Privacy-Protected Facial Biometric Verification Using Fuzzy Forest Learning

Richard Jiang, Ahmed Bouridane, *Senior Member, IEEE*, Danny Crookes, *Senior Member, IEEE*, M. Emre Celebi, *Senior Member, IEEE*, and Hua-Liang Wei

***Abstract*—Although visual surveillance has emerged as an effective technolody for public security, privacy has become an issue of great concern in the transmission and distribution of surveillance videos. For example, personal facial images should not be browsed without permission. To cope with this issue, face image scrambling has emerged as a simple solution for privacy-related applications. Consequently, online facial biometric verification needs to be carried out in the scrambled domain, thus bringing a new challenge to face classification. In this paper, we investigate face verification issues in the scrambled domain and propose a novel scheme to handle this challenge. In our proposed method, to make feature extraction from scrambled face images robust, a biased random subspace sampling scheme is applied to construct fuzzy decision trees from randomly selected features, and fuzzy forest decision using fuzzy memberships is then obtained from combining all fuzzy tree decisions. In our experiment, we first estimated the optimal parameters for the construction of the random forest and, then, applied the optimized model to the benchmark tests using three publically available face datasets. The experimental results validated that our proposed scheme can robustly cope with the challenging tests in the scrambled domain and achieved an improved accuracy over all tests, making our method a promising candidate for the emerging privacy-related facial biometric applications.**

***Index Terms*—Chaotic pattern, ensemble learning, face scrambling, facial biometrics, fuzzy random forest, privacy.**

I. Introduction

DUE to the demands for greater public security over the past decade, video surveillance has become a widely applied technology in the day-to-day life of public society. As a result, privacy protection [1]–[7] has become a concern for the public as well as for the legal authorities. Key information such as facial images [1]–[3], [6], [7] in surveillance videos should not be exposed when distributing videos over public networks. Face scrambling [1]–[3], [6], [7] has become a promising solution to this issue. By scrambling faces detected in surveillance videos, the privacy of subjects under public surveillance can be respected in modern security technology.

In comparison with encryption, image scrambling has two apparent advantages. First, scrambling usually has much lower computation cost than encryption, making it suitable for computing-efficient network-targeted applications. Second, encryption may undermine the purpose of public security control because its decryption depends on acquiring the encryption key. For example, a security guard who needs to check a key face in a surveillance video may not be able to do so until he/she has the decryption key. In comparison, scrambled faces using the Arnold transform can be easily recovered by manual attempts using the inverse Arnold transform with different parameters.

As a result, face scrambling becomes a compromised choice because it does not really hide information, while unscrambling is usually achievable by simple manual tries even though we do not know all the parameters. It avoids exposing individual biometric faces without really hiding anything from surveillance video. As shown in [1]–[7], scrambling has recently become popular in the research field of visual surveillance, where privacy protection is needed as well as public security.

There are many ways to perform face scrambling. For example, scrambling can be done simply by masking or cartooning [8]. However, this kind of scrambling will simply lose the facial information, and hence, face recognition becomes unsuccessful in this case. In addition, for security reasons, it is obviously not a good choice to really erase human faces from surveillance video. In contrast, the Arnold transform [9], [10], as a step in many encryption algorithms, is a kind of recoverable scrambling method. Scrambled faces can be unscrambled by several manual tries. Hence, in this study, we have chosen Arnold transform-based scrambling as our specific test platform.

Automated surveillance systems are installed with online facial biometric verification. While it may not be permitted to unscramble detected faces without authorization due to privacy-protection policies, the ability to carry out facial biometric verification in the scrambled domain becomes desirable for many emerging surveillance systems. Moreover, since unscrambling may involve parameters that are usually unknown by the online software, the need arises to carry out face recognition purely in the scrambled domain.

The task of automatically recognizing various facial images is usually a challenging task. As a result, face recognition has become a prominent research topic in image indexing [6], human–computer interaction [11]– [15], forensic biometrics [16], [17],

Manuscript received March 02, 2015; revised July 13, 2015; accepted September 08, 2015. Date of publication October 05, 2015; date of current version August 02, 2016.

R. Jiang and A. Bouridane are with the Department of Computer Science and Digital Technologies, Northumbria University, Newcastle upon Tyne NE1 8ST, U.K. (e-mail: richardjiang@acm.org; a.bouridane@northumbria.ac.uk).

D. Crookes is with the School of Electronics, Electrical Engineering and Computer Science, Institute of Electronics, Communications and Information Technology, Queen's University Belfast, Belfast BT3 9DT, U.K. (e-mail: d.crookes@qub.ac.uk).

M. E. Celebi is with the Department of Computer Science, Louisiana State University in Shreveport, Shreveport, LA 71115 USA (e-mail: e.celebi@lsus.edu).

H.-L. Wei is with the Department of Automatic Control and System Engineering, University of Sheffield, Sheffield S10 2TN, U.K. (e-mail: whualiang@sheffield.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TFUZZ.2015.2486803

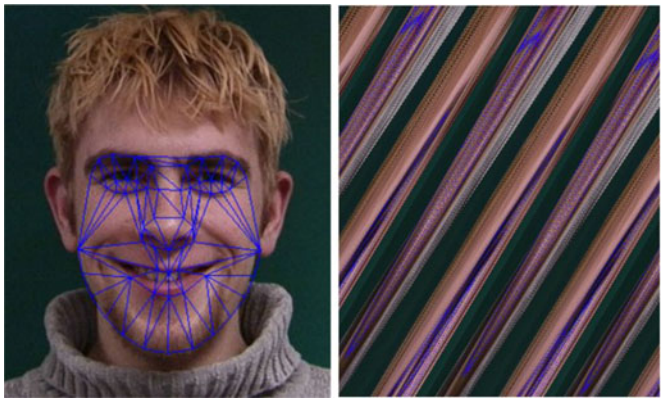


Fig. 1. Typical face verification approaches using various 2-D/3-D facial models cannot be applied in the scrambled facial domain. (Left) Typical face image with 3-D mesh model. (Right) Its scrambled image.

medical applications [18], and human cognition [19]. The challenge becomes even more substantial when such facial verification is deployed in visual surveillance systems where videos are usually captured and transmitted on an internet-based visual sensor network. In these situations, face recognition can involve third-party servers where personal privacy needs to be ensured. Further, storage and distribution of recorded surveillance videos are subject to legal constraints especially when human faces are present in videos. As a result, face scrambling will likely be adopted in these visual surveillance systems. The challenge, hence, becomes a question of how to perform face recognition in the scrambled domain without revealing the private contents [1]–[7]. Consequently, automated facial biometric verification has to be carried out in the scrambled domain. As shown in Fig. 1, a scrambled face has a very different appearance from its original facial image. The need for an effective method to handle this new challenge comes along with the new security era.

Commonly in face recognition, dimensionality reduction [20] has usually been considered as the central issue in this challenging task, and a number of methods have been introduced in the last decade, including principal component analysis (PCA) [19], independent component analysis (ICA) [21] and Fisher's linear discriminant analysis (FLD) [22]. Combined with kernel methods [23], these methods can be extended to kernel Hilbert space with a nonlinear mapping, and we then have their kernel versions such as k-PCA, k-ICA, and k-FLD. These approaches can also be applied with 2-D/3-D face modeling techniques [24]–[27], combined with various facial features [28], [29], or integrated with support vector machine (SVM) or boosting algorithms. However, it is yet very challenging to construct 3-D models automatically from 2-D images/views [30]. Besides, for face recognition in the scrambled domain, one needs a robust approach to cope with the chaotic facial patterns typical in surveillance applications.

The random forest method [31], [32] is well suited to handle randomly distributed features and, hence, excels at noise-like or chaotic pattern classification. Recent research [33] has also demonstrated that random forests can be effectively applied to the face pose normalization problem. However, in our literature search, the advantage of the random forest method has not been sufficiently exploited for face recognition, and to the best of our knowledge, very few reports on utilizing random forests for image-based face recognition are publically available. An underlying reason is that a facial image cropped from videos usually has a small number of pixels (such as 32×32), while random subspace sampling requires a larger number of features for sparse sampling.

