharply and for a radius of $R = 14$ the KNN classifier displays an error rate of 53.21%.

The error rates for the Yale B dataset are displayed in Table XVII. For a radius $R = 1$ the KNN classifier displays an error rate at 96.21% while the NBC a much lower error rate at 88.13%. For $R = 2$ both the previous classifiers drop to 95.02% and 83.32% respectively. The NC also displays a drop in error rate from 92.61% for a radius of $R = 1$ to 89.21% for $R = 2$. The mMSE is at 0.04384 for $R = 1$ and for $R = 2$ the mMSE value drops to 0.03307. The values for the mMSE in the case of the Yale B dataset are close for both the PDID-O and PDID-M method, unlike the case of the XM2VTS dataset as mentioned above. For higher values for radius R all error rates drop below 90%. For $R = 3$ the KNN and NC classifiers display a difference of 1% at 88.71% and 89.71% respectively, while the NBC remains almost stable in comparison with a radius $R = 2$ at 83.14% and the mMSe dropping to 0.02396. For values beyond $R = 3$ error rates drop sharply with a minimum of 76.51% for $R = 4$ and to a minimum of 66.14% for $R = 66.14\%$ both displayed by the KNN classifier.

As in the PDID-O method the LDA method was applied to the initial images. The results gave varying error rates that where either slightly higher than the ones with the original images and some where lower. As mentioned above the XM2VTS dataset images where resized to 40×30. In this case the radius used was $R = 0.8$. For this radius the NBC and NC classifiers displayed equal error rates for the original images and the ones with LDA giving 96.23% and 90.19% respectively. The KNN classifier displayed error rates at 90.19% for the initial images and 96.60% for the LDA. In the case of the Yale B dataset a radius of $R = 2$ was used. The NC classifier displays the same error rates at 85.89%. Error rates of the NBC classifier the error rate increases with LDA from 82.49% to 89.79%. Finally for the KNN classifier error rates from 94.52% to 89.21%.

TABLE XVI
ERROR RATES FOR PDID-M (XM2VTS)

	Classifiers		mMSE
Radius	KNN	NBC	
4	96.23 %	96.23 %	0.01954
6	90.19 %	96.23 %	0.01804
8	90.19 %	90.19 %	0.01660
10	90.19 %	90.19 %	0.01522
12	66.04 %	90.19 %	0.01390
14	53.21 %	73.58 %	0.01265

IV. METHOD COMPARISON

In this section the methods that where analyzed in the main part of the article will be compared. Firstly we will take a look at the highest error rates attained by each method as they are presented in the tables in each section. These error rates provide a high level of privacy but at the same time degrade the output image, which may not be acceptable for viewers. The highest error rates attained are presented in Tables XVIIIand Table XVIII. From the previous tables it can be seen that very high error rates can be achieved with all methods with the correct selection of each method's parameters. It may seem that the SVD-SDID method does not achieve higher error rates in comparison with the SVD-DID method, but as it is discussed in an above section the SVD-SDID method increases the error rates for specific parameters, with an increase of over 10% in one case. In the case of the Projection-DID methods, the PDID-M method achieves an error rate of 96.23 % for both classifiers while the PDID-O method achieves a higher error rate in the case of the NBC classifier. The highest error rate however heavily degrade the final quality of the output image which is not acceptable. As such we will now take a look at the error rates of each method and classifier for the

TABLE XVII
ERROR RATES FOR PDID-M (YALE B)

	Classifiers		mMSE
Radius	KNN	NBC	
1	96.76 %	88.13 %	0.04384
2	95.02 %	83.32 %	0.03307
3	88.71 %	83.15 %	0.02396
4	76.51 %	81.74 %	0.01652
5	66.14 %	81.41 %	0.01075

TABLE XVIII
HIGHEST ERROR RATES FOR THE DE-IDENTIFICATION METHODS (XM2VTS)

Method	KNN	NBC
SVD-DID	93.71 %	97.36 %
SVD-SDID	93.21 %	93.21 %
PDID-O	93.21 %	97.36 %
PDID-M	96.23 %	96.23 %

TABLE XIX
HIGHEST ERROR RATES FOR THE DE-IDENTIFICATION METHODS (YALEB)

Method	KNN	NBC
SVD-DID	97.34 %	97.51 %
SVD-SDID	97.34 %	97.51 %
PDID-O	94.94 %	92.94 %
PDID-M	96.76 %	88.13 %

TABLE XX
RECOMMENDED PARAMETER ERROR RATES FOR THE DE-IDENTIFICATION METHODS (XM2VTS)

Method	KNN	NBC
SVD-DID	90.57 %	93.21 %
SVD-SDID	90.57 %	93.21 %
PDID-O	90.57 %	93.58 %
PDID-M	90.19 %	90.19 %

TABLE XXI
RECOMMENDED PARAMETER ERROR RATES FOR THE DE-IDENTIFICATION METHODS (YALEB)

