Sparse Representation for Face Recognition

Neeraj Gangwar

Under the guidance of
Dr. Debashis Ghosh
Associate Professor

Department of Electronics and Communication Engineering
Indian Institute of Technology Roorkee

June 2014

Contents I

Contents II

Introduction

- A system of equations is described as

$$\mathbf{y} = \mathbf{D}\mathbf{x} \tag{1}$$

  Here $\mathbf{y} \in \mathbb{R}^n$, $\mathbf{D} \in \mathbb{R}^{n \times k}$ and $\mathbf{x} \in \mathbb{R}^k$.
- Depending on the values of n and k, this system can be categorized as
  1. Underdetermined System
  2. Has a unique solution
  3. Overdetermined System
- In the case of an underdetermined system, equation (1) doesn't have a unique solution.

- To get a unique solution, additional constraints have to be imposed.

$$\arg\min_{\mathbf{x}} \quad f(\mathbf{x}) \quad \text{subject to} \quad \mathbf{y} = \mathbf{D}\mathbf{x} \tag{2}$$

- For $f(\mathbf{x}) = \|\mathbf{x}\|_2^2$

$$\hat{\mathbf{x}} = \mathbf{D}^T \left(\mathbf{D}\mathbf{D}^T\right)^{-1} \mathbf{y} = \mathbf{D}^+ \mathbf{y} \tag{3}$$

Sparse Representation

- For $f(\mathbf{x}) = \|\mathbf{x}\|_0$, equation (2) becomes a sparse representation problem.
- It can be formally described as

$$P_0 : \quad \arg\min_{\mathbf{x}} \ \|\mathbf{x}\|_0 \quad \text{subject to} \quad \mathbf{y} = \mathbf{D}\mathbf{x} \tag{4}$$

- This problem is NP-Hard.
- Over the years, algorithms to find approximate solutions have been proposed.

Algorithms to find Sparse Representation

- A basic approach to solve (4) is an exhaustive search over all possible combinations of dictionary atoms.
- Above problem is known as l_0 *minimization method* or basis pursuit and is NP-Hard.
- So l_1 *minimization method* is used instead which is a convex problem

$$P_1 : \quad \underset{\mathbf{z} \in \mathbb{R}^N}{\arg\min} \; ||\mathbf{z}||_1 \quad \text{subject to} \quad \mathbf{y} = \mathbf{Dz} \tag{5}$$

- If the sparsest solution of P_0 is sufficiently sparse, P_1 converges to the same solution as P_0.

- To solve basis pursuit problem, many regression algorithms are used such as LASSO and LARS.
- Basis pursuit methods are computationally costly.
- To overcome this issue, greedy approaches were devised. These methods are sub-optimal and sometimes fail to give the correct solutions.
- For a very low value of sparsity, these algorithms give a good approximate solution.

Classification Problem

- Classification problem can be described as *using labeled training samples from L distinct classes to correctly determine to which class a new sample belongs to.*
- Face recognition is a popular classification problem in computer vision.
- **Notations**
  - Columns $\mathbf{d}_{i,j} \in \mathbb{R}^n, 1 \leq j \leq k_i$ of a matrix $\mathbf{D}_i \in \mathbb{R}^{n \times k_i}$ represent the training images.
  - Images are assumed to be grayscale of size $w \times h$. So $n = w \times h$.
  - Given L classes, a dictionary $\mathbf{D}$ is formed by concatenating $\mathbf{D}_i$

$$\mathbf{D} = [\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3 \ldots \mathbf{D}_L] \tag{6}$$

Test Image as a Linear Combination of Training Images
[SRC]

- It has been observed that the face images under different illuminations and varying expressions lie on a linear subspace.
- A test image of class i, $\mathbf{y} \in \mathbb{R}^n$ can be written as

$$\mathbf{y} = x_{i,1}\mathbf{d}_{i,1} + x_{i,2}\mathbf{d}_{i,2} + x_{i,3}\mathbf{d}_{i,3} + \ldots x_{i,k_i}\mathbf{d}_{i,k_i} \tag{7}$$

