Synthesis-based Robust Low Resolution Face Recognition

Sumit Shekhar, Student Member, IEEE, Vishal M. Patel, Member, IEEE, and Rama Chellappa, Fellow, IEEE

Abstract—Recognition of low resolution face images is a challenging problem in many practical face recognition systems. Methods have been proposed in the face recognition literature for the problem which assume that the probe is low resolution, but a high resolution gallery is available for recognition. These attempts have been aimed at modifying the probe image such that the resultant image provides better discrimination. We formulate the problem differently by leveraging the information available in the high resolution gallery image and propose a generative approach for classifying the low-resolution probe image. An important feature of our algorithm is that it can handle resolution change along with illumination variations. Furthermore, we also kernelize the algorithm to handle non-linearity in data and present a joint sparse coding technique for robust recognition at low resolutions. The effectiveness of the proposed method is demonstrated using standard datasets and a challenging outdoor face dataset. It is shown that our method is efficient and can perform significantly better than many competitive low resolution face recognition algorithms.

Index Terms—Low-resolution face recognition, dictionary learning, image relighting, non-linear dictionary learning, joint sparse coding.

I. INTRODUCTION

Face recognition (FR) has been an active field of research in biometrics for over two decades [1]. Current methods work well when the test images are captured under controlled conditions. However, quite often the performance of most algorithms degrades significantly when they are applied to the images taken under uncontrolled conditions where there is no control over pose, illumination, expressions and resolution of the face image. Image resolution is an important parameter in many practical scenarios such as surveillance where high resolution cameras are not deployed due to cost and data storage constraints and further, there is no control over the distance of faces from the camera. Figure 1 illustrates a practical scenario where one is faced with a challenging problem of recognizing humans when the captured face images are of very low resolution (LR).

Many methods have been proposed in the vision literature that can deal with this resolution problem in FR. Most of these methods are based on application of super-resolution (SR) technique to increase the resolution of images so that the recovered higher-resolution (HR) images can be used for recognition. One of the major drawbacks of applying SR techniques is that there is a possibility that recovered HR images may contain some serious artifacts. This is often the case when the resolution of the image is very low. As a result, these recovered images may not look like the images of the same person and the recognition performance may degrade significantly.

In practical scenarios, the resolution change is also coupled with other variations such as pose change, illumination and expression. Algorithms specifically designed to deal with LR images quite often fail in dealing with these variations. Hence, it is essential to include these parameters while designing a robust method for low-resolution FR. To this end, in this paper, we present a generative approach to low-resolution FR that is also robust to illumination variations based on learning class specific dictionaries. One of the major advantages of using generative approaches is that they are known to have reduced sensitivity to noise than the discriminative approaches [1]. Furthermore, we kernelize the learning algorithm to handle non-linearity in the data samples and present a joint sparse coding framework for robust recognition.

The training stage of our method consists of three main steps. In the first step of the training stage, given HR training samples from each class, we use an image relighting method to generate multiple images of the same subject with different lighting so that robustness to illumination changes can be realized. In the second step, the resolution of the enlarged gallery images from each class is matched with that of the probe image. Finally, in the third step, class and resolution specific dictionaries are trained for each class. For the testing phase, a novel LR image is projected onto the span of the atoms in each learned dictionary. The residual vectors are then used to classify the subject. A flowchart of the proposed algorithm is shown in Figure 2.

Fig. 1: A typical image in remote face recognition.
The key contributions of our work are:

1. We propose a synthesis-based method for LR FR that is robust to illumination variations, and a dictionary learning framework for classification at low resolutions.
2. We extend our method from linear to non-linear case by learning a dictionary in the high-dimensional feature space using kernel methods.
3. A joint non-linear dictionary learning method is proposed for LR FR that shares common sparse codes between HR and LR dictionaries.

A preliminary version of this work appeared in [2] which described the item 1 above. In this paper, we explore items 2 and 3 and present an extensive experimental evaluation of the proposed methods.

A. Paper organization

The rest of the paper is organized as follows: In Section II, we review a few related works. In Section III, the proposed approach is described and in Section IV, experimental results are demonstrated. Finally, Section VI concludes the paper with a brief summary and discussion.

II. PREVIOUS WORK

In this section, we review some of the recent FR methods that can deal with low resolution. We also briefly discuss the relevant sparse coding literature.

A. SR-based approaches

SR is the method of estimating HR image \( x \) given downgraded image \( y \). The LR image model is often given as

\[
y = BHx + \eta,
\]

where \( B \), \( H \) and \( \eta \) are the downsampling matrix, the blurring matrix and the noise, respectively. Earlier works for solving the above problem were based on taking multiple LR inputs and combining them to produce the HR image. A classical work by Simon and Baker [3] showed that the methods using multiple LR images using smooth priors would fail to produce good results as the resolution factor increases. They also proposed a face hallucination method for super-resolving face images. Subsequently, there have been works using single image for SR such as example-based SR [4], SR using neighborhood embedding [5] and sparse representation-based SR [6]. While these methods can be used for super-resolving the face images and subsequent recognition, methods have also been proposed for specifically handling the problem for faces.

In particular, an eigen-face domain SR method for FR was proposed by Gunturk et al in [7]. This method proposes to solve the FR at LR using SR of multiple LR images using their PCA domain representation. Given an LR face image, Jia and Gong [8] propose to directly compute a maximum likelihood identity parameter vector in the HR tensor space that can be used for SR and recognition. Hennings-Yeomans et al. [9] presented a Tikhonov regularization method that can combine the different steps of SR and recognition in one step. Wilman et al. [10] proposed a relational learning approach for super-resolution and recognition of low resolution faces.

