Learning Dictionary via Subspace Segmentation for Sparse Representation

Jianzhou Feng, Li Song, Xiaokang Yang and Wenjun Zhang

Institute of Image Comm. & Information Proc., Shanghai Jiaotong University

International Conference on Image Processing 2011

Outline

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

1. Introduction
2. Sparse Representation via Subspace Segmentation
   - Structured Representation Model
   - Subspace Segmentation Algorithm
   - Dictionary Constructing Algorithm
3. Experimental Results
   - Sparse Representation
   - Image Patch Denoising
4. Summary

Outline

1. Introduction
2. Sparse Representation via Subspace Segmentation
   - Structured Representation Model
   - Subspace Segmentation Algorithm
   - Dictionary Constructing Algorithm
3. Experimental Results
   - Sparse Representation
   - Image Patch Denoising
4. Summary

Sparse Representation Model

Image patch $y \in \mathbb{R}^n$
Overcomplete dictionary $\mathbf{D} = \{d_j\}_{j=1}^m$
Sparsity L

Sparse Representation Model

Image patch $y \in \mathbb{R}^n$
Overcomplete dictionary $\mathbf{D} = \{d_j\}_{j=1}^m$
Sparsity L

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

$$|\mathrm{supp}(x)| \leq L \ll n \leq m \tag{2}$$

Sparse Representation Model

Image patch $y \in \mathbb{R}^n$
Overcomplete dictionary $\mathbf{D} = \{d_j\}_{j=1}^{m}$
Sparsity L

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

$$|\mathrm{supp}(x)| \leq L \ll n \leq m \tag{2}$$

Open problems and the solution:

Sparse Representation Model

Image patch $y \in \mathbb{R}^n$
Overcomplete dictionary $\mathbf{D} = \{d_j\}_{j=1}^m$
Sparsity L

$$y = \mathbf{D}x \quad (1)$$

$$|\mathrm{supp}(x)| \leq L \ll n \leq m \quad (2)$$

Open problems and the solution:

1. The number of atom selection choice $\binom{m}{L}$ is exponentially large.

Sparse Representation Model

Image patch $y \in \mathbb{R}^n$
Overcomplete dictionary $\mathbf{D} = \{d_j\}_{j=1}^m$
Sparsity L

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

$$|\text{supp}(x)| \leq L \ll n \leq m \tag{2}$$

Open problems and the solution:

1. The number of atom selection choice $\binom{m}{L}$ is exponentially large.(*Putting structure into sparse model.*)

Sparse Representation Model

Image patch $y \in \mathbb{R}^n$
Overcomplete dictionary $\mathbf{D} = \{d_j\}_{j=1}^m$
Sparsity L

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

$$|\text{supp}(x)| \leq L \ll n \leq m \tag{2}$$

Open problems and the solution:

1. The number of atom selection choice $\binom{m}{L}$ is exponentially large.(*Putting structure into sparse model.*)
2. Pre-set dictionary size may be improper.

Sparse Representation Model

Image patch $y \in \mathbb{R}^n$
Overcomplete dictionary $\mathbf{D} = \{d_j\}_{j=1}^m$
Sparsity L

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

$$|\mathrm{supp}(x)| \leq L \ll n \leq m \tag{2}$$

Open problems and the solution:

1. The number of atom selection choice $\binom{m}{L}$ is exponentially large.(*Putting structure into sparse model.*)
2. Pre-set dictionary size may be improper. (*Setting the dictionary size adapt to the training data.*)

Outline

1 Introduction

2 Sparse Representation via Subspace Segmentation
- Structured Representation Model
- Subspace Segmentation Algorithm
- Dictionary Constructing Algorithm

3 Experimental Results
- Sparse Representation
- Image Patch Denoising

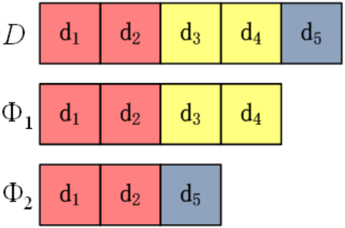
4 Summary

Structured Representation Model

Putting structure into sparse model: The atom selection choice Λ is restrict to be a subset of Γ_k, where Γ_k is one of the K subsets of $\{1, 2, \cdots, m\}$ with $|\Gamma_k| \ll m$.