In this paper, we propose a fuzzy forest learning (FFL) scheme to tackle the scrambled face recognition challenge. In our proposed scheme, a center-surround prior map is applied to guide the random sampling in the scrambled domain, and a fuzzy decision-making mechanism is introduced to weight tree decisions via their fuzzy membership vectors. We then carried out an experimental validation on several scrambled face databases to show the effectiveness of our proposed fuzzy scheme over scrambled facial images.

In the remainder of the paper, Section II introduces the basics of facial biometric verification in the scrambled domain, Section III proposes the construction of a fuzzy random forest, and Section IV describes the fuzzy forest decision-making scheme. Section V is a discussion of the parameters in the FFL, and Section VI presents experimental results on three face datasets. Conclusions are drawn in Section VII.

II. Facial Biometric Verification in Scrambled Domain

A. Face Scrambling Using Arnold Transform

Digital image scrambling can turn an image into a chaotic and meaningless pattern after transformation. It is a preprocessing step for hiding the information of the digital image, which is also known as information disguise. Image scrambling technology depends on data hiding technology, which provides non-password security algorithm for information hiding. The image after scrambling is chaotic, and as a result, the visual information is hidden from the public eye and privacy is then protected to a degree even if the visual contents are browsed or distributed over a public network.

Among the various image scrambling methods, the Arnold scrambling algorithm has the properties of simplicity and periodicity. The Arnold transform [9], [10] was proposed by V. I. Arnold in the research of ergodic theory; it was also called cat-mapping before it was applied to digital images. It has been popular in image scrambling because of its simplicity and ease of use. In this paper, we use this scrambling method to set up the test environment of our algorithm in the scrambled face domain.

In the Arnold transform, a pixel at the point (x, y) is shifted to another point (x', y′) as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \bmod N. \tag{1}$$

which is called 2-D Arnold scrambling. The recursive and iterative application of the Arnold transform can be defined as follows:

$$P_{\mathrm{xy}}^{k+1} = AP_{\mathrm{xy}}^{k}, P_{\mathrm{xy}}^{k} = (x, y)^{T}. \tag{2}$$

Here, the input is pixel $(x, y)^T$ after the kth Arnold transform, P_{xy}^{k+1} on the left is the output for the $(k+1)$th Arnold transform.

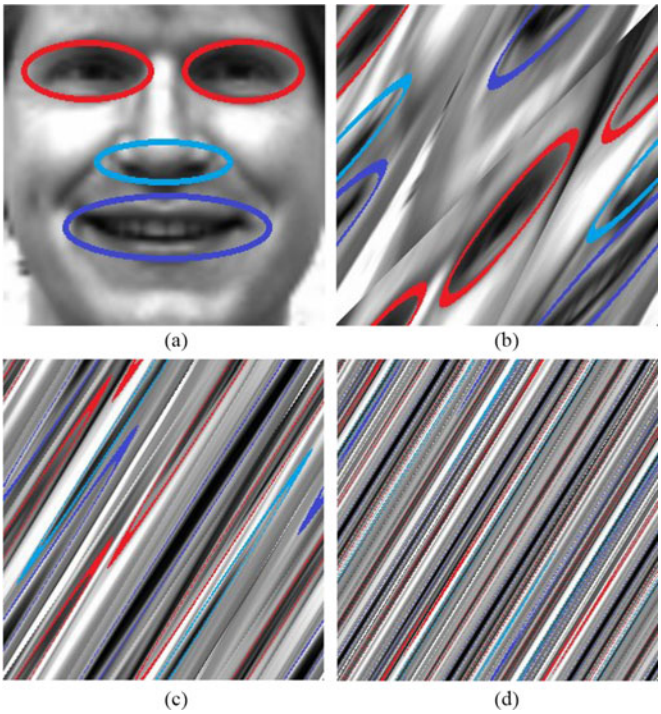


Fig. 2. Face scrambling by Arnold transform. (a) Semantic facial components. (b) After one Arnold transform. (c) After two Arnold transforms. (d) After three Arnold transforms.

k represents the number of iterations, where $k = 0, 1, 2$, and so on.

By replacing the discrete lattice for transplantation, the Arnold transform produces a new image after all of the points of the original image have been traversed. In addition to its simplicity, Arnold scrambling also has the properties of being cyclic and irreversible. This implies the facial information is kept entirely after scrambling, even though it appears as a chaotic pattern.

Unlike encryption, scrambling does not really hide information from access. In fact, for surveillance systems, encryption is not encouraged because any unbreakable hiding of information will undermine the purpose of security surveillance. Hence, scrambling is more welcome than encryption in the public surveillance paradigm, where privacy is concerned. It only prevents unwanted exposure of individual faces.

Fig. 2(a) shows a face with its facial components (i.e., eyes, nose, and mouth) circled by different colors. Fig. 2(b) shows the scrambled face after one iteration of the Arnold transform, where it can be seen that facial components have been drastically distorted. Fig. 2(c) and (d) shows the scrambled faces after two and three iterations of the Arnold transform. In comparison with Fig. 2(b), the scrambled faces in (c) and (d) are more difficult to identify by human eyes. In this study, we use three iterations of the Arnold transform to scramble all faces.

B. Challenges in Scrambled Facial Biometric Verification

Classical face recognition algorithms usually can maximize their performance by exploiting facial components. As shown in Fig. 1(a), a face can be easily modeled by a 3-D mesh that can help attain better face recognition accuracy. However, after a face is scrambled, it is even barely recognizable by human eyes. Fig. 3 shows such a case. Before scrambling, faces are easily recognized by the human eye. After scrambling, faces become extremely hard for the human eye to identify or recognize. It is even impossible to find the eyes and mouth in the scrambled patterns. Visual features are somehow randomly scattered in the result space by the scrambling process. As a result, face recognition has to be a pure data-driven classification issue, without utilizing semantic facial components or applying 2-D/3-D face models to the scrambled image.

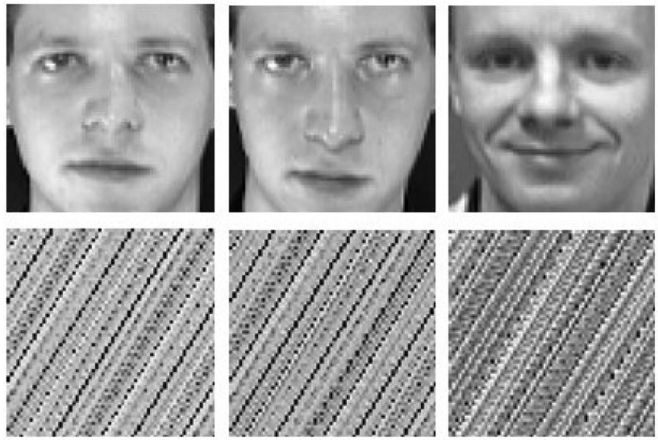


Fig. 3. Although it is easy for a human eye to recognize a face from others, it become extremely difficult in the scrambled domain. First row: three faces of two subjects. Second row: their scrambled facial images.

To find an effective method for this randomly scattered distortion, in this paper, we introduce a fuzzy random forest learning scheme to cope with this challenge. In our method, a random subspace sampling method is applied to extract a subset of features for each fuzzy decision tree. Such random sampling is expected to overcome the scattered distortion and effectively carry out face recognition on a sparse set of features.

III. Forest Learning of Scrambled Facial Biometrics

A. Priori-Based Biased Subspace Sampling

Subspace sampling in random forest reconstruction aims to improve accuracy by exploiting the power of multiple classifiers. In the random subspace selection, a small number of dimensions from a given feature space is selected in each pass, while each classifier is based on the randomized selection of a lower dimensional subspace. With respect to a set of selected subspaces, each tree generalizes its classification in the lower dimensional subspace for both the training data and the test data.

If we select k dimensions out of n, there are $K = n!/\{k!(n\text{-}k)!\}$ such selections that can be made, and with each selection, a decision tree can be constructed. While K can be a large number, for a practical random forest implementation, only a small number of trees (for example 100) are randomly selected to construct a forest. Unlike many other methods suffering from the curse of dimensionality, the high dimensionality of a feature space provides more choices than are needed in practice.

