

Robust Face Recognition Using Probabilistic Facial Trait Code

Ping-Han Lee[†], Gee-Sern Hsu[§], Szu-Wei Wu[‡], and Yi-Ping Hung[‡]

[†]Department of Computer Science and Information Engineering, National Taiwan University

[‡]Graduate Institute of Networking and Multimedia, National Taiwan University

[§]Department of Mechanical Engineering, National Taiwan University of Science and Technology

Abstract. Recently, Facial Trait Code (FTC) was proposed for solving face recognition, and was reported with promising recognition rates. However, several simplifications in the FTC encoding make it unable to handle the most rigorous face recognition scenario in which only one facial image per individual is available for enrollment in the gallery set and the probe set includes faces under variations caused by illumination, expression, pose or misalignment. In this study, we propose the Probabilistic Facial Trait Code (PFTC) with a novel encoding scheme and a probabilistic codeword distance measure. We also propose the Pattern-Specific Subspace Learning (PSSL) scheme that encodes and recognizes faces robustly under aforementioned variations. The proposed PFTC was evaluated and compared with state-of-the-art algorithms, including the FTC, the algorithm using sparse representation, and the one using Local Binary Pattern. Our experimental study considered factors such as the number of enrollment allowed in the gallery, the variation among gallery or probe set, and reported results for both identification and verification problems. The proposed PFTC yielded significant better recognition rates in most of the scenarios than all the states-of-the-art algorithms evaluated in this study.

1 Introduction

Face recognition remains a popular topic in the recent two decades, and many algorithms were proposed. Ahonen et al. proposed using Local Binary Pattern (**LBP**) for face recognition [1]. In their algorithm, a face was spatially decomposed into several non-overlapping blocks. The histograms of the local binary patterns extracted from these blocks were concatenated to form a new feature vector. For the application to face recognition, the distance between two faces was evaluated using the *weighted Chi square distance* between their feature vectors. The LBP approach has been reported to be quite robust to the facial variations caused by illumination and expression changes [1]. A face recognition algorithm using Sparse Representation proposed by Wright et al. [2] has aroused some attention recently. They treated a test sample as a sparse linear

combination of training samples, and computed this sparse representation using l^1 -minimization. The face recognition algorithm proposed in [2], denoted as **SRC**, yielded robust results when faces were occluded or corrupted. It is worthy to mention that both algorithms, LBP and SRC, do not involve a training stage that learns knowledge specific to human faces.

In contrast to LBP and SRC, the Facial Trait Code (**FTC**) proposed by Lee et al. learned some face-specific knowledge. It has been shown in [3] that patterns exist in many of the facial components and can be extracted, and the extracted patterns can be used to decompose and encode a face. The facial components with patterns good for discriminating faces were called *facial traits*, and the associated face coding scheme was called the *Facial Trait Code*. It is reported the patterns learned from a small set of human faces can be generalized to human face unseen in the training stage, and FTC yielded comparable face identification rates with the LBP approach, and significantly better verification rates [3].

Although the three algorithms were reported to be successful, they are limited in some aspects. The SRC approach assumed any probe, an image in the probe set¹, could be represented as a linear combination of the gallery images. It is not known whether SRC performs well when there is only few, or even one image per individual allowed for enrollment in gallery. Like SRC, the LBP approach does not learn any illumination- or expression-invariants among human faces. If the probes were taken under conditions (i.e. illumination or facial expression) significantly different from those the images in gallery were taken, the performance of the LBP approach is expected to degrade noticeably. FTC does not require many samples per individual for enrollment, and it learns a way to encoding faces robustly under illumination variation. However, it suffers from the following aspects:

1. Because the distance measure in the FTC code space is given by Hamming distance, the similarities between one facial trait patch and all facial trait patterns except the most similar one are ignored.
2. Because the impacts from illumination, expression, and pose² variations upon the facial trait patterns have not been studied thoroughly, a systematic way to encompass these impacts into the FTC has yet to develop.
3. Because of the above and a few simplifications, the FTC cannot handle the most rigorous face recognition scenario in which only one facial image per individual is available for enrollment in the gallery set and the probe set includes faces under variations in illumination, expression and pose (e.g. John has a smiling, uniformly-lit face enrolled in the database, and we want to recognize John's face which is left-lit and with neutral expression).

¹ According to the FRVT 2006[4], the target set contains facial images for enrollment to the face recognition system, and the probe set contains images to be recognized. If only one image per individual is allowed for enrollment, then the target set is called the gallery set. In this paper, we do not distinguish between the target set and the gallery, and we will use the term *gallery* in the rest of this paper.

² Or, out-of-plane rotation. We will use the two terms interchangeably in the rest of this paper.

