Distinguishable De-identified Faces

Zongji Sun, Li Meng*, and Aladdin Ariyaeeinia
School of Engineering and Technology, University of Hertfordshire, Hatfield, AL10 9AB, UK
*Email:.l.1.meng@herts.ac.uk

***Abstract*—The *k*-anonymity approach adopted by *k*-Same face de-identification methods enables these methods to serve their purpose of privacy protection. However, it also forces every *k* original faces to share the same de-identified face, making it impossible to track individuals in a *k*-Same de-identified video. To address this issue, this paper presents an approach to the creation of distinguishable de-identified faces. This new approach can serve privacy protection perfectly whilst producing de-identified faces that are as distinguishable as their original faces.**

I. INTRODUCTION

Over the recent years, a growing number of service providers are starting to share and utilize multimedia content for research, business, security and many other purposes. For example, surveillance footages are often used as evidence in law enforcement. Healthcare homes are sharing videos of patients to conduct analysis of common symptoms as well as to inform research for better diagnoses and treatments. Online storage and sharing of personal images and videos have become an integral part of the modern, mobile life in the Cloud age. However, this evolvement has inevitably ignited concerns about the privacy of information in the course of storing and/or distributing such data. This growing concern and the associated legal and ethical responsibilities have led to considerable interest and effort in the field of face de-identification over the last decade. The aim of face de-identification is to conceal the original identities of the faces captured in a given image or video with new facial identities generated synthetically.

To date, the most cited face de-identification methods are solutions in the *k*-Same family, where privacy protection is achieved by implementing the theory of *k*-anonymity [1]. Examples of *k*-Same algorithms include *k*-Same-Pixels [2], *k*-Same-Eigen [2], *k*-Same-M [3] and *k*-Same-furthest [4]. In a *k*-Same face de-identification process, the set of original face images is firstly partitioned into clusters of size *k*. For each cluster formed, an aggregate face is then used to de-identify all the *k* members in the cluster. In other words, each group of *k* original faces would share the same de-identified face and hence the name '*k*-Same' for this family of methods. Typically, the performance of a face de-identification method is evaluated in terms of the recognition rate of its de-identified faces against the originals. The lower the recognition rate, the better the privacy protection is. As each de-identified face is replicated *k* times for all the originals in its cluster and each de-identified face can only be matched with one original during recognition, the correct matching for each de-identified face can at best be 1 in *k* times. This *k*-anonymity approach has enabled *k*-Same methods to guarantee a recognition rate lower than $1/k$ while this guaranteed recognition rate has been further reduced to zero, i.e. perfect privacy protection has been achieved, by the *k*-Same-furthest method [4] - a new addition to the *k*-Same family.

Although the *k* repetition of each de-identified sample has enabled the *k*-Same methods to serve their purpose of privacy protection, the repetition also makes the *k*-Same de-identified faces undistinguishable. To be more specific, several different individuals will appear to be the same person, making it impossible to track an individual in a *k*-Same de-identified video. To address this common drawback of *k*-Same methods, this paper presents a new method for face de-identification. This new method guarantees perfect privacy protection by adopting the approach of the *k*-Same-furthest method, while facilitating the provision of unique distinguishable de-identified faces across all individuals.

The remainder of the paper is structured as follows. Section 2 reviews the *k*-Same framework and the *k*-Same-furthest method. Section 3 describes the new distinguishable face de-identification method and proves that this new method can always achieve a zero recognition rate. Section 4 evaluates the proposed algorithm's ability to protect privacy through experiments and compare the diversity of the de-identified face images with that of the originals. Finally, the findings of this work are summarized and further discussed in Section 5.

II. *K*-SAME DE-IDENTIFICATION

A. Face De-identification Definition and Notations

To facilitate comparisons, this paper adopts the notations from Newton et al.'s paper [1] on the first *k*-Same method. This set of notations has been used in many succeeding publications on face de-identification, including the *k*-Same-furthest method [4]. The definition of face de-identification given in [1] is quoted here.

Definition Face De-identification (Definition 2.6 in [1]). Let $\mathbf{H}$ and $\mathbf{H}_\mathrm{d}$ be face sets, $\Gamma \in \mathbf{H}$, $\Gamma_\mathrm{d} \in \mathbf{H}_\mathrm{d}$, $f: \mathbf{H} \to \mathbf{H}_\mathrm{d}$ be a function that attempts to conceal the identity of the subject of the original face image; and, $f(\Gamma) = \Gamma_\mathrm{d}$ but $\Gamma \neq \Gamma_\mathrm{d}$ (element-wise). f is termed face de-identification. Γ_d is a de-identified image.

B. k-Same-furthest Face De-identification

The *k*-Same's guarantee of a recognition rate lower than $1/k$ is achieved by replicating each Γ_d k times regardless of the clustering of $\mathbf{H}$ [2]. This implies that even random clustering would not affect the effectiveness of *k*-Same in terms of privacy protection. However, to minimize information loss, all *k*-Same methods form clusters in $\mathbf{H}$ with homogeneous faces [1] and calculate each de-identified face Γ_d as the average of a cluster.

