# Face Recognition via Sparse Representation

Wabartha - Durand - Darmet - Vayer

March 28, 2017

# Overview

# Plan

# Article's origins

- University of Illinois
- PHD work of John Wright. Work on Face and object recognition. Student Member of IEEE. 16405 citations since 2012. Now work for Columbia University
- Published in : IEEE transactions on pattern analysis and machine intelligence (TPMAI) in 2009 (a reference specialised in computer vision and patter analysis/recognition) -
- Article cited by 6386 since 2009 -
- IEEE stand for Institute of Electrical and Electronics Engineers : almost a monopoly in the area.

# Contributions

The main contribution of the article is concerning the sparse approach for image processing which leads to :

- features extraction (eigenfaces,FisherFaces,...) is not crucial anymore
- occlusion, corruption and noises robust
- fast (light computation)
- Sparsity Concentration Index (efficently detect invalid images)

# Plan

# Face recognition

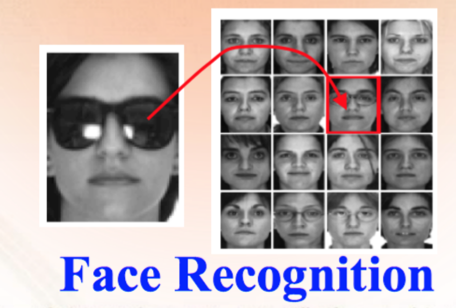



Figure: Face recognition

# Face recognition : our example

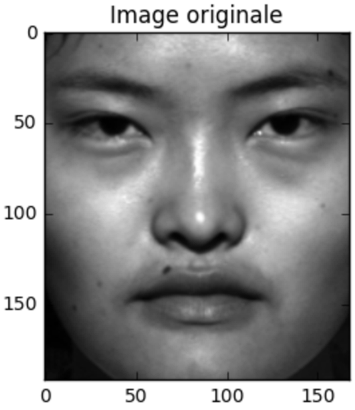



Figure: Face recognition

# Symbols

- Let's consider $k$ distinct object classes in the training data (k faces)
- An image of size $w \times h$ is indentified as a vector $v \in \mathbb{R}^m$ where $m = wh$ given by stacking it columns
- The $n_i$ given training samples, taken from the i-th class (face) are arranged as columns of a matrix $A_i = [v_{i,1}, v_{i,2}, ..., v_{i,n_i}] \in \mathbb{R}^{m \times n_i}$
- So the columns of $A_i$ are the training face images of the i-th subject

# Sparse Linear Combination

Given some training samples of the i-th object class, $A_i = [v_{i,1}, v_{i,2}, ..., v_{i,n_i}] \in \mathbb{R}^{m \times n_i}$, any new test sample $y \in \mathbb{R}^m$ **from the same class** will be described as as linear combination of the training samples associated with object i :

$$y = \alpha_{i,1} v_{i,1} + \alpha_{i,2} v_{i,2} + ... + \alpha_{i,n_i} v_{i,n_i}$$

for some scalars $\alpha_{i,j} \in \mathbb{R}$

# Sparse Linear Combination

Since the membership i of a test sample is unknown we define a new matrix A as the concatenation of $n$ training samples of all $k$ object classes :

$$A = [A_1, A_2, ..., A_k]$$

# Sparse Linear Combination

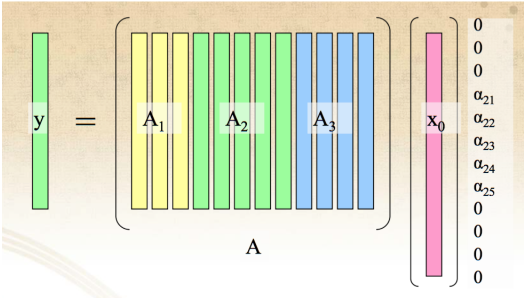



Figure: Sparse Linear Representation

$y = Ax_0 \in \mathbb{R}^m$ where
$x_0 = [0, ...., 0, \alpha_{i,1}, \alpha_{i,2}, ..., \alpha_{i,n_i}, 0, ..., 0] \in \mathbb{R}^m$

# To solve $y = Ax$ via $L^0$

- The idea is **to seek the sparsest solution** to $y = AX$
- Method :
  - We can do it by solving :

$$\hat{x}_0 = \text{argmin} \, \|x\|_0 \text{ subject to } Ax = y$$

  - Where $\|.\|_0$is the $L^0$ norm which is the number of nonzero entries in a vector
- Problem : this problem is NP-hard, and difficult even to approximate (combinatorial optimization)