In this paper, we propose the *Probabilistic Facial Trait Code* (PFTC) that solves all of the above issues. A comprehensive performance evaluation of some state-of-the-art algorithms as well as the proposed PFTC are also given. All the aforementioned algorithms, LBP, SRC and FTC, were evaluated under several different scenarios, which differ in the number of enrollment allowed in the gallery set, the variations among the gallery and the probe set, and the definitions of training set, gallery and probe set. The results in AR dataset [5] in which faces were taken under *controlled* variations, as well as a mixed dataset in which faces were taken *uncontrolled* conditions were given. The proposed PFTC yielded significantly better recognition rates than the three recent face recognition algorithms in most of the scenarios considered in this paper.

This paper begins with an introduction to Facial Trait Code (Sec. 2). The development of the Probabilistic Facial Trait Code was given in Sec. 3. A comparative study on the face recognition performance using the PFTC and other algorithms were reported in Sec. 4. The conclusion and contribution of our study were summarized in Sec. 5.

2 Introduction to Facial Trait Code

Two face datasets are needed for the construction of FTC, one is the *Trait Extraction Set*, denoted as **TES**, and the other the *Trait Variation Set*, denoted as **TVS**. The former consists of a large number of frontal facial images with neural expression and evenly distributed illumination, and is used to determine the facial traits and the patterns in each trait (Sec. 2.1). The latter consists of facial images taken under various illumination conditions, and is used as an add-on to the facial trait samples so that each trait pattern can have samples with illumination changes (Sec. 2.2). For the Facial Trait Code and the proposed Probabilistic FTC, samples in **TES** were used to extract *patch patterns* and to select *facial traits*. When facial traits and associated patterns were determined, samples in both **TVS** and **TES** were used to train trait-specific SVMs.

2.1 Facial Trait Extraction and Associated Codewords

A local patch on a face can be specified by a rectangle bounding box $\{x, y, w, h\}$, where x and y are the 2-D pixel coordinates of the bounding box's upper-left corner, and w and h are the width and height of this bounding box, respectively. A large amount of patches with different sizes and locations on faces can be defined, and slightly more than a couple thousands of patches for a face with 80x100 pixels in size are used in [3]. In the following, we assume M patches in total obtained from a face.

Assuming K faces available from the TES, and all faces aligned by the centers of both eyes, we will obtain a stack of K patch samples in each patch. To cluster the K patch samples in each patch stack, the Principal Component Analysis (PCA) followed by the Linear Discriminant Analysis (LDA) [6] are applied to extract the features. It is assumed that these low dimensional patch features in

each patch stack can be modeled by a Mixture of Gaussian (MoG), then the unsupervised clustering algorithm proposed by Figueiredo and Jain [7] can be applied to identify the MoG patterns in each patch stack. Assuming M patch stacks are available, this algorithm can cluster the low dimensional patch features into k_i clusters in the i -th patch stack, where $i = 1, 2, \dots, M$. The k_i clusters in the i -th patch stack were considered the patterns existing in this patch stack, and they are called the **patch patterns**.

A scheme was proposed in [3] that selects some combination of the patches with their patch patterns able to best discriminate the individuals in the TES by their faces. This scheme first define a matrix, called **Patch Pattern Map** (PPM), for each patch. PPM shows which individuals' faces reveal the same pattern at that specific patch. Let PPM_i denote the PPM for the i -th patch, $i = 1, 2, \dots, M$. PPM_i will be $L \times L$ in dimension in the case with L individuals, and the entry at (p, q) , denoted as $PPM_i(p, q)$, is defined as follows: $PPM_i(p, q) = 0$ if the patches on the faces of the p -th and the q -th individuals are clustered into the same patch pattern and $PPM_i(p, q) = 1$ otherwise.

Given N patches and their associated PPM_i 's stacked to form a $L \times L \times N$ dimensional array, there are $L(L-1)/2$ N -dimensional binary vectors along the *depth* of this array because each PPM_i is symmetric matrix and one can only consider the lower triangular part of it. Let $v_{p,q}$ ($1 \leq q < p \leq L$) denote one of the N -dimensional binary vectors, then $v_{p,q}$ reveals the local similarity between the p -th and the q -th individuals in terms of these N local patches. More unities in $v_{p,q}$ indicates more differences between this pair of individuals, and on the contrary, more zeros shows more similarities in between.

