Model-Based Face De-Identification

Ralph Gross, Latanya Sweeney
Data Privacy Lab, School of Computer Science
Carnegie Mellon University, USA
rgross@cs.cmu.edu, latanya@privacy.cs.cmu.edu

Fernando de la Torre, Simon Baker
Robotics Institute
Carnegie Mellon University, USA
{ftorre, simonb}@cs.cmu.edu

Abstract

Advances in camera and computing equipment hardware in recent years have made it increasingly simple to capture and store extensive amounts of video data. This, among other things, creates ample opportunities for the sharing of video sequences. In order to protect the privacy of subjects visible in the scene, automated methods to de-identify the images, particularly the face region, are necessary. So far the majority of privacy protection schemes currently used in practice rely on ad-hoc methods such as pixelation or blurring of the face. In this paper we show in extensive experiments that pixelation and blurring offers very poor privacy protection while significantly distorting the data. We then introduce a novel framework for de-identifying facial images. Our algorithm combines a model-based face image parameterization with a formal privacy protection model. In experiments on two large-scale data sets we demonstrate privacy protection and preservation of data utility.

1. Introduction

Due to the continuously falling costs of video capture equipment, it is becoming possible to record, store and process large quantities of video data. As a consequence, an increasing number of research projects aim at continuously observing and monitoring people in private spaces. The Caremedia project at CMU for example captures and analyzes video data recorded in a nursing home facility to support medical personnel in diagnosing and treating behavioral problems of the elderly [5]. The Aware Home project at Georgia Tech equipped a house with an extensive sensor network (including video cameras) with a similar goal of monitoring elderly people [1]. Privacy concerns of non-consenting subjects however limit the abilities of researchers to exchange raw data and often require labor intensive manual post-processing to remove portions of the data. These are examples of a growing number of applications in which valuable video data can not be shared due to fear of re-identification. Out of this situation the need for automatic methods to remove identifying information from images, particulary the face region, arises. The goal is to remove as much identifying information as necessary while preserving as much of the data utility as possible.

Previous work on de-identifying facial images falls in one of two categories: ad-hoc methods such as "blurring" or "pixelation" [20] or formal methods such as k-Same [21] or k-Same-Select [12]. Both types of approaches have shortcomings which we address in this paper. We first propose a new algorithm, k-Same-M, which combines a model-based face parameterization with a formal privacy protection model. We demonstrate that the algorithm achieves privacy protection similar to previously proposed methods, while producing de-identified images of much higher quality. We furthermore show that the proposed k-Same-M algorithm better preserves data utility than the previously proposed k-Same algorithm or ad hoc methods such as blur filtration. We furthermore describe two completely automatic algorithms to attack the protection provided by pixelation.

The remainder of this paper is organized as follows. In Section 2 we survey related work. Section 3 defines face de-identification along with other concepts used in the paper. Section 4 introduces model-based face de-identification and describes the k-Same-M algorithm. In Section 5 we evaluate the privacy protection afforded by the k-Same-M algorithm and examine the resulting data utility. Finally, Section 6 examines the popular algorithm of pixelation and shows its inadequacy for protecting privacy.

2. Related Work

While there is a rich body of work on privacy protection for field-structured data [2], specifically medical data [25], comparatively little has been done in the context of video surveillance. The majority of work on images or image sequences applies simple distortion methods such as "pixelation" (image subsampling) or "blurring" (smoothing the image with e.g. a Gaussian filter with large variance) to obfuscate parts or all of the image [7,17,20,27]. The PrivacyCam architecture proposed by Senior et al. [24] suppresses automatically segmented foreground objects in the scene and

Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)
0-7695-2646-2/06 $20.00 © 2006 IEEE

cryptographically secures access to the altered video stream produced by the system. While some of these techniques have been shown to reduce the capability of *human observers* to identify people or actions in the scene [7, 20, 27], no formal privacy guarantees are made.

In more recent work on face de-identification by Newton et al. [21] the k-Same algorithm is introduced, which offers formal privacy protection guarantees. k-Same is based on the k-anonymity framework introduced by Sweeney [25]. The algorithm guarantees that each de-identified face image could be representative of k faces in the gallery, therefore limiting face recognition performance to $1/k$. Unlike the methods discussed above, [21] also addresses the problem of adaptive recognition in which an adversery mimics the obfuscation technique employed by a privacy-protection algorithm. Newton et al. experimentally showed that ad-hoc methods fail to protect privacy in the case of (manually guided) adaptive recognition, whereas the k-Same algorithm offers protection guarantees even in this case.

In a different approach, Phillips [22] proposed an algorithm for privacy protection of facial images through reduction of the number of eigenvectors used in reconstructing images from basis vectors. A direct trade-off between privacy protection and data utility is established through the introduction of the privacy operating characteristic (POC), a plot similiar to a receiver operating characteristic (ROC) often used in pattern classifier design [11].

