

Face Recognition via Sparse Representation

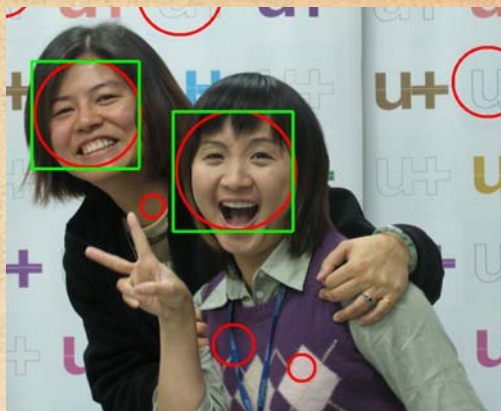
John Wright, Allen Y. Yang, Arvind,
S. Shankar Sastry and Yi Ma

IEEE Trans. PAMI, March 2008



National Taiwan University
CMLAB , since 1991

Research About Face



Face Detection



Face Alignment

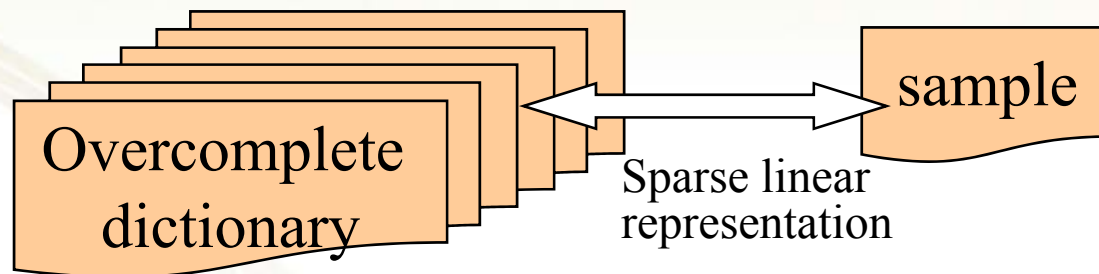


Face Recognition



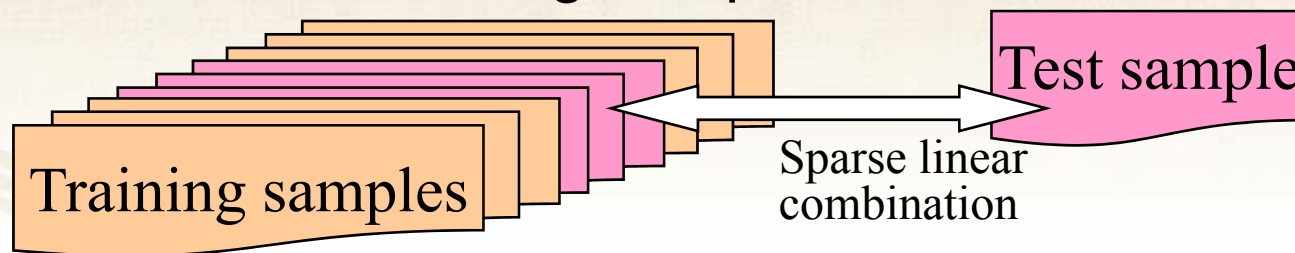
Sparse Representation

- What is it?
 - Represent a **sample** with respect to an **overcomplete dictionary**
 - The representation is sparse and linear
- Advantages
 - More concise (Compression)
 - Naturally Discriminative (**Classification**)



Classification via sparse representation

- Sample \rightarrow test sample
- Overcomplete dictionary \rightarrow training samples
- The test sample can be represented as a **linear combination** of training samples only from its class
- This representation is naturally “**sparse**”, compared to the whole training samples.

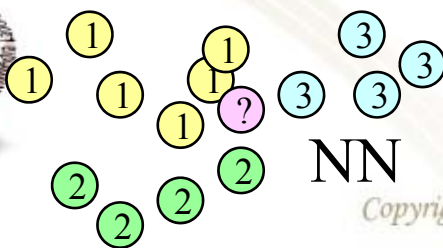


- The representation can be recovered efficiently via ℓ^1 -norm minimization
- Seeking **the sparsest representation** automatically discriminates different classes in the training set.

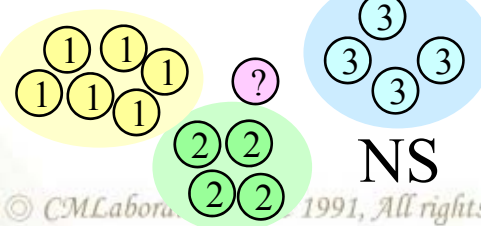


Comparison with related approaches

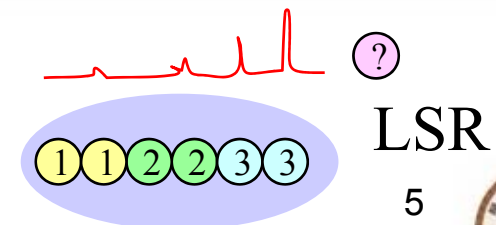
- **Nearest Neighbor** classifies the test sample based on the best representation in terms of a single training sample.
- **Nearest Subspace** classifies the test samples based on the best linear representation in terms of all the training samples in each class
- **Linear Sparse Representation** considers all possible supports (within each class or across multiple classes) and adaptively chooses the minimal number of training samples needed to represent each test sample.



NN



NS



LSR

5



New Insights 1: The Role of Feature Extraction

- Traditionally
 - Good feature provide more information for classification
 - Various features have been investigated for projecting the high-dimensional test image into low-dimensional feature spaces
 - E.g. Eigenface, Fisherface, Laplacianface
 - Lack of guidelines to decide which feature to use
- Recently, with the theory of compressed sensing
 - The choice of features is no longer critical
 - What is critical
 - The dimension of the feature space is sufficiently large
 - The sparse representation is correctly computed



New Insights 2: Robustness to Occlusion

- Traditionally
 - Occlusion poses a significant obstacle to robust, real-world face recognition
- In this work
 - Since the error corrupts **only a fraction** of the image pixels, and is therefore “**Sparse**” in the standard basis given by individual pixels.
 - It can be handled uniformly within the proposed frameworks.



Outline

- Introduction
- Classification Based on Sparse Representation
- Two Fundamental Issues in Face Recognition
- Experimental Results
- Conclusion & Future works



Symbols

- There are k distinct object classes in the training data
- The n_i given training samples, taken from the i -th class, are arranged as columns of a matrix

$$A_i = [v_{i,1}, v_{i,2}, \dots, v_{i,n_i}] \in \mathbb{R}^{m \times n_i}$$

- In the context of face recognition, we will identify a $w \times h$ grayscale image with the vector $v \in \mathbb{R}^m$ ($m = wh$) given by stacking it columns; the columns of A_i are then the training face images of the i -th subject.



A. Test Sample as a Sparse Linear Combination of Training Samples

- Observation:
 - The images of faces under varying lighting and expression lie on a special low-dimensional subspace — **face subspace**.
- Assumption:
 - The training samples from a single class do lie on a subspace



- Given sufficient training samples of the i -th object class, $A_i = [v_{i,1}, v_{i,2}, \dots, v_{i,n_i}] \in \mathbb{R}^{m \times n_i}$, any new (test) sample $y \in \mathbb{R}^m$ from the same class, will approximately lie in the linear span of the training samples associated with object i :

$$y = \alpha_{i,1}v_{i,1} + \alpha_{i,2}v_{i,2} + \dots + \alpha_{i,n_i}v_{i,n_i} \quad (1)$$

- For some scalars $\alpha_{i,j} \in \mathbb{R}, j = 1, 2, \dots, n_i$



- Since the membership i of the test sample is initially unknown, we define a new matrix A for the entire training set as the concatenation of the n training samples of all k object classes:

$$A = [A_1, A_2, \dots, A_k] = [v_{1,1}, v_{1,2}, \dots, v_{k,n_k}] \quad (2)$$

- Then the linear representation of y can be written in terms of all training samples as

$$y = Ax_0 \in \mathbb{R}^m \quad (3)$$

where

$x_0 = [0, \dots, 0, \alpha_{i,1}, \alpha_{i,2}, \dots, \alpha_{i,n_i}, 0, \dots, 0]^T \in \mathbb{R}^m$
 is a coefficient vector whose entries are zero except those associated with i -th class.



$$y = \begin{bmatrix} \begin{matrix} \text{A}_1 & \text{A}_2 & \text{A}_3 \end{matrix} \\ A \end{bmatrix} \begin{bmatrix} x_0 \\ \alpha_{21} \\ \alpha_{22} \\ \alpha_{23} \\ \alpha_{24} \\ \alpha_{25} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



- Since the entries of the x_0 encode the identity of the test sample y , we can find the associated class of y by solving the linear system of equations $y = Ax$
- Remarks
 1. A more discriminative classifier from such a “global” representation can be obtained, which demonstrate its superiority over those “local” methods (NN or NS) both for identifying objects represented in the training set and for rejecting outlying samples that do not arise from any of the classes presented in the training set.
 2. These advantages can come without an increase in the order of growth of complexity: the complexity remains linear in the size of training set.