B. Metric learning-based approaches

Though LR face images are directly not suitable for face recognition purpose, it is also not necessary to super-resolve the images before recognition, as the problem of recognition is not the same as SR. Based on this motivation, some different approaches to this problem have been suggested. The method of Coupled Metric Learning [11] attempts to solve this problem by mapping the LR image to a new subspace, where higher recognition can be achieved. A similar approach for improving the matching performance of the LR images using multidimensional scaling was recently proposed by Biswas et al. in [12]–[14]. Further, Ren et al. [15] used coupled kernel methods for low-resolution face recognition. A coupled Fisher analysis method was proposed by Sienna et al. [16]. Lei et al. [17], also proposed a coupled discriminant analysis framework for heterogenous face recognition.

C. Other methods

There have been several attempts to solve the problem of unconstrained FR using videos. In particular, Arandjelovic and Cipolla [18] use a video database of LR face images with variations in pose and illumination. Their method combines a photometric model of image formation with a statistical model of generic face appearance variation to deal with illumination. To handle pose variation, it learns local appearance manifold structure and a robust same-identity likelihood.

A change in resolution of the image changes the scale of the image. Scale change has a multiplicative effect on the distances in image. Hence, if the image is represented in log-polar domain, a scale change will lead to a translation in the said domain. Based on this, a FR approach has been suggested by Hotta et al. in [19] to make the algorithm scale invariant. This method proposes to extract shift-invariant features in the log-polar domain.
Additionally, a support vector data description method for LR FR has been described in [20]. 3D face modeling has also been used to address the LR face recognition problem [21] [22]. Choi et al. [23] present an interesting study on the use of color for degraded face recognition.

D. Sparse Coding

In recent years, sparse representation-based classification method (SRC) has emerged as a powerful tool for various classification problems. Wright et al. [24] proposed the seminal SRC algorithm for face recognition. It was shown that by exploiting the inherent sparsity of data, one can obtain improved recognition performance over traditional methods especially when data are contaminated by various artifacts such as illumination variations, disguise, occlusion, and random pixel corruption. A review of linear and non-linear dictionary-based algorithms for face recognition is presented in Patel et al. [25]. Further, a framework for joint sparse coding has been used for various tasks, like super-resolution [6] and cross-view recognition [26]. The motivation for using joint sparse coding in such tasks is due to being able to transfer the sparse codes between high and low resolution image patches [6] or combine information from multiple views [26]. In this paper, we propose a method for learning joint dictionaries for HR and corresponding LR gallery images for robust recognition at low resolutions.

III. PROPOSED APPROACH

In this section, we present the details of the proposed low-resolution FR algorithm based on learning class specific dictionaries.

A. Image Relighting

As discussed earlier, the resolution change is usually coupled with other parameters such as illumination variation. In this section, we introduce an image relighting method that can deal with this illumination problem in LR face recognition. The idea is to capture various illumination conditions using the HR training samples, and subsequently use the expanded gallery for recognition at low resolutions.

Assuming the Lambertian reflectance model for facial surface, the HR intensity image \( I^H \) is given by the Lambert’s cosine law as follows:

\[
I^H(i, j) = \rho(i, j) \max(n(i, j)^T s, 0),
\]

(1)

where \( I^H(i, j) \) is the pixel intensity at location \( (i, j) \), \( s \) is the light source direction, \( \rho(i, j) \) is the surface albedo at location \( (i, j) \), \( n(i, j) \) is the surface normal of the corresponding surface point. Given the face image, \( I^H \), image relighting involves estimating \( \rho \), \( n \) and \( s \) which is an extremely ill-posed problem. To overcome this, we use 3D facial normal data [27] to first estimate an average surface normal, \( \bar{n} \). Further, the model is non-linear due to the \( \max \) term in (1). However, the shadow points do not reveal any information about albedo. Hence, we neglect the \( \max \) term in further discussion. The albedo, \( \rho \) and source directions \( s \) can now be estimated as follows:

- The source direction can be estimated using \( \bar{n} \) following a linear Least Squares approach [28]:

\[
\hat{s} = \left( \sum_{i,j} \bar{n}(i,j)\bar{n}(i,j)^T \right)^{-1} \sum_{i,j} I^H(i, j)\bar{n}(i, j).
\]

- An initial estimate of albedo, \( \rho^0 \) can be obtained as:

\[
\rho^0(i, j) = \frac{I^H(i, j)}{\bar{n}(i,j)^T \hat{s}}
\]

- The final albedo estimate is obtained using minimum mean square approach based on Wiener filtering framework [29]:

\[
\hat{\rho} = E(\rho|\rho^0),
\]

where, \( E(\rho|\rho^0) \) denotes the minimum mean square estimate (MMSE) of the albedo.

Using the estimated albedo map, \( \hat{\rho} \) and average normal, \( \bar{n} \) we can generate new images under any illumination condition using the image formation model (1). It was shown in [30] that an image of an arbitrarily illuminated face can be approximated by a linear combination of face images in the same pose, illuminated by nine different light sources placed at pre-selected positions.

Hence, the image formation equation can be rewritten as

\[
I^H = \sum_{k=1}^9 \alpha_k I^H_k,
\]

(2)

where

\[
I^H_k(i, j) = \rho(i, j) \max(n(i, j)^T s_k, 0),
\]

and \( \{s_1, \cdot \cdot \cdot , s_9\} \) are pre-specified illumination directions. Since, the objective is to generate HR gallery images which will be sufficient to account for any illumination in the probe image, we generate images under pre-specified illumination conditions and use them in the gallery. Figure 3 shows some relighted HR images along with the corresponding LR images and the estimated albedo. Furthermore, as the condition is true irrespective of the resolution of LR image, the same set of gallery images can be used for all resolutions.

