Sub clustering K-SVD: Size variable Dictionary learning for Sparse Representations

JianZhou Feng, Li Song, XiaoKang Yang and Wenjun Zhang
Institute of Image Communication & Information Processing, Shanghai Jiao Tong University, China
{bravefjz,song_li,xkyang,zhangwenjun}@sjtu.edu.cn

Abstract

Sparse signal representation from overcomplete dictionaries have been extensively investigated in recent research, leading to state-of-the-art results in signal, image and video restoration. One of the most important issues is involved in selecting the proper size of dictionary. However, the related guidelines are still not established. In this paper, we tackle this problem by proposing a so-called sub clustering K-SVD algorithm. This approach incorporates the subtractive clustering method into K-SVD to retain the most important atom candidates. At the same time, the redundant atoms are removed to produce a well-trained dictionary. As for a given dataset and approximation error bound, the proposed approach can deduce the optimized size of dictionary, which is greatly compressed as compared with the one needed in the K-SVD algorithm.

Highlights

In this paper, we propose a so-called Sub clustering K-SVD algorithm, which characterizes its improvement on K-SVD method in two main aspects: (1) An error-driven mechanism is introduced to the dictionary update stage, achieving a better reconstruction result. (2) Priority of the atoms guides the refinement of the dictionary. Thus the most important atoms are retained and well refined.

The proposed scheme

Inputs: data points Y= $\{y_i\}_{i=1}^N$, expected RMSE E.

Initializing:

For i=1,…,N, $w_i = \|y_i\|_2$ and $d_i^0 = \frac{y_i}{w_i}$

Initial dictionary and first group of atoms $D^0=G^0=SC(\{w_i\},\{d_i^0\})$

Set initial OMP RMSE=E^0. Set J=0, Iter=$Iter_{max}$.

Repeat until Iter=0

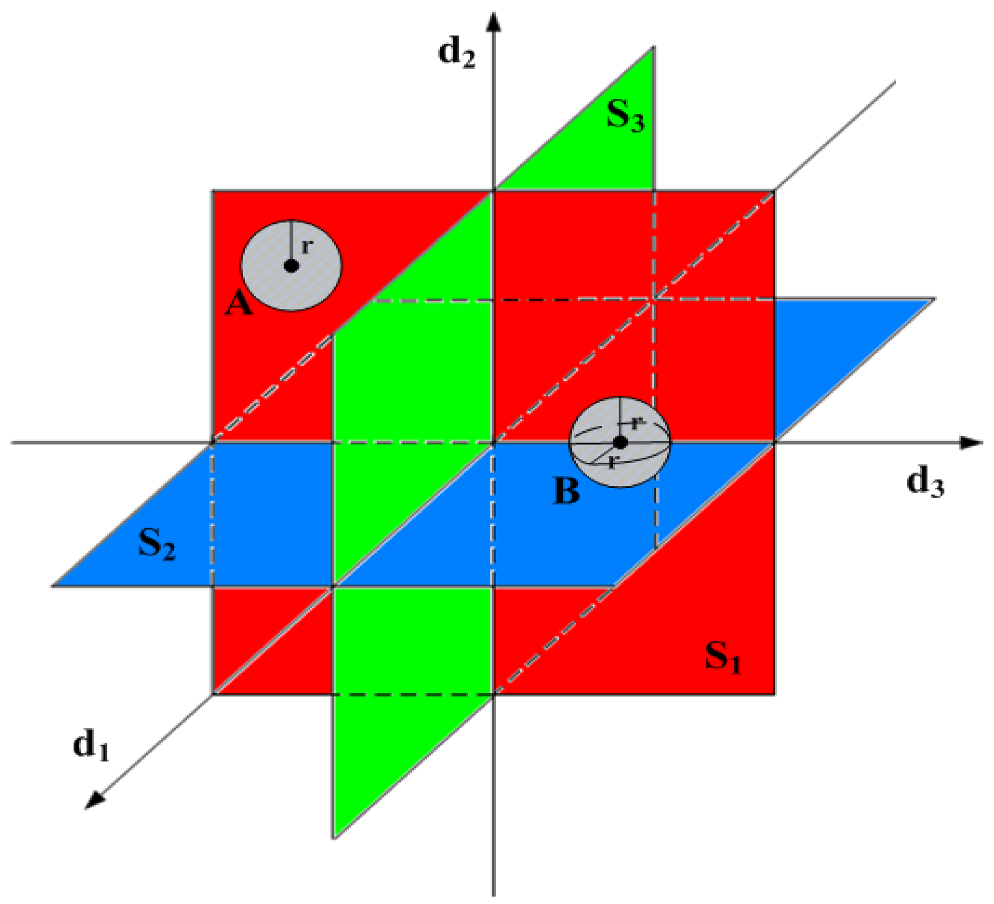
- $(X_J, count, E^{real}) = OMP(Y, D^J, E^J, T_0)$
  count: the number of image patches whose sparsity > T_0 under noise level E^J
  E^{real} : the real RMSE after OMP
- If (count > N/10 and E^{real}> E)
  time to find new group of atoms
  Residuals $\hat{r}_i = y_i - \hat{y}_i$, $\hat{y}_i$ is the approximation of y_i using $T_0/2$ atoms
  For i=1,…,N, $w_i = \|\hat{r}_i\|_2$ and $d_i^{J+1} = \frac{\hat{r}_i}{w_i}$
  $G^{J+1} = SC(\{w_i\},\{d_i^{J+1}\})$
  $G^{J+1} = G^{J+1} \setminus D^J$: an atom in G^{J+1} too close to an atom in D^J should be cut
  $D^{J+1} = D^J \cup G^{J+1}$ and $index = \{i \mid d_i \in G^{J+1}\}$
  $X_{J+1}=OMP(Y,D^{J+1},E^J,T_0)$
  Else
  $D^{J+1}=D^J$, $G^{J+1}=G^J$ and $X_{J+1}=X_J$
- If count < N/10, $E^{J+1}=\max(\delta E^J, E)$, $\delta<1$ is the learning step
- If E^{real}<E, Iter = Iter-1
- $(X_{J+1}, D'^{J+1}) = KSVD(X_{J+1}, D^{J+1})$
- $G'^{J+1} = \{d'_i \mid i \in index\}$
- For $i \in index$ $w_i = \|x^i_{J+1}\|_1$ and $G'^{J+1} = SC(\{w_i\}, G'^{J+1})$
- $D^{J+1} = \{d'_i \mid i \notin index\} \cup G'^{J+1}$