To Solve $y=Ax$

- If $m>n$, the system of equations $y=Ax$ is “**over-determined**” and the correct x_0 can usually be found as its “unique” solution.
- However, in robust face recognition, the system $y=Ax$ is typically “**under-determined**,” and so its solution is not unique.
- Conventionally, this difficulty is resolved by choosing the minimum **ℓ^2 -norm solution**,

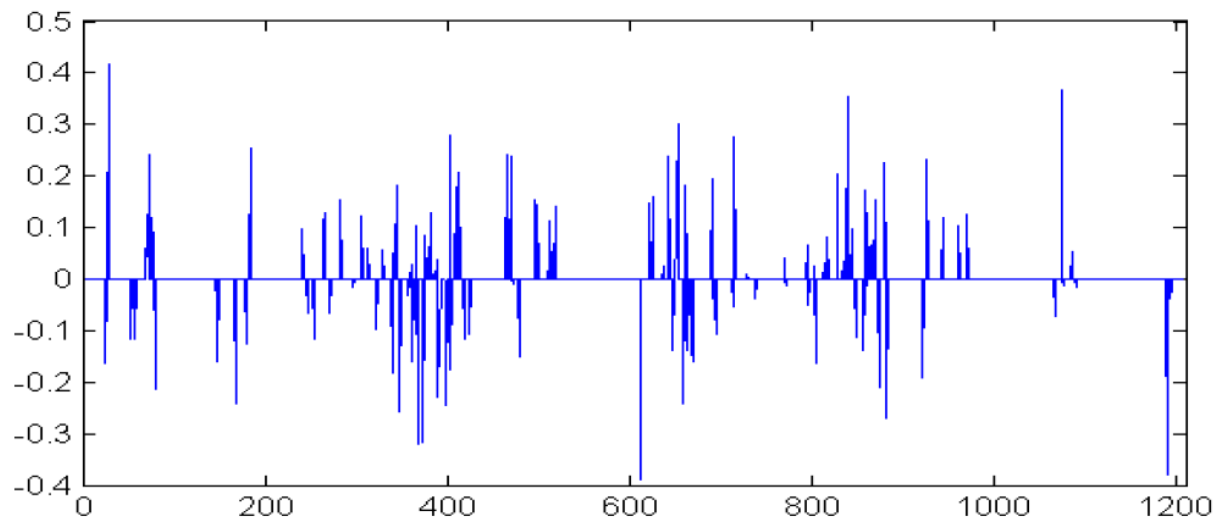
$$(\ell^2) : \hat{x}_2 = \arg \min \|x\|_2 \quad \text{subject to } Ax = y \quad (4)$$



$$(\ell^2) : \hat{x}_2 = \arg \min \|x\|_2 \text{ subject to } Ax = y$$

Problems of ℓ^2

- Eqn.(4) can be easily solved (via the pseudo-inverse of A), the solution \hat{x}_2 is not especially informative for recognizing the test image y .
- As shown in Example 1 (Fig.4), \hat{x}_2 is generally **dense**, with large nonzero entries corresponding to training samples from many different classes.



Observations

- To solve the above difficulty, the following simple observation is exploited:
 - A valid test sample y can be sufficiently represented using **only** the training samples from the **same class**.
 - This representation is naturally sparse if the number of object classes k is reasonably large. The **more sparse** the recovered x_0 is, the easier will it be to **accurately** determine the identity of the test sample y .



To solve $y=Ax$ via ℓ^0

- Idea
 - Seek the sparsest solution to $y=Ax$
- Method
 - Solving the following optimization problem:

$$(\ell^0) : \hat{x}_0 = \arg \min \|x\|_0 \text{ subject to } Ax = y \quad (5)$$
 - Where $\|\cdot\|_0$ denotes the ℓ^0 -norm, which counts the number of nonzero entries in a vector
- Reason
 - It has been proven that whenever $y=Ax$ for some x with less than $m/2$ nonzeros, x is the unique sparse solution: $\hat{x}_0 = x$.
- Problem
 - solving (ℓ^0) is NP-hard, and difficult even to approximate. — combinatorial optimization!



$$(\ell^0) : \hat{x}_0 = \arg \min \|x\|_0 \text{ subject to } Ax = y$$

B. Sparse Solution via ℓ^1 -Minimization

- Recent development in the emerging theory of sparse representation and compressed sensing reveals that if the solution x_0 sought is **sparse enough**, the solution of the Eqn.(5) is equal to the solution of the following ℓ^1 -minimization problem:

$$(\ell^1) : \hat{x}_1 = \arg \min \|x\|_1 \text{ subject to } Ax = y \quad (6)$$

- This problem can be solved in **polynomial time** by standard linear programming method.



As long as the number of nonzero entries of x_0 is a small fraction of the dimension m , ℓ^1 -minimization will recover x_0 .



Dealing with noise

- Since **real data are noisy**, the model(3) can be modified to explicitly account for small, possibly dense noise, by writing:

$$y = Ax_0 + z \quad (7)$$

- Where $z \in \mathbb{R}^m$ is a noise term with bounded energy $\|z\|_2 < \varepsilon$. The sparse solution x_0 can still be approximately recovered by solving the following stable l^1 -minimization problem:

$$(\ell_s^1) : \hat{x}_1 = \arg \min \|x\|_1 \text{ subject to } \|Ax - y\|_2 \leq \varepsilon \quad (8)$$

- This convex optimization problem can be efficiently solved via second-order cone programming

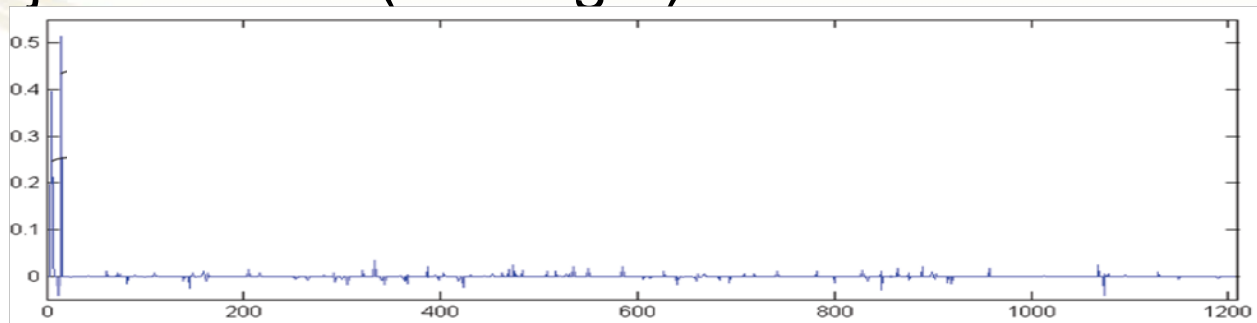


$$(\ell^1) : \hat{x}_1 = \arg \min \|x\|_1 \text{ subject to } Ax = y$$

$$(\ell_s^1) : \hat{x}_1 = \arg \min \|x\|_1 \text{ subject to } \|Ax - y\|_2 \leq \varepsilon$$

C. Classification Based on Sparse Representation

- Given a new test sample y from one of the classes in the training set, we first compute its sparse representation \hat{x}_1 via (6) or (8).
- Ideally, the nonzero entries in the estimate \hat{x}_1 will all be associated with the columns of A from a **single** object class i , and we can easily assign the test sample y to that class.
- However, noise and modeling error may lead to small nonzero entries associated with **multiple** object classes (see Fig.3)



- Based on the global, sparse representation, one can design many possibly classifiers to resolve this. For instance, we can **simply assign** y to the object class with the single **largest entry** in \hat{x}_1 .
- However, such heuristics do not harness the subspace structure associated with images in face recognition.
- To better harness such linear structure, we instead classify y based on how well the **coefficients associated with all training samples** of each object reproduce y



Methods

- For each class i , let $\delta_i : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be the characteristic function which selects the coefficients associated with the i -th class.
- For $x \in \mathbb{R}^n$, $\delta_i(x) \in \mathbb{R}^n$ is a new vector whose only nonzero entries are the entries in x that are **associated with class i** .
- Using only the coefficients associated with the i -th class, one can approximate the given test sample y as $\hat{y}_i = A\delta_i(\hat{x}_1)$.
- We then classify y based on these approximations by assigning it to the object class that minimizes the residual between y and \hat{y}_i :

$$\min_i r_i(y) = \|y - A\delta_i(\hat{x}_1)\|_2 \quad (9)$$



$$\hat{y}_2 = \begin{bmatrix} \begin{matrix} \text{A}_1 & \text{A}_2 & \text{A}_3 \end{matrix} \\ A \end{bmatrix} \begin{bmatrix} \delta_2(\hat{x}_1) \end{bmatrix}$$

$\delta_2(\hat{x}_1)$	\hat{x}_1
0	a_{11}
0	a_{22}
0	a_{33}
a_{21}	a_{21}
a_{22}	a_{22}
a_{23}	a_{23}
a_{24}	a_{24}
a_{25}	a_{25}
0	a_{31}
0	a_{32}
0	a_{33}
0	a_{34}

Algorithm 1: Sparse Representation-based Classification (SRC)

1: Input: a matrix of training samples

$A = [A_1, A_2, \dots, A_k] \in \mathbb{R}^{m \times n}$ for k classes, a test sample $y \in \mathbb{R}^m$, (and an optional error tolerance $\varepsilon > 0$.)

2: Normalize the columns of A to have unit ℓ^2 -norm.

3: Solve the ℓ^1 -minimization problem:

$$\hat{x}_1 = \arg \min_x \|x\|_1 \text{ subject to } Ax = y$$

(Or alternatively, solve

$$\hat{x}_1 = \arg \min_x \|x\|_1 \text{ subject to } \|Ax - y\|_2 \leq \varepsilon \quad)$$

4: Compute the residuals

$$r_i(y) = \|y - A\delta_i(\hat{x}_1)\|_2, \text{ for } i = 1, \dots, k$$

5: Output: $identity(y) = \arg \min_i r_i(y)$.