The binary vector $v_{p,q}$ motivated the authors in [3] to apply the Error Correcting Output Code (ECOC) [8] to their study. If each individual's face is encoded using the most discriminant patches, then the induced set of $[v_{p,q}]_{1 \leq q < p \leq L}$ can be used to define the minimum and maximum Hamming distance [9] among all encoded faces in the corresponding code space. The $v_{p,q}$ with the least (most) of unities gives the minimum (maximum) Hamming distance. To maximize the robustness against possible recognition errors in the decoding phase, authors in [3] proposed an Adaboost algorithm to maximize the d_{min} , the minimum Hamming distance, for the determination of the most discriminating from the overall patches³. Patches that best discriminate faces of different individuals were called the **facial traits**, and the associated patch patterns were dubbed as the **distinctive trait patterns**.

Assuming N facial traits selected from the the overall M patches, and each has k_j , $j = 1, 2, \dots, N$, trait patterns, one can now define the codewords in FTC. Each codeword is of length N and n -ary where n is the largest number of the trait patterns found in one single trait, and each digit in a codeword is an integer number indicating a trait pattern. In summary, given a large collection of faces as the TES, one can define N facial traits, $\sum_{j=1}^N k_j$ trait patterns, and $\prod_{j=1}^N k_j$ faces (or FTC codewords).

³ Due to the page limit, please refer to [3] for the details on this Adaboost algorithm.

2.2 FTC Encoding and Decoding

To apply the FTC to face recognition, the images in a gallery set are firstly encoded into **gallery codes** using a trait-specific SVM (Support Vector Machines) classifier [10] able to classify each facial trait into a symbolized trait pattern. This SVM classifier can be made using the trait samples from both the TES and the TVS for encompassing possible variations in each trait. In the decoding phase when a probe, an image from a probe set, is given, it is also firstly encoded into a **probe code**, and then matched against the gallery codes using Hamming distance as the measure. Given two codewords, one is a gallery code $\mathbf{g}_c = [g_1 g_2 \dots g_N]$ and the other probe code $\mathbf{p}_c = [p_1 p_2 \dots p_N]$, the Hamming distance is give by the code difference $\mathbf{d}_c = [d_1 d_2 \dots d_N]$ where $d_i = 0$ if $p_i = g_i$, and $d_i = 1$ otherwise. Then the Hamming distance between \mathbf{g}_c and \mathbf{p}_c is given by $D(\mathbf{g}_c, \mathbf{p}_c) = \sum_{i=1}^N d_i$.

3 Probabilistic Facial Trait Code

FTC suffers from the following aspects: the simplified integer codewords ignore the similarities between one facial trait patch and all facial trait patterns; a systematic way to handle the impacts from illumination, expression, or pose is yet to be developed for robust encoding. In this paper, we proposed the Probabilistic Facial Trait Code, or **PFTC**, that considers and solves the above issues. The primary differences from the original FTC include the following:

1. Instead of using only one integer to denote a facial trait in the encoding phase, we allow the probabilities of the trait belonging to all trait patterns in the codeword. The gain, to be shown in our experimental study, is the superb accuracy, on the price of more memory occupied by such a codeword. This leads to a complete revision of the distance measure between codewords and also the encoding and decoding schemes, as will be described in Sec. 3.1.
2. In addition to the maximization of the discrimination of the faces in the TES, we also maximize the discrimination of the trait patterns in each trait, based on a new TVS that includes variations caused by illumination, expression, and pose, as well as a new *Trait Enrichment Set (TRS)* that includes imprecisely aligned trait patches. This scheme makes the proposed PFTC robust in recognizing faces under aforementioned variations, and it will be described in 3.2.

3.1 Probabilistic Encoding and Decoding

Consider a N -trait FTC codeword $\mathbf{g} = [g_1 g_2 \dots g_N]$, where each g_i is an integer representing the g_i -th pattern that the i -th trait sample belongs to. It will take the form $\mathbf{G} = [G_1 G_2 \dots G_N]$ in the PFTC, where each G_i is a vector with each element measuring the probability of this trait sample to one specific pattern of that trait. The dimension of G_i is the same as the number of the patterns extracted from the TES for that trait. We measure the probability of a trait

sample belonging to a trait pattern using its distance to the support hyperplane given by the SVM classifier of that trait pattern. To make G_i a distribution, we normalize its magnitude so that $\|G_i\| = 1$. The distance between two PFTC codewords, $\mathbf{G}_a = [G_{a,1}G_{a,2}\dots G_{a,N}]$ and $\mathbf{G}_b = [G_{b,1}G_{b,2}\dots G_{b,N}]$, is defined as the following:

$$D(\mathbf{G}_a, \mathbf{G}_b) = \sum_{i=1}^N B(\mathbf{G}_{a,i}, \mathbf{G}_{b,i}) \quad (1)$$

where $B(G_{a,i}, G_{b,i})$ is the *Bhattacharyya distance* [11] between the two pattern distributions $G_{a,i}$ and $G_{b,i}$ at the i -th trait.