978-1-4799-6026-2/15/$31.00 ©2015 IEEE

The *k*-Same-furthest paper named all the *k*-Same methods prior to it '*k*-Same-closest'. This is due to the fact that all these methods de-identify an original face with the centroid of its own cluster (i.e. the cluster that is closest to the origial face). However, instead of maximizing the loss/removal of identity information, this approach actually minimizes identity loss. When no overlapping exists between any two clusters, the algorithms will always lead to a recognition rate equal to the theoretical maximum of 1/*k*. When overlapping exists between two clusters the centroid of a cluster can be closest to an original face from the overlapping cluster, reducing the recognition rate of *k*-Same-closest in such special cases. This is confirmed by the experimental results published for the *k*-Same-closest methods where their recognition rates tend to stay just below the 1/*k* curve. To maximize the removal of identity information in the original face images, the *k*-Same-furthest method de-identifies each original image $\Gamma \in \mathbf{H}$ with the average of the cluster that is, identity-wise, furthest away from it. The *k*-Same-furthest de-identification process is iterative. In each iteration, two clusters are formed and de-identified with the average of the other cluster. The clustering process in *k*-Same-furthest ensures a maximum distance between these two clusters. Fig. 1 illustrates an iteration of the *k*-Same-furthest de-identification process with a 2D data set. To best serve its goal of privacy protection, this approach is adopted here in the proposed method for the purpose of achieving perfect privacy protection with the de-identified faces.

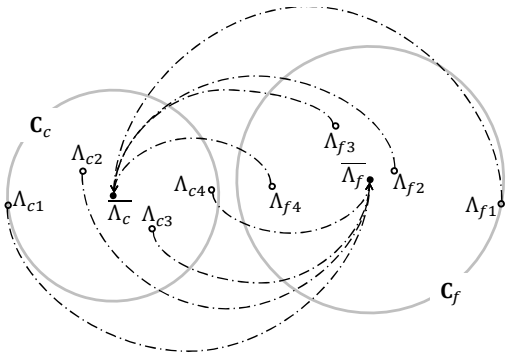

Fig. 1. An iteration of the *k*-Same-furthest de-identificaiton process with an example data set, where original samples Λ_{fi} in cluster $\mathbf{C}_f$ are de-identified as $\overline{\Lambda_c}$ (the centroid of cluster $\mathbf{C}_c$) and original samples Λ_{ci} in $\mathbf{C}_c$ are de-identified as $\overline{\Lambda_f}$ (the centroid of cluster $\mathbf{C}_f$). $i = 1,2,..k$ with $k = 4$ here.

III. Distinguishable De-identified Faces

A. The propsed k-Diff-furthest Algorithm

This section presents an approach to distinguishable de-identified faces. On the one hand, it adopts the iterative process of *k*-Same-furthest in terms of forming two clusters of size *k* in each iteration and swapping their centroids. On the other hand, instead of de-identifying each complete cluster with the same face, this new approach generates a unique (different) de-identified face for each of the *k* original faces in a cluster. It is hence named *k*-Diff-furthest. Fig. 2 outlines the process flow of the proposed *k*-Diff-furthest algorithm.

Algorithm: *k-Diff-furthest*($\mathbf{H}$, k)

Inputs: A person specific face set $\mathbf{H}$ and the privacy constraint k, with $|\mathbf{H}| \geq 2k$
An Active Appearance Model AAM($\cdot$)

Output: De-identified face set $\mathbf{H}_\mathrm{d}$and its AAM projection $\mathbf{M}_\mathrm{d}$

Uses: A face cluster $\mathbf{C}_c = \{\Lambda_{ci}\}$ with a centroid at $\overline{\Lambda_c}$ and a radius of r_c, a face cluster $\mathbf{C}_f = \{\Lambda_{fi}\}$ with a centroid at $\overline{\Lambda_f}$ and a radius of r_f, and dist$(\overline{\Lambda_c}, \overline{\Lambda_f})$ which is the distance between $\overline{\Lambda_c}$ and $\overline{\Lambda_f}$.