B. Low Resolution Dictionary Learning

In LR face recognition, given labeled HR training images, the objective is to identify the class of a novel probe LR face image. Suppose that we are given \( C \) distinct face classes and a set of \( m_i \) HR training images per class, \( i = \{1, \cdot \cdot \cdot , C\} \). Here, \( m_i \) corresponds to the total number of images in class \( i \) including the relighted images. We identify an \( l_H \times q_H \) grayscale image as an \( N_H \)-dimensional vector, \( x_H \), which can be obtained by stacking its columns, where \( N_H = r_H \times q_H \). Let

\[
X^H_i = [x^H_{i,1}, \cdot \cdot \cdot , x^H_{i,m_i}]
\]

be an \( N_H \times m_i \) matrix of training images corresponding to the \( i^{th} \) class. For resolution and illumination robust recognition, the matrix \( X^H_i \) is pre-multiplied by downsampling \( B \) and blurring \( H \) matrices. Here, \( H \) has a fixed dimension of \( N_L \times N_H \) and \( B \) will be of size \( N_L \times N_H \), where \( N_L = r_L \times q_L \), the
LR probe being a grayscale image of $r_L \times q_L$. The resolution specific training matrix, $X_i^L$ is thus created as

$$X_i^L = BHX_i^H \triangleq (X_i^H) \downarrow.$$  

Given this matrix, we seek the dictionary that provides the best representation for each element in this matrix. One can obtain this by finding a $K$-atom dictionary $D_i \in \mathbb{R}^{N_L \times K}$, and a sparse matrix $\Gamma_i \in \mathbb{R}^{K \times m_i}$, that minimizes the following representation error

$$(\hat{D}_i, \hat{\Gamma}_i) = \arg \min_{D_i, \Gamma_i} \|X_i^L - D_i \Gamma_i\|_F^2 \quad \text{subject to} \quad \|\gamma_k\|_0 \leq T_0 \quad \forall \, k,$$

where $\gamma_k$ represent the columns of $\Gamma_i$ and the $\ell_0$ sparsity measure $\|\cdot\|_0$ counts the number of nonzero elements in the representation. Here, $\|A\|_F$ denotes the Frobenius norm defined as $\|A\|_F = \sqrt{\sum \sum |A(i,j)|^2}$. Many approaches have been proposed in the literature for solving such optimization problem. In this paper, we adapt the K-SVD algorithm [31] for solving (4) due to its simplicity and fast convergence. The K-SVD algorithm alternates between sparse-coding and dictionary update steps. In the sparse-coding step, $D_i$ is fixed and the representation vectors $\gamma_k$s are found for each example $x_{i,m}$. Then, with fixed a $\Gamma_i$, the dictionary is updated atom-by-atom in an efficient way. See [31] for more details on the K-SVD dictionary learning algorithm.

1) Classification:: Given an $r_L \times q_L$ LR probe, it is column-stacked to give the column vector $y$. It is projected onto the span of the atoms in each $D_i$ of the $C$ class dictionary, using the orthogonal projector

$$P_i = D_i(D_i^T D_i)^{-1} D_i^T.$$

The approximation and residual vectors can then be calculated as

$$\hat{y}_i = P_i y = D_i \alpha_i$$

and

$$r_i(y) = y - \hat{y}_i = (I - P_i) y,$$

respectively, where $I$ is the identity matrix and

$$\alpha_i = (D_i^T D_i)^{-1} D_i^T y$$

are the coefficients. Since the K-SVD algorithm finds the dictionary, $D_i$, that leads to the best representation for each examples in $X_i^L$, $\|r_i(y)\|_2$ will be small if $y$ were to belong to the $i^{th}$ class and large for the other classes. Based on this, we can classify $y$ by assigning it to the class, $d \in \{1, \cdots, C\}$, that gives the lowest reconstruction error, $\|r^d(y)\|_2$:

$$d = \text{identity}(y) = \arg \min_i \|r_i(y)\|_2.$$  

2) Generic Dictionary Learning:: The class-specific dictionary, $D_i, i = 1, \cdots, C$ learnt above can be extended to use features other than intensity images. Specifically, the dictionary can be learnt using features like Eigenbasis, $F_i^H$ extracted from training matrix $X_i^H$. However, as equation (3) does not hold for $F_i^H$, the resolution specific feature matrix $F_i^L$ is directly extracted using $X_i^L$. Our Synthesis-based LR FR (SLRFR) algorithm is summarized in Figure 4.

C. Non-linear Dictionary Learning

The class identities in the face dataset may not be linearly separable. Hence, we also extend the SLRFR framework to the kernel space. This essentially requires the dictionary learning model to be non-linear [32].

Let $\phi^L : \mathbb{R}^{N_L} \rightarrow G$ be a non-linear mapping from $N_L$ dimensional space into a dot product space $G$. A non-linear dictionary can be trained in the feature space $G$ by solving the following optimization problem

$$(\hat{A}_i, \hat{\Gamma}_i) = \arg \min_{A_i, \Gamma_i} \|\phi^L(X_i^L) - \phi^L(X_i^L)A_i\Gamma_i\|_F^2 \quad \text{subject to} \quad \|\gamma_k\|_0 \leq T_0 \quad \forall \, k,$$

where

$$\phi^L(X_i^L) = [\phi^L(x_{i,1}^L), \cdots, \phi^L(x_{i,m_i}^L)].$$
Given a LR test sample $y$ and $C$ training matrices $\{X_i^H\}_{i=1}^C$ corresponding to HR gallery images.

**Procedure:**
- For each training image, use the relighting approach described in section III-A to generate multiple images with different illumination conditions and use them in the gallery.
- Learn the best dictionaries $D_i$, to represent the resolution specific enlarged training matrices, $X_i^L$, using the K-SVD algorithm, where $X_i^L = (X_i^H)^\dagger$.
- Compute the approximation vectors, $\tilde{y}_i$, and the residual vectors, $r_i(y)$, using (5) and (6), respectively for $i = 1, \cdots, C$.
- Identify $y$ using (8).