Set J=J+1

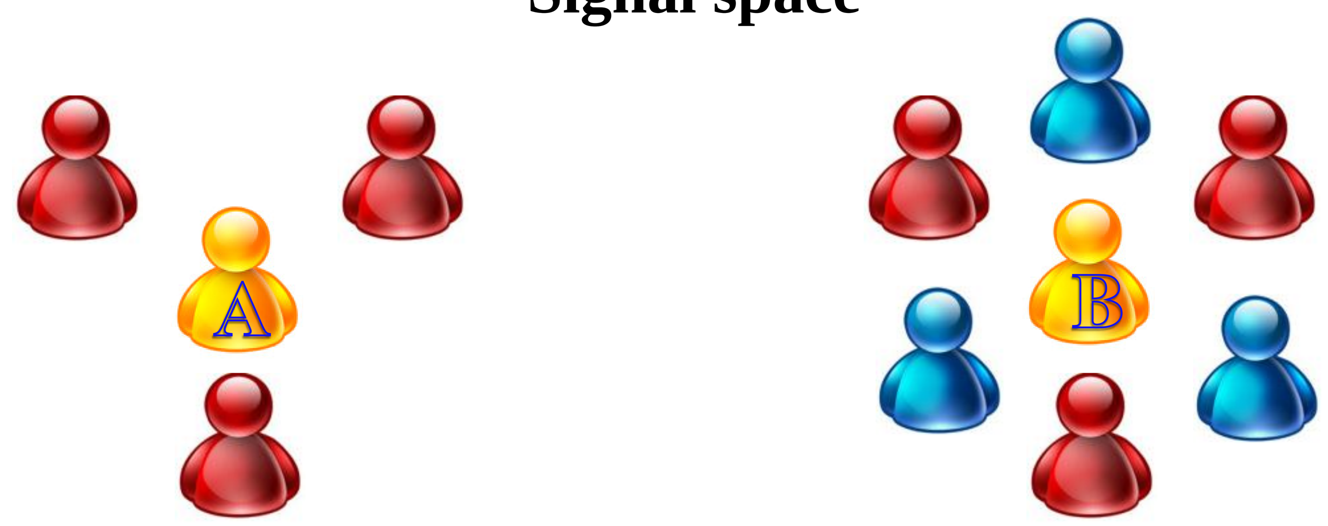
Toy example for explanation

The dictionary learning capabilities of K-SVD and the ability to find dominant atoms of subtractive clustering algorithm can be combined to learn better dictionaries with proper size.

