Linear Regression Classification (LRC) based face recognition achieves high accuracy while being highly efficient. As with most other linear-subspace-based methods, the faces of a subject are assumed to reside on a linear manifold; however, where occlusion or disturbances are involved, this assumption may be inaccurate. In this paper, a manifold-learning procedure is used to expand on conventional LRC by excluding faces not fitting the original assumption (of linearity), thereby localizing the manifold subspace, increasing the accuracy over conventional LRC and reducing the number of faces for which the regression must be performed. The algorithm is evaluated using two standard databases and shown to outperform the conventional LRC.

1. Introduction

Face recognition is an active research area, with much attention being directed towards pixel intensity map-linear-subspace classification algorithms such as Nearest Feature Line (NFL) [2], Nearest Feature Subspace (NFS) [3], Sparse Representation Classification (SRC) [9] and Linear Regression Classification (LRC) [5]. These algorithms achieve high recognition rates with varying levels of complexity and efficiency. Of particular note is LRC, which represents the state-of-the-art, and has shown to be exceptionally accurate and efficient [5]. SRC generally has better accuracy, however is much slower than LRC and is not suitable for real-time processing.

The assumption common to the linear subspace based algorithms is that the test and training images of a subject’s face are all contained in the same linear subspace [3, 5, 9]. However, it is well known that the manifold, which may have to accommodate various facial expressions, orientations and lighting conditions, is not necessarily linear; it is even less likely considering potentially misaligned or partially occluded images.

If the nonlinearizing content of the manifold is reduced through manifold learning, better performance of the linear solution is usually observed [1]. The proposed Locality-Regularized Linear Regression Classification (LLRC) embeds the manifold learning system into LRC. The experimental results confirm the effectiveness of using locality-regularization in LRC, which effectively increases its robustness to facial appearance changes and achieves favorable results, both in accuracy and efficiency, in human face recognition.

2. Related work

Linear Regression Classification.

Let there be \(N\) vectorized face images \(X = [x_1, x_2, \ldots, x_N] \in \mathbb{R}^{d_0 \times N}\) containing \(C\) different individuals with each face vector having \(d_0\) elements. Let \(x^c_i \in \mathbb{R}^{d_0}\) denote the \(i^{th}\) face of the \(c^{th}\) individual, \(X_c \in \mathbb{R}^{d_0 \times N_c}\) be the image collection for the \(c^{th}\) class, and let \(y \in \mathbb{R}^{d_0}\) denote the query image. If \(y\) belongs to the \(c^{th}\) class then it is assumed to be able to be represented by a linear combination of images in the same class, i.e.

\[
y = X_c \beta_c,
\]

where \(\beta_c \in \mathbb{R}^{d_0}\) is the vector of parameters projecting the test image onto the linear space spanned by the training data images, \(X_c\). Solving (1) for \(\beta_c\) using a pseudo-inverse gives the least-squares sense approximation:

\[
\hat{\beta}_c = (X_c^T X_c)^{-1} X_c^T y.
\]

The test vector may now be represented in terms of the training images as \(y = X_c \hat{\beta}_c\). From this estimate,
the residual reconstruction error for the class can be shown as in (3), where \( I \in \mathbb{R}^{d_0 \times d_0} \) is the identity matrix.

\[
    r_c = \left\| \left( \hat{X}_{c}^\top \hat{X}_{c} \right)^{-1} \hat{X}_{c}^\top \mathbf{y} \right\|^2.
\]  
(3)

Finally, LRC classifies the query image \( y \) as belonging to the class having the least error as in (4).

\[
    \hat{c} = \arg\min_c (r_c).
\]  
(4)

LRC has an impressive performance and a simple and elegant implementation based on the assumption that all faces in \( X_c \) reside in a linear space.

3. Locality-regularized linear regression based classification

Figure 1 shows two example synthetic manifolds with face images for each subject distributed within each manifold. Suppose that the query image \( y \), shown in the figure as a square belongs to subject 2. It is likely that \( y \) lies on the plane (the black triangle) spanned by the images \( a_1, a_2, a_3 \) of subject 1. In such case, LRC considers all gallery images within the manifolds and will misclassify the query as belonging to subject 1.