Example 1

(l^1 -Minimization versus l^2 -Minimization)

- Randomly select half of the 2,414 images in the **Extended Yale B database** as the training set, and the rest for testing.
- In this example, we sub-sample the images from the original 192x168 to size 12x10. The pixel values of the down-sampled image are used as 120-D features – stacked as columns of the matrix A in the algorithm. Hence matrix A has size 120x1207, and the system $y = Ax$ is underdetermined. (See Fig. 3)
- Algorithm 1 achieves an overall recognition rate of **92.1%** across the Extended Yale B database





Testing Input



Feature Extraction

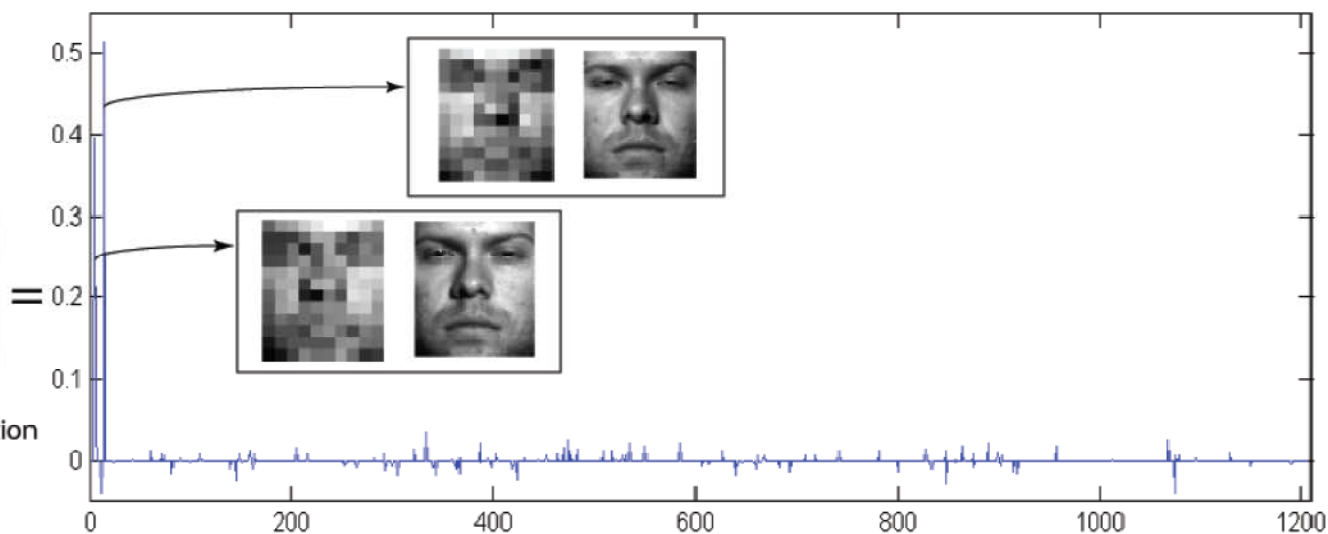
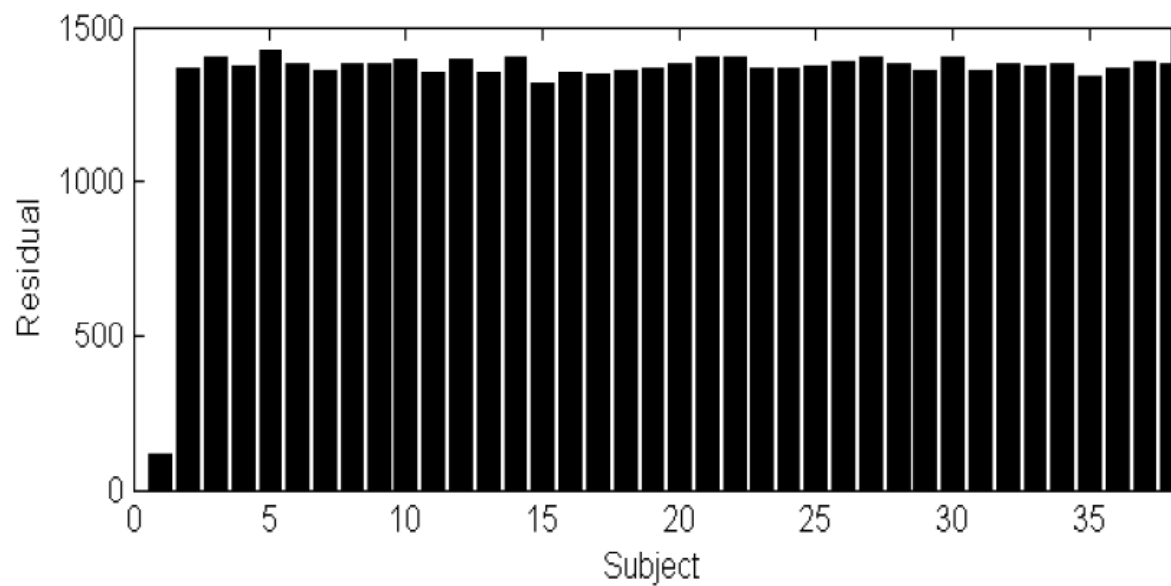


Fig. 3



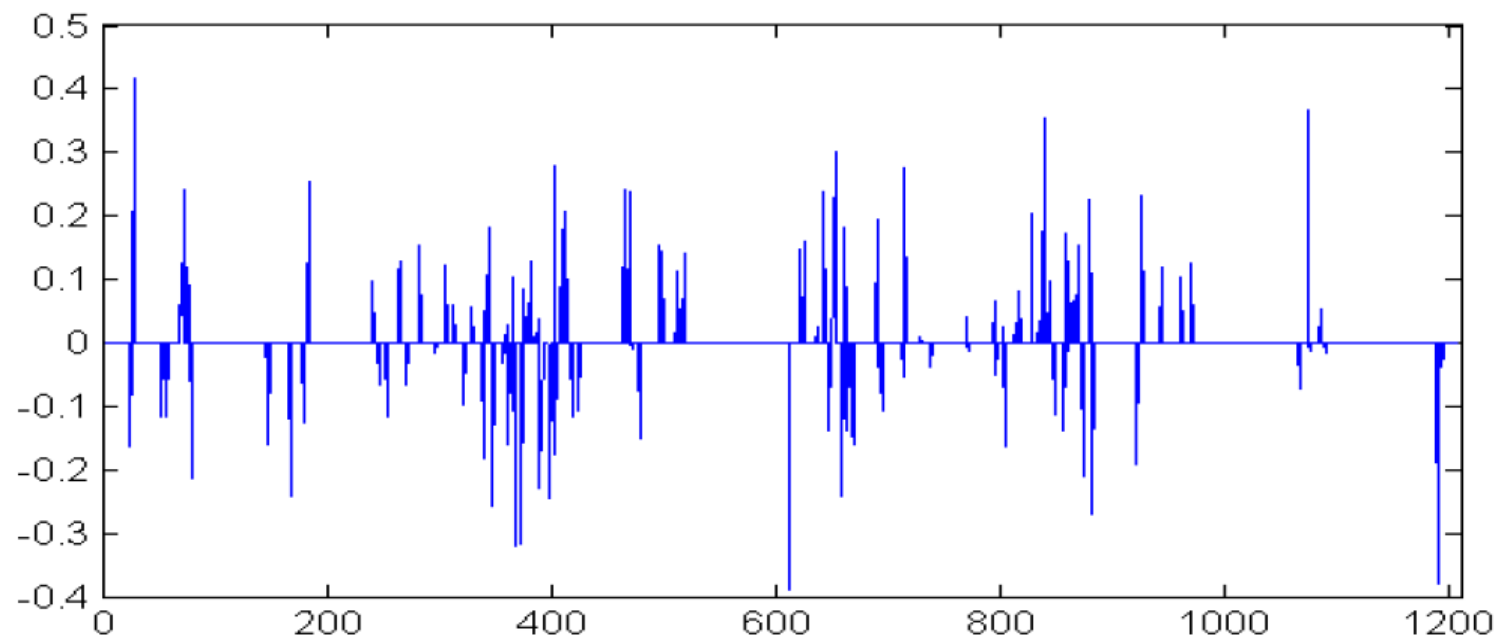
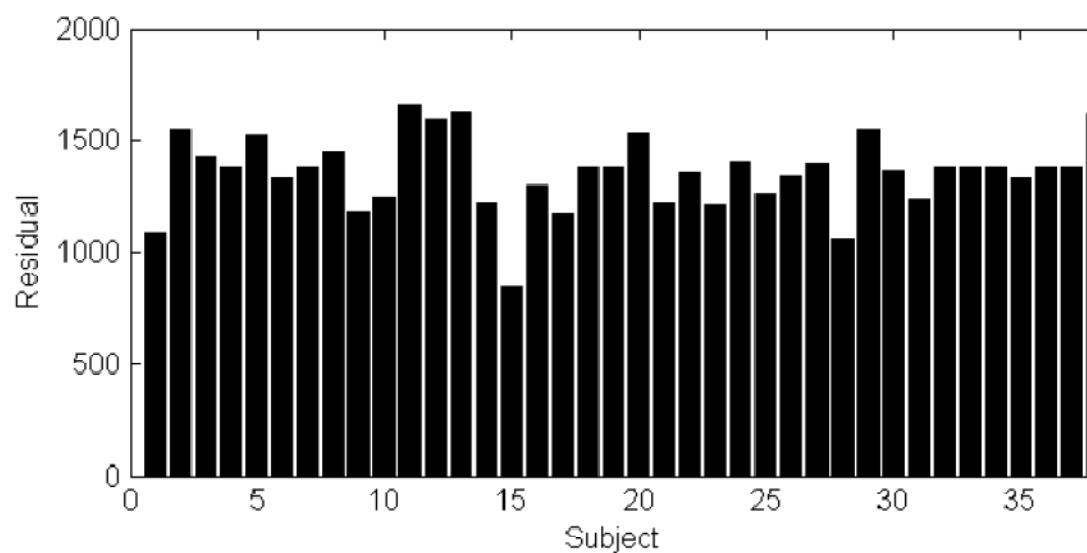


Fig. 4



D. Validation Based on Sparse Representation

- Fact
 - It is important for recognition systems to detect and **reject** invalid test samples (**outliers**).
- Reason
 - The input test image could be even of a subject which is **not a face at all**.
- Conventional classifiers' methods (e.g. NN or NS)
 - Use the residuals $r_i(y)$ for validation
 - Accepts or rejects a test sample based on how small the smallest residual is
- Problem
 - Each residual $r_i(y)$ only measures **similarity** between the **test sample** and **each** individual **class**. (Does not concern other classes)



Validation methods in this work

- In the sparse representation paradigm, the coefficients \hat{x}_1 are computed **globally**, in terms of images of all classes. In a sense, it can harness the joint distribution of all classes for validation.
- → the coefficients \hat{x} are better statistics for validation than the residuals. (See Example 2)



Example 2

(Concentration of Sparse Coefficients)

- Randomly select an irrelevant image from Google, and down-sample it to 12x10.
- Compute the sparse representation of the image against the same training data as in Example 1. (See Fig. 5)
- The distribution of the estimated sparse coefficients \hat{x} contains important information about the validity of the test image:
 - A **valid** test image should have a sparse representation whose nonzero entries concentrate mostly on **one** subject, whereas an **invalid** image has sparse coefficients spread widely among **multiple** subjects.