In order to overcome the trade-off between privacy protection and data utility the k-Same-Select algorithm was introduced as direct extension of the k-Same algorithm [12]. It was shown that the k-Same-Select algorithm preserves data utility (as measured by the accuracy of gender and facial expression classifiers) while guaranteeing privacy protection. However, the k-Same-Select algorithm, like *all* algorithms discussed above, is strictly appearance-based and works directly on the pixel level. As a consequence, commonly appearing mismatches in image alignment (the correspondence between pixels in images) lead to poor quality in de-identified images.

3. Face Recognition and Face De-Identification

3.1. Defiitions

In this section we provide definitions of all basic concepts related to face de-identification. See Figure 1 for an overview.

Definition 3.1 (Image Coding) *For a given grayscale or color input image $I \in \mathbb{R}^n$ we define image coding as function $f^c_{\mathcal{A}} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ which maps an n-dimensional input image to an m-dimensional representation using the auxiliary information $\mathcal{A}$.*

Note that grayscale images of dimension $h \times w$ are converted into vectors of dimension $n = h * w$ through rasterscanning. For color images we concatenate the vectors obtained by rasterscanning the different color channels, so that $n = h*w*3$. In this work we consider two different coding schemes:

Example 3.1 (Appearance-Based Coding) *Using the (typically manually established) location of at least three feature points (e.g. center of the eyes and tip of the nose) appearance-based coding geometrically normalizes the face so that the feature points are located in the same positions across images. This is typically done by estimating the parameters of an affine transform (accounting for translation, rotation and scale) between the current and target feature point locations and applying the transform to all pixels in the image. The images are then usually cropped to the same fixed dimensions.*

Example 3.2 (Model-Based Coding) *In model-based coding a previously learned generative model is fitted to the input image by changing the model parameters until the difference between the original image and the image reconstructed from the model is minimal. The model parameter vector is then used as encoding of the input image.*

In the following we assume that all face images are coded, either appearance-based or model-based. For notational ease we simply refer to them as images.

Definition 3.2 (Image Sets) *We distinguish a number of different sets of facial images. The* gallery *contains face images of individuals known to a face recognition algorithm: $\mathcal{G} = \{G_1, G_2, \ldots, G_l\}$. The* probe *set contains images of unknown people: $\mathcal{P} = \{P_{(1,1)}, P_{(1,2)}, \ldots, P_{(1,m_1)}, \ldots, P_{(n,1)}, P_{(n,2)}, \ldots, P_{(n,m_n)}\}$. The goal of face recognition is therefore the correct linking of images in the probe set to images in the gallery set. We furthermore define the generic face set $\mathcal{F} = \{F_{(1,1)}, \ldots, F_{(1,s_1)}, \ldots, F_{(r,s_r)}\}$, where we assume that $\mathcal{F} \cap \mathcal{G} = \mathcal{F} \cap \mathcal{P} = \emptyset$.*

This definition of image sets follows the standard established for the FERET evaluations [23].

Definition 3.3 (Face Recognition) *We define face recognition as function $\Phi : \mathbb{R}^m \times (\mathbb{R}^m)^u \rightarrow \{1, 2, \ldots, u\}$ which, given a probe image $p \in \mathcal{P}$ and a gallery $\mathcal{G}$, with $|\mathcal{G}| = u$ outputs the index of the subject most likely to correspond to the subject seen in the probe image: $\Phi(p, \mathcal{G}) = j, 1 \leq j \leq u$. By convention we extend Φ to apply as well to a probe set $\mathcal{P}$ with $|\mathcal{P}| = v$ as $\Phi : (\mathbb{R}^m)^v \times (\mathbb{R}^m)^u \rightarrow (\{1, 2, \ldots, u\})^v$.*

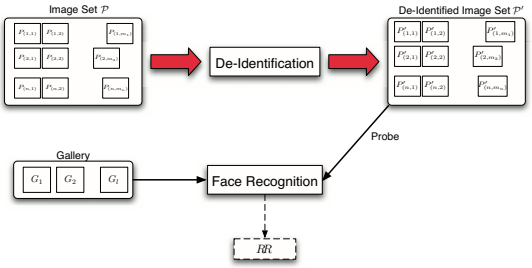


Figure 1. Overview of the de-identification framework. Using the de-identification function Φ, the image set $\mathcal{P}$ is mapped to the de-identified image set $\mathcal{P}'$. We evaluate the performance of the de-identification function Φ in terms of privacy protection afforded by measuring face recognition performance (*RR*) and in terms of preserved data quality.