# The main result : recover via $L^1$

## Sparse solution for $L^0$ minimization via $L^1$

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

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

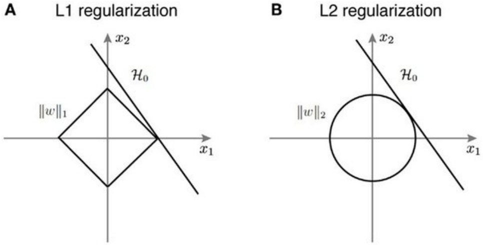
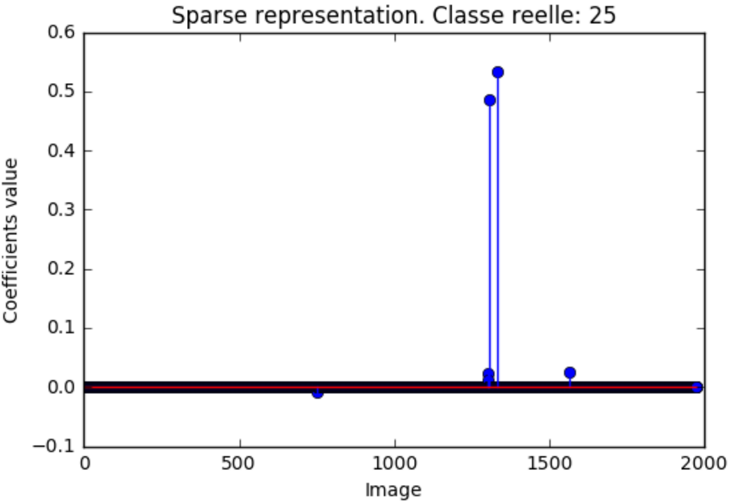



Figure: Sparse solution

# Face recognition : our example



# Plan

# Methods

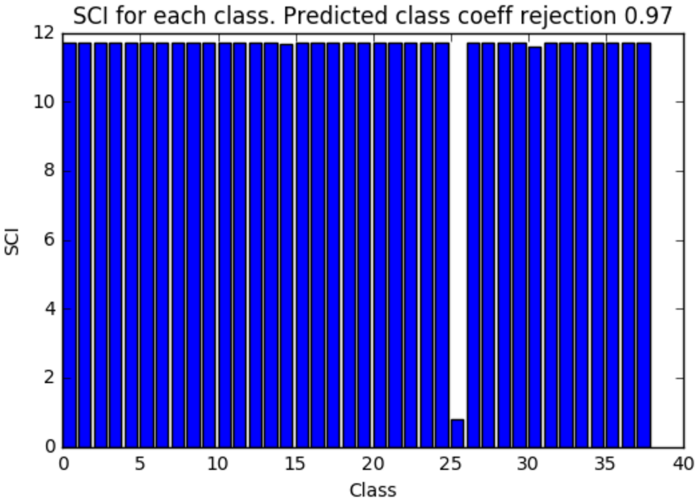
- For each class $i$, we denote $\delta_i : \mathbb{R}^n \rightarrow \mathbb{R}^n$ the characteristic function which selects the coefficients associated with the i-th class
- Using only the coefficients associated with the i-th class, we can approximate the test sample $y$ as $\hat{y}_i = A\delta_i(\hat{x}_1)$
- We classify y based on the approximation that minimizes the residual $\|y - A\delta_i(\hat{x}_1)\|_2$

# Algorithm 1: Sparse representation based Classification (SRC)

## SRC

1. Input : a matrix of training samples $A = [A_1, A_2, ..., A_k] \in \mathbb{R}^{m \times n}$ for k classes and a test sample $y \in \mathbb{R}^m$
2. Normalize the columns of A to have unit $L^2$ norm
3. Solve the $L^1$ minimization problem :
$$\hat{x}_1 = \text{argmin} \, \|x\|_1 \text{ subject to } Ax = y$$
4. Compute the residuals
for all $i = 1, ...k$, $r_i(y) = \|y - A\delta_i(\hat{x}_1)\|_2$
5. $\mathcal{C}_y = \text{argmin}_i \, r_i(y)$

# Face recognition : our example



# Face recognition : our example

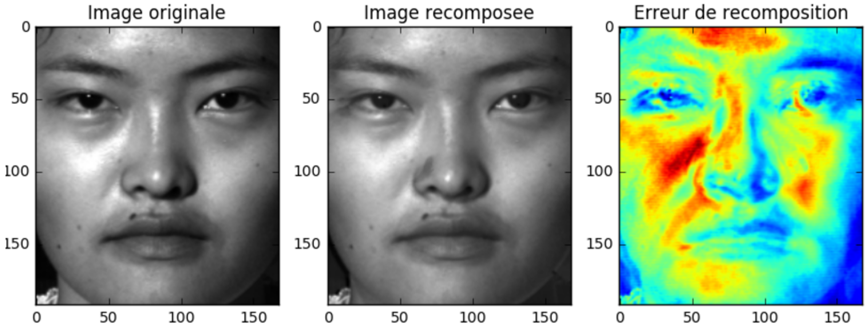



Figure: Face recomposition

# Algorithm 2: SRC with noise (relaxed problem)

- Since the real data are noisy we can rewrite the model :

$$y = Ax_0 + z$$

  where $z \in \mathbb{R}^m$ is a noise term with bounded energy, i.e, $\|z\|_2 < \epsilon$
- The sparse solution $x_0$ can still be approximately recovered by solving the stable $L^1$ minimization problem (eq to LASSO) :

$$\hat{x}_1 = \text{argmin} \|x\|_1 \text{ subject to } \|Ax - y\|_2 \leq \epsilon$$