![Fig. 4: The SLRFR algorithm.](image)

In (9) we have used the following model for the dictionary in the feature space,
\[ D_i = \phi_i^L(X_i^L)A_i, \]

Since it can be shown that the dictionary lies in the linear span of the samples $\phi_i^L(X_i^L)$, where $A_i \in \mathbb{R}^{m_i \times K}$ is a matrix with $K$ atoms [32]. This model provides adaptivity via modification of the matrix $A_i$. Through some algebraic manipulations, the cost function in (9) can be rewritten as,
\[ ||\phi_i^L(X_i^L) - \phi_i^L(X_i^L)A_i\Gamma_i||_F^2 = \text{tr}((I - A_i\Gamma_i)^T K^L(X_i^L, X_i^L)(I - A_i\Gamma_i)), \]
\[ (10) \]
where $K^L$ is a kernel matrix whose elements are computed from
\[ \kappa(i, j) = \phi_i^L(x_i^L)^T \phi_j^L(x_j^L). \]

It is apparent that the objective function is feasible since it only involves a matrix of finite dimension $K^L \in \mathbb{R}^{m_i \times m_i}$, instead of dealing with a possibly infinite dimensional dictionary.

An important property of this formulation is that the computation of $K^L$ only requires dot products. Therefore, we are able to employ Mercer kernel functions to compute these dot products without carrying out the mapping $\phi_i^L$. Some commonly used kernels include polynomial kernels
\[ K^L(x, y) = \langle x, y \rangle + c \]
and Gaussian kernels
\[ K^L(x, y) = \exp\left(-\frac{||x - y||^2}{\sigma^2}\right), \]
where $c$, $d$ and $\sigma$ are parameters.

Similar to the optimization of (4) using the linear K-SVD [31] algorithm, the optimization of (9) involves sparse coding and dictionary update steps in the feature space which results in the kernel dictionary learning algorithm [32]. Details of the optimization can be found in [32] and Appendix A.

1) **Classification:** Let $\{A_i\}_{i=1}^C$ denote the learned dictionaries for $C$ classes. Let $z \in \mathbb{R}^{NL}$ be a vectorized LR probe image $z$ of size $r_L \times q_L$. We first find coefficient vectors $\gamma_i \in \mathbb{R}^K$ with at most $T$ non-zero coefficients such that $\phi_i^L(X_i^L)A_i\gamma_i$ approximates $z$ by minimizing the following problem
\[ \min_{\gamma_i} ||\phi_i^L(z) - \phi_i^L(X_i^L)A_i\gamma_i||_2^2 \text{ s.t. } ||\gamma_i||_0 \leq T, \]
\[ (11) \]
for all $i = 1, \cdots, C$. The above problem can be solved by the Kernel Orthogonal Matching Pursuit (KOMP) algorithm [32]. The reconstruction error is then computed as
\[ r_i = ||\phi_i^L(z) - \phi_i^L(X_i^L)A_i\gamma_i||^2 = K^L(z, z) - 2K^L(z, X_i^L)A_i\gamma_i + \gamma_i^T A_i^T K^L(X_i^L, X_i^L)A_i \gamma_i, \]
\[ (12) \]
where,
\[ K^L(z, X_i^L) = [\kappa(z, x_{i1}^L), \kappa(z, x_{i2}^L), \cdots, \kappa(z, x_{im_i}^L)]. \]

Similar to the linear case, once the residuals are found, we can classify $z$ by assigning it to the class, $d \in \{1, \cdots, C\}$, that gives the lowest reconstruction error, $||r_i(y)||_2$:
\[ d = \text{identity}(y) = \arg \min_i ||r_i(y)||_2. \]
\[ (13) \]
Our kernel Synthesis-based LR FR (kerSLRFR) algorithm is summarized in Figure 5.

![Fig. 5: The kerSLRFR algorithm.](image)

Given a LR test sample $y$ and $C$ training matrices $\{X_i^H\}_{i=1}^C$ corresponding to HR gallery images.

**Procedure:**
- For each training image, use the relighting approach described in section III-A to generate multiple images with different illumination conditions and use them in the gallery.
- Learn non-linear dictionaries $A_i$, to represent the resolution specific enlarged training matrices, $X_i^L$, using the kernel dictionary learning algorithm 9, where $X_i^L = (X_i^H)^\dagger$, $i = 1, \cdots, C$.
- Compute the sparse codes, $\gamma_i$ and the residual vectors, $r_i^\gamma(y)$, using (11) and (12), respectively for $i = 1, \cdots, C$.
- Identify $y$ using (13).

**D. Joint Non-linear Dictionary Learning**

In the previous sections, we described methods to learn resolution-specific dictionaries for linear and non-linear cases. However, even though dictionaries can capture class-specific variations, the recognition performance would go down at low resolutions. Hence, information available in the HR training images must be exploited to make the method robust. To enable this, we propose a framework of learning joint dictionaries for HR and corresponding LR images. We achieve this
through sharing sparse codes between HR and LR dictionaries. This regularizes the learned LR dictionary to output similar sparse codes as HR dictionary, thus, making it robust. The proposed formulation is described as follows. An overview of the proposed approach is also shown in Figure 6.