We evaluate face recognition performance by computing cumulative match characteristics (CMC) for a given face recognition function Φ and gallery and probe sets $\mathcal{P}$ and $\mathcal{G}$. The CMC contains for each rank (position in the similarity ordering of the face recognition function output) the likelihood that the algorithm returns the correct answer at or below this rank [6]. The simplest way of reporting the performance of face recognition algorithms is to report rank 1 recognition rates.

Definition 3.4 (Face De-Identification) *We define face de-identification as a function* $\Psi_{\mathcal{A}} : \mathbb{R}^m \rightarrow \mathbb{R}^m$*, which associates each input image with a vector of equal dimensionality:* $\Psi_{\mathcal{A}}(p) = p', p, p' \in \mathbb{R}^m$*, using the auxiliary information* $\mathcal{A}$*.*

The implicit goal of a face de-identification method Ψ is to remove identifying information from face images, so that $CMC(\Phi, \Psi(\mathcal{P}), \mathcal{G}) < CMC(\Phi, \mathcal{P}, \mathcal{G})$.

One formal approach to privacy protection that has gained popularity in recent years is k-anonymity [25]. The idea underlying k-anonymity is to ensure that every data attribute that could be used to identify a particular user relates indiscriminatively to at least k elements in the dataset. Applied to sets of faces we therefore define (following [21]):

Definition 3.5 (k-Anonymized Probe Set) *We call a de-identified probe set of face images* k-anonymized*, if for every probe image there exist at least* k *images in the gallery to which the probe image corresponds.*

Maximum privacy protection is trivially achieved by suppressing the data, e.g. by setting each pixel value in an image to 0. We strive to guarantee privacy protection while preserving as much of the original signal as possible.

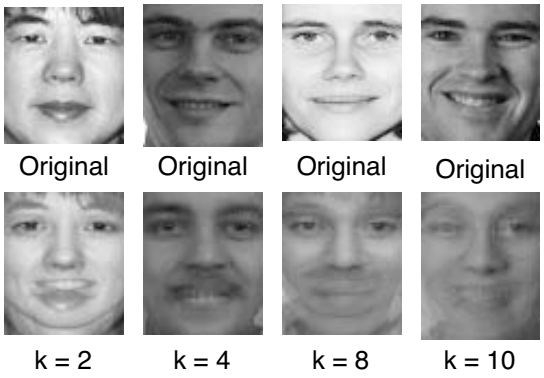


Figure 2. Faces de-identified using the k-Same algorithm. Due to the appearance-based nature of the algorithm errors in image alignment lead to poor image quality in the de-identified images.

4. Model-Based De-Identification

This section introduces the proposed algorithm, k-Same-M. We motivate the work by demonstrating significant shortcomings of the previously proposed k-Same algorithm [21]. We furthermore give a brief introduction to Active Appearance Models.

4.1. Shortcomings of Appearance-Based Methods

The algorithms introduced in [21] and [12] are both appearance based, operating entirely in the image space. Since both algorithms compute averages of images, artifacts due to misalignments of the images involved are inevitable, even when images are aligned based on a small number of feature points (e.g. the eyes and the tip of the nose). This leads to a reduction of data utility at higher levels of privacy protection (see Section 5). As shown in Figure 2 strong "ghosting" artifacts are visible.

4.2. Background: Active Appearance Models

Active Appearance Models (AAMs) are generative parametric models that have been used successfully for modelling and tracking faces [9, 19]. In the following we give a brief description on constructing and fitting AAMs.

4.2.1 Definition and Model Construction

The *2D shape* of an AAM is defined by a 2D triangulated mesh and in particular the vertex locations of the mesh. Mathematically, the shape $\mathbf{s}$ of an AAM is defined as the 2D coordinates of the n vertices that make up the mesh: $\mathbf{s} = (x_1, y_1, x_2, y_2, \ldots, x_n, y_n)^{\mathrm{T}}$. AAMs allow linear shape variation. This means that the shape matrix $\mathbf{s}$ can be expressed as a base shape $\mathbf{s}_0$ plus a linear combination

of m shape matrices $\mathbf{s}_i$:

$$\mathbf{s} = \mathbf{s}_0 + \sum_{i=1}^{m} p_i \, \mathbf{s}_i \tag{1}$$

where the coefficients p_i are the shape parameters. AAMs are computed from training data consisting of a set of images with the shape mesh hand marked on them [9]. After geometrically aligning the training shapes using the *Procrustes* algorithm [10], Principal Component Analysis (PCA) [18] is applied to the aligned training meshes. The base shape $\mathbf{s}_0$ is the mean shape and the matrices $\mathbf{s}_i$ are the (reshaped) eigenvectors corresponding to the m largest eigenvalues.