Steps:

1 $\mathbf{H}_\mathrm{d} = \emptyset$
2 $\mathbf{M} = \mathrm{AAM}(\mathbf{H})$, $\mathbf{M}_\mathrm{d} = \emptyset$
3 For each $\Lambda_i \in \mathbf{M}$ do:
4 $\quad \mathbf{C}_c = \Lambda_i$, and remove Λ_i from $\mathbf{M}$
5 $\quad \overline{\Lambda_c} = \Lambda_i$
6 $\quad$ Select from $\mathbf{M}$ the face Λ_{f1} that is furthest away from Λ_i
7 $\quad \mathbf{C}_f = \Lambda_{f1}$, and remove Λ_{f1} from $\mathbf{M}$
8 $\quad \overline{\Lambda_f} = \Lambda_{f1}$
9 $\quad$ While $|\mathbf{C}_c| < k$ and $|\mathbf{C}_f| < k$ do:
10 $\quad\quad$ Select from $\mathbf{M}$ the face Λ_f that is closest to $\overline{\Lambda_f}$
11 $\quad\quad$ Add Λ_f to $\mathbf{C}_f$
12 $\quad\quad$ Select from $\mathbf{M}$ the face Λ_c that is closest to $\overline{\Lambda_c}$
13 $\quad\quad$ Add Λ_c to $\mathbf{C}_c$
14 $\quad\quad$ Update $\overline{\Lambda_c}$, $\overline{\Lambda_f}$, r_c, r_f and dist$(\overline{\Lambda_c}, \overline{\Lambda_f})$
15 $\quad\quad$ If dist$(\overline{\Lambda_c}, \overline{\Lambda_f}) < r_c + r_f$ then
16 $\quad\quad\quad$ Remove Λ_f from $\mathbf{C}_f$
17 $\quad\quad\quad$ Remove Λ_c from $\mathbf{C}_c$
18 $\quad\quad\quad$ Update $\overline{\Lambda_f}$ and $\overline{\Lambda_c}$
19 $\quad\quad\quad$ Break from while loop
20 $\quad\quad$ Endif
21 $\quad\quad$ Remove Λ_f and Λ_c from $\mathbf{M}$
22 $\quad$ Loop
23 $\quad$ If $|\mathbf{M}| \leq 2$ then
24 $\quad\quad$ If dist$(\Lambda_i, \overline{\Lambda_c}) >$ dist$(\Lambda_i, \overline{\Lambda_f})$
25 $\quad\quad\quad$ Add Λ_i to $\mathbf{C}_f$ and remove Λ_i from $\mathbf{M}$
26 $\quad\quad$ Else
27 $\quad\quad\quad$ Add Λ_i to $\mathbf{C}_c$ and remove Λ_i from $\mathbf{M}$
28 $\quad\quad$ Endif
29 $\quad$ Endif
30 $\quad \Delta\Lambda = \overline{\Lambda_c} - \overline{\Lambda_f}$
31 $\quad$ For each $\Lambda_{ci} \in \mathbf{C}_c$ do:
32 $\quad\quad$ Compute $\Lambda_\mathrm{d} = \Lambda_{ci} - \Delta\Lambda$
33 $\quad\quad$ Add Λ_d to $\mathbf{M}_\mathrm{d}$ to de-identify Λ_{ci}
34 $\quad$ Next
35 $\quad$ For each $\Lambda_{fi} \in \mathbf{C}_f$ do:
36 $\quad\quad$ Compute $\Lambda_\mathrm{d} = \Lambda_{fi} + \Delta\Lambda$
37 $\quad\quad$ Add Λ_d to $\mathbf{M}_\mathrm{d}$ to de-identify Λ_{fi}
38 $\quad$ Next
39 Next
40 $\mathbf{H}_d = \mathrm{AAM}^{-1}(\mathbf{M}_d)$

Fig. 2. The process flow of the proposed *k*-Diff-furthest algorithm.

The proposed *k*-Diff-furthest algorithm transforms the given person-specific face set $\mathbf{H}$ from the RGB pixel-based

space to a pre-trained Active Appearance Model (AAM) [5, 6] feature space where all the faces are aligned to a common face. It has been shown that representing original faces in an AAM space and performing face de-identification there can prevent ghost artefacts in the de-identified face effectively [3]. The AAM representation of the original face set $\mathbf{H}$ is denoted as $\mathbf{M}$ and its de-identified version as $\mathbf{M}_\mathrm{d}$.

As shown in Fig. 2, the *k*-Diff-furthest process is iterative. Like the *k*-Same-furthest method, *k*-Diff-furthest completes two tasks in each iteration. The first task is to form two clusters $\mathbf{C}_c$ and $\mathbf{C}_f$ in $\mathbf{M}$ for the given original face Λ_i, where Λ_i is the trigger of the current iteration and $\mathbf{C}_c$ is formed with faces closest to Λ_i while $\mathbf{C}_f$ with those furthest from it. Once an original face is assigned to a cluster, it is removed from $\mathbf{M}$. The second task of each iteration is to generate a de-identified face Λ_d for each original in $\mathbf{C}_c$ and $\mathbf{C}_f$.