Let \( \Phi^H : \mathbb{R}^{N_H} \rightarrow G \) be a non-linear mapping from \( N_H \) dimensional space into a dot product space \( G \). We seek to learn dictionaries \( \Phi^H \in \mathbb{R}^{m_i \times K} \) and \( \Phi^L \in \mathbb{R}^{m_i \times K} \) by solving the optimization problem:

\[
\begin{align*}
(\hat{\Phi}^H, \hat{\Phi}^L, \hat{\Gamma}) &= \arg\min_{\Phi^H, \Phi^L, \Gamma} \left\{ \|\Phi^H(X^H) - \Phi^H(X^H)\Gamma \|^2_F \\
&+ \lambda \|\Phi^L(X^L) - \Phi^L(X^L)\Gamma \|^2_F \right\} \\
\text{subject to } &\|\gamma_k\|_0 \leq T_0 \forall k,
\end{align*}
\]

(14)

where, \( \lambda > 0 \) is a hyperparameter. This can be re-formulated as:

\[
\begin{align*}
(\hat{\Phi}^H, \hat{\Phi}^L, \hat{\Gamma}) &= \arg\min_{\Phi^H, \Phi^L, \Gamma} \left\{ \|\Phi_1(X^H, X^L) - \Phi_2(X^H, X^L)\Gamma \|^2_F \\
&+ \lambda \|\gamma_k\|_0 \leq T_0 \forall k,
\end{align*}
\]

(15)

where,

\[
\Phi_1(X^H, X^L) = \left[ \begin{array}{c} \Phi(X^H) \\ \sqrt{\lambda}\Phi(X^L) \end{array} \right], \quad \Phi_2(X^H, X^L) = \left[ \begin{array}{c} \Phi(X^H) \\ 0 \end{array} \right],
\]

\[
\Phi_1(X^H, X^L) = \left[ \begin{array}{c} \Phi(X^H) \\ \sqrt{\lambda}\Phi(X^L) \end{array} \right]
\]

The optimization problem (14) can be solved in a similar way as (9) using a modified version of kernel K-SVD algorithm [32]. Details of the method are presented in Appendix A.

1) Classification: Let \( \{\Phi_i^L\}_{i=1}^C \) denote the learned dictionaries for \( C \) classes. Then a low resolution probe \( z \in \mathbb{R}^{NL} \) can be classified using the KOMP algorithm [32], as described in (11), (12) and (13), by substituting \( \{\Phi_i^L\}_{i=1}^C \) for dictionary term. The proposed algorithm referred to as joint kernel SRLFR (jointKerSRLFR) is summarized in Figure 7.

Given a LR test sample \( y \) and \( C \) training matrices \( \{X^H_i\}_{i=1}^C \) corresponding to HR gallery images.

**Procedure:**

- For each training image, use the relighting approach described in section III-A to generate multiple images with different illumination conditions and use them in the gallery.
- Learn the dictionaries \( \Phi^H_i \) and \( \Phi^L_i \) to jointly represent the HR and LR training matrices, \( X^H_i \) and \( X^L_i \), where \( X^L_i = (X^H_i)_{\downarrow} = 1, \cdots, C \) respectively using the joint kernel dictionary algorithm.
- Using the learnt dictionary \( \Phi^L_i \), compute the sparse codes, \( \gamma_i \) and the residual vectors, \( r_i \), using (11) and (12) respectively for \( i = 1, \cdots, C \).
- Identify \( y \) using (13).

**IV. EXPERIMENTS**

To demonstrate the effectiveness of our method, in this section, we present experimental results on various face recognition datasets. We demonstrate the effectiveness of the proposed recognition framework, as well as compared with metric learning [11], [12] and SR-based [9], [10] methods. For all the experiments, we learnt the dictionary elements using the PCA features.

**A. FRGC Dataset**

We also evaluated on Experiment 1 of the FRGC dataset [33]. It consists of 152 gallery images, each subject having one gallery and 608 probe images under controlled setting. A separate training set of 183 images is also available which was used to learn the PCA basis.

1) Implementation: The resolution of the HR image was fixed at \( 48 \times 40 \) and the probe images at resolutions of \( 10 \times 8 \) and \( 7 \times 6 \) were created by smoothing and downsampling the HR probe images. From each gallery image, 5 different illumination images were produced, which were flipped to...
give 10 images per subject. The experiments were done at resolutions of 10 × 8 and 7 × 6, thus validating the method across resolutions. We also tested the CLPM algorithm [11] and PCA performances on the expanded gallery to get a fair comparison. We also report the recognition rate for PCA using the original gallery image to demonstrate the utility of gallery extension at low resolutions. Results from other algorithms are also tabulated. We chose RBF kernel for testing kerSLRFR and jointKerSLRFR and set $\lambda = 1$ for jointKerSLRFR. The kernel parameter, $\sigma$ was obtained through cross-validation for both HR and LR data. The dictionary size, $K$ was set to 7 and the sparsity, $T_0$ was taken as 4. We used the nearest neighbor method for classification using PCA features and CLPM [11] method.

2) Observations: Figure 8 and Table I show that the proposed methods clearly outperforms previous algorithms. The proposed algorithm, SLRFR improves the CLPM algorithm for all the resolutions, while kerSLRFR further boosts the performance. The jointKerSLRFR shows the best performance for all the methods. The joint sparse coding framework, clearly helps in improving performance at low resolutions. Further, PCA using the extended gallery set also improves the performance over using a single gallery image. This shows that our method of gallery extension can be coupled with the existing face recognition algorithms to improve performance at low resolutions.

Fig. 8: Recognition Rates for FRGC data with probes at low resolutions

3) Sensitivity to noise:: Low resolution images are often corrupted with noise. Thus, sensitivity of noise is important in assessing performance of different algorithms. Figure 9 shows the recognition rate for different algorithms with increasing noise level. It can be seen that CLPM shows a sharp decline with increasing noise, but the proposed approaches SLRFR, kerSLRFR and jointKerSLRFR are stable with noise. This is because the CLPM algorithm learns a model tailored to noise-free low resolution images, whereas the generative approach in the proposed methods leads to stable performance with increasing noise.