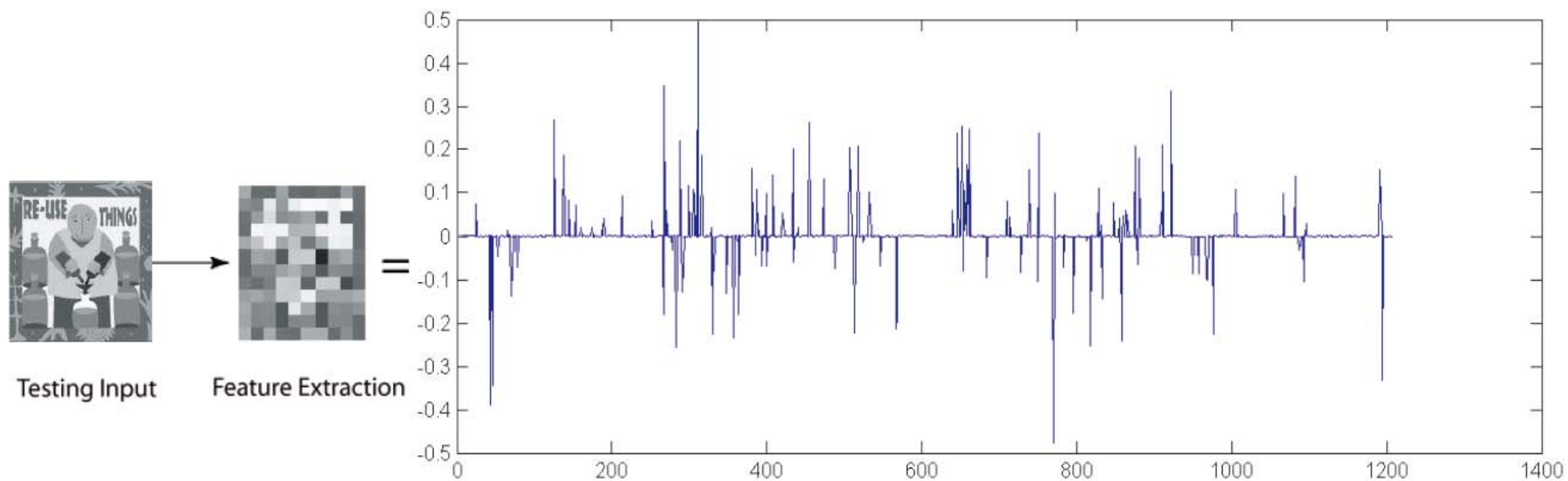
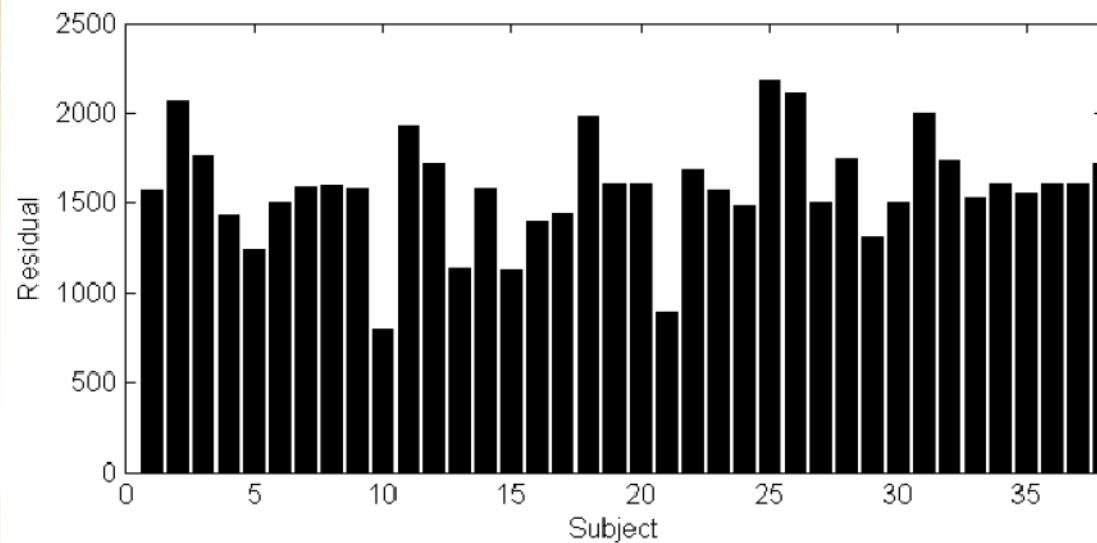


Fig. 5



Definition 1

(Sparsity Concentration Index): a quantitative measure

$$SCI(x) = \frac{k \cdot \max_i \|\delta_i(x)\|_1 / \|x\|_1 - 1}{k - 1} \in [0, 1] \quad (10)$$

- For a solution \hat{x} found by Algorithm 1, if $SCI(\hat{x}) = 1$, the test image is represented using only images from a **single** object, and if $SCI(\hat{x}) = 0$, the sparse coefficients are spread evenly over **all** classes.
- We choose a threshold $\tau \in (0, 1)$ and accept a test image as valid if $SCI(\hat{x}) \geq \tau$, and otherwise reject as invalid.
- In step 5 of Algorithm 1, one may choose to output the identity of y only if it passes this criterion.



Outline

- Introduction
- Classification Based on Sparse Representation
- Two Fundamental Issues in Face Recognition
- Experimental Results
- Conclusion & Future works



A. The Role of Feature Extraction

- Traditional role
 - Crucial
 - Reason
 - The **choice of feature** transformation affects the success of the classification algorithm.
 - Result
 - Researchers developed a wide variety of increasingly complex feature extraction methods (including nonlinear and kernel features)
- The role in this work
 - To reduce data dimension and
 - To reduce computational cost



Methods to apply feature extraction

- Since most feature transformations involve only **linear operations** (or approximately so), the projection from the **image space** to the **feature space** can be represented as a matrix $R \in \mathbb{R}^{d \times m}$, with $d \ll m$

- Applying R to both sides of equation (3) yields:

$$\tilde{y} = Ry = RAx_0 \in \mathbb{R}^d \quad (11)$$

- In practice, the dimension d of the feature space is typically chosen to be much smaller than n . In this case, the system of equations $\tilde{y} = RAx \in \mathbb{R}^d$ is **underdetermined** in the unknown $x \in \mathbb{R}^n$.



Methods to apply feature extraction (Con'd.)

- Nevertheless, as the desired solution x_0 is sparse, we can hope to recover it by solving the following reduced ℓ^1 -minimization problem:

$$(\ell_r^1) : \hat{x}_1 = \arg \min \|x\|_1 \text{ subject to } \|RAx - \tilde{y}\|_2 \leq \varepsilon \quad (12)$$

- for a given error tolerance $\varepsilon > 0$.
- Thus, in Algorithm 1, the matrix A of training images is now replaced by the matrix $RA \in \mathbb{R}^{d \times n}$ of d -dimensional features; the test image y is replaced by its features \tilde{y} .



A surprising phenomenon: (the blessing of dimensionality)

- If the solution x_0 is **sparse enough**, then with overwhelming probability, it can be correctly recovered via ℓ^1 -minimization from any sufficiently large number d of linear measurements $\tilde{y} = RAx_0$. More precisely, if x_0 has $t \ll n$ nonzeros, then with overwhelming probability,

$$d \geq 2t \log(n/d) \quad (13)$$

random linear measurements are sufficient for ℓ^1 -minimization (12) to recover the correct sparse solution x_0

- Random features can be viewed as a less-structured counterpart to classical face features, such as Eigenfaces or Fisherfaces.



Definition 2 (Randomfaces)

- Consider a transform matrix $R \in \mathbb{R}^{d \times m}$ whose entries are independently sampled from a **zero-mean normal distribution** and each row is normalized to unit length. The row vectors of R can be viewed as d random faces in \mathbb{R}^m .
- Randomfaces are extremely efficient to generate, as the transformation R is independent of the training dataset.



Remark

- As long as the correct sparse solution x_0 can be recovered, Algorithm 1 will always give **the same classification result, regardless of the feature** actually used.
- Thus, when the dimension of feature d exceeds the above bound (13), one should expect that the recognition performance of Algorithm 1 with different features **quickly converges**, and the choice of an “optimal” feature transformation is no longer critical:
 - Even random projections or downsampled images should perform as well as any other carefully engineered features.



B. Robustness to Occlusion or Corruption

- By ECC theory, **redundancy** is essential to detecting and correcting errors
 - It is possible to correctly recognize face image
 - the **number of image pixels** is typically far **greater** than the **number of subjects** that have generated the images (redundant!)
 - We should work with the highest possible resolution
 - feature extraction discard useful information that could help compensate for the occlusion
 - The most redundant, robust, or informative is the **original** images.