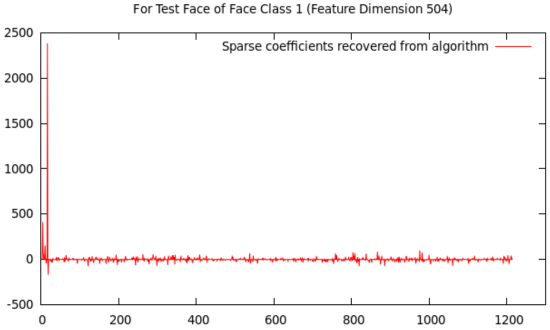
- As a linear combination of all training samples

$$\mathbf{y} = \mathbf{D}\mathbf{x} \tag{8}$$

  Here $\mathbf{x} = [0, 0, \ldots x_{i,1}, x_{i,2} \ldots x_{i,k_i}, 0, 0 \ldots 0]$.
- If L is large enough, equation (8) becomes a sparse representation problem.

Sparse Representation based Classification

- Reconstructed coefficients for a test image of class 1 looks like



- **Observation:** Large coefficients are concentrated on class 1.

- One method for classifying the test samples is to use concentration of the reconstructed coefficients.
- For each class i, let $\delta_i : \mathbb{R}^n \to \mathbb{R}^n$ be a function that selects coefficients associated with class i and makes all other entries zero.
- A metric for the concentration of coefficients on class i is defined as

$$\alpha_i = \frac{\|\delta_i(\mathbf{x})\|_1}{\|\mathbf{x}\|_1} \tag{9}$$

- If value of α_i exceed a pre-defined threshold, label i is assigned to the test image $\mathbf{y}$.

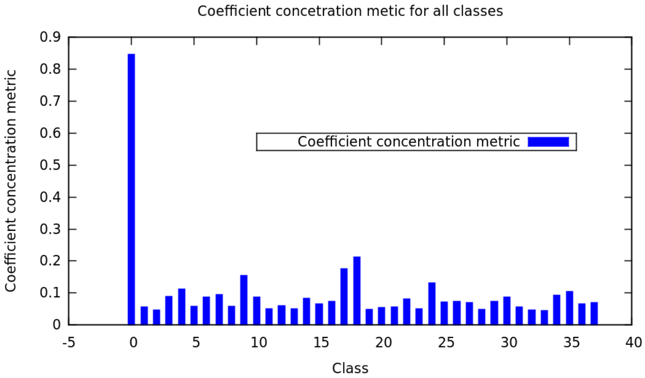


Figure : Coefficient concentration for a test image of class 1

- Another approach for classification is to use residuals w.r.t. different classes.
- Using coefficients associated with class i, test image can be approximated as $\hat{\mathbf{y}}_i = \mathbf{D}\delta_i(\mathbf{x})$.
- $\mathbf{y}$ is assigned to the class i using

$$\min_i r_i(\mathbf{y}) = ||\mathbf{y} - \mathbf{D}\delta_i(\mathbf{x})||_2 \tag{10}$$

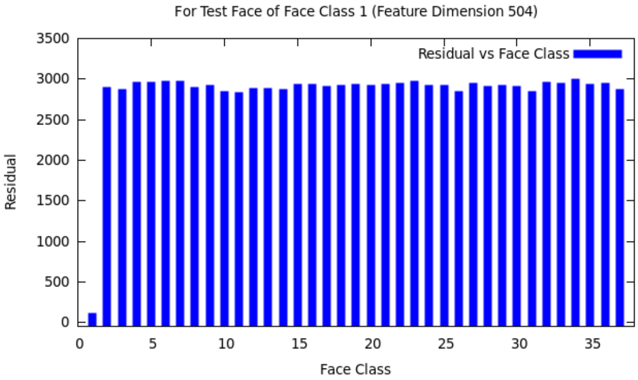
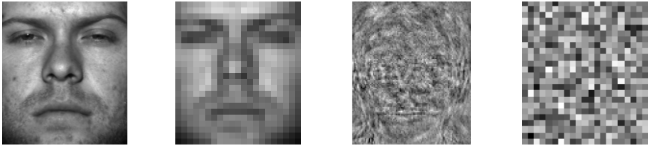


Figure : Residual for a test image of class 1

Feature Extraction

- Following feature extraction techniques are used for all the simulations
  1. Eigenfaces
  2. Randomfaces
  3. Downsampling



(a) (b) (c) (d)