We denote the integer codewords described in [3] as the **hard codewords**, and denote the proposed ones as the **probabilistic codewords**. The recognition results using both types of codewords will be reported and compared in our experimental study. The probabilistic codewords require a bit more storage space and computation than the hard ones, in exchange of superior recognition accuracy. TABLE 1 gives a comparison between the hard and the probabilistic codewords. Consider a FTC with 64 facial traits, and each facial trait has at most 64 trait patterns. A corresponding hard codeword requires only 48 Bytes for storage, while a probabilistic codewords requires 16 KBytes, when *single precision* numbers are used.

codeword type	hard	probabilistic
data structure	N integers	N k-by-1 real arrays (typically, $k < 100$)
encoding complexity	N SVM classifications	
decoding complexity	N integer comparisons	N Bhattacharyya distance calculations

Table 1. Comparison between the hard and probabilistic codewords encoded by N facial traits.

3.2 Trait Enrichment Set and the Pattern-Specific Subspace Learning

Similar to FTC, for the application of the PFTC to face recognition problem, a given face is encoded into a codeword, and matched, or decoded, into one of the gallery codeword. Ideally, if two faces belong to the same individual, they should be encoded exactly the same. However, in practice the facial variations in illumination conditions, expressions, poses or even the misalignments of facial images pose challenges for robust encoding. To recognize face correctly under a very strict scenario where only one facial image is allowed for enrollment and probe images are under aforementioned variations (e.g. John has a smiling, uniformly-lit face enrolled in the database, and we want to recognize John’s face which is

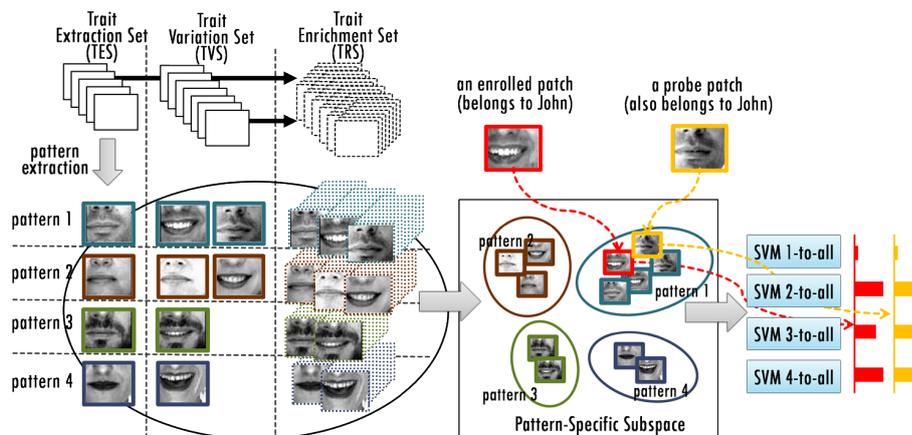


Fig. 1. Illustration of the pattern-specific subspace learning scheme. In this illustration, we extract four patterns for this trait using the facial images in TES. With patches in TVS and TRS added, we learn the pattern-specific subspace where patches of different patterns are well separated. Given a patch from an enrolled face, it is projected into the pattern-specific subspace, and then its distance to the four one-to-the-rest SVMs are calculated and concatenated to form the trait-specific probabilistic distribution. The associated distribution of a probe patch is also calculated and compared with the enrolled distribution.

left-lit and with neutral expression), PFTC is required to encode faces of the same individual *similarly*, no matter under what conditions these facial images are taken.

The *Pattern-Specific Subspace Learning* scheme, denoted as **PSSL**, proposed in this paper is the solution to this requirement. We learn subspaces in which trait patterns can be best discriminated and impacts of variations in illuminations, expressions, poses and misalignments are minimized. The proposed PFTC using the PSSL scheme is illustrated in Fig. 1, and it is composed of the following steps:

1. We collect a large set of faces taken under variations caused by illumination, facial expression and pose and add them into the **TVS**.⁴
2. The facial trait pattern extraction is the same as that in the [3]: extraction of patch patterns using the clustering approach given in [7] and then the Adaboost selection scheme upon the *PPMs* using the faces from the **TES**.
3. All facial images in this work were aligned by centers of eyes, and thus the localization errors of eyes lead to the misalignment of facial images in practice. To handle this problem, for each image in the TES and TVS, we allowed the center of one eye to have 2 pixel offset from the *true center* in

⁴ The **TVS** in [3] contained only facial images taken under minor illumination variations.