B. CMU-PIE dataset

The PIE dataset [34] consists of 68 subjects in frontal pose and under different illumination conditions. Each subject has 21 face images under different illumination conditions.

1) Implementation: We chose the first 34 subjects with 6 randomly chosen illuminations as the training set to learn PCA basis. For the remaining 34 subjects and the 15 illumination conditions, the experiment was done by choosing one gallery image per subject and taking the remaining as probe images. The procedure was repeated for all the images and the final recognition rate was obtained by averaging over all the images. The size of the HR images was fixed to 48 × 40. The LR images were obtained by smoothing followed by downsampling the HR images. For each gallery image, 10 images under different illuminations produced using gallery extension method and the corresponding flipped images were added to the gallery set. The RBF kernel was chosen for kerSLRFR and jointKerSLRFR and the kernel parameter, $\sigma$ was set through cross-validation. We set $\lambda = 1$ for all the experiments.

Fig. 9: Recognition Rates for FRGC data across increasing noise levels at 10 × 8 LR probe resolutions

2) Observations: Figures 10, 11 and Table II show that the proposed method clearly outperforms previous algorithms. The proposed algorithms shows over 20% improvement over the MDS method [12] and 8% better than the CLPM method at rank one recognition rate, for the probe resolution of 7 × 6. The kerSLRFR and jointKerSLRFR methods report better performance than VLR algorithm [10] at 7 × 6 resolution. Further,
TABLE I: Comparisons for rank one recognition rate of FRGC dataset

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>6 × 6</td>
<td>-</td>
<td>55.0%</td>
<td>-</td>
<td>45.1%</td>
<td>60.7%</td>
<td>62.9%</td>
<td>64.7%</td>
<td>65.2%</td>
</tr>
<tr>
<td>7 × 6</td>
<td>-</td>
<td>55.3%</td>
<td>-</td>
<td>49.7%</td>
<td>65.5%</td>
<td>66.4%</td>
<td>71.2%</td>
<td>73.6%</td>
</tr>
<tr>
<td>9 × 7</td>
<td>58.0%</td>
<td>-</td>
<td>-</td>
<td>56.1%</td>
<td>70.2%</td>
<td>72.2%</td>
<td>76.4%</td>
<td>78.1%</td>
</tr>
</tbody>
</table>

TABLE II: Comparisons for rank one recognition rate of PIE dataset. Note that VLR* [10] uses multiple gallery images while training.

The LR images were obtained by smoothing followed by downsampling the HR images to 14 × 10. We also tested the performance of the CLPM [11] and PCA algorithms on the expanded gallery to get a fair comparison. Results from other algorithms are also tabulated.

2) Observations: Figure 12 shows the CMC curve for the first 5 ranks. Clearly, the proposed approaches outperform other methods. SLRFR gives better rank one performance than the CLPM algorithm, while kerSLRFR and jointKerSLRFR further increase the recognition over all the ranks. This further demonstrates that the proposed algorithms can also handle variations like expression change in the LR probe.

C. AR Face dataset

We also tested the proposed algorithms on the AR Face dataset [35]. The AR face dataset consists of faces with varying illumination and expression conditions, captured in two sessions. We evaluated our algorithms on a set of 100 users. Images from the first session, seven for each subject, were used as training and gallery and the images from the second session, again seven per subject, were used for testing.

1) Implementation: To test our method and compare with existing metric-learning based methods [11] [12], we chose the first 30 subjects from the first session as the training set. For the remaining 70 subjects, the experiment was done by choosing one gallery image per subject from the first session and taking the corresponding images from session 2 as probes. The procedure was repeated for all the 7 images in the session 1 and the final recognition rate was obtained by averaging over all the runs. The size of the HR images was fixed to 55 × 40.

The LR images were obtained by smoothing followed by downsampling the HR images to 14 × 10. We also tested the performance of the CLPM [11] and PCA algorithms on the expanded gallery to get a fair comparison. Results from other algorithms are also tabulated.

2) Observations: Figure 12 shows the CMC curve for the first 5 ranks. Clearly, the proposed approaches outperform other methods. SLRFR gives better rank one performance than the CLPM algorithm, while kerSLRFR and jointKerSLRFR further increase the recognition over all the ranks. This further demonstrates that the proposed algorithms can also handle variations like expression change in the LR probe.

D. Outdoor Face Dataset

We also tested our method on a challenging outdoor face dataset. The database consists of face images of 18 individuals at different distances from the camera. We chose a subset of 90 low-resolution images, which were also corrupted with blur, illumination, and pose variations. 5 high-resolution, frontal, and well-illuminated images were taken as the gallery set for each subject. The images were aligned using 5 manually selected facial points. Automatic alignment of LR faces using landmarks is a challenging problem by itself and we will explore in a separate paper. The gallery resolution was fixed at 120 × 120 and the probe resolution at 20 × 20. Figure 13 shows some of the gallery images and the low quality probe images. The recognition rates for the dataset are shown.
in Table III. We compare our method with the Regularized Discriminant Analysis (RDA) [36] and CLPM [11]. For the reg LDA comparison, we first used the PCA as a dimensionality reduction method to project the raw data onto an intermediate space, then we used the RDA to project the PCA coefficients onto a final feature space.