Figure : (a) Original (b) Downsampling (c) Eigenfaces (d) Randomfaces

Simulation Results

[SRC]

- Comparison of performance for different feature extraction techniques for extended Yale B database

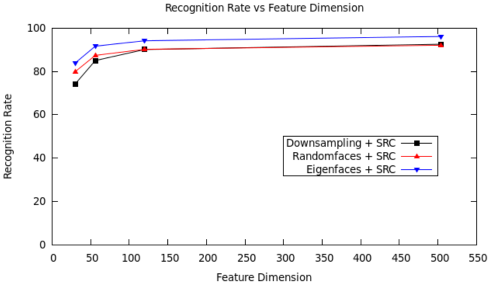


Figure : Recognition rate for Extended Yale B Database

- It gives almost similar performance with all of them. So any technique can be used to save computation cost.

- Comparison between different feature extraction techniques for extended Yale B database

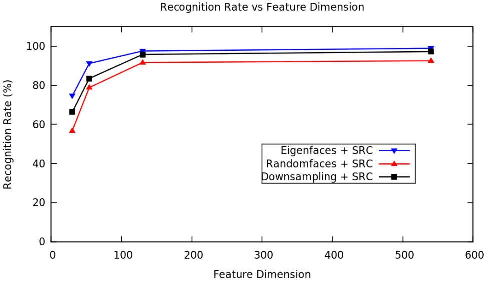


Figure : Recognition rate for AR face database

- Confusion matrices

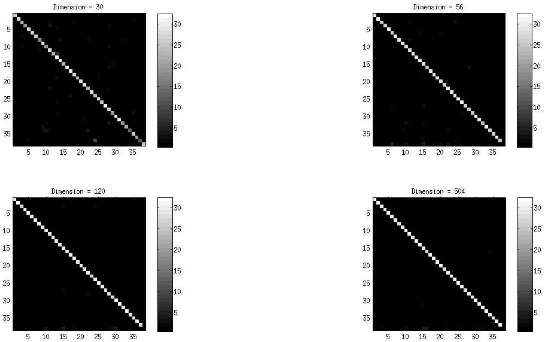


Figure : Confusion matrices for Yale extended B database (Downsampling)

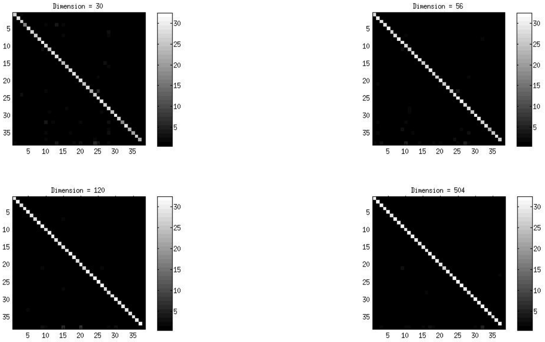


Figure : Confusion matrices for Yale extended B database (Randomfaces)

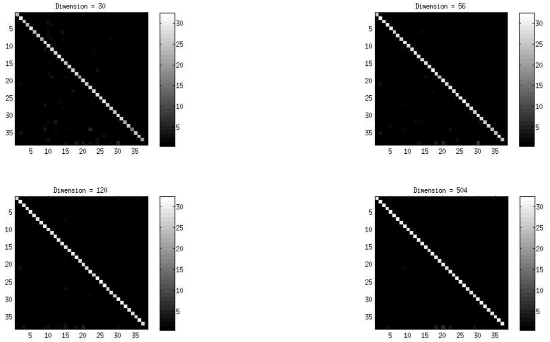


Figure : Confusion matrices for Yale extended B database (Eigenfaces)

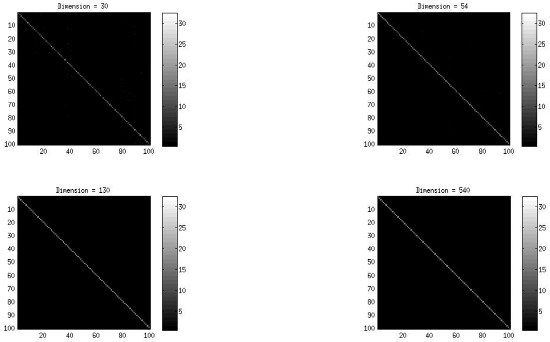


Figure : Confusion matrices for AR database (Downsampling)