The *appearance* of the AAM is defined within the base mesh $\mathbf{s}_0$. Let $\mathbf{s}_0$ also denote the set of pixels $\mathbf{u} = (u, v)^{\mathrm{T}}$ that lie inside the base mesh $\mathbf{s}_0$. The appearance of the AAM is then an image $A(\mathbf{u})$ defined over the pixels $\mathbf{u} \in \mathbf{s}_0$. AAMs allow linear appearance variation. This means that the appearance $A(\mathbf{u})$ can be expressed as a base appearance $A_0(\mathbf{u})$ plus a linear combination of l appearance images $A_i(\mathbf{u})$:

$$A(\mathbf{u}) = A_0(\mathbf{u}) + \sum_{i=1}^{l} \lambda_i \, A_i(\mathbf{u}) \tag{2}$$

where the coefficients λ_i are the appearance parameters. The appearance images A_i are usually computed by applying PCA to the shape normalized training images [9, 19]. We furthermore define the vector $c = \{p_1, \ldots, p_n, \lambda_1, \ldots, \lambda_m\}$ as concatenation of the shape parameters p_i and the appearance parameters λ_i.

4.2.2 Model Fitting

Fitting a AAM is usually formulated [19] as minimizing the sum of squares difference between the model instance $A(\mathbf{x}) = A_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i A_i(\mathbf{x})$ and the input image warped back onto the base mesh $I(\mathbf{W}(\mathbf{x}; \mathbf{p}))$:

$$\sum_{\mathbf{x} \in \mathbf{s}_0} \left[A_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i A_i(\mathbf{x}) - I(\mathbf{W}(\mathbf{x}; \mathbf{p})) \right]^2 \tag{3}$$

where the sum is performed over all of the pixels $\mathbf{x}$ in the base mesh $\mathbf{s}_0$. A number of fitting algorithms have been proposed for the minimization of the expression in Eqn. 3 ranging from the very efficient project-out algorithm [19] and the less efficient but more accurate simultaneous inverse compositional algorithm [3] to efficient *robust* fitting algorithms capable of dealing with occlusion [13].

4.3. Model-Based k-Same: k-Same-M

For subject images within the modelling space of a given AAM the model is able to describe the image with high accuracy, meaning that the face image reconstructed from the model parameters is very close in appearance to the original face image. Coupled with the generative nature of the AAM it is intuitive to perform face de-identification in the space of the *model parameters* instead of the image space. We therefore extend the previously proposed k-Same algorithm [21] to the k-Same-M algorithm by performing k-Same on AAM model parameter vectors. Intuitively, k-Same-M works by computing the average of k AAM parameter vectors computed from a set of faces and replacing the vectors with the average vector. See algorithm box 4.1 for a definition of k-Same-M. It has been shown previously [21] that image sets de-identified with the k-Same algorithm are k-anonymized. The same holds for parameter sets de-identified using the k-Same-M algorithm.

input : Face set $\mathcal{M}_o$, privacy constant k, with $|\mathcal{M}_o| \geq k$, Active Appearance Model $\mathcal{A}$
output: De-identified face set $\mathcal{M}_d$

$\mathcal{M}_d \leftarrow \emptyset$
$\mathcal{M}'_o \leftarrow \emptyset$
for $i \in \mathcal{M}_o$ **do**
  Compute parameter representation c_i of i with respect to AAM $\mathcal{A}$ and add to $\mathcal{M}'_o$
end
for $c \in \mathcal{M}'_o$ **do**
  if $|\mathcal{M}'_o| < 2k$ **then**
    $k = |\mathcal{M}'_o|$
  end
  Select the k vectors $\{c_1, \ldots, c_k\} \in \mathcal{M}'_o$ that are closest to c according to L_2 norm.
  $avg \leftarrow \frac{1}{k} \sum_{m=1}^{k} c_m$
  Add k copies of avg to $\mathcal{M}_d$
  Remove $c_1, \ldots, c_k$ from $\mathcal{M}'_o$
end

Algorithm 4.1: k-Same-M Algorithm.

Figure 3 shows example images of faces de-identified using k-Same-M. The images are of visually higher quality in comparison to the images shown in Figure 2. Notice that no "ghosting" artifacts are present.

5. Experiments

To evaluate the privacy protection afforded by the proposed k-Same-M algorithm we conducted recognition experiments using Principal Component Analysis [26].