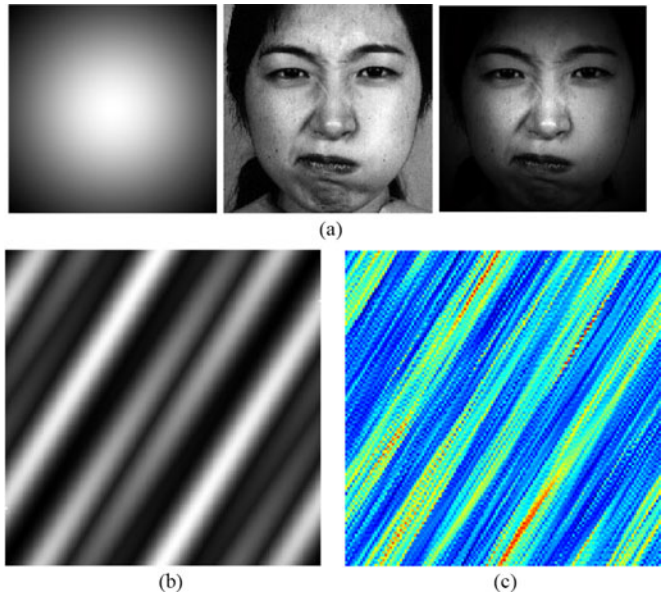


Fig. 4. Biased random sampling based on the center-biased prior map. (a) Center-surround distribution of facial features. (b) Biased weighting in the scrambled domain. (c) Hit map of biased random sampling by 100 trees.

Contrary to the well-known Occam's Razor principle, random forest can take advantage of high dimensionality, and it improves the generalization accuracy as it grows in complexity. Hence, a sophisticated strategy to construct a high-dimensional feature space is usually favored by the random forest method.

In face recognition, human vision usually pays more attention to central features [34] (such as eyes and mouth regions in facial images). As shown in Fig. 4(a), one can give central features more weight, given that mostly central features form the basic inference elements for human vision to recognize a face. Naturally, in this paper, we consider a biazed randomization strategy toward the central facial features. Considering the maximum multiplication factor as ω_s, the repetition of each feature is defined as

$$\omega_k = 1 + \text{round}\left(\omega_s \exp\left(-\sqrt{x^2+y^2}\right)\right). \tag{3}$$

Here, ω_s is a weighting factor, x and y are coordinates normalized to the center of the image, and ω_k is a center-surround weighting map, as shown in the left image in Fig. 4(a). Fig. 4(b) shows the scrambled weight map of the center-surround weight map in Fig. 4(a).

Given the scrambled facial feature space F, and a scrambled priori map ω_k shown in Fig. 4(b), we can then construct a new larger feature space by multiplying each feature according to their importance. Then, we can have a new set of features (pixels or data dimensions) as

$$F_{\text{new}} = \left\{ \underbrace{f_1, \ldots, f_1}_{\omega_1}, \ldots, \underbrace{f_k, \ldots, f_k}_{\omega_k}, \ldots\ldots \right\}. \tag{4}$$

Then, randomization is applied to extract a subset of features from the new feature space F_{new} for each tree to form the forest.

In the random selection procedure, for each pixel, a higher ω_k means higher repetition in F_{new} and so is more likely to be included in each random tree. Fig. 4(c) gives an example of a hit map in the construction of 100 trees, where jet color map is used to visualize the hit map on features f_k. Fig. 5 shows the features randomly selected by 100 trees in the feature space, where each row line stands for a tree, and blue dots denote the selected features from the whole feature space for each tree.

B. Fuzzy Tree Construction in Random Forest

After the features are selected for each tree, we can then construct a fuzzy decision tree based on the selected subspace. For each tree τ_j, we apply a method called local sensitive discriminant analysis (LSDA [35], an extended graph embedding approach similar to LPP [36] and LFDA [37], [38]) to project the selected facial feature space $F^{\{j\}}$ into an eigenvector-based subspace. LSDA has been shown to be an effective method for handling face classification [35]. Compared with LPP, LSDA has fewer parameters to tune and, hence, is easier to use for our purpose.

The decision tree is then constructed in the dimension-reduced eigensubspace. The trees constructed in each selected subspace are fully split using all training data. They are, therefore, perfectly correct on the training set by construction, assuming no intrinsic ambiguities in the samples. There are many kinds of splitting functions for tree construction, such as average mutual information [39], oblique hyperplanes [40], simulated annealing [41], perceptron training [42], or SVM-based hyperplane [31]. Piecewise linear or nearest-neighbor splits can be obtained by various kinds of supervised or unsupervised clustering. There are also many variations of each popular method. Each splitting function defines a model for projecting classification from the training samples on to unclassified points in the space.

In our fuzzy tree construction, we employ a simple piecewise linear split, with a Voronoi tessellation of the feature space. Samples are assigned based on nearest-neighbor matching to chosen anchor points. The anchor points are selected as the training samples that are closest to the class centroids. These trees can have a large number of branches and can be very shallow. The number of leaves is the same as the number of training samples.

Fig. 6 illustrates the fuzzy tree constructed for this purpose. In the fuzzy decision of each tree, the membership of a query sample to each node is computed, and subsequently, a fuzzy membership is computed with respect to every leaf (namely a training sample), and the final output of a fuzzy tree is a vector of memberships to all leaves, instead of a simple binary decision. Consequently, for a fuzzy tree τ_j and an input x, there is an output as a vector of membership; let the probability that x belongs to class $z_k (z_k = 1, 2, \ldots, K_c)$ be denoted by $\hat{P}(z_k|\tau_j(x))$; then, the overall likelihood will be estimated as

$$\alpha(z_k|\tau_j, x) = \frac{P(z_k|\tau_j, x)}{\sum_i P(z_i|\tau_j, x)}. \tag{5}$$

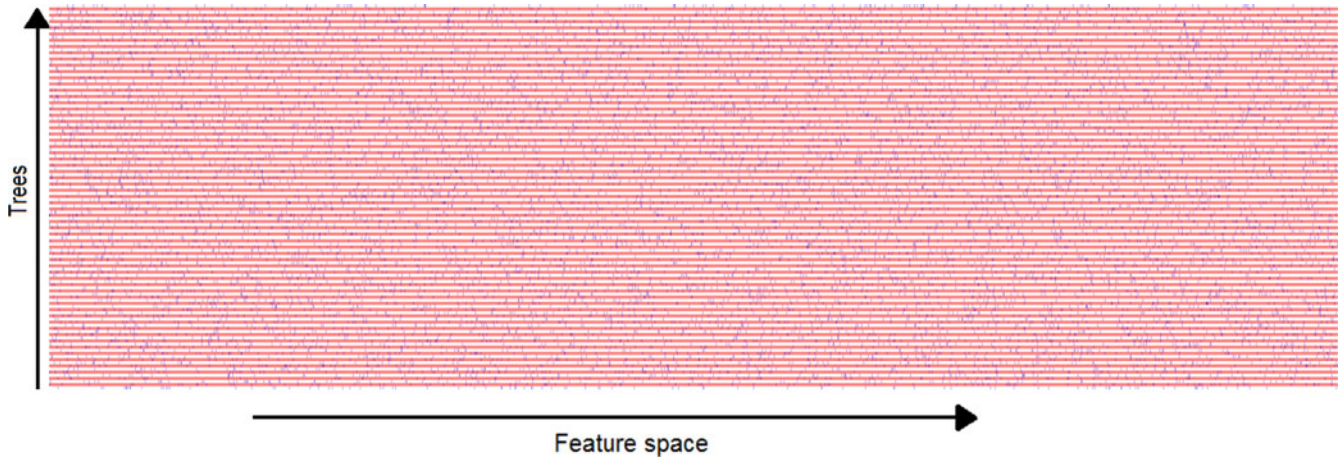


Fig. 5. Example of random subspace selection of 100 trees in the scrambled facial feature space. Each tree selects 5% features only. Each row line stands for a tree, and blue dots denote the selected features from the whole space for each tree.

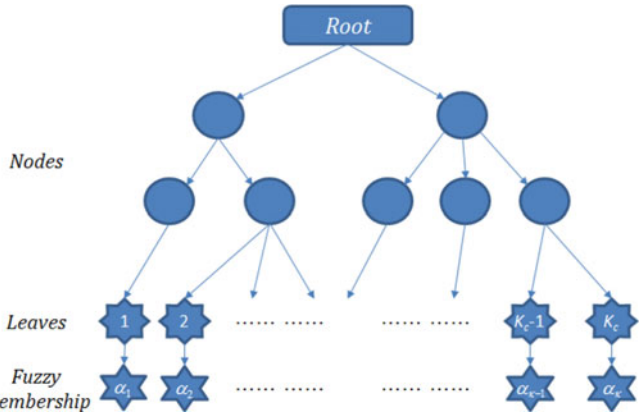


Fig. 6. Fuzzy tree structure used in our forest learning.

which is the fraction of class c points over all points that are assigned to τ_j (x) (in the training set), where z_k denotes the kth leaf in the decision tree.