![Figure 1: Showing the classification ambiguity of LRC, which uses all test faces. By using only the k-NN (in this case k=3), the test face y is correctly classified.](image)

Inspired by [7], we wish to employ the more plausible case of local linearity rather than assume global linearity of points in the classes. We propose to enforce locality regularization on LRC by only including the \( k \) closest images to the query image measured using Euclidean distance. By rejecting all but the \( k \)-NN faces, \( a_1, a_2, a_3 \) are not in the set and are not considered to belong to the query image \( y \); consequently subject 2 is correctly selected.

To this end, let us define a locality-linearized subset of \( X_c \), containing only the \( k \) closest neighbors to the test image, as \( \hat{X}_c = \{ \hat{x}_1, \hat{x}_2, \ldots, \hat{x}_k \} \in \mathbb{R}^{d_0 \times k} \), for subject \( c \). Solving (1) but substituting \( \hat{X}_c \) in place of \( X_c \) produces \( \hat{\beta}_c^* = (\hat{X}_c^\top \hat{X}_c)^{-1} \hat{X}_c^\top \mathbf{y} \), with a reconstruction error of

\[
    r_c = \left\| \left( \hat{X}_c^\top \hat{X}_c \right)^{-1} \hat{X}_c^\top \mathbf{y} \right\|^2.
\]  
(5)

The classification is now done using (4), as per LRC.

Manifold learning is employed to find the optimum \( k \) value for a given training gallery set via leave-one-out cross-validation with \( k \) in \{2, 4, 6, 8, 10\}; the trial returning the highest accuracy corresponds to the \( k \) value chosen to classify the probe image.

3.1. Kernelization

The main idea of Kernelized LLRC (KLLRC) is to map the original face samples into a high-dimensional Hilbert space \( \mathcal{F} \) and apply LLRC on the Hilbert space. The mapping function \( \Phi : \mathcal{X} \to \mathcal{F} \) is typically nonlinear and may increase the separability of data. The form of \( \Phi \) is not necessarily known explicitly; Mercer’s theorem [8] tells us the mapping function could be determined by a kernel function \( K : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \) that

\[
    K(x, z) = \langle \Phi(x), \Phi(z) \rangle.
\]  
(6)

It is now obvious that (3) can be expressed in terms of inner products between the training face set \( X_c \) and itself or the test face \( y \). Let us define

\[
    \Psi_{i,j} = \langle \Phi(\hat{x}_i), \Phi(\hat{x}_j) \rangle = K(\hat{x}_i, \hat{x}_j)
\]  
(7)

and let \( K \in \mathbb{R}^{k \times k} \) refer to the inner product vector that

\[
    \Psi_c = \langle \Phi(\hat{x}_c), \Phi(\mathbf{y}) \rangle = K(\hat{x}_c, \mathbf{y}).
\]  
(8)

It is then straightforward to arrive at

\[
    r_c = K(\mathbf{y}, \mathbf{y}) - \kappa \Psi^{-1} \kappa,
\]  
(9)

where all the individual terms appear in the form of the kernel function, hence the kernelization of LLRC is feasible.

There are numerous types of kernel functions [8], though we adopt the most popular one, the RBF kernel, the definition of which is given by

\[
    K(x, z) = \exp\left(-\gamma \| x - z \|^2\right)
\]  
(10)

Manifold learning is performed for KLLRC as for LLRC, trialing cross-validation for each \( k \) in \{2, 4, 6, 8, 10\} and each \( \gamma \) in \{1/16, 1/4, 1, 4, 16\}.

3.2. Block-based Algorithm

It should be noted that for any test image incorporating an occlusion mode, there may not be a training image in a class that will wholly resemble the test image; in fact, images in the class may actually detract from the match accuracy. For this reason, dividing each of the test and training images into 8 blocks (i.e. 4x2 partitions) and performing the matching algorithm blockwise allows poorly-matched blocks to be excluded from the result by voting on the blockwise predicted class results.