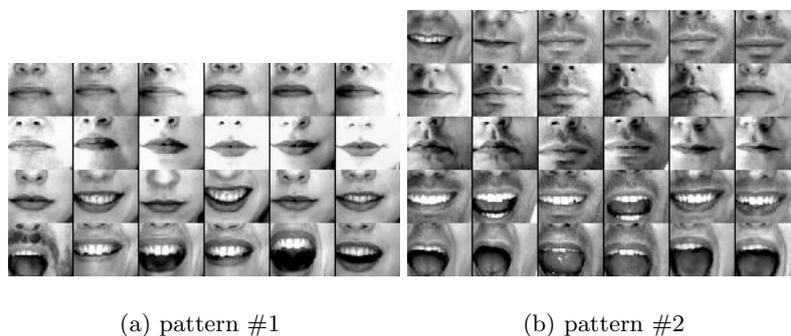
each direction, up, down, right, left. This gave 5 possible eye centers for an eye, and both eyes resulted in 25 possible eye-center pairs. We cropped the face using these 25 pairs, left the one with true centers in the TES or TVS, and moved the rest 24 to a new set defined in this paper, the *Trait Enrichment Set (TRS)*.⁵

4. Since the individuals in the **TVS** and **TRS** must be in **TES**, each *variational* facial trait sample from the **TVS** and **TRS** must have at least one corresponding *neural* trait sample⁶ in **TES**. We merged the neural trait samples and the variation trait samples in each trait, and then applied the Linear Discriminant Analysis (LDA) to extract the features that maximize the scattering between different *trait patterns* while minimize the scattering within each trait pattern. This step forced each trait feature to include the trait’s variations caused by illumination, expression, pose, and misalignment.
5. For each trait pattern in a facial trait, a SVM classifier was trained using the *one-to-the-rest* scheme with the LDA features extracted in Step-4. This gave the same number of SVM classifiers as that of the patterns existing in the facial trait.
6. In the encoding phase, a gallery face was firstly decomposed into a set of facial traits, and each trait was encoded using its normalized distances to the hyperplanes of all SVM classifiers for that specific trait. This step converts a gallery face into a gallery codeword.
7. In the decoding phase, a probe face was also converted into a probe codeword in the same manner, and then the distance measure given by (1) was used to determine if it matched any gallery codeword.

As shown in Fig. 1, assume we have an enrolled patch and a probe patch both belong to John, and John’s patch belongs to the first pattern. The enrolled patch is taken under uniform lighting with mouth smiling, and it is the only enrolled patch for John. Although we do not know the actual appearance of a left-lit patch with neutral expression of John, as long as we have a training individual whose patches are under the same condition and also belong to the first pattern, we can recognize John’s probe patch shown in Fig. 1 through the proposed PSSL scheme. This scheme exploits an observation on human facial images: *if the neutral patches (i.e. uniformly-lit patches with neutral expressions) of two individuals are classified into the same pattern, the patches of the two individuals under the same illumination condition will have similar appearance, so will those make the same facial expression*. Fig. 2 illustrates this observation. The proposed PSSL scheme grants the PFTC the ability to handle the most strict scenario where only one facial image is allowed for enrollment and probe images are under aforementioned variations, and it will be validated in our experimental study in Sec. 4.

⁵ Similarly, one can tolerate larger degree of misalignment of the eyes by allowing the center of one eye to have more than two pixel offset.

⁶ an evenly lighted frontal face without facial expression



(a) pattern #1

(b) pattern #2

Fig. 2. Illustration of patches of two patterns under different illumination conditions and facial expressions. (a) Patches belong to three different individuals that are classified into pattern #1. The first row: neutral patches; the second row: patches under a similar illumination condition; the third and the fourth row: patches make two facial different expressions. (b) Patches belong to three different individuals that are classified into pattern #2. The first row: neutral patches; the second and the third row: patches under two different illumination conditions; the fourth and the fifth row: patches make two different facial expressions.

4 Performance Evaluation—A Comparative Study

The training set, the gallery (or target) set, and the probe set are generally three disjoint sets. The training set is composed of the TES, TVS and TRS datasets for building up the facial traits and their associated trait patterns. If the gallery set happens to be the training set, i.e., the trait patterns are all learned from the gallery set, the performance of the FTC is expected to reach its best. It would be interesting to study the difference in performance between this best case and the general case that the trait patterns are already defined from the training set, and the gallery set can only be encoded using the training-set defined trait patterns. Therefore, two test protocols are considered as follows:

- **Protocol-1:** the training set and gallery set are the same;
- **Protocol-2:** the training set and gallery set are two disjoint sets.