An obvious merit of using this fuzziness is to avoid wrong decisions being made at the early stage of a single tree, and it gives more space for the optimal forest decision. Fig. 7 shows an example of fuzzy tree decision. In the visualized image, each column line stands for the computed fuzzy memberships from a fuzzy tree. In total, 300 trees are displayed in the image. The color stands for the value of the initial estimated fuzzy membership to a class z_k (corresponding to the vertical coordinate) estimated by a tree τ_j (corresponding to the horizontal coordinate).

IV. Fuzzy Forest Decision

A. Weights of Fuzzy Tree Decision

The process of building a forest from the features leads to many interesting theoretical questions, such as the number of sub-spaces needed to achieve a certain accuracy, the number of randomized trees needed to balance between speed and accuracy, and the way to combine all the trees together. Different trees can be constructed if different feature dimensions are selected at each split, while the use of randomization when selecting the dimensions is merely a convenient way to explore the possibilities.

List I. Fuzzy Forest Learning

Train Procedure:
Input:
  T: Scrambled train dataset;
  L: Labels of the dataset;
Output:
  F: Constructed forest of decision trees;
Process:
Construct a new feature space F_{new} using center-biased map multiplied with the weighting factor ω_s;
Loop for K trees
  Randomly generate n index numbers;
  Using the n index to subsample from F_{new};
  Learn discriminant features via LSDA;
  Construct the tree in the dim-reduced subspace;
End Loop;
Test Procedure:
Input:
  F: Constructed forest of decision trees;
  Q: Scrambled query image;
Output:
  z: the most likely label;
  φ: the final fuzzy memberships to all classes;
Process:
Loop for K trees
  Subsampling into the same subspaces for each tree;
  Project features via LSDA eigenvectors;
  Calculate the membership α_k over all classes;
End Loop;
Compute w_k of each tree using fuzzy membership;
Combine all trees according to their fuzzy weights;
Obtain the final φ and final decision z;

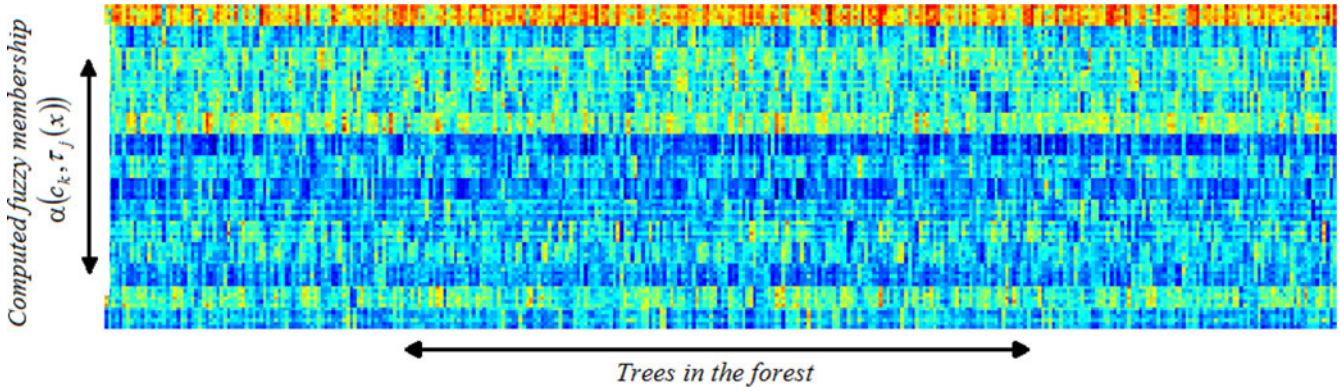


Fig. 7. Fuzzy decisions by each tree in the forest. Each column line stands for the computed fuzzy memberships from a fuzzy tree.

Basically, in the construction of the random forest, an ensemble learning algorithm needs to pay attention to two aspects: 1) how to select proper features/subspaces to generate random trees; and 2) how to guarantee a good combination of tree decisions, which means the decision from each tree needs to be weighted in a rational and effective way.

To combine the decision trees in the random forest for face recognition, we propose a method to weigh a tree via its cross validation in the forest. Given *N* classes and *K* trees, the decision from a tree can be repeated *K/N* times by random chance. We can then estimate the confidence of a tree from its decision by comparing against other trees in the forest by using Kullback–Leibler divergence [14], [36]

$$w_k = \sum_j D_{\mathrm{KL}}\left(\alpha\left(z|\tau_i\right)||\alpha\left(z|\tau_j\right)\right) \tag{6}$$

where

$$D_{\mathrm{KL}}\left(\alpha_m||\alpha_n\right) = \sum_k \alpha_{m,k} \ln \frac{\alpha_{m,k}}{\alpha_{n,k}}. \tag{7}$$

With the above formula, the trees having nonconsensus decision will be given a reduced weight from cross validation via Kullback–Leibler divergence.

B. Fuzzy Forest Decision

A motivation to build multiple classifiers originates from the method of cross validation, where random subsets are selected from the training set and a classifier is trained using each subset. Such methods can help avoid the tantalizing problem of overfitting to some extent by withholding part of the training data. A similar idea has been exploited in bootstrapping [43] and boosting [44]. In boosting, the creation of each subset is dependent on previous classification results, and the final decision combination is based on weighted individual classifiers. Similarly, a random forest consists of a number of trees that need to be combined.

The theory of stochastic discrimination [45] has suggested that classifiers can be constructed by combining many components of weak discriminative power with generalization. Classification accuracies are then related to the statistical properties of the combination function. The capability to build classifiers of arbitrary complexity while increasing generalization accuracy is shared by all this type of methods, and decision forest is one such method.

While the forest is based on random selection of subspaces, it is difficult to determine those trees having better accuracies than others, due to the nature of randomness. In our combination procedure, we use the weighting function in (6), and the fuzzy decision from each tree is then weighted as

$$\widetilde{P}\left(z_k|\tau_j, x\right) = w_k \alpha\left(z_k|\tau_j, x\right). \tag{8}$$

The final discriminant function is defined as

$$\varphi\left(z_k|x\right) = \frac{1}{K}\sum_k \widetilde{P}\left(z_k|\tau_j\left(x\right)\right) \tag{9}$$

and the decision rule is to assign x to class c for which φ is the maximum:

$$z\left(x\right) = \underset{z_k}{\operatorname{argmax}}\, \varphi\left(z_k|x\right). \tag{10}$$

For a random forest, the forest decision is usually based on a plurality vote among the classes decided by each tree. In our scheme, the vote from each tree is fuzzy, and the forest decision is based the combination of weighted memberships estimated from each tree, where odd decisions are neutralized in the fuzzy forest decision process.

Fig. 7 shows an example of the initial estimated fuzzy membership vectors of 100 trees. Here, the membership is visualized by "*jet*" colormap. Fig. 8(a) gives the weighting scores of all 25 trees computed from (6) using Kullback–Leibler divergence among their membership vectors. Fig. 8(b) shows a case where a direct average made a wrong decision and the proposed fuzzy combination corrected it. Here, the test (see Section V) is based on the Yale face dataset. By using the proposed fuzzy combination, the likelihood of the wrong choice at the 13th leaf in the trees is decreased (shown as a red downward arrow) and the correct choice at the 78th leaf is increased (shown as a red upward arrow). As a result, the wrong decision is corrected from the 13th label to the 78th label, thanks to the proposed fuzzy combination.

