Yongxin Taylor Xi, Peter J. Ramadge
yxi@princeton.edu
Pixl Lab, Department of Electrical Engineering, Princeton University, NJ 08544

Abstract

Programming a computer to learn to recognize a human face has been a challenging problem for decades. Critical to current appearance-based solutions is dimensionality reduction—using training examples to identify effective projections of salient features that distinguish individuals, e.g., leading PCA, and FDA eigenvectors. The faces of each individual form a manifold in a very high dimensional space. Our current research is to explore how new ideas form compressive sensing and manifold learning can help identify the salient features of this manifold. Least angle regression algorithm is applied to learn the effective projections and enhance the recognition performances for all of the three databases that we experimented.

Face recognition Problem definition

Given limited amount of training images of K individuals, our objective is to construct efficient computer algorithms and apply them to the training database, such that they can determine the identity of a new face image accurately.

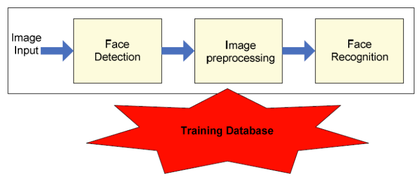


Fig.1 Flowchart of face recognition system

Necessity of dimensionality reduction

Dimensionality reduction is usually a must for a learning system with very high dimensional data. It can not only overcome the overfitting problem thus increase the learning performance, but also release the burden of computation.

Open challenges

The main challenge of face recognition is that the variance of the face images within one person might be larger than the variance across different people, for example: facial expression change, illumination change, aging of the individual, rotation of the head, etc.

So we hope to find ways to learn projections which capture a certain type of variance, e.g., the variance among different individuals, so that recognition will be easier because other variances will not come into play.

The learning problem

Training database:

- Suppose there are K individuals in the database;
- We have ni images for the ith individual to learn from;
- Let n=n1+n2+…+nk be the total number of images.

Preprocessing of the images:

- Proper alignment (fix two eyes for all images)
- Cropping into P1xP2 matrices
- Normalization
- Vectorization (transform into vector in $\mathbb{R}^P, P = P1 \times P2$)

Data matrix X:

$$X^T = \begin{pmatrix} {x_1}^T \\ \vdots \\ {x_n}^T \end{pmatrix}, x_i \in \mathbf{R^p}, \text{p around } 2000$$

Label matrix Y: $y_i = \begin{pmatrix} 0 \dots 01_k 0 \dots 0 \end{pmatrix}$, if $x_i \in$ Class k

Solve for: matrix B, $B \in \mathbf{R}^{p \times K}$ such that it is a good approximation to Y, and more importantly, given new image vector x, $X^T B$ had better give a good prediction of its label. Note that the problem is decomposable into K sub-problems.

Recognition protocol

Assign class j to new x, if: $j = argmin_i \|x^T B - y_i\|, i \in \{1, \dots, K\}$

Regularization

For n<p, the problem is underdetermined, i.e., there exist lots of solutions. Regularization is therefore needed for good prediction.

$$\min_{b_i} \|X^T b_i - y^i\|^2 + \lambda \|b_i\|_a$$
$$a = 1, 2$$
$$\lambda > 0, i = 1, \dots, K$$

L2 norm (a=2): standard regularized least squares (RLS)
Easy to solve; But solutions not sparse.
L1 norm (a=1): Lasso
More difficult to solve (but still convex problem)
Sparse solutions

L1 norm is preferred because sparse solution B can help explain the scientific insight of the X-Y relationship.

Model-building algorithms

Instead of solving an optimization problem, several linear model-building algorithms directly solve for sparse vector b given (X, y) and budget m (# of non-zero terms in b), i.e., they look for b which approximate X well to y, using only m coefficients in b. We are most interested in the Lars algorithm, because of its efficiency and good prediction accuracy. An additional merit of Lars is that it can be used to produce Lasso solutions.