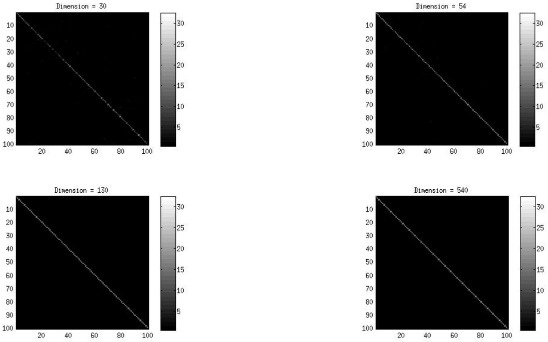


Figure : Confusion matrices for AR database (Randomfaces)

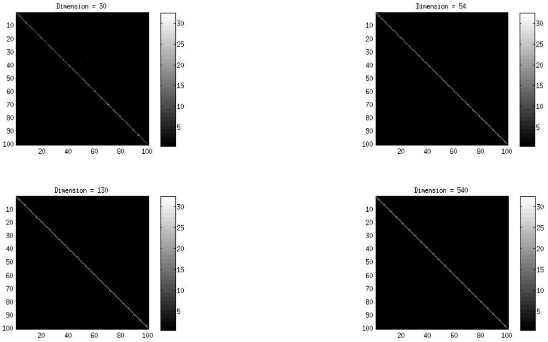


Figure : Confusion matrices for AR database (Eigenfaces)

Handling an Irrelevant Test Image
[SRC]

- One important aspect of any face recognition algorithm is to discard an invalid test image.
- Reconstructed coefficients for an invalid test image

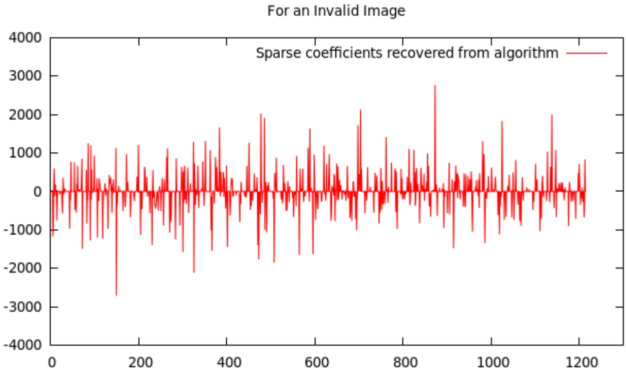


Figure : Reconstructed coefficients for an invalid image

- **Observation:** Coefficients are pretty scattered.
- A sparsity concentration index (SCI) is used to check if a test image is valid.

$$\mathsf{SCI}(\mathbf{x}) = \frac{k \cdot \max_i ||\delta_i(\mathbf{x})||_1 / ||\mathbf{x}||_1 - 1}{k-1} \quad \in [0,1] \tag{11}$$

- An image is discarded if SCI is below a threshold value.

Dictionary Learning

- What if the number of training examples is very large?
- **Solution:** Dictionary Learning
- It's a method of learning dictionary atoms suitable for the training data, rather than using the training data directly.
- **Advantage:** Size of dictionary can be controlled depending on computational capabilities.
- Popular Algorithms
  1. Method of optimal directions
  2. K-SVD
- We used K-SVD for dictionary learning.
- A dictionary learning algorithm tries to solve

$$\min_{\mathbf{D}} ||\mathbf{Y} - \mathbf{DX}||_F \quad \text{subject to} \quad ||\mathbf{x}_i||_0 \leq k_0 \quad 1 \leq i \leq N \tag{12}$$

K-SVD: A Dictionary Learning Algorithm

Algorithm 1: The K-SVD Algorithm

Data: Input singals in form of a matrix $\mathbf{Y}$.
Initialization : Set the dictionary matrix $\mathbf{D}^{(0)} \in \mathbb{R}^{n \times k}$ with l_2 normalized columns.
$J = 1$.
until *Stopping criterion is met*, **do**