![Example images from the outdoor face dataset](image)

**Fig. 13:** Example images from the outdoor face dataset (a) HR gallery images (b) LR probe images

<table>
<thead>
<tr>
<th>Method</th>
<th>Recognition Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCA</td>
<td>58.9%</td>
</tr>
<tr>
<td>reg LDA [36]</td>
<td>60%</td>
</tr>
<tr>
<td>CLPM [11]</td>
<td>67.7%</td>
</tr>
<tr>
<td>SLRFR</td>
<td>71.1%</td>
</tr>
<tr>
<td>kerSLRFR</td>
<td>71.1%</td>
</tr>
<tr>
<td>jointKerSLRFR</td>
<td>71.1%</td>
</tr>
</tbody>
</table>

**TABLE III:** Performance for the Outdoor Face Dataset

1) **Observations:** It can be seen from the table that SLRFR outperforms other algorithms on this difficult outdoor face dataset. The kerSLRFR algorithm further improves the performance, however, the jointKerSLRFR doesn’t improve it further. This may be because this is a challenging dataset containing variations other than LR, like pose, blur, etc. The CLPM algorithm performs rather poorly on this dataset, as it is unable to learn the challenging variations in the dataset.

**V. Computational Efficiency**

All the experiments were conducted using 2.13GHz Intel Xeon processor on Matlab programming interface. The gallery extension step using relighting took an average of 2s per gallery image of size $48 \times 40$. The SLRFR method took on an average 0.07s to train each class, while classification of a probe image was done in an average of 0.1s at the resolution of $7 \times 6$. Similarly, kerSLRFR and jointKerSLRFR took 1s to train each class and 0.5s to classify at $7 \times 6$ resolution. Thus, the proposed algorithm is computationally efficient. Further, as the extended gallery can be used for all resolutions, it can be computed once and stored for a database.

**VI. Discussion and Conclusion**

We have proposed an algorithm which can provide good accuracy for LR face images, even when a single HR gallery image is provided per person. While the method avoids the complexity of previously proposed algorithms, it is also shown to provide state-of-the-art results when the LR probe face differs in illumination from the given gallery image. Further, we also show good results for a dataset with expression variations and a challenging outdoor face dataset. The idea of exploiting the information in a HR gallery image is novel and can be used to extend the limits of remote face recognition. We have also proposed a non-linear extension of the algorithm and a joint sparse coding framework for robust recognition at low resolutions. In future, we plan to extend our approach to handle variations like pose, alignment, etc which can affect the recognition at low resolutions. Discriminative framework for the proposed algorithms can also be explored as a future direction.

**ACKNOWLEDGMENT**

This work was supported by ...

**APPENDIX A**

Here, we will describe the kernel dictionary learning algorithm [32] and the framework for the proposed joint kernel dictionary learning algorithm (jointKerKSVD).

**A. Kernel Dictionary Learning**

The optimization problem (9) can be solved in two stages.

1) **Sparse Coding:** Here, $A_i$ is kept fixed while searching for the optimal sparse code, $\Gamma_i$. The cost term in (9) can be written as:

$$\| \phi^L(X_i^L) - \phi^L(X_i^L)A_i\Gamma_i \|^2_F = \sum_{k=1}^{m_i} \| \phi^L(x_{i,j}^L) - \phi^L(x_{i,j}^L)A_i\gamma_j \|^2_F,$$

where, $\gamma_j$ is the sparse code for $x_{i,j}^L$. Hence, the optimization problem can be broken up into $m_i$ different sub-problems:

$$\arg\min_{\gamma_j} \| \phi^L(x_{i,j}^L) - \phi^L(x_{i,j}^L)A_i\gamma_j \|^2_F$$

subject to $\|\gamma_j\|_0 \leq T_0 \forall j$.

We can solve this using kernel orthogonal matching pursuit (KOMP). Let $I_k$ denote the set of selected atoms at iteration $k$, $\hat{x}_k$ denote the reconstruction of the signal, $\phi^L(x_{i,j}^L)$ using the selected atoms, $r_k$ being the corresponding residue and $\gamma_{j,k}$ the estimated sparse code at $k^{th}$ iteration.  
1) Start with $I_0 = 0$, $\hat{x}_k = 0$, $\gamma_{j,k} = 0$.
2) Calculate the residue as:

$$\phi^L(x_{i,j}^L) - \phi^L(x_{i,j}^L)\hat{x}_k = r_k.$$

3) Project the residue on atoms not selected and add the atom with maximum projection value to $I_k$:

$$\tau_t = (\phi^L(x_{i,j}^L) - \phi^L(X_i^L)\hat{x}_k)^T(X_i^L a_t) = (K^L(x_{i,j}^L, X_i^L) - \hat{x}_k K^L(X_i^L, X_i^L))a_t, \ t \notin I_k.$$  

(16)

Update the set $I_k$ as:

$$I_{k+1} = I_k \cup \arg\max_{t \notin I_k} |\tau_t|.$$  

(17)
4) Update the sparse code, $\gamma_{k+1}$ and reconstruction, $\hat{x}_{k+1}$ as:

$$
\gamma_{j,k+1} = ((\phi^L(X_{i,j}^L)A_{i,k+1})^T(\phi^L(X_{i,j}^L)A_{i,k+1}))^{-1}
(\phi^L(X_{i,j}^L)A_{i,k+1})^T \phi^L(x_{i,j}^L) = (A_{i,k+1}^T K^L(x_{i,j}^L, X_{i,j}^L) A_{i,k+1})^{-1}
(\gamma^L(x_{i,j}^L, X_{i,j}^L)A_{i,k+1})^T, 
$$

$$
\hat{x}_{k+1} = A_{i,k+1} \gamma_{j,k+1}.
$$

5) $k \leftarrow k + 1$; Repeat steps 2-4 $T_0$ times.