Method	KNN	NBC
SVD-DID	97.34 %	97.51 %
SVD-SDID	97.34 %	97.51 %
PDID-O	89.96 %	72.61 %
PDID-M	95.02 %	93.32 %

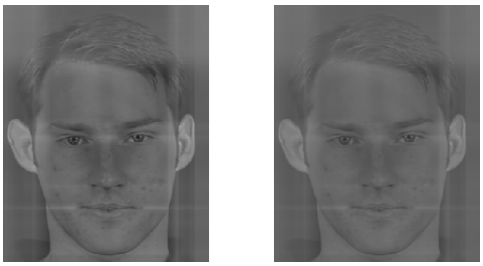

Fig. 21. Images de-identified using Left: the SVD-DID method, Right: the SVD-SDID method

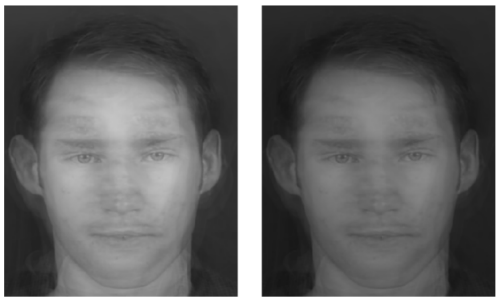

Fig. 22. Images de-identified using Left: the PDID-M method, Right: the PDID-O method

recommended values of the parameters in each method.

Beginning with the SVD-DID and SVD-SDID methods the recommended parameters are $N = 2$, $r = 10$, $d = 0.5$ and $\alpha = 0.5$. In the case of the PDID-O method the radius has a value of $R = 70$ in the XM2VTS database and $R = 10$ in the YaleB database. Finally for the PDID-M method $R = 10$ for XM2VTS and $R = 2$ for the YaleB database. These parameter values provide an acceptable visual result and at the same time a high level of privacy protection. The error rates for these parameter values are presented in Table XX and Table XXI and the visual results can be compared in Figure 21 and Figure 22. From these results it is evident for the recommended parameters high levels of privacy can be attained. In the case of the XM2VTS database in Table XX there is a small variation in the error rates, all of which are above 90%. This does not apply in the case of the YaleB database where there is greater variation between the error rates. In Table XXI the error rates vary from 72.61% for the PDID-O method and the NBC classifier to 97.51% For the two SVD-DID methods and the same classification algorithm.

From the above discussion it can be inferred that both approaches in facial image de-identification can provide a high level of privacy. Visually from Figures 21 and 22 it can be seen that in all cases artifacts are introduced in the output images, while there is also a variation in the luminosity of each pixel. The visual artifacts are introduced due to the filtering steps for the SVD-DID methods and especially during the SVD-MSF step where all parts of the image are sharpened. In the case of the Projection-DID methods artifacts are introduced in the averaging with the mean image in the case of PDID-O method and in the PDID-M method from the center of the hypersphere on which the input images are projected. The drop in luminosity is caused from the reduction of the highest singular values to zero in step SVD-CZ for the SVD-DID methods and from the projection in the Projection-DID methods. These effects can be observed in both figures, where the left image appears to be brighter that the one on the right and perhaps slightly more clear due to the differences between the methods. Depending on whether privacy is vital or not and whether the aim is to preserve the majority of the visual data the right combination of parameter values can be selected. This means that in order to achieve high error rates from the classifiers a compromise must be made in the image quality which will suffer. If privacy is not a concern a higher image quality can be attained leading to lower error rates.

V. Conclusions

In this article we have described and analyzed two methods for de-identifying facial images. These methods aim to limit the effectiveness of face identification methods, while retaining part of the initial visual quality. From the results above, it can be deducted that using the appropriate parameter values, a high level of privacy can be attained. For the SVD-DID method in the case of the YaleB database, the highest error rate achieved was 97.51% and the highest error rate for the XM2VTS database was 93.71%. Despite the high error rate, the end product of these methods can be characterized as acceptable for everyday use. This method, when applied to the initial images, tend to have a smoothing effect on the image, while introducing visual artifacts. Also, by applying the various methods and filters there exists the tendency to darken the image, which is counterbalanced, by adding a constant value to the output image, in order to preserve adequate visual information so that the faces can be identified by human viewers. The combination of these effects reduces the identification accuracy of automatic face identity classifiers. From the error rates and visual results we can conclude that the proposed SVD-DID method serves the purpose of protecting privacy and providing a visually acceptable output.