C. Overview

Fig. 9 gives an overview of the proposed FFL scheme for the scrambled facial verification. Given a training dataset, faces are

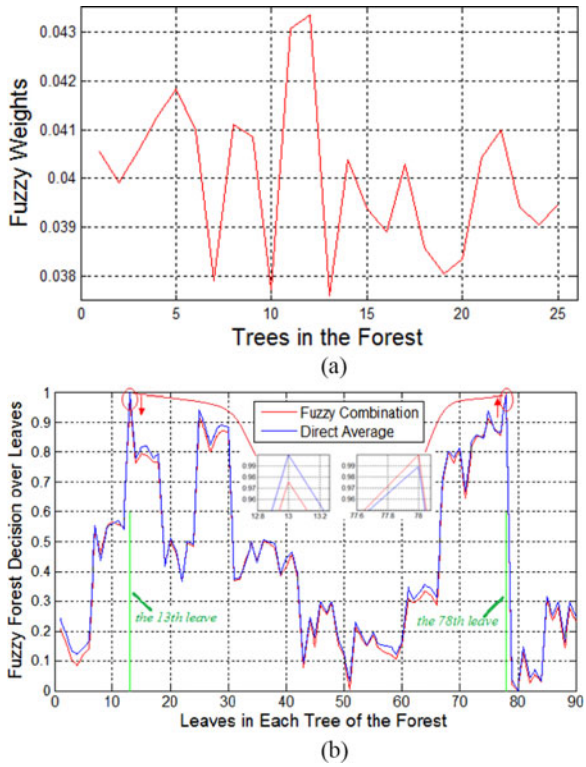


Fig. 8. Fuzzy weights of trees estimated by KL divergence. (a) Computed fuzzy weights of all trees in the forest. (b) Sample case of fuzzy decision versus direct average.

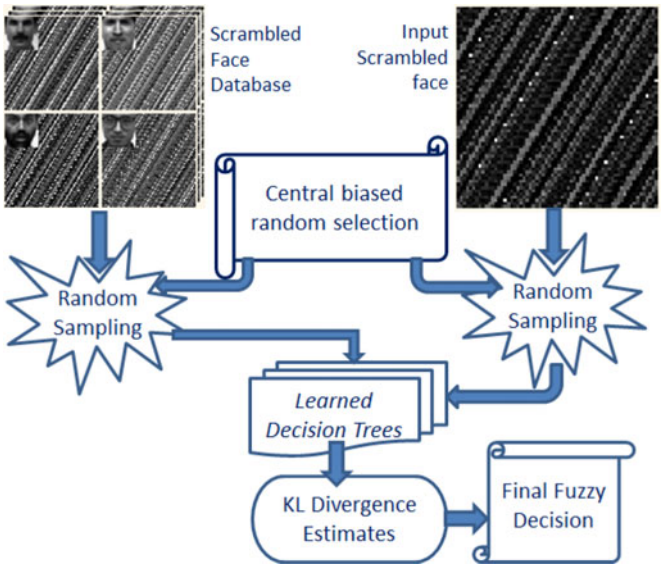


Fig. 9. Schematic view of the proposed approach.

scrambled and forwarded to the FFL scheme. The procedure then randomly selects the features from the scrambled domain with biased weights toward central features, and a number of fuzzy trees are constructed based on the selected features, where LSDA is applied to further extract discriminant features from randomly selected features.

After a scrambled face is input as a test, each tree computes a fuzzy vector of membership and forwards it to the forest decision process. The forest decision procedure then weighs each tree via their total Kullback–Lieder divergences from all other trees, while the final decision is based on a fuzzy combination of all trees. List I gives the pseudocode of the proposed method.

V. Parameters in Fuzzy Forest Learning

Before we go further for experimental validation of our proposed method, we need to answer several critical questions. How many trees are we are going to use? How many features should we select for a tree? What is the best value for the biased factor ω_s in (3)? Does the fuzzy decision via KL divergence really work better than direct averaging? These questions could be pursued to lead to deeper theoretical analysis. In this paper, however, we instead treat these questions in a practical way, and try to optimize these parameters using several experiments.

In our experiment, we ran our tests on the Yale dataset [22]. The Yale dataset has 15 subjects and each subject has six sample faces. With this small dataset, we carried out the face verification experiments by splitting the small dataset into training and test datasets, where the training dataset has five facial images per subject. We then varied the parameters and ran experiments to see which parameter values gave the lowest error rates. Fig. 10 shows our experimental results.

In Fig. 10(a), the bias factor ω_s is varied from 0 (no bias) to 5.5. Here, 100 trees are constructed and the sampling ratio is set to 5%. It can be clearly seen that by increasing the bias factor, the error rate is reduced from 12.0% to 8.8% around $\omega_s = 3.25$. Obviously, from the test, it is shown that the biased sampling did improve the classification accuracy.

In Fig. 10(b), the sample ratio is varied from 0.5% to 10.5%, and it can be seen that the error rate decreases to 8.0% when the sample ratio is tuned from 0.5% to 3.25%, and it then rises back slowly toward the baseline (12%, the error rate for the original LSDA method) when the sample ratio is increased. Here, 100 trees are generated to form the forest and ω_s is set to 3.25. From this experiment, we can also see that the random forest does not necessarily work better than a single tree-based method if its parameters are not selected properly.

Fig. 10(c) gives the experiment on varying the number of trees. Setting the sample ratio to 3.25% and ω_s to 3.25, the number of trees was varied from 3 to 145. We can see that the error rate tends to decrease when the number of trees is increased, and its fluctuation becomes smaller as well. When the number of trees is increased to 80, the error rate is further reduced to 7.7%. Basically, more trees mean more computing time. Provided we have a stable lowest error rate, using fewer trees is usually a favorite choice.

Fig. 10(c) also shows a comparison between direct average (the blue curve) and fuzzy combination (the red curve). It can be seen that fuzzy combination can attain better accuracy consistently in the tests. Fig. 8(b) illustrates how this can be achieved by showing one case in the test of Fig. 10(c). Using the proper fuzzy combination, the likelihood is reduced with respect to the wrong choice (the 13th leaf) and increased with respect to the correct choice (the 78th leaf). Consequently, a correct decision is attained by the fuzzy combination.

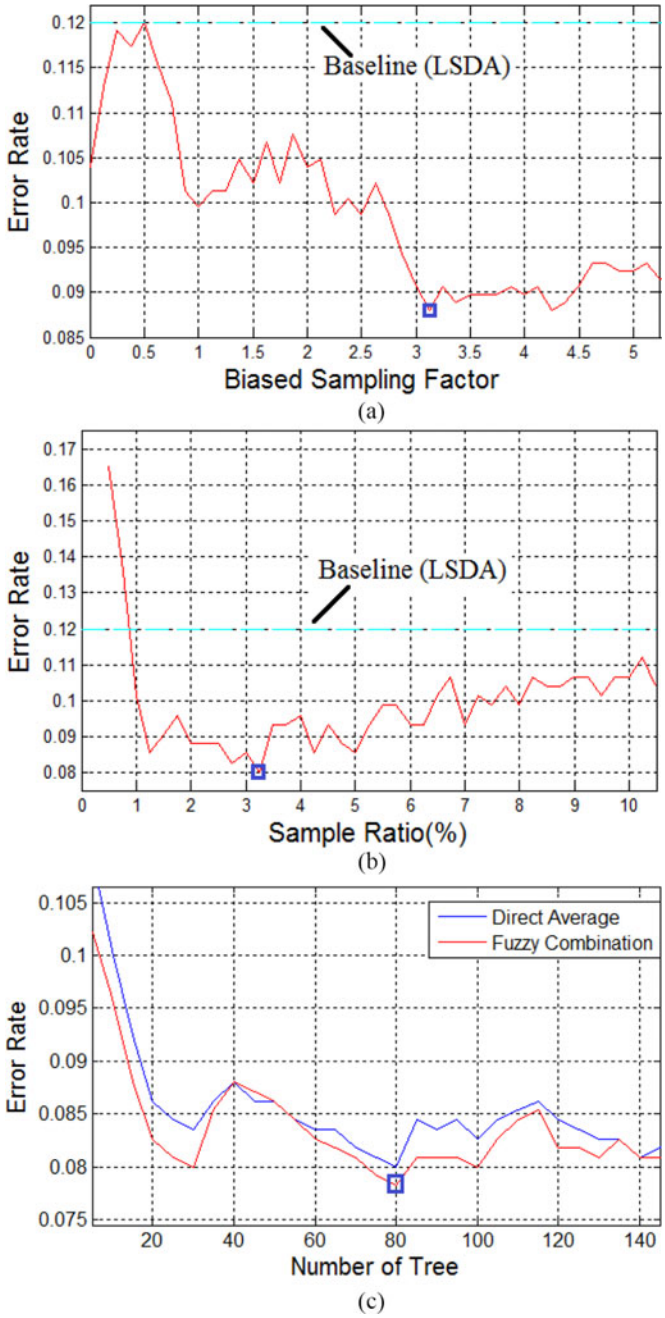


Fig. 10. Learning parameters in our proposed FFL scheme. (a) Effect of varying the biased sampling factor ω_s. (b) Effect of varying the sampling ratio in FFL. (c) Results for different numbers of trees in FFL.