In order to achieve a privacy protection level guaranteed to be better than $1/k$, all the *k*-Same-closest methods [4] demand that each cluster formed in the de-identification process must contain at least k members. In contrast to these methods, *k*-Same-furthest guarantees perfect privacy protection regardless of the value of k, by preventing overlapping between the two clusters formed in each iteration. *k*-Diff-furthest adopts this same approach and guarantees perfect privacy protection regardless of the value of k. Whenever a new member is added to the two clusters $\mathbf{C}_c$ and $\mathbf{C}_f$ each, *k*-Diff-furthest checks to see whether overlapping is caused by these two new members (line 15). If so, both new members are removed from their clusters and the clustering loop for both $\mathbf{C}_c$ and $\mathbf{C}_f$ is stopped as adding any other remaining face to $\mathbf{C}_c$ or $\mathbf{C}_f$ would cause even more overlapping between the two clusters. As a result, the size of the clusters formed in the *k*-Diff-furthest process might be smaller than k. Part B of this section proves the effectiveness of this approach in terms of privacy protection.

To maximize identity loss, *k*-Same-furthest de-identifies the originals in $\mathbf{C}_c$ and $\mathbf{C}_f$ by swapping the cluster centers. Whilst the same approach is adopted in *k*-Diff-furthest, the two methods differ in the way how the de-identified faces are computed. The *k*-Same-furthest algorithm implements *k*-anonymity and uses the average of one cluster as the de-identified face for all the faces in the other cluster. Lines 30-39 in Fig. 2 details how the de-identified faces are computed in *k*-Diff-furthest. As detailed by the pseudo code in Fig. 2 and illustrated by the example in Fig. 3, the de-identification step in *k*-Diff-furthest is equivalent to moving original faces Λ_i in $\mathbf{C}_c$ to their new centroid $\overline{\Lambda_f}$ with their relative locations to the centroid unchanged, i.e.

$$\text{vector}\,(\overline{\Lambda_c}\Lambda_i)^{\rightarrow} = \overrightarrow{\overline{\Lambda_f}\Lambda_\mathrm{d}} \qquad (1).$$

The same applies to the original faces in $\mathbf{C}_f$. Through this approach, *k*-Diff-furthest generates a unique de-identified face for each original face in $\mathbf{H}$ and retains the diversity of $\mathbf{H}$ in $\mathbf{H}_\mathrm{d}$. As it is assumed that original faces in $\mathbf{H}$ are distinguishable, *k*-Diff-furthest ensures that the de-identified faces in $\mathbf{H}_\mathrm{d}$ are equally distinguishable.

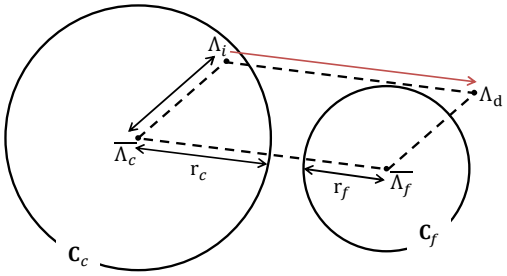

Fig. 3. Computation of the de-identified face $\Lambda_{\mathrm{d}i}$ for an original face Λ_i in the *k*-Diff-furthest process, illustrated with a simplified example in a 2D space.

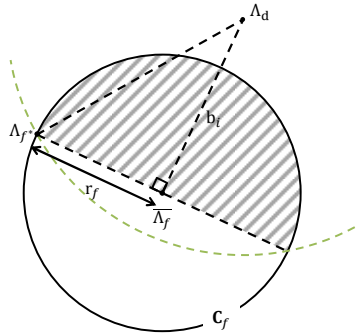

Fig. 4. Illustration of Theorem 1 in a 2D space.

B. Correctness of the k-Diff-furthest Algorithm

Theorem 1. Given a privacy constraint $k > 1$; a person-specific face set $\mathbf{H}$ with $|\mathbf{H}| \geq 2k$; and a face set $\mathbf{H}_d = k\text{-Diff-furthest}(\mathbf{H}, k)$, k-Diff-furthest() is effective with respect to the following claim for any face image $\Gamma_d = k\text{-Diff-furthest}(\Gamma, k)$ for $\Gamma \in \mathbf{H}$:

Given that *k*-Diff-furthest() uses $\mathrm{dist}(\Gamma_1, \Gamma_2)$ to measure the identity distance between any two faces Γ_1 and Γ_2, there cannot exist any face recognition software that measures identity distance with $\mathrm{dist}(\Gamma_1, \Gamma_2)$ to correctly recognize the subject of Γ_d as Γ.