Both protocols were tested on the AR face database [5] and a dataset composed of samples from several face databases. The tests on the AR database gave a way to compare the FTC against other algorithms with reported performance on AR database. However, to reveal that the facial traits can be better defined from a large set of faces collected from different resources, and compared with other algorithms in performance, a *mixed* dataset was used. The mixed dataset is composed of images collected from AR [5], FERET [12], FRGC [13], FVI [16], PIE [14] and XM2VTS [15]. The mixed dataset includes 6405 facial images from

Dataset	#subjects	#faces	#face/subject	note
AR [5]	126	1764	14	all facial images without occlusion such as sunglasses and scarf
FERET [12]	200	400	2	ba (neutral) and bk (with illumination variation)
FRGC [13]	201	1957	1 ~ 25	1957 images with their illumination conditions, facial expressions and poses manually annotated
FVI	38	760	20	10 images from one session and 10 from another session. Images in the two sessions were collected two weeks apart.
PIE [14]	43	344	8	random 8 images for each person taken under room lights on with flash lights of different directions
XM2VTS [15]	295	1180	4	the speech shot
total	903	6405	7.09	

Table 2. The mixed dataset: the details of the samples selected from the six datasets.

903 individuals taken under uncontrolled illumination conditions, facial expressions, and poses, or using different image acquisition devices. The pose variations cover at most 20° toward both sides, up and down. The details of the samples selected from the six datasets are given in TABLE 2. All facial images in both datasets were aligned to the centers of the eyes, and normalized to 80x100 pixels in size.

Two typical face recognition tasks were carried out: identification and verification. In identification, each probe image had one unique match to identify in the gallery set. In verification, each probe image with a claimed subject were both presented to the verification algorithm, which would either accept or reject the claim. A claim would be rejected when the probe failed to match the claimed subject, no matter whether the subject of the probe was in the gallery set or not.

We compared the FTC’s performance with *LBP* [1] and the algorithm using Sparse Representation (*SRC*) [2]. *SRC* is acknowledged as one of the most potential approaches for face recognition published recently. Our comparison also included on algorithm using local patches [17], one algorithm using *ECOC* [18], as well as two baseline methods, *Eigenface* [19] and *Fisherface* [20]. The early version of FTC [3] with hard codewords was also included in this comparison and dubbed as *FTC*. The proposed PFTC using both PSSL scheme and probabilistic codewords is dubbed as *PFTC*. For the *SRC* algorithm, we implemented the **Eigen+SRC** described in [2], which applied the PCA features. The implementation of *ECOC* applied [127, 15, 27] binary BCH code to generate the codewords⁷.

TABLE 3 summarizes the three experiments performed in this study. The *training set* is the set of images for training algorithms: it was used to extract the

⁷ According to our unpublished results, *ECOC* did not achieve good recognition rates when using codewords less than 127 bits.

E	P	D	training	gallery	probe		SPE	descriptions of the data partitions
					enrollee	imposter		
1	1	AR	A_1 126S, 882F		A_2 126S, 882F	-	7	A_1 and A_2 , two disjoint sets from AR, parted by different shotting time.
2			B_1 63S 882F	B_{2a} 63S 63F~441F	B_{2b} 63S 441F	-	1~7	B_1 and B_2 , two disjoint sets from AR, parted by different subject. B_2 further divided into B_{2a} and B_{2b} , parted by different shotting time.
3	2	mixed	C_1 304S 2630F	C_{2a} 303S 303F	C_{2b} 303S 975F	C_3 296S 1950F	1	C_1 , C_2 and C_3 , three disjoint sets from the mixed dataset, parted by different <i>subject</i> . C_2 further divided into C_{2a} and C_{2b} , parted by different shotting time.

Table 3. The test protocols. 'E', 'P', 'D', and 'SPE' stands for 'Experiment', 'Protocol', 'Dataset', and 'the number of Sample-Per-Enrollee' respectively. 'S' and 'F' stands for 'Subject' and 'Face' respectively.

eigen-component in *Eigenface*, *Fisherface*, and *SRC*; it was divided into **TES** and **TVS** in FTC-based algorithms. The *probe* set contains faces for testing. If a *probe*, a face in the probe set, also exists in the gallery, then it is known as an *enrollee*; otherwise, it will be referred to as an *imposter*.