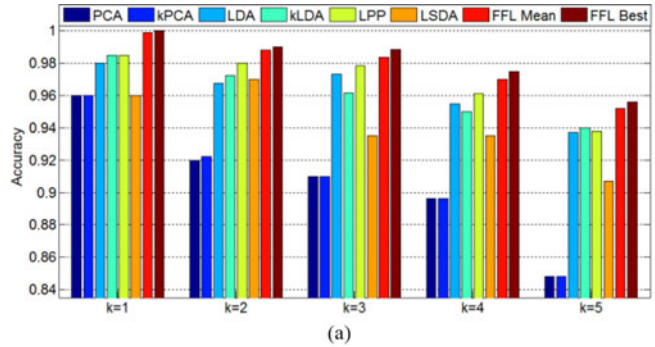
Fig. 11. Three face databases used in our benchmark test. (a) Subjects in the ORL database. (b) Subjects in the PIE database. (c) Subjects in the CK+ dataset.

VI. Experimental Results

A. Experimental Conditions

To investigate the performance of the proposed scheme, we have carried out systematic experiments on three databases: ORL database [46], PIE database [47], and PUBFIG wild face database [48]. Fig. 11 shows typical faces in these databases and their scrambled images. The ORL database has 40 subjects with ten faces each at different poses. The CMU PIE database has 41 368 faces, comprising 68 classes with about 170 faces per class (we use 100 faces per subject, similar to [36]). PUBFIG database [48] contains wild faces selected from internet. It is very similar to LFW database [49], but it provides standard cropped faces. As has been shown [49], background textures in LFW can help achieve a higher accuracy. While we consider facial region recognition only, PUBFIG fits better with our purpose.

In our experiment, all code was implemented in MATLAB, and ran on a PC with 2.7-GHz dual-core Intel CPU. In our experiment, we have used a test scheme called *leave-k-out* [50]. If each subject has N faces in a dataset, we leave k faces out of the training dataset for testing. As a result, the benchmark test will have $(N-k)$ training faces per subject. Selecting k samples from N faces will have C_N^k choices. To make it feasible, we just chose consecutive k faces from N samples, and then, we have N tests in turn for a *leave-k-out* experiment. The accuracy is the average of all N tests. It is noted that the consecutive splitting will usually have a large difference between test and train datasets, because

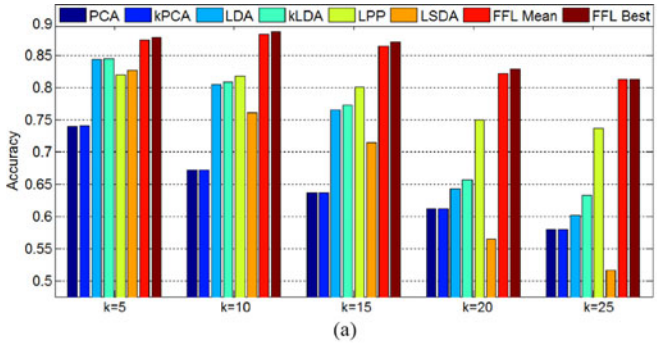


(a)

PCA	kPCA	LDA	kLDA	LPP	LSDA	FFL Mean	FFL Best
90.7	90.7	96.3	96.2	96.8	94.1	97.9	98.2

(b)

Fig. 12. Test results on the ORL dataset. (a) *Leave-k-out* tests. (b) Overall accuracy of all *k tests.*



(a)

PCA	kPCA	LDA	kLDA	LPP	LSDA	FFL Mean	FFL Best
64.8	64.9	73.2	74.3	78.5	67.7	85.1	85.6

(b)

Fig. 13. Test results on the PIE dataset. (a) *Leave-k-out* tests. (b) Over all accuracy of all *k tests.*

faces in datasets usually change consecutively, and the first k faces are usually very different from the last $(N - k)$ faces.

Our benchmark tests aim to verify whether or not our proposed fuzzy random forest learning scheme can enhance the accuracy of face recognition. In our approach, we have proposed a computing-efficient data-driven facial biometric classification method. Hence, we compare our approach with a number of typical data-driven methods, including PCA [21], LDA [22], kPCA [23], kLDA [23], LPP [36], and LSDA [35].

Because random forest is based on random feature selection, it may give a different accuracy each time. As a result, we ran each test ten times and recorded the average (FFL-Mean) and maximum (FFL-Best) accuracies for comparison. It is also noted that for each test, there were N subtests due to the different selection of the *leave-k-out* scheme. Therefore, in total, we have $10 \times N$ subtests for each *leave-k-out* test.

B. Validation on ORL Dataset

The ORL database has ten faces per subject. In our *leave-k-out* test, k varies from 2 to 6. In total, we have five k-tests, where each k-test has 10×10 subtests. The final accuracy is the average on all 100 subtests.

Fig. 12(a) shows all *leave-k-out* tests, where k varied from 2 to 6. We can see that our proposed FFL method attained the best accuracy in all k tests. It is also observed that FFL-mean is very close to FFL-best. In our test, each forest has nearly 80 trees, and such a large number of trees can effectively quench the stochastic fluctuation in the random feature selection. Besides, each k-test has 10×10 subtests, making it statistically similar.

Fig. 12(b) lists the overall accuracy by averaging all k tests. Here, we included PCA, LDA, kPCA, kLDA, LPP, and LSDA for comparison because they are typical data-driven face recognition technology. We can see that our FFL attained the best accuracy over all k-tests—around 98%, while LPP came second to this at 96.8%. kLDA and LDA attained similar accuracy around 96.2%, LSDA attained 94.1%, and kPCA and PCA had an accuracy of 90.7%.

C. Validation on PIE Dataset

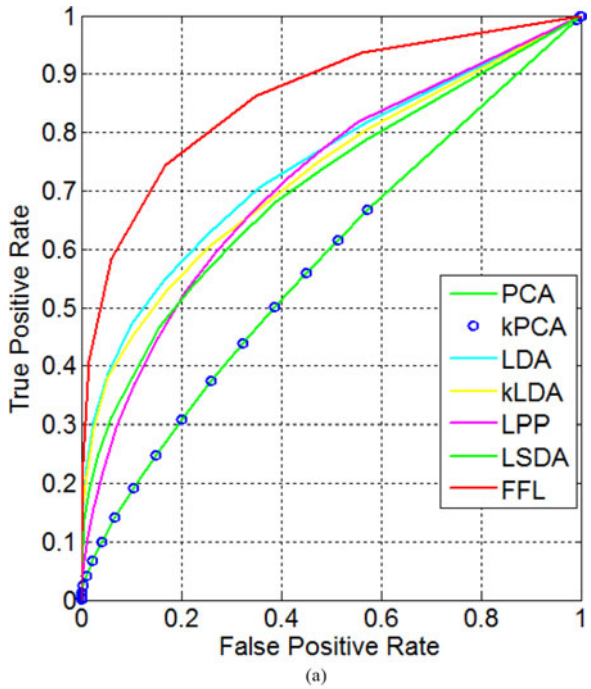
In this benchmark test, 50 faces per subject and in total 3350 faces from the PIE dataset were used. In the test scheme, k faces from 50 samples per subject are selected as test samples, and the rest are used as training samples. In our experiment, we repeatedly selected k faces (consecutively) from 50 samples ten times and carried out ten subtests per k test. Random forest can vary from time to time due to its random mechanism. As before, for each subtest, we ran ten times and used both average and best accuracy to evaluate our FFL classifier.

Fig. 13(a) shows all *leave-k-out* tests on the PIE dataset, where k varied from 5 to 25. We can see that our proposed FFL method attained the best accuracy in all k tests. It is also observed that FFL-mean is very close to FFL-best. It is also noticed that when k rises to 25, LSDA, LDA, and kLDA have the largest drop in accuracy. In comparison, the proposed FFL method attained steady performance even when the number of available training samples is reduced.