Structured Representation Model

Putting structure into sparse model: The atom selection choice Λ is restrict to be a subset of Γ_k, where Γ_k is one of the K subsets of $\{1, 2, \cdots, m\}$ with $|\Gamma_k| \ll m$.
Toy example:



$m = 5$, $K = 2$, $\Gamma_1 = \{1, 2, 3, 4\}$ and $\Gamma_2 = \{1, 2, 5\}$. For $L = 3$, the choice number decrease from $\binom{5}{3} = 10$ to $\binom{4}{3} + \binom{3}{3} = 5$.

Structured Representation Model

Learning dictionary under the structured model:

Learning dictionary under the structured model:

1. *Subspace segmentation:* Segmenting the training data that lie on different subspaces.

Structured Representation Model

Learning dictionary under the structured model:

1. *Subspace segmentation:* Segmenting the training data that lie on different subspaces.
2. *Dictionary construction:* Determine the shared atoms of different subspaces.

Subspace Segmentation Algorithm

Prior work for segmentation:

- *expectation maximization*(EM)
- K-means
- K-subspaces
- Generalized PCA

Our segmentation algorithm is a modification of the K-subspaces algorithm.

The K-subspaces Clustering Algorithm

1. **Initialization**
   Start with a random collection $\{S_1, \cdots, S_K\}$ of K subspaces of dimension d, where $S_k \subset \mathbb{R}^n$. Each subspace S_k is represented by one of its orthonormal basis, U_k (represented as a n-by-d matrix).

The K-subspaces Clustering Algorithm

1. **Initialization**
   Start with a random collection $\{S_1, \cdots, S_K\}$ of K subspaces of dimension d, where $S_k \subset \mathbb{R}^n$. Each subspace S_k is represented by one of its orthonormal basis, U_k (represented as a n-by-d matrix).

2. **Cluster Assignment**
   We define an operator $P_k = I_{n \times n} - U_k U_k^T$ for each subspace S_k. Each sample y_i is assigned a new label $L(y_i)$ such that

$$L(y_i) = argmin_k \|P_k y_i\|_2^2 \tag{3}$$

The K-subspaces Clustering Algorithm

1. **Initialization**
   Start with a random collection $\{S_1, \cdots, S_K\}$ of K subspaces of dimension d, where $S_k \subset \mathbb{R}^n$. Each subspace S_k is represented by one of its orthonormal basis, U_k (represented as a n-by-d matrix).
2. **Cluster Assignment**
   We define an operator $P_k = I_{n\times n} - U_k U_k^T$ for each subspace S_k. Each sample y_i is assigned a new label $L(y_i)$ such that

$$L(y_i) = argmin_k \|P_k y_i\|_2^2 \quad (3)$$

3. **Cluster Update**
   Let S'_k be the set of samples labeled as k. We apply SVD (Singular Value Decomposition) to samples in S_k to form the new basis U'_k. Stop when $S'_k = S_k$ for all k. Other wise, go to Step 2.

Modification

Unlike the K-subspace algorithm, we allow different subspace to have different dimension d_k.

Modification

Unlike the K-subspace algorithm, we allow different subspace to have different dimension d_k.
So, in the Cluster Assignment step, we replace the label function by

$$L'(y_i) = argmin_k \|P_k y_i\|_2^2 + \lambda d_k. \tag{4}$$

The left part restricts the approximation error and the right part makes the representation as sparse as possible.

Modification

Unlike the K-subspace algorithm, we allow different subspace to have different dimension d_k.
So, in the Cluster Assignment step, we replace the label function by

$$L'(y_i) = argmin_k \|P_k y_i\|_2^2 + \lambda d_k. \tag{4}$$

The left part restricts the approximation error and the right part makes the representation as sparse as possible.
In the Cluster Update step, we retain d_k columns for the new basis U'_k.

Modification

As for the initialization step, we make our algorithm *adaptive to the training data*.

Modification

As for the initialization step, we make our algorithm *adaptive to the training data*.

The subspace number K is not predefined. It is initialized to be 0. The data set is initialized as $Y = \{y_i\}_{i=1}^N$

Modification

As for the initialization step, we make our algorithm *adaptive to the training data*.

The subspace number K is not predefined. It is initialized to be 0. The data set is initialized as $Y = \{y_i\}_{i=1}^N$

Iteration:

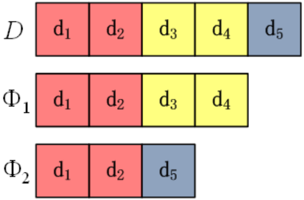
1. $K = K + 1$.
2. Find the largest neighborhood Ω in Y.
3. Find the subspace with base U_K that most data in Ω lie on.
4. Set all the data lie on U_K in Y to be S_K.
5. $Y = Y - S_K$.