Proof. As stated, *k*-Diff-furthest() measures $\mathrm{dist}(\Gamma_1, \Gamma_2)$ in an AAM feature space as $\mathrm{dist}(\Lambda_1, \Lambda_2)$. The following proves that the de-identified face $\Lambda_\mathrm{d} = k\text{-Diff-furthest}(\Lambda_i, k)$ for any $\Lambda_i \in \mathbf{C}_c$ will be recognized as an original $\Lambda_j \in \mathbf{C}_f$, i.e.

$$\min_{\Lambda_j \in \mathbf{C}_f}\{\mathrm{dist}(\Lambda_j, \Lambda_\mathrm{d})\} < dist(\Lambda_i, \Lambda_\mathrm{d}) \qquad (2).$$

As demonstrated in Fig. 4, the shaded half of $\mathbf{C}_f$ must contain at least an original face Λ_j. Otherwise, the centroid $\overline{\Lambda_f}$ would have shifted into the un-shaded half of $\mathbf{C}_f$. Within the shaded half of $\mathbf{C}_f$, the furthest point to Λ_d is Λ_{f^*}, giving that

$$\min_{\Lambda_j \in \mathbf{C}_f}\{\mathrm{dist}(\Lambda_j, \Lambda_\mathrm{d})\} \leq \mathrm{dist}(\Lambda_{f^*}, \Lambda_\mathrm{d}). \qquad (3)$$

According to the Triangle Inequality Theorem,

$\mathrm{dist}(\Lambda_{f^*}, \Lambda_\mathrm{d}) < r_f + \mathrm{dist}(\Lambda_\mathrm{d}, \overline{\Lambda_f})$ where $\mathrm{dist}(\Lambda_\mathrm{d}, \overline{\Lambda_f}) = \mathrm{dist}(\Lambda_i, \overline{\Lambda_c}) \leq r_c$.

Therefore,

$$\mathrm{dist}(\Lambda_{f^*},\ \Lambda_{\mathrm{d}}) < r_f + r_c. \tag{4}$$

The condition on line 15 of Fig. 2 means that in *k*-Diff-furthest()

$$r_c + r_f \leq \mathrm{dist}(\overline{\Lambda_c},\ \overline{\Lambda_f}) \tag{5}$$

Combining (3), (4) and (5) gives

$$\min_{\Lambda_j \in \mathbf{C}_f}\{\mathrm{dist}(\Lambda_j,\ \Lambda_{\mathrm{d}})\} < \mathrm{dist}(\overline{\Lambda_c},\ \overline{\Lambda_f}) \tag{6}$$

As illustrated in Fig. 3, (1) stands for any original face $\Lambda_i \in \mathbf{C}_c$. This gives $\mathrm{dist}(\Lambda_i,\ \Lambda_{\mathrm{d}}) = \mathrm{dist}(\overline{\Lambda_c},\ \overline{\Lambda_f})$.

Equation (2) and hence **Theorem 1 are proved**. Although the above proves Theorem 1 for any $\Lambda_i \in \mathbf{C}_c$. The same stands for any $\Lambda_i \in \mathbf{C}_f$.

Theorem 1 means that as long as the same distance measure is used in the face recognition software, *k*-Diff-furthest() guarantees to thwarts the face recognition software for every face in its $\mathbf{H}_d$ regardless of the value of *k*.

It is worth mentioning that the last two original faces in $\mathbf{M}$ might not comply with (5). When an original face does not satisfy condition (5), its de-identified version will not satisfy (2) and might be recognized correctly as its original face. As explained in part A of this section, the *k*-Diff-furthest process is iterative. Once an original face is de-identified, it is removed from $\mathbf{M}$. Clustering is carried out with the originals remaining in $\mathbf{M}$. This above mentioned situation may only occur to the last two original faces in $\mathbf{M}$ when they cannot join $\mathbf{C}_c$ or $\mathbf{C}_f$ without breaking condition (5). Although each of these two remaining original faces may form a cluster on its own, de-identification of these single-member clusters would be equivalent to a simple exchange of identities between these two faces. The de-identified face of person 1 would be identical to the original face of person 2. Obviously, this is not acceptable. To prevent this, single-member clusters are avoided in the *k*-Diff-furthest by the process defined by lines 23-29 in Fig. 2. However, the compromise of this no single-member cluster policy in *k*-Diff-furthest means that privacy protection can no longer be guaranteed for the last two remaining original faces in $\mathbf{M}$ when they cannot join $\mathbf{C}_c$ or $\mathbf{C}_f$ without breaking condition (5).

IV. Experiments

A. Dataset

Experiments in this work were conducted with face images from the IMM [7] and the LFPW datasets [8]. Manual annotations of facial landmarks are given as the ground-truth data within the IMM dataset. For the LFPW dataset, manual annotations are provided by the 300 Faces In-the-Wild (300-W) Challenge [9]. The 68 points mark-up are followed in the annotation of both datasets (see row I of Fig. 5). The LFPW dataset contains faces with uncontrolled head pose, facial expressions and illumination; whereas the IMM dataset contains face images of 40 individuals captured in a controlled environment with six images per individual (frontal neutral, frontal happy, left rotated neutral, right rotated neutral, spot light neutral and free style). All the 783 24-bit colour face images in the LFPW dataset and all the 40 frontal neutral face images in the IMM dataset were used to train/construct the AAM feature space in this work.