- *Sparse coding stage*: Any sparse decomposition algorithm to compute sparse vectors $\mathbf{x}_i$ for each column $\mathbf{y}_i$ over the dictionary computed in the last iteration.
- *Codebook update*: For each column $j = 1, 2, \ldots k$ in $\mathbf{D}^{J-1}$. Update it as
  - Obtain the examples that use this atom, $\omega_j = \{i \mid 1 \leq i \leq N,\ \mathbf{x}_T^j \neq 0\}$.
  - Compute the error matrix, $\mathbf{E}_j$ by

$$\mathbf{E}_j = \mathbf{Y} - \sum_{i \neq j} \mathbf{d}_i \mathbf{x}_T^i$$

  - Restrict $\mathbf{E}_j$ by choosing columns corresponding to ω_j and obtain $\mathbf{E}_j^R$.
  - Use SVD to update dictionary column $\mathbf{d}_j$ and $\mathbf{x}_R^j$.
- $J = J + 1$

Label Consistent K-SVD

- For classification, a discriminative dictionary is needed.
- LC-KSVD is a discriminative dictionary learning algorithm.
- This algorithm is defined as

$$\langle \mathbf{D}, \mathbf{B}, \mathbf{X} \rangle = \arg\min_{\mathbf{D}} \; \|\mathbf{Y} - \mathbf{DX}\|_F + \alpha \|\mathbf{S} - \mathbf{BX}\|_F$$

- $\mathbf{S}$ is discriminative sparse code matrix.

$$\mathbf{S} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{13}$$

Solution
[LC-KSVD]

- Equations describing LC-KSVD can be reduced

$$\langle \mathbf{D}, \mathbf{B}, \mathbf{X} \rangle = \min_{\mathbf{D},\mathbf{B},\mathbf{X}} \left\| \begin{pmatrix} \mathbf{Y} \\ \sqrt{\alpha}\,\mathbf{S} \end{pmatrix} - \begin{pmatrix} \mathbf{D} \\ \sqrt{\alpha}\,\mathbf{B} \end{pmatrix} \mathbf{X} \right\|_F^2$$

$$\text{subject to} \quad ||\mathbf{x}_i||_0 \leq k_0 \quad 1 \leq i \leq N$$

- It can also be written as

$$< \mathbf{D}_{new}, \mathbf{X} >= \min_{\mathbf{D}_{new}} ||\mathbf{Y}_{new} - \mathbf{D}_{new}\mathbf{X}||_2$$

$$\text{subject to} \quad ||\mathbf{x}_i||_0 \leq k_0 \quad 1 \leq i \leq N$$

- Above equation can be solved by K-SVD.

Classification
[LC-KSVD]

- Once dictionary $\mathbf{D}$ is obtained, a classifier $\mathbf{W}$ is trained by using ridge regression.
- Formally, it can be described as

$$\mathbf{W} = \arg\min_{\mathbf{W}} \|\mathbf{H} - \mathbf{W}\mathbf{X}\|_F + \lambda \|\mathbf{W}\|_2 \tag{14}$$

which can be solved to obtain

$$\mathbf{W} = \mathbf{H}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \lambda\,\mathbf{I})^{-1} \tag{15}$$

Here $\mathbf{H}$ contains true label information of input data.

Proposed Classification Approach

[LC-KSVD]

- To classify a test sample $\mathbf{y}$, the steps involved are as follows
  1. Compute sparse representations over the learned dictionary $\mathbf{D}$.
  2. Obtain the product $\mathbf{b} = \mathbf{B}\mathbf{x}$.
  3. For class i, compute concentration parameter
$$\alpha_i = \frac{\|\delta_i(\mathbf{b})\|_1}{\|\mathbf{b}\|_1} \tag{16}$$
  4. Assign $\mathbf{y}$ to the class that maximizes α_i.