Previous methods

- Motivation
 - It is difficult to extraction the information encoded in the redundant data
- Methods
 - Focus on **spatial locality**
 - Computed **Local features** from only a small fraction of the image pixels that are clearly less likely to be corrupted by occlusion than **holistic features**.
 - Examples: ICA, LNMF, Local Binary Patterns, Gabor wavelets, etc.
- Problem
 - no bases or features are more spatially localized than the **original** image pixels themselves.
 - The role of feature extraction in achieving spatial locality is **questionable**



In this work

- We modified the above linear model (3) as

$$y = y_o + e_o = Ax_o + e_o \quad (14)$$

- where $e_o \in \mathbb{R}^m$ is a vector of errors – a fraction, ρ , of its entries are nonzero. The nonzero entries of e_o model which pixels in y are corrupted or occluded.
- Assume that the corrupted pixels are a relatively small portion of the image. The error vector e_o , like the vector x_o , then has sparse nonzero entries. Since $y_o = Ax_o$, we can rewrite (14) as

$$y = [A, I] \begin{bmatrix} x_o \\ e_o \end{bmatrix} = Bw_o \quad (15)$$

- Here, $B = [A, I] \in \mathbb{R}^{m \times (n+m)}$, so the system $y = Bw$ is **always underdetermined** and does not have a unique solution for w .



The rate of occlusion

- From the above discussion about the sparsity of x_0 and e_0 , the correct generating $w_0 = [x_0, e_0]$ has at most $n_i + \rho m$ nonzeros.
- We might therefore hope to recover w_0 as the sparsest solution to the system $y = Bw$.
- In fact, if the matrix B is in general position, then as long as $y = B\bar{w}$ for some \bar{w} with less than $m/2$ nonzeros, \bar{w} is the unique sparsest solution.
- Thus, if the occlusion e covers less than $\frac{m - n_i}{2}$ pixels, $\approx 50\%$ of the image, the sparsest solution to $y = Bw$ is the true generator, $w_0 = [x_0, e_0]$.

$$\frac{m}{2} - n_i$$



- More generally, one can assume that the corrupting error e_0 has a sparse representation with respect to some basis $A_e \in \mathbb{R}^{m \times n_e}$.
- That is, $e_0 = A_e u_0$ for some sparse vector $u_0 \in \mathbb{R}^m$.
- Here, we have chosen the special case $A_e = I \in \mathbb{R}^{m \times m}$ as e_0 is assumed to be sparse with respect to the natural pixel coordinates.
- If the error e_0 is instead more sparse w.r.t. another basis, e.g., Fourier or Haar, we can simply redefine the matrix B by appending A_e (instead of the identity I) to A and instead seek the sparsest solution w_0 to the equation:

$$y = Bw \text{ with } B = [A, A_e] \in \mathbb{R}^{m \times (n+n_e)} \quad (16)$$



- In this way, the same formulation can handle more general classes of (sparse) corruption. That is

$$(l_e^1) : \hat{w}_1 = \arg \min \|w\|_1 \text{ subject to } Bw = y \quad (17)$$

- That is, in Algorithm 1, we replace the image matrix A with the extended matrix $B = [A, I]$ and x with $w = [x, e]$.



- If y is an image of subject i , the ℓ^l -minimization (17) cannot guarantee to correctly recover $w_0 = [x_0, e_0]$ if

$$n_i + |\text{support}(e_0)| > d/3 \quad (18)$$

- Generally, $d \gg n_i$, so (18) implies that the largest fraction of occlusion under which we can hope to still achieve perfect reconstruction is 33%.



- Once the sparse solution $\hat{w}_1 = [\hat{x}_1, \hat{e}_1]$ is computed, setting $y_r = y - \hat{e}_1$ recovers a clean image of the subject with occlusion or corruption compensated for.
- To identify the subject, we slightly modify the residual $r_i(y)$ in Algorithm 1, computing it against the recovered image y_r :

$$r_i(y) = \|y_r - A\delta_i(\hat{x}_1)\|_2 = \|y - \hat{e}_1 - A\delta_i(\hat{x}_1)\|_2 \quad (19)$$



Outline

- Introduction
- Classification Based on Sparse Representation
- Two Fundamental Issues in Face Recognition
- **Experimental Results**
- Conclusion & Future works

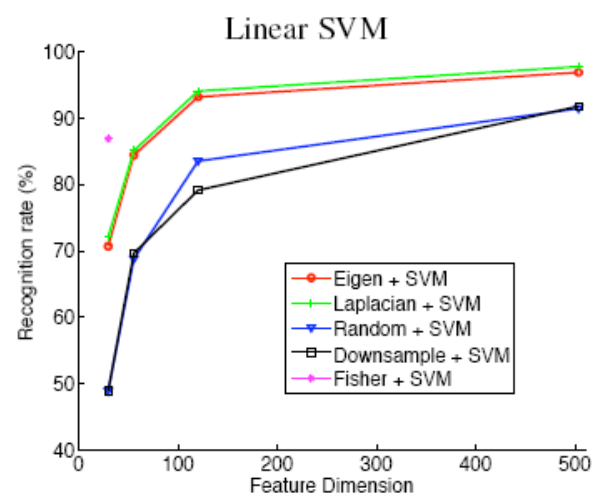
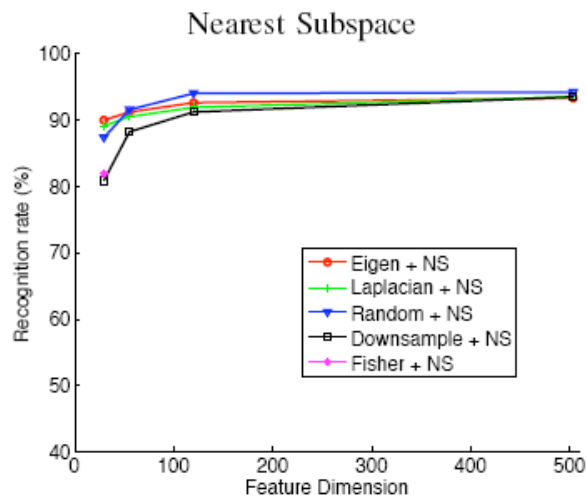
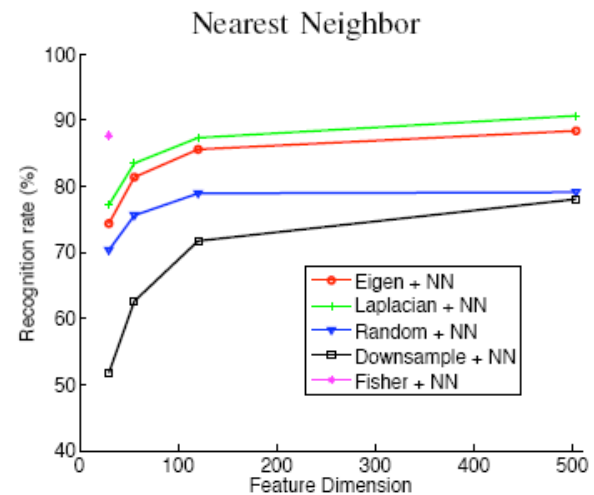
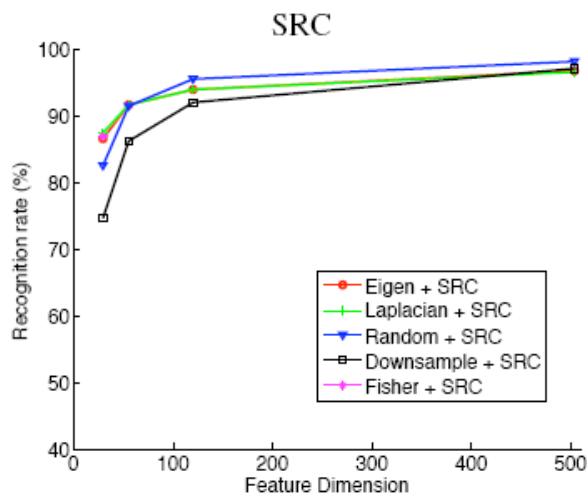


Databases

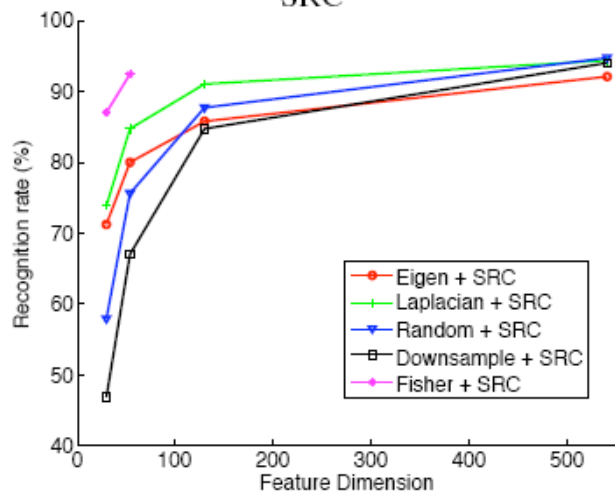
- Extend Yale B
 - 2,414 frontal-face images of 38 individuals
 - Images are cropped to 192x168 pixels
 - Captured under various laboratory controlled lighting conditions
- AR Face Data Base
 - Over 4,000 frontal-face images of 126 individuals
 - Including illumination change, different expressions and facial disguise
 - Cropped to 165x120 pixels