- Forward selection algorithm
- Stagewise selection algorithm
- Least angle regression algorithm (Lars)

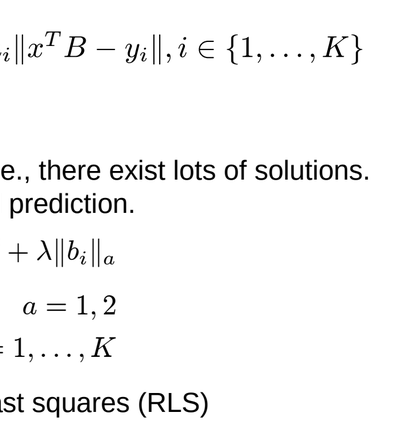


Fig.2 Performance comparision among three algorithms

For these model building algorithms,

$$X^T = \begin{pmatrix} {x_1}^T \\ \vdots \\ {x_n}^T \end{pmatrix} = \begin{pmatrix} x^1 \dots x^p \end{pmatrix}, x^i \in \mathbf{R}^n$$

$$y \in \{y^1, \dots, y^K\}, y^i \in \mathbf{R}^n$$

x^is are called regressors/predictors
y is called the response vector
$X^T b$ is a linear combination of the regressors

Preprocessing of the model-building algorithms

- Subtract mean of data X (so that regressors centered around origin)
- Normalize the length of each regressor

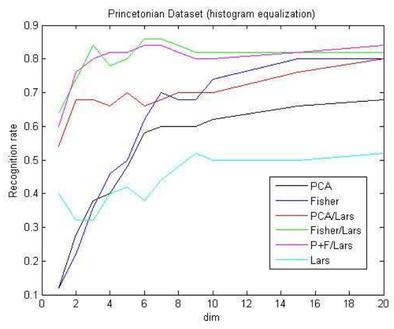


Fig. 5 Recognition rate of methods, Princetonian Database

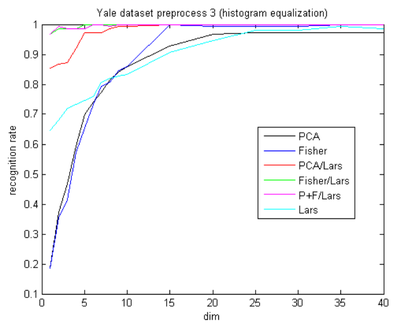


Fig. 6 Recognition rate of methods, Yale Face Database B

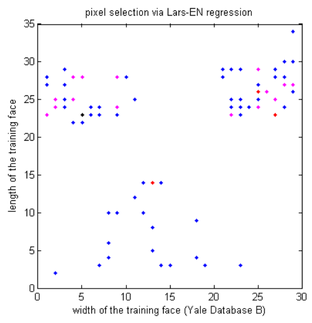


Fig. 7 the pixels of a face image selected by Lars algorithm. Red stands for higher frequency (being selected 4 or 5 times).

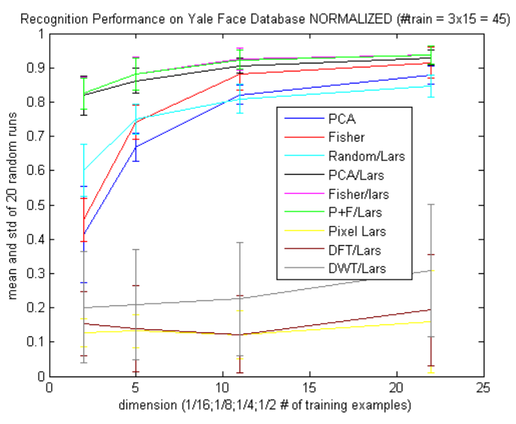


Fig. 8 Recognition rate of methods, Yale Face Database
Training images/person=3.

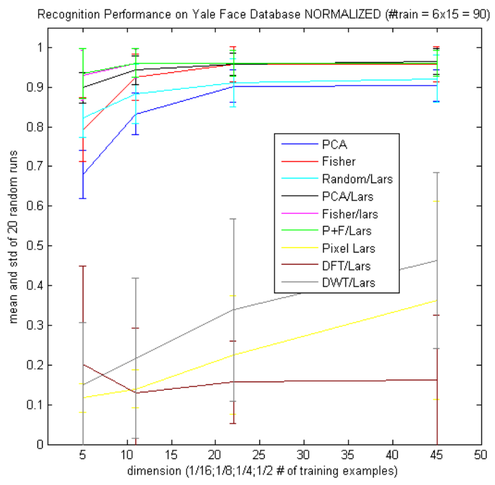


Fig. 9 Recognition rate of methods, Yale Face Database
Training images/person=6.