While [5] proposes the DEF competition method
using the block with the least reconstruction error (ignoring all other blocks) to decide the outcome, we found that majority voting performed significantly better with our dimensionally reduced set.

The training stage of the LLRC and KLLRC is performed for each block in isolation to obtain optimum parameter settings for each block.

4. Experimental Results

To evaluate the effectiveness of the proposed LLRC and KLLRC approaches, we conducted comparative experiments on two publicly available face databases, namely AR [4] and FERET [6].

4.1. Recognition accuracies on AR database

There are 100 individuals in AR (cropped version) dataset. Each subject consists of 26 face images, with different lighting conditions, facial expressions and occlusion modes, see Figure 2. Two sessions are recorded with a two-week time interval; the first is used as the training gallery; the second as the test set.

![Figure 2: Sample images from the AR database (cropped version).]

Table 1: Recognition accuracy for AR-100 dataset classifying over 4x2 blocks using majority voting. Note LDA is unsuitable for dimensions >100 with this dataset.

<table>
<thead>
<tr>
<th>Dimension reduction method</th>
<th>Dimensions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>20  40  50  100  200</td>
</tr>
<tr>
<td>Random LRC</td>
<td>51.8  81.5  84.4  86.6  86.5</td>
</tr>
<tr>
<td>Random NN</td>
<td>72.2  77.3  79.6  80.4  82.0</td>
</tr>
<tr>
<td>Random LLRC</td>
<td>73.7  80.6  84.4  85.2  87.0</td>
</tr>
<tr>
<td>Random KLLRC</td>
<td>75.5  81.1  84.7  85.8  87.0</td>
</tr>
<tr>
<td>PCA LRC</td>
<td>59.0  81.0  82.0  84.8  84.3</td>
</tr>
<tr>
<td>PCA NN</td>
<td>68.5  72.2  72.8  84.8  84.3</td>
</tr>
<tr>
<td>PCA LLRC</td>
<td>70.3  82.1  83.2  84.5  85.0</td>
</tr>
<tr>
<td>PCA KLLRC</td>
<td>73.8  81.0  82.4  83.3  83.7</td>
</tr>
<tr>
<td>LDA LRC</td>
<td>36.9  82.7  85.5  90.9  -</td>
</tr>
<tr>
<td>LDA NN</td>
<td>79.5  86.5  86.8  90.8  -</td>
</tr>
<tr>
<td>LDA LLRC</td>
<td>81.3  87.7  88.8  91.2  -</td>
</tr>
<tr>
<td>LDA KLLRC</td>
<td>82.2  87.8  89.1  90.9  -</td>
</tr>
</tbody>
</table>

This configuration presents a more difficult problem than using a selected subset or using cross-validation method [5]. Consequently all accuracy results will be less than those reported in [5].

All images are first downsampled to 40x40 pixels before being partitioned into 4x2 blocks and having dimensionality reduced using PCA, LDA or a random matrix.

The results in Table 1 show that LLRC and KLLRC consistently outperform LRC for all but one case. The improvement is most noticeable at low dimensions, gaining 44.4% and 45.3% respectively for LRC and KLLRC.

4.2. Sensitivity to misalignment

The AR database (non-cropped version of 120 subjects) was used to evaluate the effect of misalignment on the recognition rate of the proposed approaches. Before the probe images were aligned and cropped, the eye locations were offset by two normally distributed zero-mean random vectors ($\Delta X_1, \Delta Y_1$) and ($\Delta X_2, \Delta Y_2$). A sample of the perturbed images are shown in Figure 3. The recognition rates as a function of the eye offset variance $\sigma^2$ are illustrated in Figure 4.

![Figure 3: Sample images from the AR database perturbed by $\sigma^2$ of {0, 1, 3, 5, 9} (from left to right).]

Figure 4 shows the improvement of LLRC and KLLRC over LRC for all cases except for PCA reduction with $\sigma$ less than 1.22 pixels.