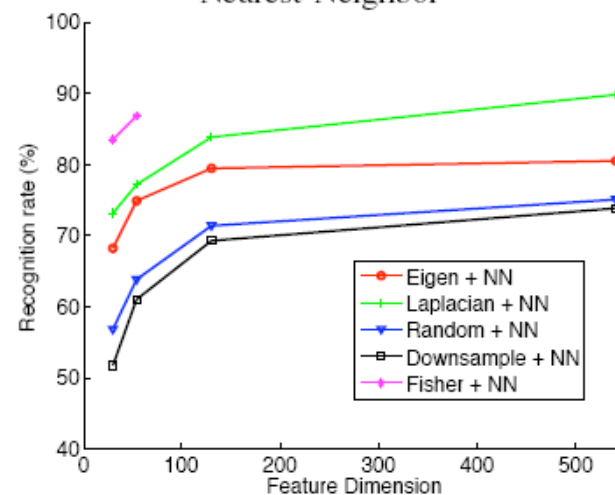
1. Feature extraction and classification methods



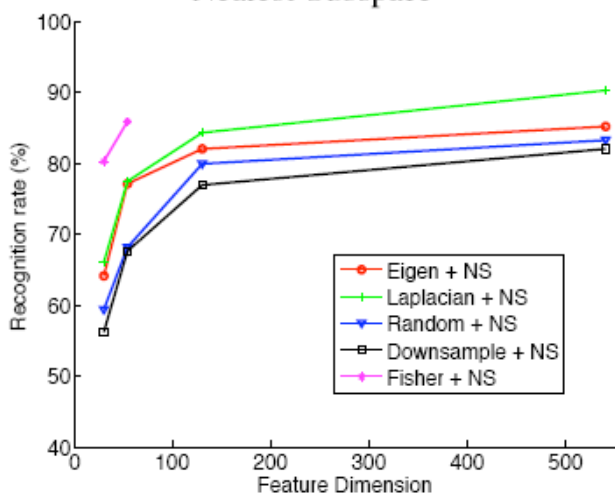
SRC



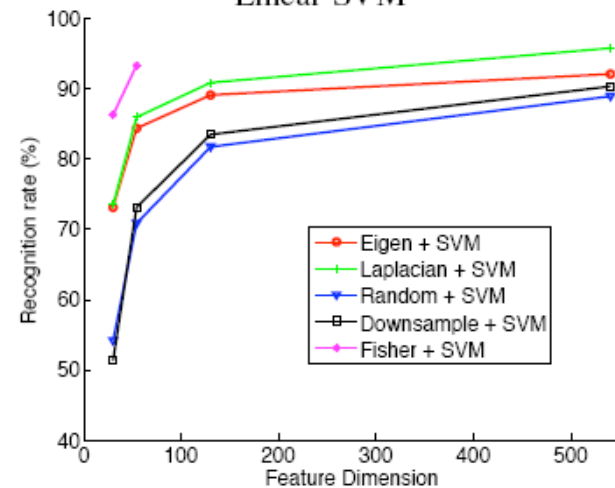
Nearest Neighbor



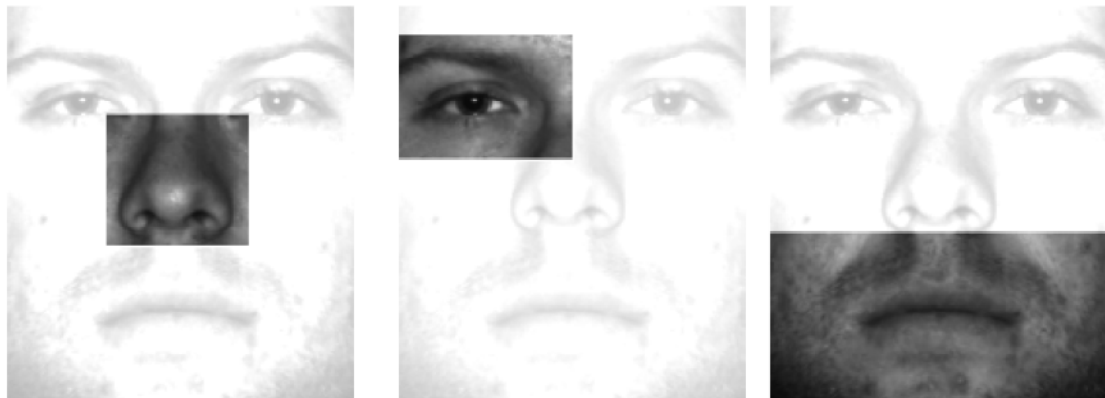
Nearest Subspace



Linear SVM



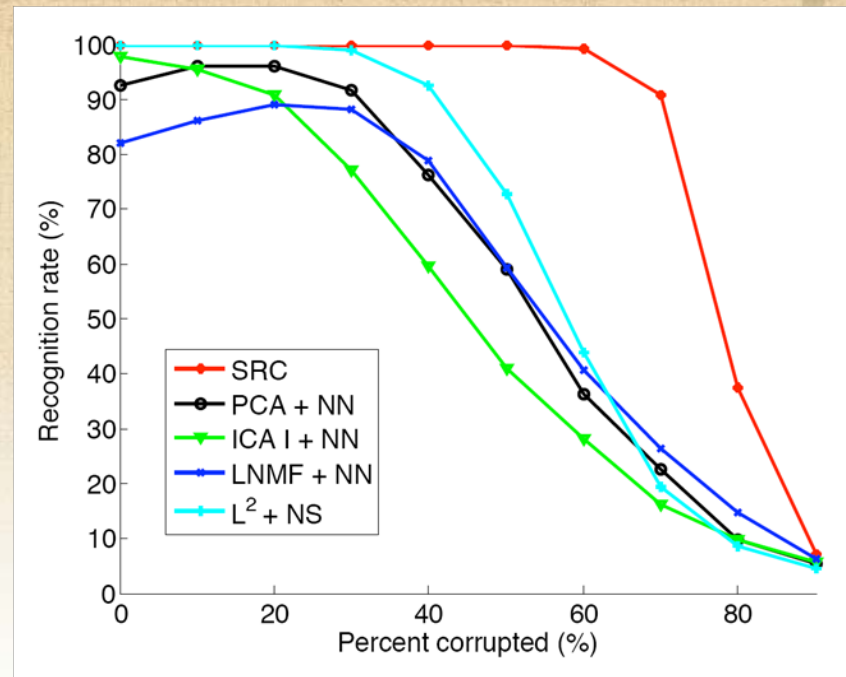
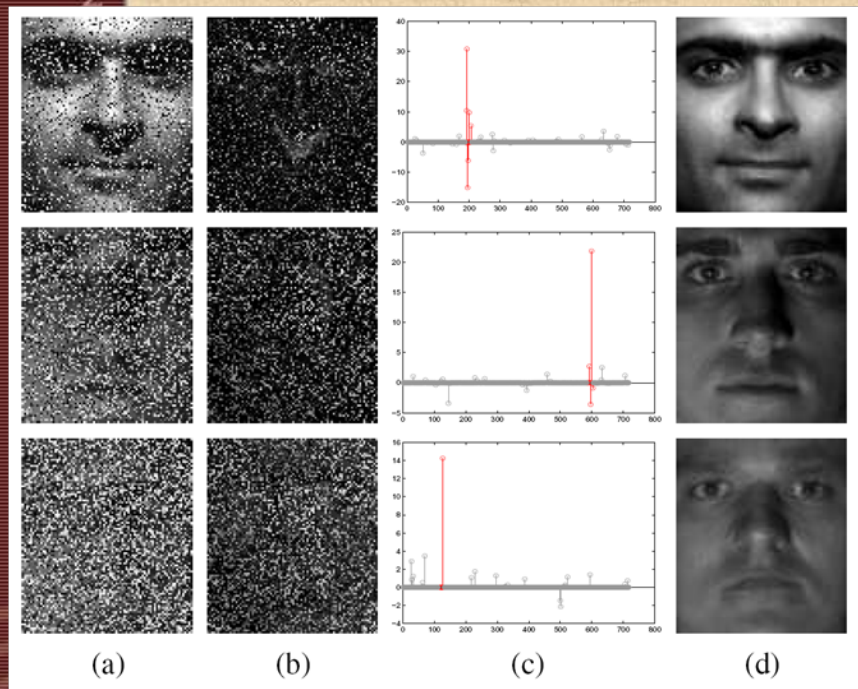
2. Recognition with partial feature



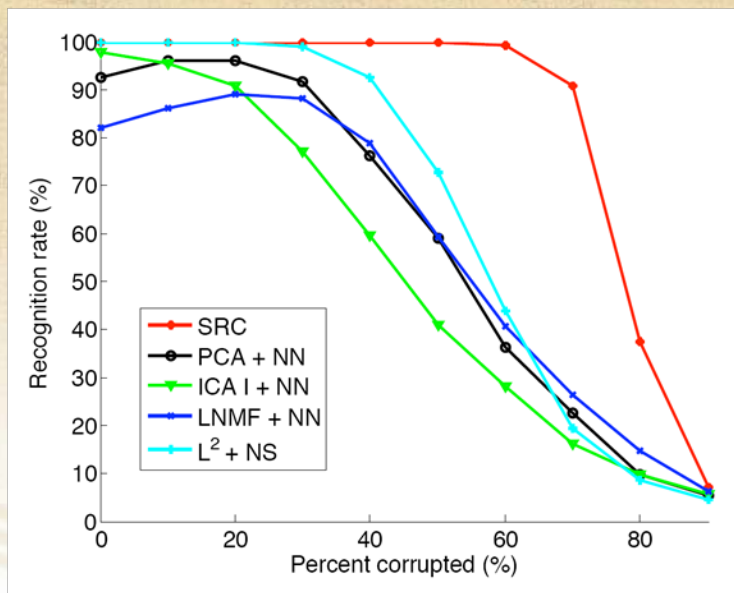
Features	Nose	Right Eye	Mouth & Chin
Dimension (d)	4,270	5,040	12,936
SRC	87.3%	93.7%	98.3%
NN	49.2%	68.8%	72.7%
NS	83.7%	78.6%	94.4%
SVM	70.8%	85.8%	95.3%

Top: example features. Bottom: Recognition rates of SRC, NN, NS, and SVM on the Extended Yale B database.