Fig. 13(b) lists the overall accuracy by averaging all k tests. We can see that PCA and kPCA have the lowest accuracy of around 65%, LSDA has an accuracy around 67.7%, LDA and kLDA attained similar accuracy around 74%, and LPP attained an accuracy at 78.5%. In comparison, our FFL method attains a far better accuracy of around 85% in this test.

D. Validation on PUBFIG Dataset

The PUBFIG dataset has been developed for benchmark tests to compare various algorithms against the human vision system. Its typical benchmark test can have as many as 20 000 pairs of faces for comparison. However, in the surveillance-targeted scrambled domain, human perception can barely recognize any scrambled faces, making it meaningless to carry out this human-targeted comparison test. On the other hand, in surveillance applications, users (such as police) usually have a set of wanted faces in their training datasets on the server side, making it more like a *leave-k-out* experiment. For this reason, we need to design a suitable evaluation scheme for this work.



(a)

PCA	kPCA	LDA	kLDA	LPP	LSDA	FFL
30.7	30.7	57.7	56.3	52.1	48.6	76.6

(b)

Fig. 14. Test results on PUBFIG wild faces. (a) TPR-FPR curves. (b) TPR at FPR = 20%.

In our experiments, we have selected 52 subjects with 60 faces each and split them randomly into test and training datasets, with each having $30 \times 52 = 1560$ faces. We have then test all data-driven methods by comparing each test face against all training faces. In total, we have $1560 \times 1560 = 2.4$ million pairs of estimated likelihood values, which form a likelihood matrix of 1560×1560 elements. Then, we have varied the thresholds on the likelihood matrix and counted how many pairs below the threshold are false positive and/or true positive. False positive rates (FPR) and true positive rates (TPR) can then be computed accordingly, and we can have the ROC curves (FP versus TP) as our evaluation criteria.

Fig. 14(a) gives our test results on all methods. It is observed that PCA has given worse performance than it did on LFW [49]. This implies that this test is even harder than the standard LFW test in [48] (at least it is true for eigenfaces). From the comparison results, we can clearly see that the proposed FFL method appears to have better performance in this test on real-world faces, with significantly better TPR consistently over other data-driven methods. Fig. 14(b) gives the TPR at FPR = 20%. PCA and kPCA attain a low accuracy of around only 30%, while FFL attains an accuracy of around 76.6%, about 20% higher than LSDA, LPP, LDA, and kLDA.

VII. Conclusion

In this paper, we have successfully developed a robust FFL scheme for facial biometric verification in the scrambled domain. In our proposed scheme, to extract the features from scrambled face images robust, a biased random subspace sampling scheme is applied to construct fuzzy decision trees from randomly selected features. Then, a fuzzy forest decision is obtained from all fuzzy trees by the weighted combination of their fuzzy decision vectors of membership. Our experiments using three public datasets have successfully validated that the proposed FFL scheme can robustly cope with challenging tests in the scrambled domain, and it consistently attained the best accuracy over all datasets, making our method a promising candidate for emerging privacy-related facial biometric applications, especially for public visual surveillance systems where face scrambling is applied.

It is worth highlighting that our approach is not dependent on any semantic face models or 3-D templates. Although face specific features targeted toward semantic/3-D face modeling can enhance accuracy, face modeling from images and facial component detection needs extra computation time and can also easily introduce extra errors. Instead, our approach is based purely on data-driven classification and can easily be applied to other similar chaotic pattern classification cases, such as texture classification in image analysis or factor analysis of stock prices. In our future work, we plan to investigate the use of our method in these applications.

References

[1] A. Melle and J.-L. Dugelay, "Scrambling faces for privacy protection using background self-similarities," in *Proc. IEEE Int. Conf. Image Process.*, 2014, pp. 6046–6050.
[2] Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft, "Privacy-preserving face recognition," in *Proc. 9th Int. Symp. Privacy Enhancing Technol.*, 2009, pp. 235–253.
[3] T. Honda, Y. Murakami, Y. Yanagihara, T. Kumaki, and T. Fujino, "Hierarchical image-scrambling method with scramble-level controllability for privacy protection," in *Proc. IEEE 56th Int. Midwest Symp. Circuits Syst.*, 2013, pp. 1371–1374.
[4] S. Hosik, W. De Neve, and Y. M. Ro, "Privacy protection in video surveillance systems: Analysis of subband-adaptive scrambling in JPEG XR," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 2, pp. 170–177, Feb. 2011.
[5] F. Dufaux and T. Ebrahimi, "Scrambling for video surveillance with privacy," in *Proc. Conf. Comput. Vision Pattern Recog. Workshop*, Washington, DC, USA, 2006, pp. 106–110.
[6] F. Dufaux, "Video scrambling for privacy protection in video surveillance: Recent results and validation framework," *Proc. SPIE*, vol. 8063, p. 806302, 2011.
[7] T. Winkler and B. Rinner, "Security and privacy protection in visual sensor networks: A survey," *ACM Comput. Surveys*, vol. 47, no. 1, p. 2, 2014.
[8] A. Erdlyi, T. Bart, P. Valet, T. Winkler, and B. Rinner, "Adaptive cartooning for privacy protection in camera networks,"in *Proc. Int. Conf. Adv. Video Signal Based Surveillance*, 2014, pp. 44–49.
[9] Y. Wang and T. Li, "Study on image encryption algorithm based on arnold transformation and chaotic system,"in *Proc. 2010 Int. Conf. Intell. Syst. Des. Eng. Appl.*, 2010, pp. 449–451.
[10] Z. Tang and X. Zhang, "Secure image encryption without size limitation using arnold transform and random strategies," *J. Multimedia*, vol. 6, no. 2, pp. 202–206, Apr. 2011.
[11] R. Jiang, A. H. Sadka, and D. Crookes, "Multimodal biometric human recognition for perceptual human-computer interaction," *IEEE Trans. Syst. Man Cybern. C, Appl. Rev.*, vol. 40, no. 6, pp. 676–681, Nov. 2010.
[12] Z. Ju and H. Liu, "A unified fuzzy framework for human-hand motion recognition," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 5, pp. 901–913, Oct. 2011.
[13] D. J. Kim and Z. Bien, "Design of 'personalized' classifier using soft computing techniques for 'personalized' facial expression recognition," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 4, pp. 874–885, Aug. 2008.