LRC, LLRC and KLLRC are all highly susceptible to misalignment of the test images with respect to the gallery images. One solution to this problem is to add perturbed images to the gallery; unfortunately for LRC this will further dilute the effectiveness and increase computation time due to the additional unlocalized images added to each class. The extra images in the gallery will only minimally burden LLRC and KLLRC with the need to calculate the additional distances; solving will still use only an optimal subset of the available gallery images. We leave this as future work.
4.3. Recognition accuracy on FERET database

The FERET database includes 7,800 images belonging to 739 individuals. There are extreme variations in alignment, rotations, poses, illuminations and backgrounds making it an important test database. Sample images from the FERET are shown in Figure 5.

Each subject in the database has a differing number of images; this experiment is carried out on a subset of the database, i.e. 20 individuals having at least 33 images.

![Figure 5: Sample images from the FERET database.](image)

For each class, 20 images are selected randomly for the training gallery and the remaining 13 are used as test images. The result is averaged over five trials.

<table>
<thead>
<tr>
<th>Dimensions Reduction Method</th>
<th>20</th>
<th>40</th>
<th>50</th>
<th>100</th>
<th>200</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random LRC</td>
<td>13.4±1.4</td>
<td>67.9±2.6</td>
<td>70.2±3.5</td>
<td>73.1±3.3</td>
<td>73.1±2.8</td>
</tr>
<tr>
<td>NN</td>
<td>69.8±4.5</td>
<td>72.7±2.3</td>
<td>71.5±5.2</td>
<td>73.2±3.4</td>
<td>74.9±3.9</td>
</tr>
<tr>
<td>LLRC</td>
<td>72.2±3.0</td>
<td>77.6±2.4</td>
<td>76.5±2.8</td>
<td>79.3±2.1</td>
<td>77.9±2.2</td>
</tr>
<tr>
<td>KLLRC</td>
<td>73.6±3.3</td>
<td>77.1±2.7</td>
<td>77.3±4.1</td>
<td>79.7±2.1</td>
<td>79.4±2.3</td>
</tr>
<tr>
<td>PCA LRC</td>
<td>17.2±3.3</td>
<td>71.8±4.4</td>
<td>72.3±4.0</td>
<td>75.2±3.0</td>
<td>73.5±2.9</td>
</tr>
<tr>
<td>NN</td>
<td>76.4±2.0</td>
<td>76.3±2.3</td>
<td>76.0±2.9</td>
<td>74.8±3.1</td>
<td>74.9±3.1</td>
</tr>
<tr>
<td>LLRC</td>
<td>79.1±3.1</td>
<td>80.2±3.0</td>
<td>79.0±2.5</td>
<td>78.1±2.7</td>
<td>78.4±2.9</td>
</tr>
<tr>
<td>KLLRC</td>
<td>79.0±2.8</td>
<td>79.4±3.2</td>
<td>79.8±2.7</td>
<td>78.8±4.1</td>
<td>78.8±3.9</td>
</tr>
</tbody>
</table>

Table 2 shows the performance mean and standard deviation of all algorithms; it is evident that LLRC and KLLRC improve upon LRC with up to 62.8% and 62.6% respectively for 20 dimensions. The proposed algorithms maintain superiority in all cases shown.

4.4. Efficiency

The algorithms are tested against 120 subjects from the AR database, with Figure 6 showing the MATLAB running time (Intel i5 2.3GHz, 8GB RAM) in milliseconds per face. It is clear that LRC is slightly faster for low dimensions where LLRC, KLLRC are significantly more accurate. The proffered methods improve on LRC in efficiency at higher dimensions.

5. Conclusion

We have improved on a state-of-the-art face recognition algorithm in terms of both accuracy and efficiency. Through manifold learning and dynamic formulation of a gallery localized around the probe image, the subspace is made more suitable for the linear solution to the equation.

Compared with LRC, the results for the LLRC showed higher accuracy in most cases and higher efficiency was noted in the higher dimensions.

Minor improvements were gained with KLLRC which employed kernelization of the algorithm to circumvent the difficulty in classification in a low dimensional linear subspace at the cost of efficiency.

References