Experiment 1 studied the performance under Protocol-1, where the gallery set is the same as the training set. Experiment 2 studied performance variations with SPE (number of Sample-Per-Enrollee). Experiment 3 studied the performance variations caused by illumination, expression and pose under most challenging scenario where $SPE = 1$. The experimental results and discussions are listed below:

algorithm	IDT	HIT at FAR equals to			EER
		10^{-1}	10^{-2}	10^{-3}	
<i>Eigenface</i> [19]	77.8	90.1	67.8	48.5	10.2
<i>Fisherface</i> [20]	80.5	89.7	69.8	52.7	10.0
<i>Heisele03</i> [17]	82.2	93.7	87.3	78.1	9.82
<i>ECOC</i> [18]	83.3	92.5	88.9	82.8	7.26
<i>LBP</i> [1]	92.4	93.0	78.6	57.8	8.31
<i>SRC</i> [2]	90.5	99.7	96.9	91.0	2.0
<i>FTC</i> [3]	96.8	99.8	99.4	97.2	1.1
<i>PFTC</i>	95.0	98.9	96.2	85.6	2.4

Table 4. Results of Experiment 1 (in percentage). The 'IDT' and 'EER' stand for 'identification rate' and 'equal-error rate measured when FAR equals to FRR', respectively.

- TABLE 4 gives the results of Experiment 1. Both *FTC* and *PFTC* outperformed most algorithms, except for *SRC*. *FTC* and *PFTC* outperformed *SRC* in identification rate, but *SRC* appeared slightly better than *PFTC* in hit rate and ERR. However, *FTC* appeared to give the best overall performance.
- The identification and hit rates for Experiment 2 are shown in Fig. 3 (a) and (b), respectively. The performance of most algorithms degraded significantly when SPE decreased, except *FTC* and *PFTC*. *PFTC* (the red dotted lines) gives the best overall performance in this test.

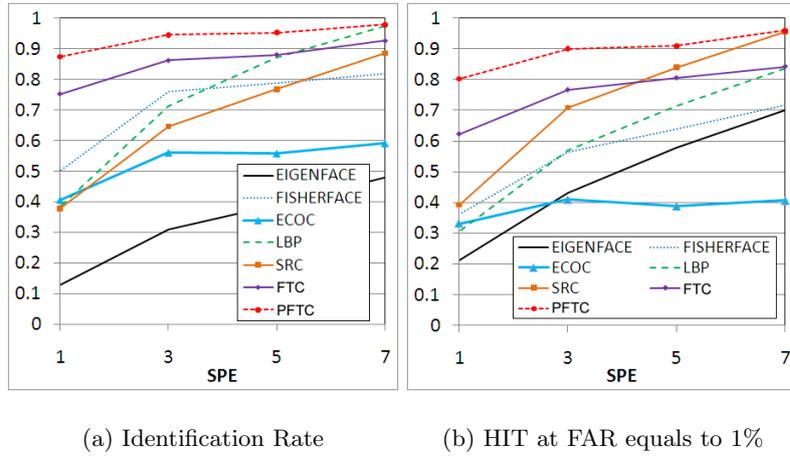


Fig. 3. Recognition results for Experiment 2.

- TABLE 5 summarizes the results of Experiment 3, in terms of identification and verification rates. With the most challenging scenario considered, where each enrollee has only one facial sample for enrollment (SPE=1), *PFTC* gave the best overall performance. It outperformed *FTC*, *SRC* and *LBP* by 14.9% 38.3% 13.4% respectively in identification rate. Similar performance was observed in hit rates and ERRs as well.

5 Conclusions and Future Works

In this paper, we proposed the Probabilistic Facial Trait Code to handle the most rigorous face recognition scenario with one gallery face per individual and probe faces under variations caused by illumination, expression and pose. The *PFTC* comes with novel encoding scheme, and it gives a much better performance than its predecessor, the [3]. Instead of using only one integer to denote a facial trait

subset	neutral set			variant set											
				illumination set			expression set			pose set			variant overall		
#faces	1806			490			410			219			1119		
algorithm	IDT	verification		IDT	verification		IDT	verification		IDT	verification		IDT	verification	
		HIT	EER		HIT	EER		HIT	EER		HIT	EER		HIT	EER
<i>Eigenface</i> [19]	54.8	51.2	21.9	19.4	14.3	30.0	47.3	34.2	22.8	52.1	54.6	21.6	36.0	29.5	25.7
<i>Fisherface</i> [20]	68.5	60.9	15.1	60.4	49.0	14.3	57.8	49.2	16.8	63.5	59.1	13.6	60.1	51.0	15.1
<i>ECOC</i> [18]	50.5	66.2	13.9	39.6	55.8	14.7	43.2	60.5	17.2	35.4	58.3	14.8	40.1	58.0	15.6
<i>LBP</i> [1]	77.7	62.1	15.1	60.4	56.1	20.2	72.6	65.6	16.5	59.4	44.1	27.6	64.7	57.2	20.3
<i>SRC</i> [2]	70.1	76.3	13.8	21.5	29.4	33.1	47.4	52.8	22.3	66.7	69.5	16.2	39.8	45.8	25.8
<i>FTC</i> [3]	80.9	85.1	6.89	68.1	77.9	7.91	60.0	66.3	9.99	58.3	74.8	8.61	63.2	73.0	8.81
<i>PFTC</i>	87.8	88.0	4.83	84.0	85.0	5.22	68.4	67.1	8.34	83.3	86.3	6.51	78.1	78.7	6.62