Simulation Results
[LC-KSVD]

- Comparison between the trained classifier and the proposed classification approach

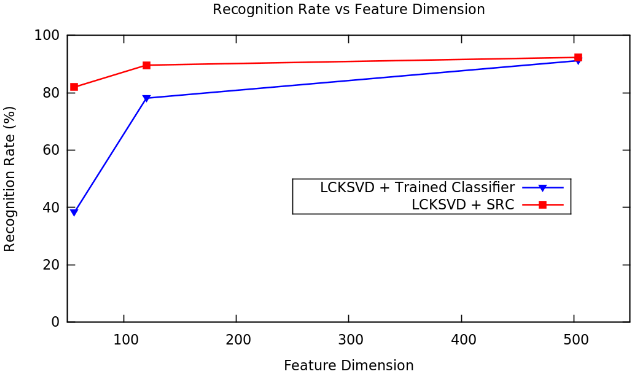


Figure : Recognition rate of LC-KSVD for extended Yale B database

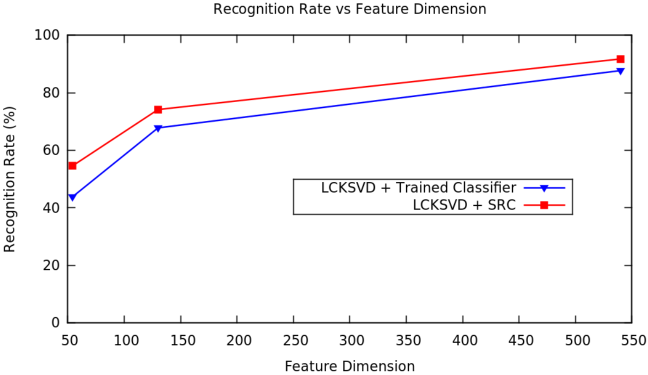


Figure : Recognition rate of LC-KSVD for AR face database

- Confusion matrix

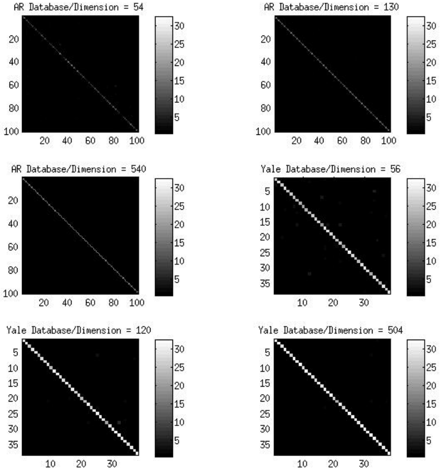


Figure : Confusion matrices for LC-KSVD

Task Driven Dictionary Learning

- Another approach to learn a discriminative dictionary.
- Task driven dictionary learning can be formulated as

$$\underset{\mathbf{D},\ \mathbf{W}}{\arg\min}\quad f(\mathbf{D},\ \mathbf{W}) + \nu\,\|\mathbf{W}\|_F^2 \tag{17}$$

where f has a form

$$f(\mathbf{D},\ \mathbf{W}) = \frac{1}{N}\sum_{i=1}^{N} l_s(z_i,\ \mathbf{W},\ \mathbf{x}_i) \tag{18}$$

- Gradient descent algorithm is used to solve this problem.

- The next important point to consider is the differentiability of f with respect to $\mathbf{D}$ and $\mathbf{W}$.

$$\begin{aligned} \nabla_{\mathbf{W}} f(\mathbf{D},\ \mathbf{W}) &= \nabla_{\mathbf{W}} l_s(\mathbf{z},\ \mathbf{W},\ \mathbf{X}) \\ \nabla_{\mathbf{D}} f(\mathbf{D},\ \mathbf{W}) &= -\mathbf{D}\boldsymbol{\beta}^* \mathbf{X}^T + (\mathbf{Y} - \mathbf{D}\mathbf{X}){\boldsymbol{\beta}^*}^T \end{aligned} \tag{19}$$

Here $\boldsymbol{\beta}^*$ is a matrix formed by concatenating the vectors β^* which are defined for a signal $\mathbf{x}_i$ as

$$\beta^*_{\Lambda^C} = 0 \quad \text{and} \quad \beta^*_{\Lambda} = (\mathbf{D}_{\Lambda}^T \mathbf{D}_{\Lambda} + \lambda_2 \mathbf{I})^{-1} \nabla_{\mathbf{x}_{i_\Lambda}} l_s(z,\ \mathbf{W},\ \mathbf{x}_i) \tag{20}$$

where $\Lambda = \{ j \mid x_{i_j} \neq 0 \}$.