2) **Dictionary update:** Once the sparse codes are calculated, the dictionary $A_i$ can be updated using kernel K-SVD or MOD methods as described in [32]. Here, we use the MOD to update the dictionary as follows:

$$
A_i = \Gamma_i^T (\Gamma_i \Gamma_i^T)^{-1}.
$$

The dictionary atoms are now normalized to unit norm in feature space:

$$
A_{i,j} = \frac{A_{i,j}}{\sqrt{A_{i,j}^T K^L(x_{i,j}^L, X_{i,j}^L) A_{i,j}}}, \ j = 1, \cdots, K.
$$

B. Joint kernel dictionary learning

The optimization problem (14) can be solved in a similar way as the kernel dictionary learning problem in two alternative steps:

1) **Sparse Coding:** Here, we keep $A^H_i$ and $A^L_i$ fixed and learn the joint sparse code $\Gamma_i$. The cost term in (15) can be written as:

$$
||\Phi_1(x^H_{i,j}, x^L_{i,j}) - \Phi_2(x^H_{i,j}, x^L_{i,j})\tilde{A}_i\Gamma_i||_F^2 = 
\sum_{j=1}^{m_i}||\Phi_1(x^H_{i,j}, x^L_{i,j}) - \Phi_2(x^H_{i,j}, x^L_{i,j})\tilde{A}_i\gamma_j||_F^2,
$$

where, $\gamma_j$ is the sparse code for $x^L_{i,j}$. Thus, the optimization can be broken up into $m_i$ sub-problems:

$$
\arg\min_{\gamma_j} ||\Phi_1(x^H_{i,j}, x^L_{i,j}) - \Phi_2(x^H_{i,j}, x^L_{i,j})\tilde{A}_i\gamma_j||_F^2
$$

subject to $||\gamma_j||_0 \leq T_0 \ \forall \ j$.

This is similar to the original kernel dictionary learning formulation, with the signal $\phi^L(x_{i,j}^L)$ replaced by $\Phi_1(x^H_{i,j}, x^L_{i,j})$. Thus, the above problem can be solved using similar procedure as KOMP. Let $I_k$ denote the set of selected atoms at iteration $k$, $\hat{x}^H_{i,k}$ denote the reconstruction of the signal, $\Phi_1(x^H_{i,j}, x^L_{i,j})$ using the selected atoms, $r_k$ being the corresponding residue and $\gamma_{j,k}$ the estimated sparse code at $k^{th}$ iteration.

1) Start with $I_0 = \emptyset$, $\hat{x}^H_{i,0} = 0$, $\gamma_{j,k} = 0$.
2) Calculate the residue as:

$$
\Phi_1(x^H_{i,j}, x^L_{i,j}) = \Phi_2(x^H_{i,j}, x^L_{i,j})\hat{x}^H_{i,k} + r_k.
$$

3) Project the residue on atoms not selected and add the atom with maximum projection value to $I_k$:

$$
\tau_k = (\Phi_1(x^H_{i,j}, x^L_{i,j}) - \Phi_2(x^H_{i,j}, x^L_{i,j})\hat{x}^H_{i,k})^T
(\Phi_2(x^H_{i,j}, x^L_{i,j})\alpha_k)
 = (\mathcal{K}^1 - (\hat{x}^H_{i,k})^T \mathcal{K}^2)\hat{a}_k, \ t \notin I_k,
$$

where,

$$
\mathcal{K}^1 = \Phi_1(x^H_{i,j}, x^L_{i,j})^T \Phi_1(x^H_{i,j}, x^L_{i,j})
 = \begin{bmatrix}
\mathcal{K}_H & 0 \\
0 & \lambda \mathcal{K}_L
\end{bmatrix},
$$

and,

$$
\mathcal{K}^2 = \Phi_2(x^H_{i,j}, x^L_{i,j})^T \Phi_2(x^H_{i,j}, x^L_{i,j})
 = \begin{bmatrix}
0 & \lambda \mathcal{K}_L
\end{bmatrix}.
$$

Update the set $I_k$ as:

$$
I_{k+1} = I_k \cup \arg \max_{t \notin I_k} |\tau_t|.
$$

4) Update the sparse code, $\gamma_{j,k+1}$ and reconstruction, $\hat{x}^H_{i,k+1}$ as:

$$
\gamma_{k+1} = ((\Phi_2(x^H_{i,j}, x^L_{i,j})\hat{A}_{i,k+1})^T(\Phi_2(x^H_{i,j}, x^L_{i,j})\hat{A}_{i,k+1}))^{-1}
(\Phi_2(x^H_{i,j}, x^L_{i,j})\hat{A}_{i,k+1})^T \phi^L(x_{i,j}^L) = (\hat{A}_{i,k+1}^T \mathcal{K}^2 \hat{A}_{i,k+1})^{-1}(\mathcal{K}^1 \hat{A}_{i,k+1})^T, 
$$

$$
\hat{x}^H_{i,k+1} = \hat{A}_{i,k+1} \gamma_{j,k+1}.
$$

5) $k \leftarrow k + 1$; Repeat steps 2-4 $T_0$ times.

2) **Dictionary update:** The dictionaries $A^H_i$ and $A^L_i$ can now be obtained using the MOD method as follows:

$$
A^H_i = \Gamma_i^T (\Gamma_i \Gamma_i^T)^{-1},
$$

$$
A^L_i = \Gamma_i^T (\Gamma_i \Gamma_i^T)^{-1}.
$$

Further the dictionary atoms are normalized to unit norm in feature space:

$$
A^H_{i,j} = \frac{A^H_{i,j}}{\sqrt{(A^H_{i,j})^T \mathcal{K}^H(x^H_{i,j}, x^H_{i,j}) A^H_{i,j}}}, \ j = 1, \cdots, K,
$$

$$
A^L_{i,j} = \frac{A^L_{i,j}}{\sqrt{(A^L_{i,j})^T \mathcal{K}^L(x^L_{i,j}, x^L_{i,j}) A^L_{i,j}}}, \ j = 1, \cdots, K.
$$

REFERENCES