The second method that is based on projections on hyperspheres a good radius R for the PDID-O method was calculated using the SVDD method was used. The radii given by the SVDD gave radii values that provided high error rates and at the same time acceptable image quality. Error rates where high, attaining 93.58% for the XM2VTS dataset using the Naive Bayes Classifier and the radius $R = 67.4034$. For the Yale B dataset the highest error rate was 92.12% with the Nearest Centroid Classifier and a radius $R = 17.4241$. For the PDID-M method, the radii given by the SVDD did not provide adequate de-identification so the values where selected empirically. The highest error rates with the proposed radii where 90.19% for $R = 10$ for the XM2VTS dataset and 95.02% for $R = 2$ for the Yale B dataset. Comparing the two proposed methods it can be seen that the PDID-M method performs better compared to the PDID-O method. For similar values of mMSE (about 0.012) the minimum error rate is 26.04% for the PDID-O method and 53.21% for the PDID-M method which is more than double the error rate for PDID-O.

To summarize, from the above results it can be concluded that the SVD-DID and Projection-DID methods serve the purpose of providing privacy protection by attaining high error rates from classifiers and providing an end image that can be characterized as acceptable for everyday use.

References

[1] Pr. Agrawal, P. J. Narayanan, "Person De-identification in Videos", *in IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, 2005, pp. 299-310 .

[2] S. Tansuriyavong, S. Hanaki, "Privacy protection by concealing persons in circumstantial video image", *in Proceedings of the 2001 workshop on Perceptive user interfaces* , 2001, pp. 1-4 .

[3] R. Gross, L. Sweeney, F. de la Torre, S. Baker, "Semi-Supervised Learning of Multi-Factor Models for Face De-Identification", *in IEEE Conference on Computer Vision and Pattern Recognition, 2008, CVPR 2008*, 2008, pp. 1-8 .

[4] V. Blanz, S. Romdhani, T. Vetter, Face Identification across Different Poses and Illuminations with a 3D Morphable Model, *in Fifth IEEE International Conference on Automatic Face and Gesture Recognition, 2002. Proceedings.* ,2002, pp. 192-197

[5] Singular Value Decomposition (SVD) tutorial MIT BE.400 / 7.548, Perspectives in Biological Engineering Course (http://web.mit.edu/be.400/www/SVD/Singular_Value_Decomposition.htm)

[6] Singular Value Decomposition (SVD), Department of Computer Science & Engineering University of Nevada, CS4/791Y: Mathematical Methods for Computer Vision, Dr. George Bebis (http://www.cse.unr.edu/~bebis/MathMethods/SVD/lecture.pdf)

[7] G. H. Golub, C. F. Van Loan, Matrix Computations pg. 76 - 81, Fourth Edition, , The Johns Hopkins University Press, Baltimore, 2012, ISBN 13: 978- 1-4214-0794-4

[8] E.Newton, L.Sweeney and B.Mali, "Preserving Privacy by De-identifying Facial Images", *in IEEE Transactions on Knowledge and Data Engineering*, 2005, pp. 232-243.

[9] R. Gross, L. Sweeney, J. Cohn, F. de la Torre and S. Baker, "Face De-Identification", *in Protecting Privacy in Video Surveillance*, 2009, pp. 129-146.

[10] L. Sweeney, "k-anonymity: a model for protecting privacy", International Journal on Uncertainty,Fuzziness and Knowledge-based Systems, 10 (5), 2002, pp. 557-570.

[11] S. Mosaddegh, L. Simon and F. Jurie, "Photorealistic Face de-Identi cation by Aggregating Donors' Face Components". *in Asian Conference on Computer Vision*, 2014, Singapore, pp.1-16

[12] B. Driessen and M. Drmuth, "Achieving Anonymity Against Major Face Recognition Algorithms", *in Cryptology ePrint Archive*, 2013.

[13] P.J. Phillips, "Privacy Operating Characteristic for Privacy Protection in Surveillance Applications", *in Audio- and Video-Based Biometric Person Authentication*, 2013, pp. 869-878.

[14] D.M.Y. Sommerville, *An Introduction to the Geometry of n Dimensions*, Methuen, Dover, New York; 1958, pp. 135-137.

[15] E.W. Weisstein, "Hypersphere", *MathWorld, A Wolfram Web Resource*, 2014, http://mathworld.wolfram.com/Hypersphere.html.

[16] E.W. Weisstein, "Ball", *MathWorld, A Wolfram Web Resource*, 2014, http://mathworld.wolfram.com/Ball.html.

[17] S. Theodoridis, K. Slavakis and I. Yamada, "Adaptive Learning in a World of Projections", *in IEEE Signal Processing Magazine*, 2011, pp. 97-123.

[18] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, England; 2004, pp. 244.

[19] D.M.J. Tax and R.P.W. Duin, "Support Vector Data Description", *in Machine Learning*, vol. 54, 2004, pp 45-66.

[20] A. Georghiades, P. Belhumeur and D. Kriegman's, "From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose", *in PAMI*, 2001.

[21] K. Messer, J. Matas, J. Kittler, J. Luettin and G. Maitre, "XM2VTSbd: The Extended M2VTS Database", *in Proceedings 2nd Conference on Audio and Video-base Biometric Personal Verification (AVBPA99)* Springer Verlag, New York, 1999.