Cropped face images (row II of Fig. 5) showing only the region inside the outline of the AAM-fitted face shape were used in the experiments. Each cropped face image is represented as a 59-dimensional vector in the trained AAM feature space, with 9 dimensions for face shape and 50 for face texture. The face images in the LFPW dataset were captured under various lighting conditions. As shown on row I of Fig. 5, the three LFPW images vary in terms of overall image brightness and some of them are significantly brighter than those from the IMM dataset. Due to this large variation in brightness, the AAM feature space trained with 783 LFPW face images and 40 IMM face images in this work has its most dominant face texture feature representing the overall image brightness. Considering that this feature does not contain any identity information whereas clustering of face images in the face de-identification process should be carried out based on the identity features of the faces, the first AAM texture feature has been set arbitrarily to zero for all the face images in the testing set. This is equivalent to the brightness (or histogram) equalization procedure that is typically applied as part of the image pre-processing process in face recognition and face de-identification. The cropped images on row II of Fig. 5 are images after brightness equalization.

Like most *k*-Same face de-identification algorithms (including *k*-Same-Pixel/-Eigen [2], *k*-Same-M [3], and *k*-Same-furthest [4]), the proposed *k*-Diff-furthest algorithm focuses on the de-identification of images of neutral frontal faces. As a result, all images used in the experimental evaluation of the *k*-Diff-furthest algorithm are of neutral frontal faces. The testing set of this work consists of the 37 colour images from the IMM dataset and another three selected from the LFPW dataset (see Fig. 5 columns (d-f)). The reason for replacing the three grayscale face images in the IMM dataset is to remove the impact of the colour format on the visual quality of the de-identified faces.

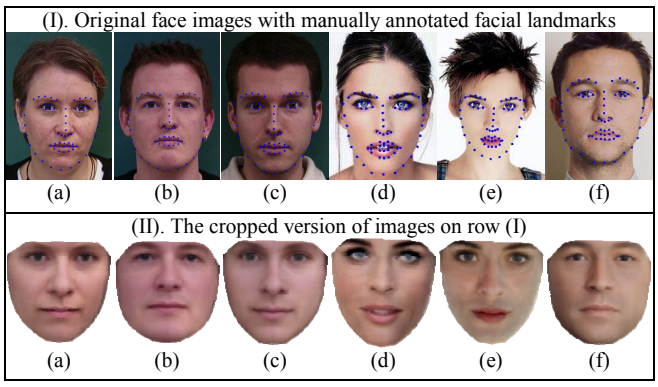

Fig. 5. Examples of face images in the testing set of this work. (a - c) are originally from the IMM dataset [6]; (d – f) from the IFPW dataset.

Although the *k*-Diff-furthest algorithm is proposed for neutral frontal faces only, additional measures can be applied to integrate the head poses, facial expressions and illumination of the original faces into the de-identified face image. For instance, replacing the single AAM in the *k*-Diff-furthest algorithm with a set of five view-based AAMs [10] will enable the algorithm to retain the horizontal head rotations in the de-identified faces. In addition, a facial expression transfer scheme [11] has been developed for the restoration of original facial expressions on the de-identified faces, which can be directly applied to the de-identified faces generated by the *k*-Diff-furthest algorithm.

B. Perfect Privacy Protection by k-Diff-furthest

1) Test design

The privacy protection ability of the proposed *k*-Diff-furthest algorithm is evaluated through recognition experiments using the Eigenface technique [12] in the AAM space, where the 40 original face images (all cropped) from the testing set are de-identified and the de-identified faces are then matched against all the original faces in the testing set. In the de-identification process, the original face that triggers each iteration (line 3 of Fig. 2) is randomly selected. All results reported are based on running the identification process 1000 times for each value of *k*.

2) Test results

Fig. 6 shows the rank-1 recognition rates of the de-identified faces against their original faces. The *k*-Same-M algorithm is a *k*-Same-closest solution. But like *k*-Diff-furthest, it also performs face de-identification in the AAM space. As expected and confirmed in Fig. 6, the recognition rate of the *k*-Same-M de-identified faces always stays synchronized with and just below the theoretical maximum of $1/k$. The same experimental results of recognition rate have been reported for all the other *k*-Same-closest face de-identification methods in their original papers [2, 3, 13], forcing all *k*-Same-closest methods to use large values of *k* in order to achieve acceptable privacy protection. The recognition rates of *k*-Diff-furthest faces on the other hand are significantly lower than those of the *k*-Same-M faces. Fig. 7 is a zoomed-in version of Fig. 6. Both Figs. 6 and 7 confirm that when single-member clusters are allowed, i.e. when all original faces satisfy condition (5), de-identified faces generated by *k*-Diff-furthest always yield a recognition rate of zero regardless of the value of *k*. When single-member clusters are not allowed and when the steps defined by lines 23-29 in Fig. 2 are carried out, the last two (out of 40) original faces may lead to a correct matching with their de-identified versions. However, as shown in Figs. 6 and 7, the probability for this to happen is lower than 0.4%.