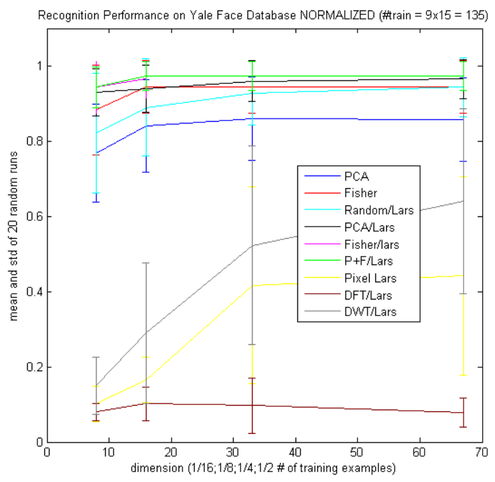
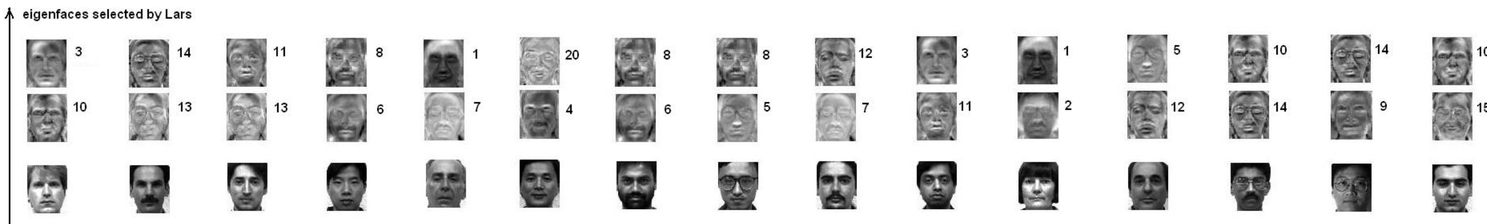


Fig. 10 Recognition rate of methods, Yale Face Database
Training images/person=9.



Lars algorithm

Given all the regressors x^i s and y;
Let: b=0
Current estimate of response $\hat{y} = X^T b = 0$
Current residual $y - \hat{y} = y$
Current active regressor group Xa=[ ];
For i=1:m

1) Add a new regressor into Xa (now of size i)
2) Compute the equiangular direction of Xa
3) Proceed until some regressor out of Xa has the same correlation with the residual
4) Update b according to Xa and step length

Lasso/Lars Relationship

With a small modification, Lars can produce sparse Lasso solutions.
Lasso constraint:
sign of any non-zero coordinate bj must agree with the sign sj of the current correlation.
Lasso modification:
If within a step, bj will change sign, then stop at bj=0, and remove xj from the active regressor group: A=A-{j}

Experiments and results

♣ **Three databases**

1) Princetonian Database (hard):

- 10 people, 5 images/person, collected from website.
- Leave-one-out

Fig. 3 Sample pictures from Princetonian Database

2) Yale Face Database B (easy):

- 10 people under 4 x 6 =24 viewing conditions
- Same expression; no eyeglass change
- Each person: 8 images for training, 16 images for testing

3) Yale Face Database (median):

- 15 people; 11 images/person: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink.

Fig. 4 Illustration of the 11 images for a person, Yale Face Database

- Conduct 20 times:
Randomly select 3/6/9 training images of each person from the database. Learn then test on the rest images.

♣ **Experiment the following face recognition methods:**

PCA, FDA, Lars directly in Pixel domain, PCA/Lars (PCA followed by Lars), Fisher/Lars (FDA followed by Lars), P+F/Lars (Lars selecting projections from both PCA and FDA), Random/Lars (Lars learning projections from random projections)

Then compare these methods under the same allowed dimension.
Results are shown in Fig. through Fig.

Conclusion and future research

For the three face databases, Lars algorithms with dimensionality reduction beat the well-known PCA and Fisher methods. Surprisingly, even Random/Lars would beat PCA and Fisher in some cases. This means that from all projection candidates (e.g., PCA eigenvectors), Lars automatically selects the projections which can best tell different individuals apart, so that salient features of individuals can be learned. In other experiments such as mood detection (happy/sad), light direction estimation (left/right), and eyeglasses detection (with/without eyeglasses), Lars has also demonstrated its potential in learning effective projections for specific tasks.

Current algorithms works for an accessible set of projections, i.e., Lars will learn projections from finite number of projection candidates produced by PCA, FDA, or random projections. However, we hope that the projection candidate space be infinite (but then it will be impractical problem) so that Lars can have a greater freedom to learn. Future research, therefore, focuses on solving the dimensionality reduction and learning projections at the same time, so that the optimizer of the optimization problem would yield the “best” projections.