Dictionary Constructing Algorithm

Task: Given a segmentation $\{S_1, \cdots, S_K\}$ and its dimension set $\{d_1, \cdots, d_K\}$, constructing the dictionary $\mathbf{D}$.

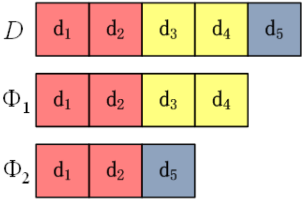
Dictionary Constructing Algorithm

Task: Given a segmentation $\{S_1, \cdots, S_K\}$ and its dimension set $\{d_1, \cdots, d_K\}$, constructing the dictionary $\mathbf{D}$.
Recall the toy example for $K = 2$.



Dictionary Constructing Algorithm

Task: Given a segmentation $\{S_1, \cdots, S_K\}$ and its dimension set $\{d_1, \cdots, d_K\}$, constructing the dictionary $\mathbf{D}$.
Recall the toy example for $K = 2$.



$\mathbf{D} = (\mathbf{B}_1|\mathbf{B}_2|\mathbf{B}_3)$ with $\Phi_1 = (\mathbf{B}_1|\mathbf{B}_2)$ and $\Phi_2 = (\mathbf{B}_1|\mathbf{B}_3)$.

$$\begin{cases} (\mathbf{B}_1|\mathbf{B}_2) = Q_1 U_1 \\ (\mathbf{B}_1|\mathbf{B}_3) = Q_2 U_2 \end{cases} \tag{5}$$

$\mathbf{B}_1$ can be obtained according to the SVD result of $U_1^T U_2$.

Dictionary Constructing Algorithm

Similarly, for $K > 2$. For $k = 1$ to K.
Suppose

$$\mathbf{D}^{(k-1)} = (\mathbf{B}_1 | \cdots | \mathbf{B}_{n_{k-1}}).$$

We obtain $\mathbf{B}_i'$ by computing SVD for $\mathbf{B}_i^T U_k$, for $i = 1, ..., n_{k-1}$.

$$\begin{aligned} \mathbf{D}^{(k)} = (&\mathbf{B}_1' | \cdots | \mathbf{B}_{n_{k-1}}' | \\ &\mathbf{B}_1 / \mathbf{B}_1' | \cdots | \mathbf{B}_{n_{k-1}} / \mathbf{B}_{n_{k-1}}' | \\ &U_k / \mathbf{B}_1' / \cdots / \mathbf{B}_{n_{k-1}}') \end{aligned} \tag{6}$$

Outline

Sparse Representation

We choose 10000 patches of size 8×8 from a given image for representation.

$L = \frac{\sum_{k=1}^{K} |S_k| d_k}{\sum_{k=1}^{K} |S_k|}$: the average number of atoms used.

n_K: the dictionary size.

Table 1: Dictionary learning results.

image	K	L	n_K	RMSE
lena	14	7.65	95	4.23
barbara	18	13.37	193	4.53
boat	21	13.32	205	4.86
peppers	18	7.87	147	4.33
man	14	24.22	211	4.87

Image Patch Denoising

For a given image I and the noisy version $I' = I + \varepsilon$, where $\varepsilon_{ij} \sim N(0, \sigma^2)$ is white noise, we collect all the 8×8 patches to be $\{y_i\}$ and $\{y_i'\}$. Total number of signal in $\{y_i\}$ ia about 250000.
Training data: Randomly selected 10000 patches from $\{y_i\}$.
For denoising: All patches in $\{y_i'\}$.

Table 2: Denoising RMSE under different σ, the column from left to right represents K-SVD, SSMS and the proposed algorithm respectively. The smallest one is bolded.

image	$\sigma = 10$			$\sigma = 20$		
lena	6.20	6.50	**5.60**	9.18	9.63	**8.30**
barbara	7.48	7.84	**6.59**	11.52	11.93	**10.22**
boat	7.55	7.94	**6.93**	11.18	11.67	**10.31**
peppers	6.52	6.76	**6.03**	9.22	9.72	**8.33**
man	8.36	9.03	**8.10**	12.40	13.11	**12.01**

Outline

Summary

We proposed a new algorithm for dictionary learning.

The learned dictionary is strongly structured with its size adaptive to the training data.

Initial supportive experiments showed its superiority and potential in image processing related applications.