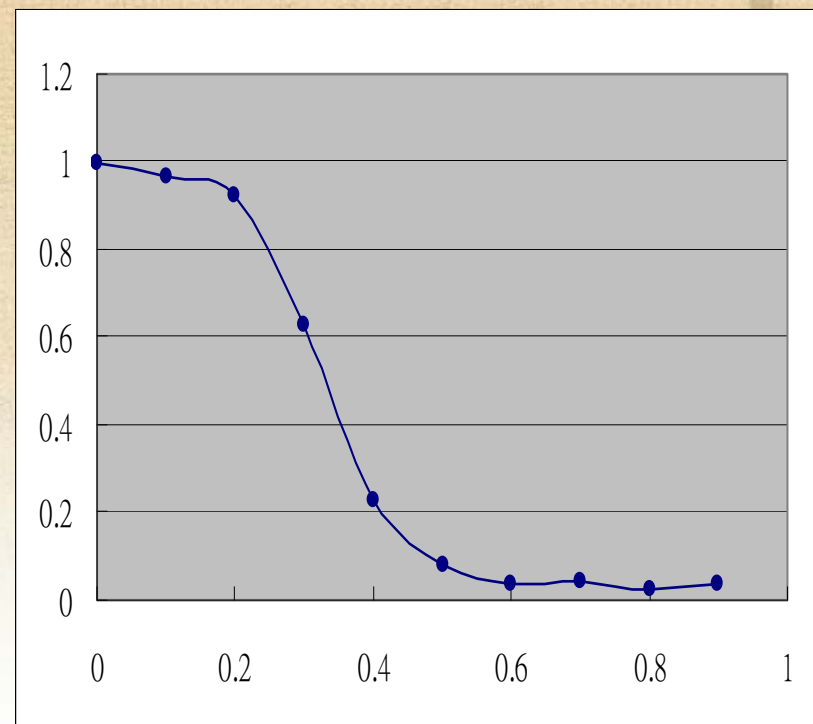
3. Recognition under random corruption



Percent corrupted	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%
Recognition rate	100%	100%	100%	100%	100%	100%	99.3%	90.7%	37.5%	7.1%



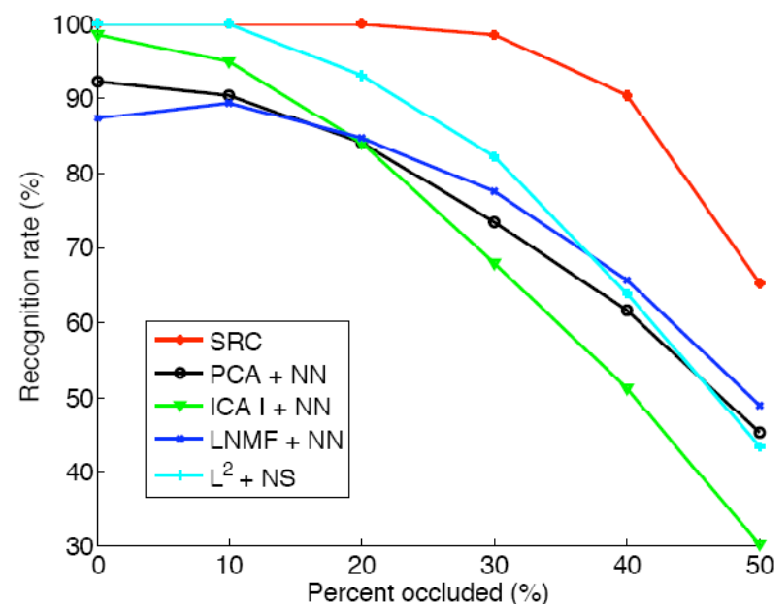
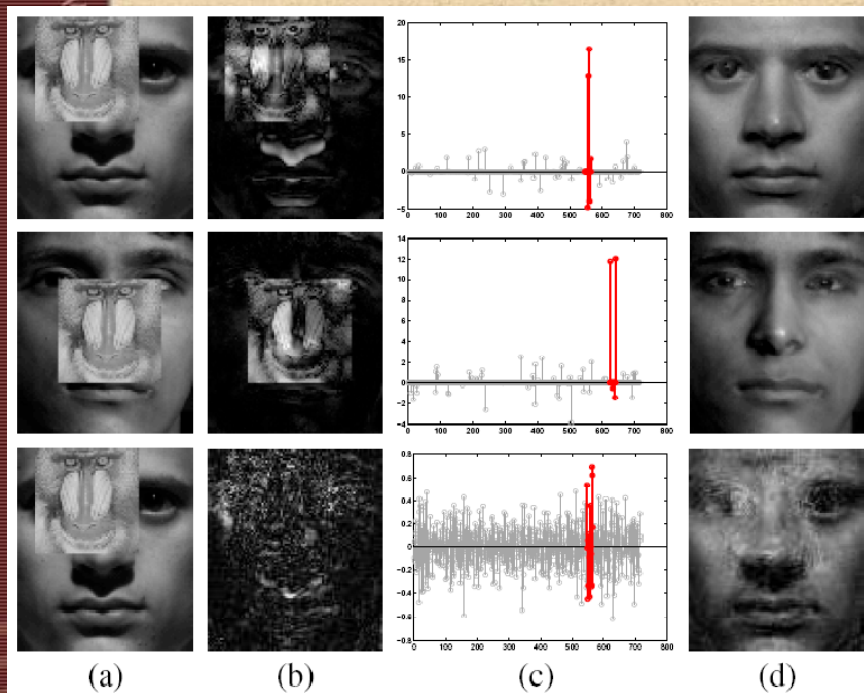
96x84 pixels



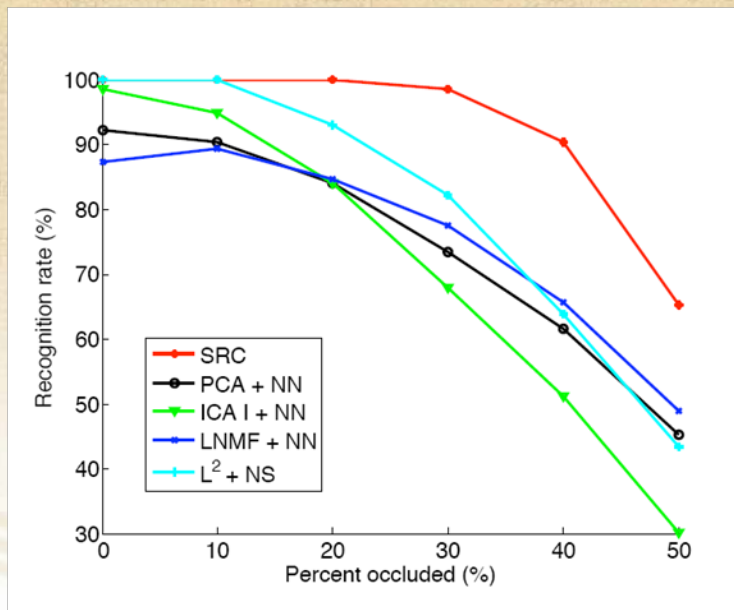
12x10 pixels



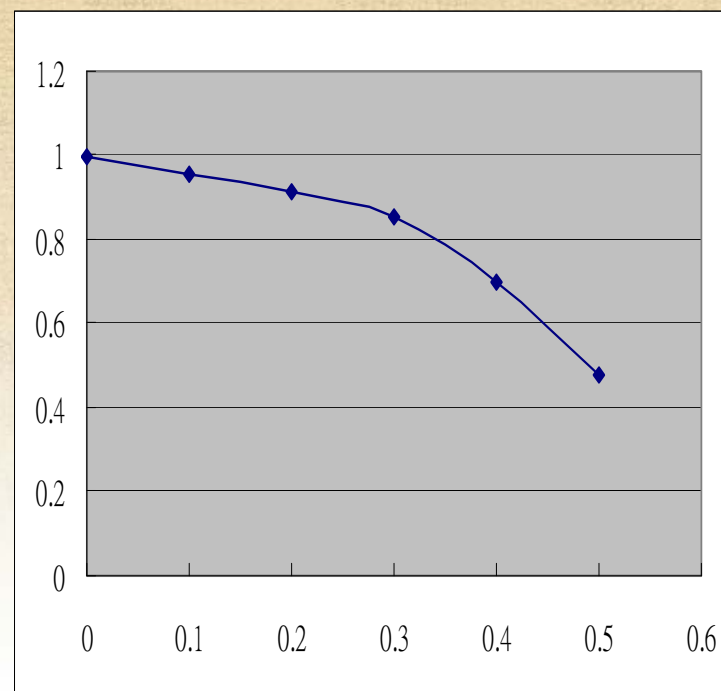
4. Recognition under varying level of contiguous occlusion



Percent occluded	0%	10%	20%	30%	40%	50%
Recognition rate	100%	100%	99.8%	98.5%	90.3%	65.3%