Algorithm 2: Gradient Descent Algorithm for Task Driven Dictionary Learning

Data: Training samples $\mathbf{Y}$ and regularization parameters λ_1, λ_2 and ν.
Initialization: J $= 1$, $\mathbf{D}_0$ and $\mathbf{W}_0$
Repeat following steps for specified number of iterations

1. Sparse coding stage: For each training sample $\mathbf{y}_i$, compute sparse representations $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$.

$$\mathbf{x}_i \leftarrow \arg\min_{\mathbf{x}_i} \ \frac{1}{2} \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 + \lambda_1 \|\mathbf{x}_i\|_1 + \lambda_2 \|\mathbf{x}_i\|_2^2$$

2. Form a matrix $\boldsymbol{\beta}^*$ by concatenating β^* for each signal as
   - Compute Λ as

$$\Lambda \leftarrow \{j \mid x_{i_j} \neq 0\}$$

   - Compute β^*

$$\beta^*_{\Lambda^C} = 0 \quad \text{and} \quad \beta^*_{\Lambda} = (\mathbf{D}_\Lambda^T \mathbf{D}_\Lambda + \lambda_2 \mathbf{I})^{-1} \nabla_{\mathbf{x}_{i_\Lambda}} l_s(z, \ \mathbf{W}, \ \mathbf{x}_i)$$

3. Update $\mathbf{D}$ and $\mathbf{W}$

$$\mathbf{W} \leftarrow \mathbf{W} - \rho(\nabla_{\mathbf{W}} l_s(\mathbf{z}, \ \mathbf{W}, \ \mathbf{X}) + \nu \mathbf{W})$$
$$\mathbf{D} \leftarrow \mathbf{D} - \rho(-\mathbf{D}\boldsymbol{\beta}^* \mathbf{X}^T + (\mathbf{Y} - \mathbf{D}\mathbf{X}){\beta^*}^T)$$

4. J $=$ J $+ 1$

The Proposed Approach
[Task Driven Dictionary Learning]

- The proposed approach uses task driven dictionary learning with SRC.
- Face classification is a multiclass classification problem and it can be solved in several ways.
- One possible way is to use a set of binary classifiers in one-vs-all setting. This approach doesn't scale well for large datasets.
- Another approach to use a multiclass loss function. This approach produces a single dictionary unlike one-vs-all setting.
- We follow the latter approach and use softmax function.

Softmax Function

[Task Driven Dictionary Learning]

- For multiclass classification, hypothesis takes the form of

$$h_{\mathbf{W}}(\mathbf{x}^{(i)}) = \begin{bmatrix} p(z=1 \mid \mathbf{x}^{(i)}, \mathbf{W}) \\ p(z=2 \mid \mathbf{x}^{(i)}, \mathbf{W}) \\ \vdots \\ p(z=L \mid \mathbf{x}^{(i)}, \mathbf{W}) \end{bmatrix} = \frac{1}{\sum_{j=1}^{L} e^{\mathbf{w}_j^T \mathbf{x}^{(i)}}} \begin{bmatrix} e^{\mathbf{w}_1^T \mathbf{x}^{(i)}} \\ e^{\mathbf{w}_2^T \mathbf{x}^{(i)}} \\ \vdots \\ e^{\mathbf{w}_L^T \mathbf{x}^{(i)}} \end{bmatrix} \tag{21}$$

- In this setting, the cost function is defined as

$$J(\mathbf{W}) = -\frac{1}{N} \left[\sum_{i=1}^{N} \sum_{j=1}^{L} 1 \left\{ z^{(i)} = j \right\} \log \frac{e^{\mathbf{w}_j^T \mathbf{x}^{(i)}}}{\sum_{l=1}^{L} e^{\mathbf{w}_l^T \mathbf{x}^{(i)}}} \right] \tag{22}$$

- The derivatives with respect to $\mathbf{W}$ and $\mathbf{x}$ are

$$\nabla_{\mathbf{w}_j}(\mathbf{W}) = -\frac{1}{N}\sum_{i=1}^{N}\left[\mathbf{x}^{(i)}\left(1\left\{z^{(i)}=j\right\} - \frac{e^{\mathbf{w}_j^T\mathbf{x}^{(i)}}}{\sum_{l=1}^{L} e^{\mathbf{w}_l^T\mathbf{x}^{(i)}}}\right)\right] \tag{23}$$

$$\nabla_{\mathbf{x}^{(i)}}(\mathbf{W}) = -\frac{1}{N}\left[\mathbf{w}_j - \frac{e^{\mathbf{w}_j^T\mathbf{x}^{(i)}}\mathbf{w}_j}{\sum_{l=1}^{L} e^{\mathbf{w}_l^T\mathbf{x}^{(i)}}}\right] \tag{24}$$

  In equation (24), j is the class of sample $\mathbf{x}$.
- For classification, sparse representation based classification (SRC) is used with the learned dictionary $\mathbf{D}$.