[14] M. Rashid, S. A. R. Abu-Bakar, and M. Mokji, "Human emotion recognition from videos using spatio-temporal and audio features," *Visual Comput.*, vol. 29, no. 12, pp. 1269–1275, Dec. 2013.
[15] M. L. Gao, L. L. Li, X. M. Sun, and D. S. Luo, "Face tracking based on differential harmony search," *IET Comput. Vision*, p. 12, Jun. 2014.
[16] R. Jiang, D. Crookes, and N. Luo "Face recognition in global harmonic subspace," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 416–424, Sep. 2010.
[17] H. Chang, Y. Yao, A. Koschan, B. Abidi, and M. Abidi, "Improving face recognition via narrowband spectral range selection using jeffrey divergence," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 1, pp. 111–122, Mar. 2009.
[18] A. Ghazanfar and D. Takahashi, "Facial expressions and the evolution of the speech rhythm," *J. Cognitive Neurosci.*, vol. 26, no. 6, pp. 1196–1207, Jun. 2014.
[19] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.
[20] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, 2014, pp. 1701–1708.
[21] B. Draper, K. Baek, M. Bartlett, and J. Beveridge, "Recognizing faces with PCA and ICA," *Comput. Vision Image Understanding*, vol. 91, nos. 1/2, pp. 115–137, 2003.
[22] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.
[23] M. H. Yang, "Kernel eigenfaces vs. kernel fisherface: Face recognition using kernel methods," in *Proc. Int. Conf. Autom. Face Gesture Recog.*, 2002, p. 215.
[24] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 681–685, Jun. 2001.
[25] R. Hsu and A. Jain, "Semantic face matching," in *Proc. IEEE Int. Conf. Multimedia Expo.*, 2002, pp. 145–148.
[26] O. Ibañez, O. Cordon, S. Damas, and J. Santamaria, "Modeling the skull–face overlay uncertainty using fuzzy sets," *IEEE Trans. Fuzzy Systems*, vol. 19, no. 5, pp. 946–959, Oct. 2011.
[27] S. Mian, M. Bennamoun, and R. Owens, "An efficient multimodal 2d-3d hybrid approach to automatic face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 11, pp. 1927–1943, Nov. 2007.
[28] Y. W. Pang, X. L. Li, Y. Yuan, D. C. Tao, and J. Pan, "Fast haar transform based feature extraction for face representation and recognition," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 3, pp. 441–450, Sep. 2009.
[29] X. Zhao and S. Zhang, "Facial expression recognition based on local binary patterns and kernel discriminant isomap," *Sensors*, vol. 11, no. 10, pp. 9573–9588, 2011.
[30] R. Jiang, M. Parry, P. Legg, D. Chung, and I. Griffiths, "Automated 3D animation from snooker videos with information theoretic optimization," *IEEE Trans. Comput. Intell. AI Games*, vol. 5, no. 4, pp. 337–345, Dec. 2013.
[31] L. Breiman, "Random forests," *Mach. Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001.
[32] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 832–844, Aug. 1998.
[33] L. Ding, X. Ding, and C. Fang, "Continuous pose normalization for pose-robust face recognition," *IEEE Signal Process. Lett.*, vol. 19, no. 11, pp. 721–724, Nov. 2012.
[34] R. Jiang and D. Crookes, "Visual saliency estimation through manifold learning," in *Proc. Natl. Conf. Artif. Intell.*, 2012, pp. 2773–2779.
[35] D. Cai, X. He, K. Zhou, J. Han, and H. Bao, "Locality sensitive discriminant analysis," in *Proc. 20th Int. Joint Conf. Artif. Intell.*, 2007, pp. 708–713.
[36] X. He, S. Yan, Y. Hu, P. Niyogi, and H. J. Zhang, "Face recognition using Laplacian faces," *IEEE Trans. Pattern Analysis Mach. Intell.*, vol. 27, no. 3, pp. 328–340, Mar. 2005.
[37] Y. Rahulamathavan, R. C.-W. Phan, J. A. Chambers, and D. J. Parish, "Facial expression recognition in the encrypted domain based on local fisher discriminant analysis," *IEEE Trans. Affective Comput.*, vol. 4, no. 1, pp. 83–92, Jan.–Mar. 2013.
[38] M. Sugiyama, "Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis," *J. Mach. Learning Res.*, vol. 8, pp. 1027–1061, 2007.
[39] I. K. Sethi and G. P. R. Sarvarayudu, "Hierarchical classifier design using mutual information," *IEEE Trans. Pattern Anal.Mach. Intell.*, vol. PAMI-4, no. 4, pp. 441–445, Jul. 1982.
[40] Y. Park and J. Sklansky, "Automated design of multiple-class piecewise linear classifiers," *J. Classification*, vol. 6, pp. 195–222, 1989.
[41] S. Murthy, S. Kasif, and S. Salzberg, "A system for induction of oblique decision trees," *J. Artif. Intell. Res.*, vol. 2, no. 1, pp. 1–32, 1994.
[42] L. Breiman, "Bagging predictors," *Mach. Learning*, vol. 24, pp. 123–140, 1996.
[43] L. Breiman, "Arcing classifier," *Ann. Statist.*, vol. 26, no. 3, pp. 801–849, 1998.
[44] L. Mason, J. Baxter, P. Bartlett, and M. Frean, "Boosting algorithms as gradient descent," in *Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 2000, vol. 12, pp. 512–518,
[45] E. M. Kleinberg, "An overtraining-resistant stochastic modeling method for pattern recognition," *Ann. Statist.*, vol. 4, no. 6, pp. 2319–2349, Dec. 1996.
[46] F. Samaria and A. Harter, "Parameterisation of a stochastic model for human face identification," presented at the Proc. 2nd IEEE Workshop Applications Appl. Computer Comput. Vision, Sarasota, FL, USA, Dec. 1994.
[47] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression (PIE) database," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, May 2002, pp. 46–51.
[48] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *Proc. Int. Conf. Comput. Vision*, 2009, pp. 365–372.
[49] G. B. Huang, V. Jain, and E. Learned-Miller, "Unsupervised joint alignment of complex images," in *Proc. IEEE 11th Int. Conf. Comput. Vision*, 2007, pp. 1–8.
[50] G. Cawley and N. Talbot, "Efficient leave-one-out cross-validation of kernel fisher discriminant classifiers," *Pattern Recog.*, vol. 36, no. 11, pp. 2585–2592, 2003.

Richard Jiang received the Ph.D. degree in computer science from Queen's University Belfast, Belfast, U.K., in July 2008.

After his Ph.D. study, he has worked with Brunel University, Loughborough University, Swansea University, University of Bath, and University of Sheffield. He joined Northumbria University, Newcastle upon Tyne, U.K., in May 2013, where he is currently a Lecturer with the Department of Computer Science and Digital Technologies. His research interests include the fields of artificial intelligence, man–machine interaction, visual forensics, and biomedical image analysis. His research has been funded by the Engineering and Physical Sciences Research Council, the Biotechnology and Biological Sciences Research Council, TSB, EU Framework Program, and industry funds. He has authored and coauthored more than 40 publications.

Ahmed Bouridane (M'98–SM'06) received the "Ingenieur d'État" degree in electronics from Ecole Nationale Polytechnque of Algiers, Algiers, Algeria, in 1982, the M.Phil. degree in electrical engineering (VLSI design for signal processing) from Newcastle University, Newcastle upon Tyne, U.K., in 1988, and the Ph.D. degree in electrical engineering (computer vision) from the University of Nottingham, Nottingham, U.K., in 1992.

From 1992 to 1994, he was a Research Developer in telesurveillance and access control applications. In 1994, he joined Queen's University Belfast, Belfast, U.K., initially as a Lecturer in computer architecture and image processing and then as a Reader in computer science. He became a Professor in image engineering and security with Northumbria University, Newcastle upon Tyne, U.K., in 2009. His research interests include imaging for forensics and security, biometrics, homeland security, image/video watermarking, and cryptography. He has authored and coauthored more than 200 publications.

Danny Crookes (M'98–SM'12) received the B.Sc. degree in Mathematics and Computer Science in 1977, and the Ph.D. degree in Computer Science in 1980, both from Queen's University Belfast.

He became Professor of Computer Engineering in 1993 at Queen's University Belfast, Belfast, U.K., and was Head of Computer Science from 1993–2002. He is currently Director of Research for Speech, Image and Vision Systems at the Institute of Electronics, Communications and Information Technology, Queen's University Belfast. His current research interests include the use of novel architectures (especially GPUs) for high performance speech and image processing. Professor Crookes is currently involved in projects in automatic shoeprint recognition, speech separation and enhancement, and processing of 4D confocal microscopy imagery. Professor Crookes has published 200 scientific papers in journals and international conferences.

M. Emre Celebi (S'05–M'07–SM'11) received the B.Sc. degree in computer engineering from the Middle East Technical University, Ankara, Turkey, in 2002, and the M.Sc. and Ph.D. degrees in computer science and engineering from the University of Texas at Arlington, Arlington, TX, USA, in 2003 and 2006, respectively.

He is currently an Associate Professor with the Department of Computer Science, Louisiana State University in Shreveport, Shreveport, LA, USA. He has conducted research in the field of image processing and analysis. He has published more than 120 articles in journals and conference proceedings. His recent research is funded by grants from the National Science Foundation.

Hua-Liang Wei received the B.Sc. degree in mathematics from Liaocheng University, Liaocheng, China, in 1989, the M.Sc. degree in automatic control theory and applications from the Beijing Institute of Technology, Beijing, China, in 1992, and the Ph.D. degree in signal processing and control engineering from the University of Sheffield, Sheffield, U.K., in 2004.

He previously held academic appointments (Lecturer and Associate Professorship) from 1992 to 2000 with the Beijing Institute of Technology. He joined the Department of Automatic Control and Systems Engineering, The University of Sheffield, in 2004, where he is currently a Lecturer. His current research interests include modeling and identification methods for nonlinear systems; NARMAX methodology and its applications; signal and image processing; pattern recognition; spatiotemporal systems; neuroimaging, data modeling and analysis; wavelets and neural networks; applications of signal processing, system identification, and modeling in complex system analysis including systems/synthetic biology and biomedical engineering; forecasting of stochastic and dynamical processes; regression analysis; and linear and nonlinear optimization.