C. Distinguishable De-identified Faces by k-Diff

1) Test design

To measure how diverse a set of face images is, the Euclidean distance between each image and every other image in the set is computed in the AAM space. This is the distance measure used in Eigenface and many other face recognition techniques. It indicates how distinguishable the faces are in terms of the facial features displayed in the images. The smaller the distance between two face images, the harder it becomes to distinguish the two faces in the images.

2) Test result and result analysis

Fig. 8 shows the histogram distribution of the facial feature distances among the original testing face images as well as their de-identified face images generated by *k*-Same-closest, *k*-Same-furthest and the proposed *k*-Diff-furthest when $k = 5$. *k*-Same-M is again used as the representative *k*-Same-closest method. There are 40 face images in the testing set, meaning each histogram in Fig. 8 shows the distribution of $C_2^{40} = 780$ facial feature distances. Table I lists the minimum, the maximum and the average distances as well as the standard deviation for each set of images. Calculation of standard deviation for both *k*-Same-closet and *k*-Same-furthest has excluded the distance at zero as this distance is given by repetitions of the same de-identified face. Fig. 9 illustrates the relationships between the computed facial feature distance and the visual difference displayed between the pair of face images.

As shown in Fig. 8, the distance distributions of the original faces and the *k*-Diff-furthest (de-identified) faces have very similar outlines, indicating that the diversity of faces in terms of their facial features are kept through the *k*-Diff-furthest face de-identification process and hence the *k*-Diff-furthest faces are as distinguishable as their original faces. This is also confirmed by the results in Table I, where the two sets of face images have very similar average and maximum distances. The higher minimum distance from the *k*-Diff-furthest de-identified faces means that the most similar pair of *k*-Diff-furthest faces (Fig. 9(d)) is more distinguishable than the most similar pair of original faces (Fig. 9(a)). The higher minimum value has also given *k*-Diff-furthest faces a slightly smaller standard deviation and a slightly more narrow distribution than the original faces. In contrast to those of the original and the *k*-Diff-furthest faces, the distance distributions for both the *k*-Same-closest and the *k*-Same-furthest faces are much more discrete. This reflects the fact that *k*-Same methods de-identify a cluster of *k* original faces using the same de-identified face. This is also indicated by the spike at zero in both histograms. In addition, the de-identified faces generated by *k*-Same methods are the centroids of clusters. The averaging effect of these de-identified faces has led to a much smaller maximum distance and a much more narrow distribution diagram for each *k*-Same method, implying that faces originally distinctively different have become much less distinguishable when being *k*-Same de-identified.

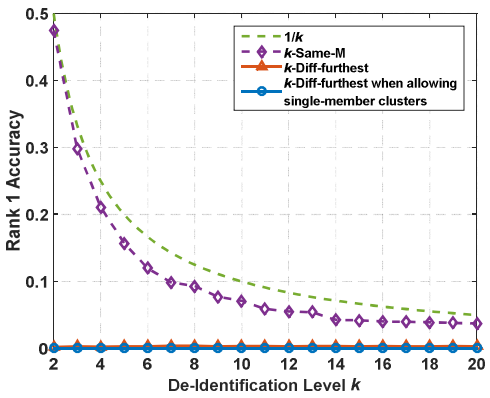

Fig. 6. Recognition rates for de-identified faces aginst their original faces.

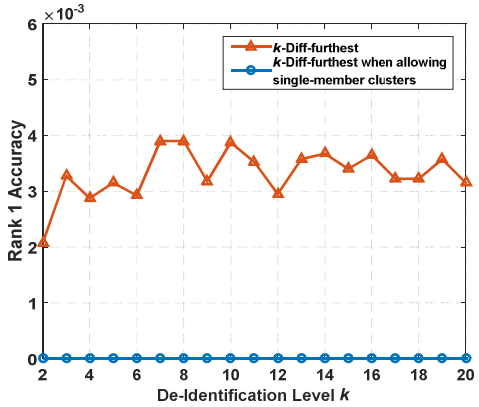

Fig. 7. Recognition rates for *k*-Diff de-identified face aginst their original faces.

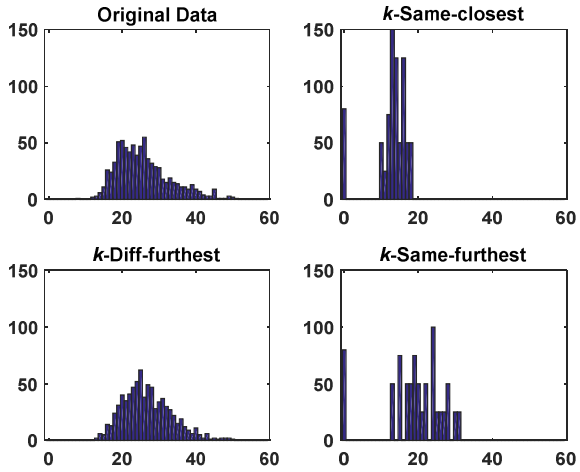

Fig. 8. Histogram of feature distances of original faces and various sets of de-identified faces when *k*=5.