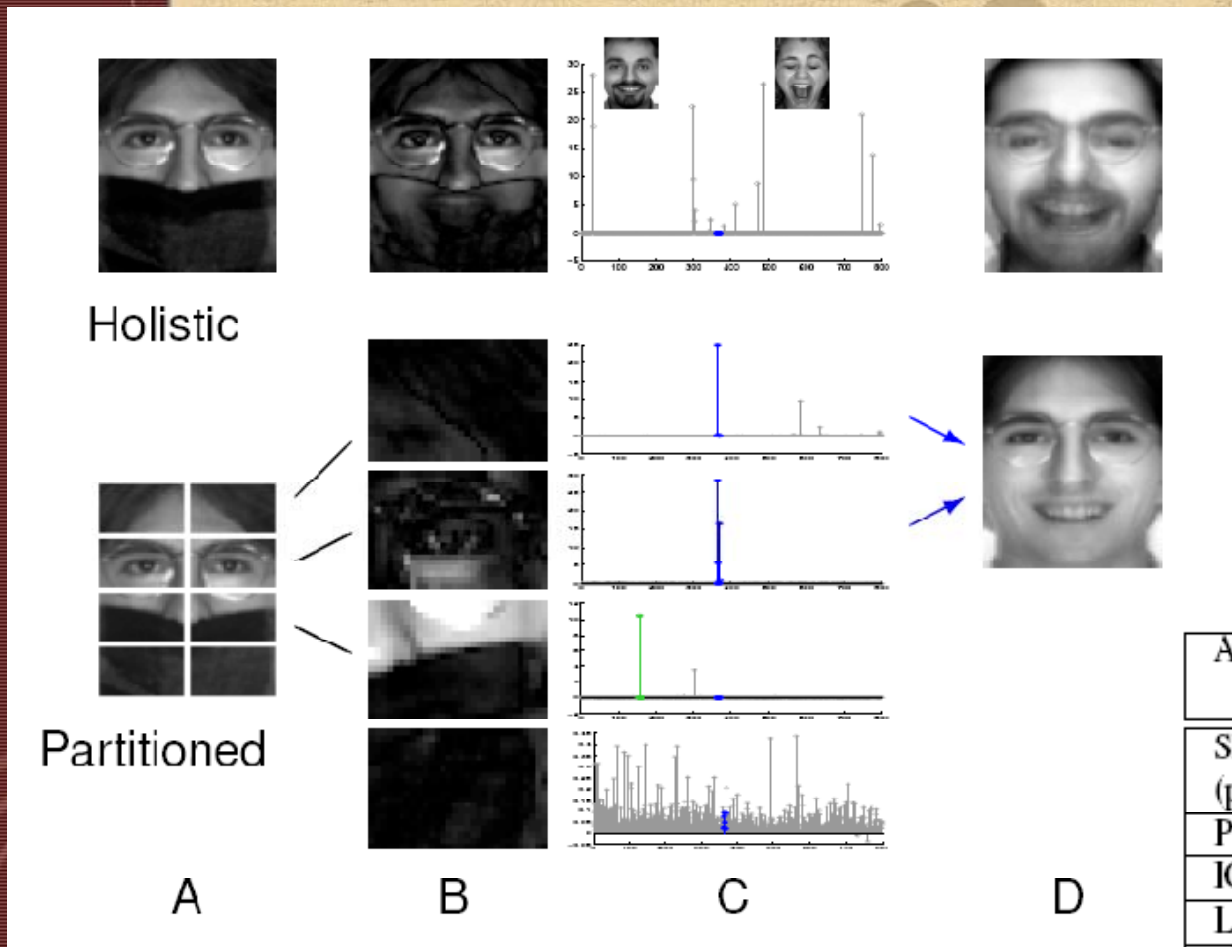
96x84 pixels



12x10 pixels



5. Partition scheme to tackle contiguous disguise



Algorithms	Rec. rate sunglasses	Rec. rate scarves
SRC (partitioned)	87.0% (97.5%)	59.5% (93.5%)
PCA + NN	70.0%	12.0%
ICA I + NN	53.5%	15.0%
LNMF + NN	33.5%	24.0%
ℓ^2 + NS	64.5%	12.5%

Remark: The use of holistic versus local features in Face Recognition

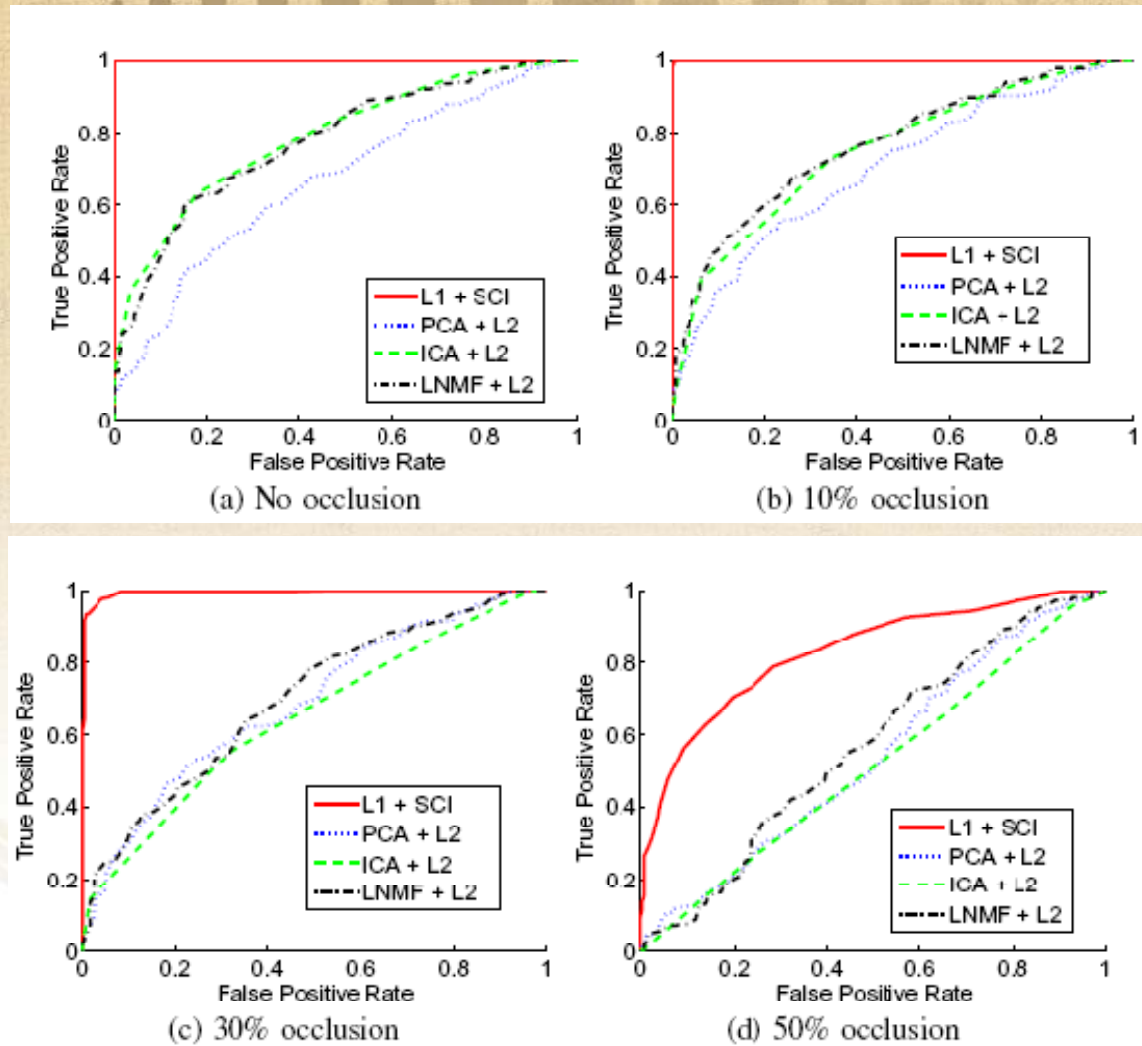
- The problem is not the choice of representing the test image in terms of a holistic or local basis, but rather how the representation is computed.
- Properly **harnessing redundancy** and **sparsity** is the key to error correction and robustness.
- Extracting local or disjoint features can only reduce redundancy, resulting in inferior robustness.



W. Zhao, et al., "Face recognition: A literature survey," ACM Computing Surveys, pp. 399–458, 2003.



6. Rejecting invalid test images



7. Robust training set design

- An important consideration in designing recognition systems is selecting the number of training images as well as the conditions (lighting, expression, viewpoint etc.) under which they are to be taken.
- The training images should be extensive enough to span the conditions that might occur in the test set: they should be “sufficient” from a pattern recognition standpoint.



K. Lee, et al., “Acquiring linear subspaces for face recognition under variable lighting,” IEEE T-PAMI, vol. 27, no. 5, pp. 684–698, 2005. 62



- This paper provides a different, quantitative measure for how “robust” the training set is:
 - the amount of worst-case occlusion the algorithm can tolerate is directly determined by how neighborly the associated polytope is.

“Polytope”: please refer to Prof. David L. Donoho’s website.





Subset 1



Subset 2



Subset 3



Subset 4

Training set	1	2	3	4
Neighborliness	1,124	1,122	1,190	1,330



Neutral (N)



Happy (H)



Angry (A)



Screaming (S)

Training set	N+H	N+A	N+S	H+A	H+S	A+S
Neighborliness	585	421	545	490	550	510



- Training sets with **wider variation** (such as lighting and expression) in the images allow greater robustness to occlusion.
- However, the training set should not contain too many similar images.
- In the language of signal representation, the training images should form an “Incoherent Dictionary.”

D. Donoho, “For most large underdetermined systems of linear equations the minimal ℓ_1 -norm solution is also the sparsest solution,” *Communication on Pure and Applied Math*, vol. 59, no. 6, pp. 797–829, 2006.



Outline

- Introduction
- Classification Based on Sparse Representation
- Two Fundamental Issues in Face Recognition
- Experimental Results
- Conclusion & Future works



Conclusion & Future works

- Conclusion
 - We proposed a high performance classification of high-dimensional data (face images) via sparsity.
 - The choice of features become less important than numbers of features dim.
 - Occlusion and corruption can be handles uniformly and robustly in this work
- Future works
 - Extend to object detection
 - Variations in object pose



Thank You!