Simulation Results
[The Proposed Approach]

- Recognition rate for extended Yale B database

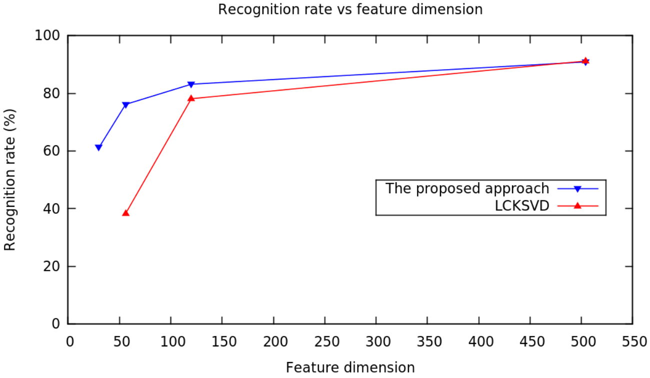


Figure : Recognition rate of proposed approach for extended Yale B database

- Recognition rate for AR face database

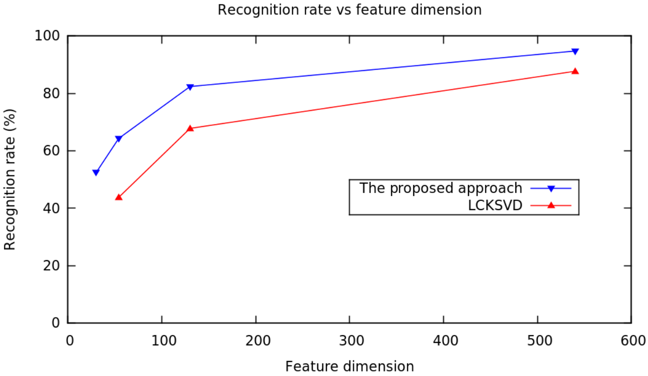


Figure : Recognition rate of proposed approach for AR face database

- Confusion matrix

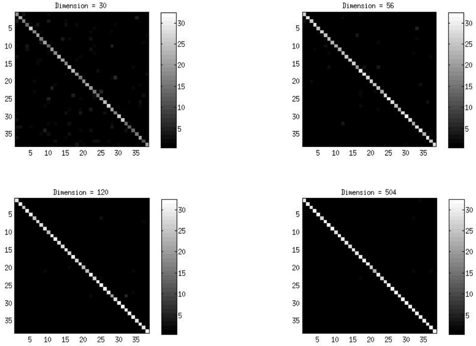


Figure : Confusion matrices for extended Yale B database (Proposed approach)

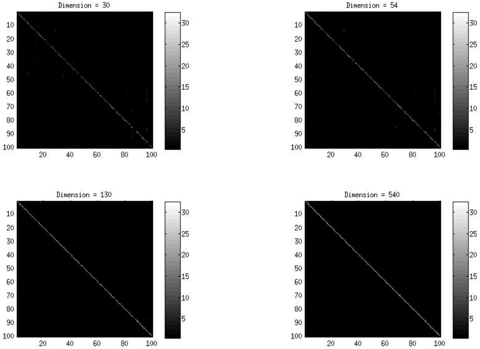


Figure : Confusion matrices for AR face database (Proposed approach)

Conclusion

- It can be concluded by the results that a sparse representation based framework gives promising results for classification.
- It has also been validated that the proposed classification for LC-KSVD gives superior results than the trained classifier.
- The proposed approach for classification performs well for both the databases and for all feature dimensions.
- It is apparent that a very good discriminative dictionary is needed for good classification.
- Future work in this field includes discriminative dictionary learning algorithms with better discriminative factor.

Thank You!