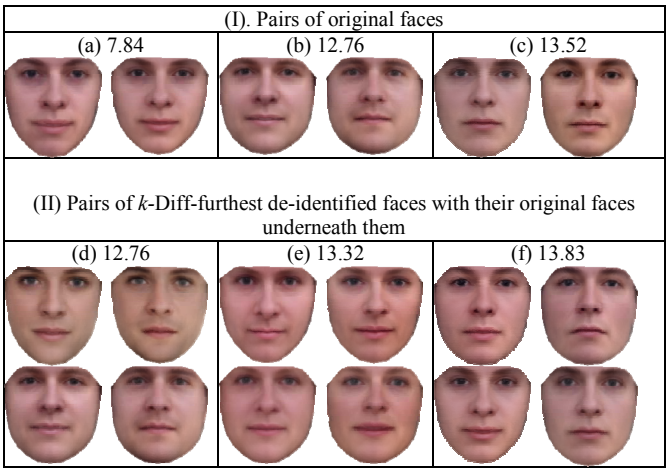

Fig. 9. Example faces to show relationships between the computed facial feature distance and the visual difference displayed. The computed facial feature distances are given in the labels for each pair of example face images.

TABLE I. FEATURE DISTANCES STATISTICS

		min	max	average	std
Original face images		7.84	50.50	25.74	7.32
De-identified face images	**k-Diff-furthest**	12.76	50.10	26.80	6.55
	k-Same-closest	0	18.47	12.64	2.17
	k-Same-furthest	0	31.17	19.10	5.00

V. DISCUSSION AND CONCLUSION

This paper presents a new approach to face de-identification named *k*-Diff-furthest. The *k*-Diff-furthest method maximizes the removal of identity information by de-identifying an original face image based on face images that are furthest from it. As proved mathematically in the paper and verified through recognition experiments, *k*-Diff-furthest can serve its purpose of privacy protection perfectly regardless of value of *k*. Furthermore, in contrast to *k*-Same face de-identification where diversity within the set of the original faces is lost, *k*-Diff-furthest maintains the diversity of the de-identified faces and keeps them as distinguishable as their original faces.

REFERENCES

[1] L. Sweeney, "*k*-Anonymity: a model for protecting privacy," Int'l J. Uncertainty, Fuzziness, and Knowledge-Based Systems, vol. 10, no. 5, pp. 557–570, 2002.

[2] E.M. Newton, L. Sweeney, and B. Malin, "Preserving privacy by de-identifying face images," IEEE Trans. Knowledge and Data Eng., vol. 12, no. 2, pp. 232 – 243, February 2005.

[3] R. Gross, L. Sweeney, F. de la Torre, and S. Baker, "Model-based face de-identification," IEEE Workshop on Privacy Research in Vision, 2006.

[4] L. Meng and Z. Sun, "Face De-Identification with Perfect Privacy Protection," Proc. of the 37th Intl. Convention MIPRO, Special Session on BiForD, pp. 1234-1239, Opatija, Croatia, May 2014.

[5] G. Edwards, C. Taylor, and T. Cootes, "Interpreting Face Images Using Active Appearance Models," Proc. FG'98, pp. 300–305, Apr. 1998.

[6] I. Matthews, and S. Baker, "Active appearance models revisited," Int'l J. Computer Vision, vol. 60, no. 2, pp. 135-164, Nov. 2004.

[7] M.M. Nordstrøm, M. Larsen, J. Sierakowski, and M.B. Stegmann, "The IMM face database - an annotated dataset of 240 face images," Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, May 2004.

[8] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman and N. Kumar, "Localizing parts of faces using a consensus of exemplars," 2011 IEEE Conf. Computer Vision and Pattern Recognition, pp. 545-552, 2011.

[9] Intelligent Behaviour Understanding Group, "300 Faces In-the-Wild Challenge (300-W)," ICCV 2013. Available online at http://ibug.doc.ic.ac.uk/resources/300-W [Accessed in June 2014].

[10] T.F. Cootes, K. Walker, and C.J. Taylor, "View-based active appearance models," Proc. Of 4th IEEE Intl. Conf. Automatic Face and Gesture Recognition, pp. 227-232, Mar 2000.

[11] L. Meng, Z.J. Sun, K.L. Bennett, and A. Ariyaeeinia, "Retaining Expressions on De-identified Faces," Proc. of the 37th Intl. Convention MIPRO, Special Session on BiForD, pp. 1252-1257, Opatija, Croatia, May 2014.

[12] M. Turk, and A. P. Pentland, "Eigenfaces for recognition," J. Cognitive Neuroscience, vol. 3, no. 1, pp. 71-86, 1991.

[13] R. Gross, E. Airoldi, B. Malin, and L. Sweeney, "Integrating utility into face de-identification," Workshop on Privacy-Enhanced Technologies, 2005.