5.1. Dataset

We used a subset of the CMU Multi-PIE face database [14] containing 249 subjects displaying a neutral and a smile expression. The images were captured within minutes of each other as part of a multi-camera, multi-flash recording. In the experiments here only images with frontal pose

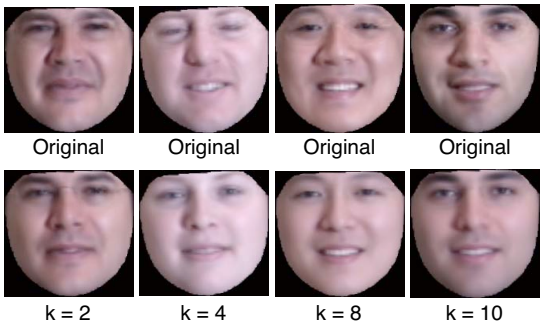


Figure 3. Face images de-identified using the proposed k-Same-M algorithm.

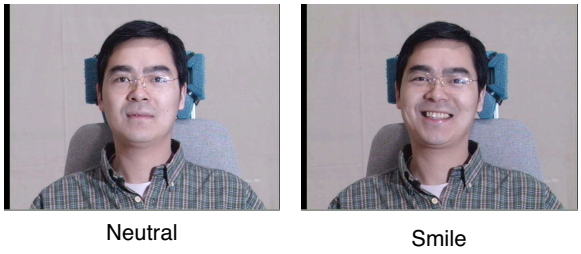


Figure 4. Example images from the dataset used in experiments for the k-Same-M algorithm.

and frontal illumination were used. AAM ground-truth was manually established for all 498 face images. See Figure 4 for example images.

5.2. Evaluation of Privacy Protection

We divided the subject pool into non-overlapping sets for training and gallery/probe, following the experimental setup established in the FERET evaluations [23]. We used 30% of the subjects for training of the PCA eigenspace with original, unaltered images. For the remaining subjects, the neutral expression images were used as gallery and the smile images as probe. All results reported are based on randomly selecting five different subject assignments into training and gallery/probe sets and computing the average recognition rate over all configurations.

Figure 5 shows rank-1 accuracies and CMC curves for face images de-identified for the different variants of the k-Same algorithm. The recognition rates stay below the theoretical maximum of $1/k$ [21] (see Figure 5(a)). The recognition accuracy of k-Same is slightly below the accuracy of k-Same-M, potentially due to the noise introduced by this algorithm. Figure 5(b) shows CMC curves for the k-Same-M algorithm for different levels of k. Performance declines steadily with increasing k, illustrating the amount of de-identification necessary to achieve a given privacy goal.

5.3. Evaluation of Data Utility

Depending on the application, many different measures of data utility are possible, ranging from simple distance metrics to the evaluation of the presence of certain image features determined by specific classifiers. Following [12] we evaluated data utility through facial expression classification. We say that more data utility is preserved if a facial expression classifier performs better on a certain image set. In experiments we employed a Support Vector Machine classifier with Radial Basis Kernel, implemented using LIBSMV [8]. We performed 5-fold cross-validation by partitioning the dataset into five near equally sized subsets, trainining in turn on four subsets and testing on the remaining fifth. The reported classification accuracy is averaged over the five experiments. Figure 6 shows classification accuracy plots for images de-identified using k-Same and k-Same-M (Figure 6(a))[1] and for images de-identified using blur filtration (Figure 6(b)). The accuracy for both variants of the k-Same algorithm is substantially higher than in the case of blur filtration. Furthermore, data utility of the images de-identified using the proposed k-Same-M algorithm is slightly better than the data utility of the previously introduced k-Same algorithm.

6. How Not To Protect Privacy: Pixelation

In the popular media, in data protection legislation [12] as well as in parts of the scientific literature, pixelation filters are considered adequate means to protect privacy [17, 27]. Previous studies have already suggested that this is not the case [12, 21]. In this section we provide further evidence by demonstrating multiple ways to achieve high recognition accuracies on pixelated images.

6.1. Image Pixelation

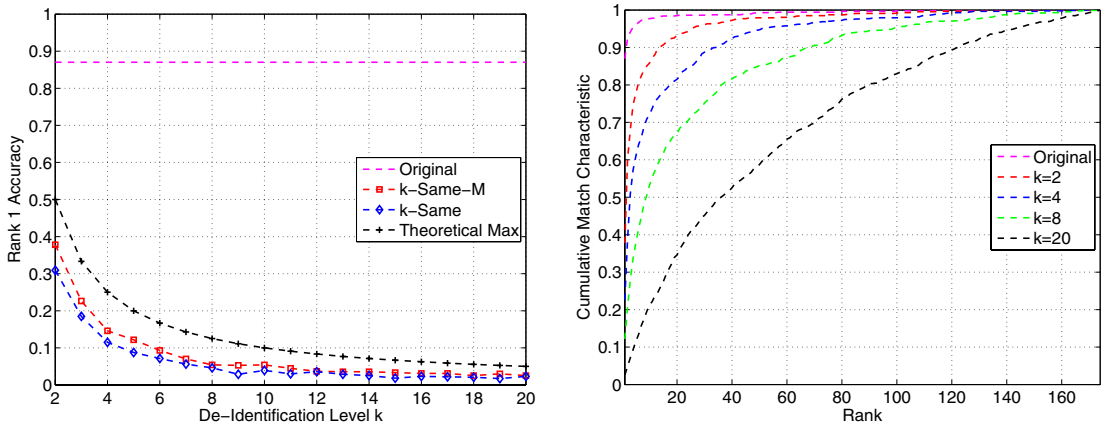
Pixelation occurs as unwanted side effect when the resolution of an image is increased using a simple method such as nearest neighbor interpolation. In the de-identification setting pixelation is performed by replacing all pixel values in all sub-blocks of given constant size in an image by the average pixel value of the block. As the size of the sub-blocks increases more information is removed from the image. See Figure 7 for example images.