Table 5. Results of Experiment 3 (in percentage). The 'IDT' stands for the identification rate; 'HIT' is the hit rate measured at FAR=0.01; 'EER' stands for the equal-error rate, the rate at which both accept and reject errors are equal. The **neutral set** includes frontal faces with no variation; the **illumination**, **expression** and **pose set** includes faces taken under variation in illumination, expressions and pose, respectively.

in the encoding phase, PFTC allows the probabilities of the trait belonging to all trait patterns in the codeword. Furthermore, we included facial images under variations caused by illumination, expression, pose and misalignment, and learnt a pattern-specific subspace, which makes the proposed PFTC robust in recognizing faces under aforementioned variations.

The extensive experimental study given in this paper also evaluated three recent face recognition algorithms, one based on local binary pattern, one using the sparse representation, and the Facial Trait Code approach, under several scenarios with different definitions of training set, gallery and probe set and different conditions facial images were taken. The proposed PFTC outperformed all the state-of-the-art algorithms compared in this paper, especially when there is only one single image per individual allowed in the gallery.

Acknowledgement

This work was supported in part by the Ministry of Economic Affairs, Taiwan, under Grant 98-EC-17-A- 02-S1-032 and by the National Science Council, Taiwan, under grant NSC 98-2221-E-002-127-MY3.

References

1. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: Application to face recognition. In: IEEE Transactions on Pattern Analysis and Machine Intelligence. (2006) 2037–2041

2. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31** (2009) 210–227
3. Lee, P.H., Hsu, G.S., Hung, Y.P.: Face verification and identification using facial trait code. *IEEE Conference on Computer Vision and Pattern Recognition* (2009) 1613–1620
4. Phillips, J.P., Scruggs, T.W., O’toole, A.J., Flynn, P.J., Bowyer, K.W., Schott, C.L., Sharpe, M.: *Frvt 2006 and ice 2006 large-scale results*. Technical report, National Institute of Standards and Technology (2007)
5. Martinez, A., Benavente, R.: *The ar face database*. Technical Report 24, CVC (1998)
6. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification* (2nd Edition). Wiley (2000)
7. Figueiredo, M., Jain, A.: unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24** (2002) 381–396
8. Dietterich, T.G., Bakiri, G.: Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research* **2** (1995) 263–286
9. Lin, S., Costello, D.J.: *Error Control Coding* (2nd Edition). Pearson Education International (2004)
10. Cristianini, N., Shawe-Taylor, J.: *An Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press (2000)
11. Bhattacharyya, A.: On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society* **35** (1943) 99–109
12. Phillips, P.J., Moon, H., Rizvi, S.A., Rauss, P.J.: The FERET evaluation methodology for face-recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22** (2000) 1090–1034
13. Phillips, P., Flynn, P., Scruggs, T., Bowyer, K., Chang, J., Hoffman, K., Marques, J., Min, J., Worek, W.: Overview of the face recognition grand challenge. In: *IEEE Conference on Computer Vision and Pattern Recognition*. (2005) 947–954
14. Sim, T., Baker, S., Bsat, M.: The CMU pose, illumination, and expression (PIE) database of human faces. Technical Report CMU-RI-TR-01-02, Carnegie Mellon University (2001)
15. Messer, K., Matas, J., Kittler, J., Luettin, J., Maitre, G.: Xm2vtsdb: The extended m2vts database. In: *Proceedings of the Second International Conference on Audio and Video-based Biometric Person Authentication*. (1999)
16. Lee, P.H., Chu, L.J., Hung, Y.P., Shih, S.W., Chen, C.S., Wang, H.M.: Cascading multimodal verification using face, voice and iris information. In: *IEEE International Conference on Multimedia and Expo, Beijing, China* (2007) 847–850
17. Heisele, B., Ho, P., Wu, J., Poggio, T.: Face recognition: component-based versus global approaches. *Computer Vision and Image Understanding* **91** (2003) 6–12
18. Kittler, J., Ghaderi, R., Windeatt, T., Matas, J.: Face verification using error correcting output codes. In: *IEEE Conference on Computer Vision and Pattern Recognition*. Volume 1. (01) I-755– I-760
19. Turk, M., Pentland, A.: Eigenfaces for recognition. *Journal of Cognitive Neuroscience* **3** (1991) 71–86
20. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19** (1997) 711–720