# Algorithm 3: SRC with feature extraction

- The feature extraction is crucial because it reduce data dimension and computational cost
- Most feature transformation involve only linear operations into a **feature space** represented by $R \in \mathbb{R}^{d \times m}$ with $d << m$
- The first problem becomes

$$\widetilde{y} = Ry = RAx_0 \in \mathbb{R}^d$$

- The main results is if $x_0$ is **sparse enough** then with overwhelming probability it can be recovered via $L^1$ minimization
- Random features can be used ! (extremely efficient to generate)

# Face recognition : our example

Figure: Feature extraction

# Face recognition : our example

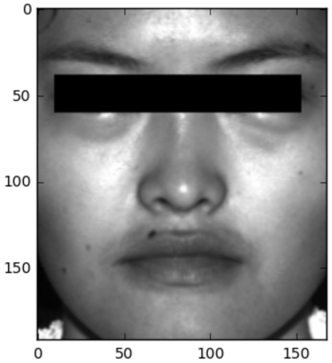



Figure: Occluded image

# Face recognition : our example

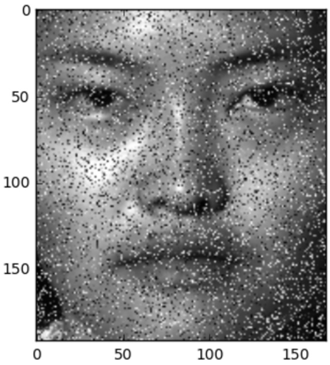


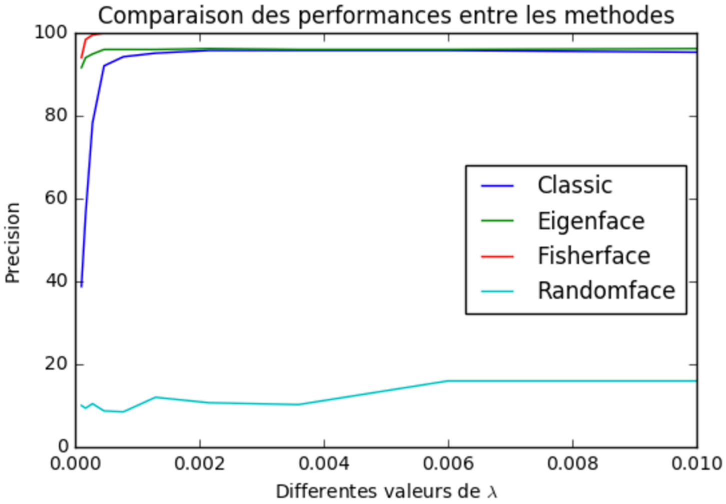

Figure: Noisy image

# Plan

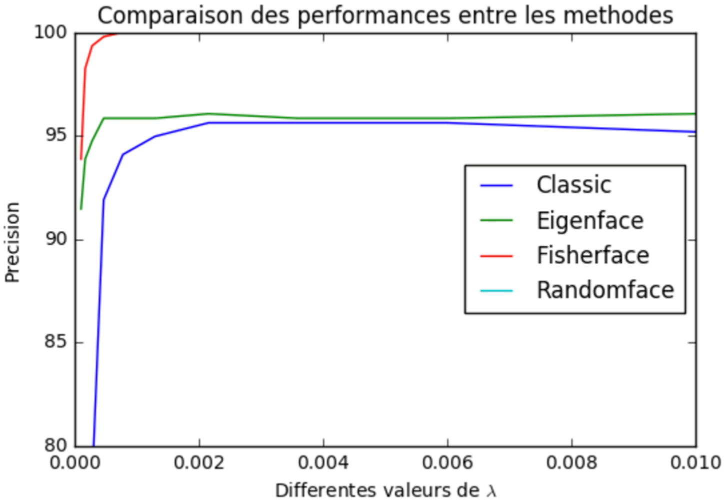
# What did we do

- Implemented sparse representation algorithm 2 (relaxed version) with LASSO
- Applied the algorithm for the situation with noise, occlusion and with different features extractions on Yale Database
- Find best $\lambda$ for LASSO with cross val
- Ilustrate and compare the algorithm with PCA + SVM approach.

# Performances



# Performances



# Plan

# Conclusion

- The algorithm worked very well but the database is very clean high preprocessing of images
- SVM quite as good and sparse as well but less interpretability
- Article's database just 100 people
- Was state of the art in 2009 and now CNN : does it better work ?