6.2. Attacks on Ad-Hoc Methods

6.2.1 Automatic Parrot Recognition

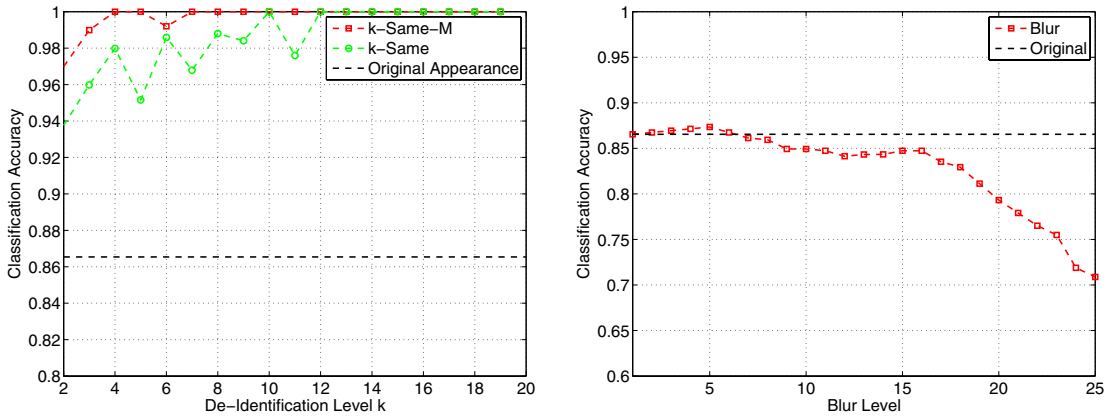
Newton et al proposed to defeat pixelation along with other ad-hoc de-identification methods by applying the same transformation on both the training images as well as the

[1] De-identification for both algorithms was performed on expression specific subsets, effectively using the "select" variant of the k-Same algorithm introduced in [12].



(a) Rank-1 Accuracy (b) CMC Curves

Figure 5. Face recognition rates for original images and images de-identified using the proposed k-Same-M algorithm as well as the previously proposed k-Same algorithm. The experiments use neutral images from the CMU Multi-PIE dataset [14] as gallery and smile images as probe. (a) Comparison of the rank-1 accuracies of the k-Same and k-Same-M algorithms. Both algorithms stay well below the theoretically predicted maximum recognition rate of $1/k$. (b) CMC curves for the k-Same-M algorithm for different levels of k. Recognition performance declines steadily with increasing k, illustrating the amount of de-identification necessary to achieve a given privacy goal.



(a) k-Same-M (b) Blur

Figure 6. Data utility as measured through facial expression classification accuracy. (a) The accuracy achieved by the proposed k-Same-M algorithm is higher than the accuracy of the previously introduced k-Same algorithm. (b) Data utility of images de-identified using blur filtration decreases with higher blur levels, demonstrating a direct trade-off between privacy protection and data utility.

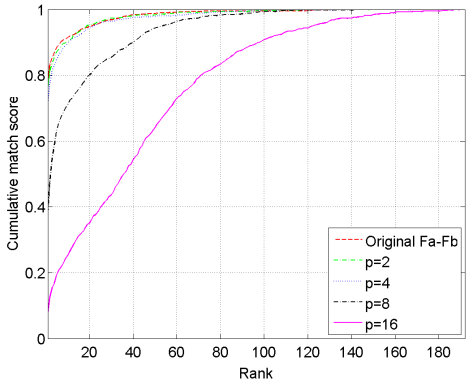
gallery images [21]. In extension of this work we developed a simple detection algorithm to determine the amount of pixelation applied to the image. The algorithm counts runs of pixels with identical or nearly identical pixel values in both the x and y direction and uses a voting scheme to determine a single pixelation value across the whole image. A similar attack on images de-identifed by blur filtration is described in [15].

6.2.2 Resolution Enhancement

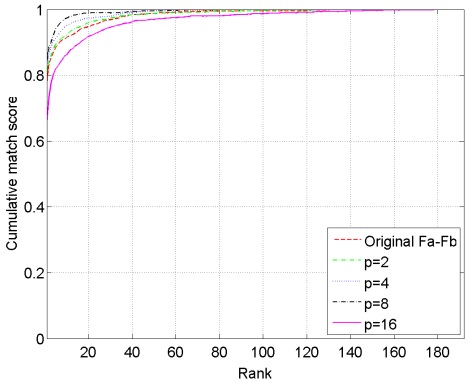
Since pixelated images are essentially low-resolution versions of the original image scaled to the size of the input image, resolution enhancement algorithms can be applied directly to defeat the protection afforded by pixelation. Much work has been devoted in recent years to resolution enhancement methods, especially for face images [4, 16]. We report results using the algorithm described in [16].

6.3. Experiments

We performed experiments on a 275 subject subset of the FERET database using *fa* images in the gallery and *fb* images as probes [23]. The recognition experiments followed the same protocol as described in Section 5.2.



(a) Original vs. Pixelated Recognition



(b) Automatic Parrot Recognition

Figure 8. Results of recognition experiments using PCA on pixelated images. (a) CMC curves for recognition on original *fa* and *fb* images as well as images pixelated to various degrees. High recognition rates are achieved for anything but the highest level of pixelation. (b) CMC curves for experiments in which the amount of pixelation in probe images is automatically detected and applied to gallery images as well. This algorithm achieves recognition rates that are even *higher* than the performance on original, unaltered images.

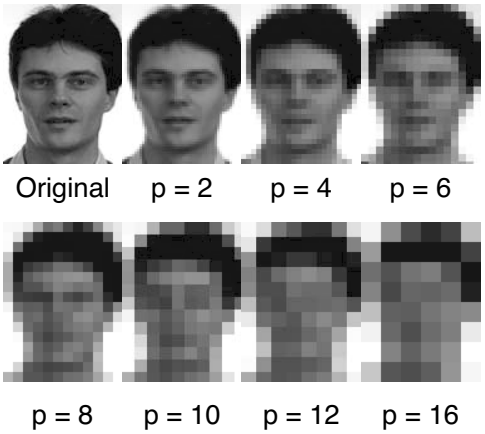


Figure 7. Original image from the FERET database [23] along with de-identified versions using different degrees of pixelation. The parameter p refers to the side length of the sub-block over which pixel values are averaged.

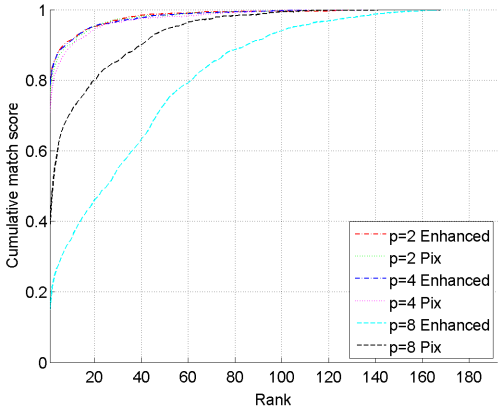


Figure 9. Comparison of recognition performance for pixelated probe images and enhanced pixelated probe images. For small levels of pixelation, enhancement improves the recognition accuracy.

6.3.1 Privacy Protection

Figure 8 shows results of recognition experiments using PCA on both the original *fa* and *fb* images as well as images pixelated to various degrees. In Figure 8(a) we see inadequate privacy protection for anything but that highest level of pixelation. Notice that for $p = 2$ and $p = 4$ recognition rates stay virtually unchanged. Even at the comparatively high pixelation level $p = 8$ a rank-1 recognition of 40% is achieved. We therefore conclude that pixelation does not adequately protect privacy.

6.3.2 Automatic Parrot Recognition

We used the algorithm described in Section 6.2.1 to automatically detect the amount of pixelation present in probe images and applied the same amount of pixelation to gallery images. Training images and the separately computed eigenspace remained unchanged. As shown in Figure 8(b) the resulting recognition rates are even *higher* than the rates achieved on the original, unaltered images for all but the highest level of pixelation.

6.3.3 Image Enhancing Methods

We applied the algorithm proposed by Hardie et al. [16] to pixelated images and compared the recognition perfor-

mance on these enhanced images with performance on pixelated images. We found small performance improvements for the enhanced images for low pixelation levels ($p = 2$ and $p = 4$) and decreases in performance for a higher pixelation level ($p = 8$). See Figure 9.

6.3.4 Conclusion

In this section we showed that even under the "best" circumstances, the privacy protection afforded by pixelation is fairly low. With simple attacks such as the automatic parrot recognition and image enhancing methods the little protection that pixelation provides can be further eroded. We conclude that pixelation is *inadequat* as a privacy protection mechanism.

7. Summary

In this paper we introduced a novel framework for the protection of privacy in facial images. We showed that our k-Same-M algorithm offers privacy protection similar to previously proposed algorithm, while producing de-identified images of much better quality. We furthermore provided additional evidence that simple ad-hoc methods such as pixelation are inadequate for protecting privacy.

8. Acknowledgements

This work was supported in part by the Data Privacy Lab at Carnegie Mellon University, DOJ award 2005-IJ-CX-K046, and by grant R01 MH051435 from the National Institute of Mental Health. Portions of the research in this paper use the FERET database of facial images collected under the FERET program.

References

[1] G. Abowd, A. Bobick, I. Essa, E. Mynatt, and W. Rogers. The aware home: Developing technologies for successful aging. In *Proceedings of the AAAI Workshop on Automation as a Caregiver*, 2002.

[2] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proceedings of the SIGMOD*, 2000.

[3] S. Baker, R. Gross, and I. Matthews. Lucas-Kanade 20 years on: A unifying framework: Part 3: Allowing linear appearance variation. Technical report, Robotics Institute, Carnegie Mellon University, 2003.

[4] S. Baker and T. Kanade. Limits on super-resolution and how to break them. *IEEE PAMI*, 24(9):1167–1183, 2002.

[5] A. Bharucha, B. Pollock, M. Dew, C. Atkeson, D. Chen, S. Stevens, and H. Wactlar. Caremedia: Automated video and sensor analysis for geriatric care. In *AMDA*, 2006.

[6] D. Blackburn, M. Bone, and P. J. Phillips. Facial recognition vendor test 2000: Evaluation Report, 2000.

[7] M. Boyle, C. Edwards, and S. Greenberg. The effects of filtered video on awareness and privacy. In *ACM CSCW*, pages 1–10, 2000.

[8] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001.

[9] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE PAMI*, 23(6):681–685, 2001.

[10] I. Dryden and K. Mardia. *Statistical Shape Analysis*. Wiley & Sons, 1998.

[11] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. Wiley-Interscience, 2000.

[12] R. Gross, E. Airoldi, B. Malin, and L. Sweeney. Integrating utility into face de-identification. In *Workshop on Privacy Enhancing Technologies (PET)*, June 2005.

[13] R. Gross, I. Matthews, and S. Baker. Constructing and fitting active appearance models with occlusion. In *First IEEE Workshop on Face Processing in Video (FPIV)*, 2004.

[14] R. Gross, I. Matthews, J. Cohn, and S. Baker. The CMU Multi-PIE face database. Technical report, Carnegie Mellon University, Robotics Institute, 2006. forthcoming.

[15] R. Gross, L. Sweeney, F. de la Torre, and S. Baker. Model-based face de-identification. Technical report, Carnegie Mellon University, School of Computer Science, 2006. forthcoming.

[16] R. Hardie, K. Barnard, and E. Armstrong. Joint map registration and high-resolution image estimation using a sequence of undersampled images. *IEEE Transactions on Image Processing*, 6(12):1621–1633, 1997.

[17] S. Hudson and I. Smith. Techniques for addressing fundamental privacy and disruption tradeoffs in awareness support systems. In *ACM CSCW*, pages 1–10, Nov 1996.

[18] M. Kirby and L. Sirovich. Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE PAMI*, 12(1):103–108, 1990.

[19] I. Matthews and S. Baker. Active appearance models revisited. *IJCV*, 60(2):135–164, 2005.

[20] C. Neustaedter, S. Greenberg, and M. Boyle. Blur filtration fails to preserve privacy for home-based video conferencing. *ACM TOCHI*, 2005.

[21] E. Newton, L. Sweeney, and B. Malin. Preserving privacy by de-identifying facial images. *IEEE Transactions on Knowledge and Data Engineering*, 17(2):232–243, 2005.

[22] P. J. Phillips. Privacy operating charateristic for privacy protection in surveillance applications. In *AVBPA*, 2005.

[23] P. J. Phillips, H. Moon, S. Rizvi, and P. Rauss. The FERET evaluation methodology for face-recognition algorithms. *IEEE PAMI*, 22(10):1090–1104, 2000.

[24] A. Senior, S. Pankati, A. Hampapur, L. Brown, Y.-L. Tian, and A. Ekin. Blinkering surveillance: enabling video surveillance privacy through computer vision. *IEEE Security and Privacy*, 3(5), 2005.

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

[26] M. Turk and A. P. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[27] Q. Zhao and J. Stasko. Evaluating image filtering based techniques in media space applications. In *ACM CSCW*, pages 11–18, 1998.