In order to show that the centers extracted by subtractive clustering are close to dictionary atoms used to generate the data space, let's consider a simple case.



Signal space



 A point from S_1

 A point from S_2

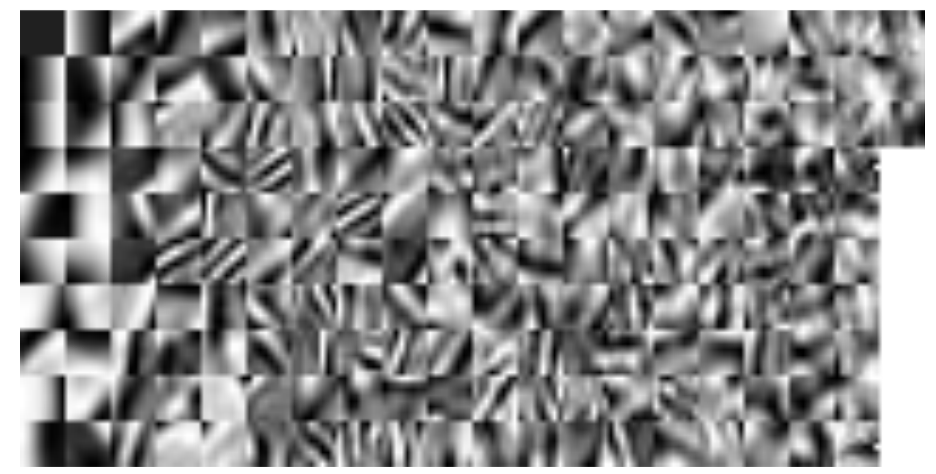
 A test point caught by clustering

This toy example suggests that SC can be used to improve K-SVD in the following two aspects. Firstly, SC can be used to initialize the dictionary with data points of clustering center. Secondly, we can use SC to pruning similar atoms group or seldom used atoms learned during K-SVD iteration.

EXPERIMENTAL RESULTS

Natural image experiments:

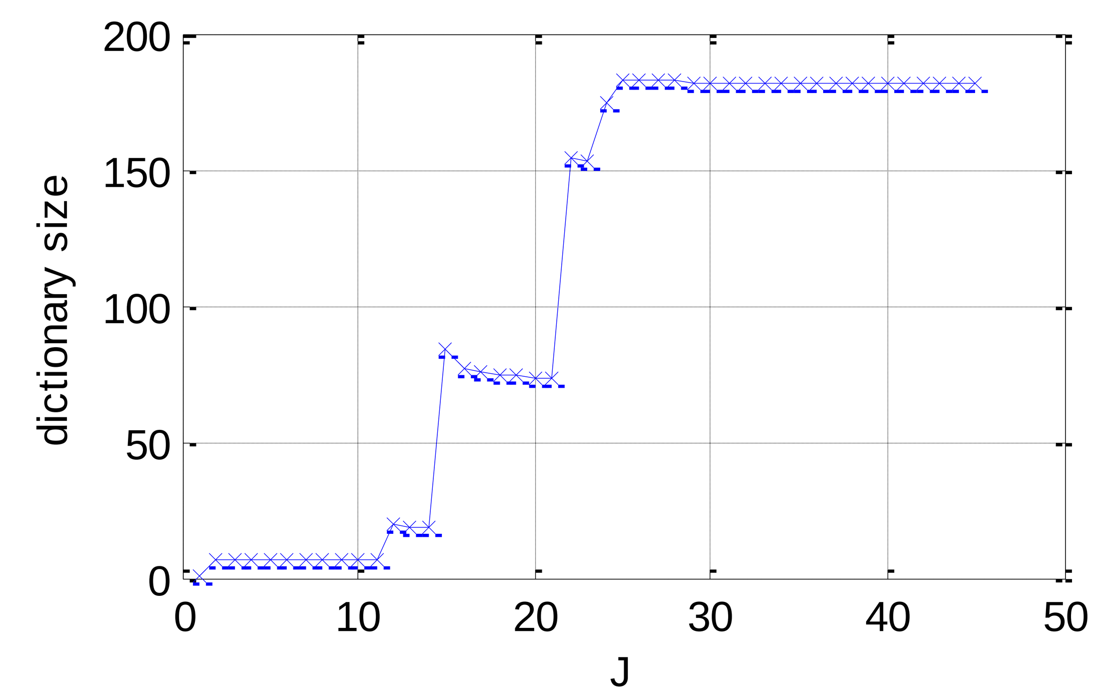
Dictionaries learned from *lena* training data

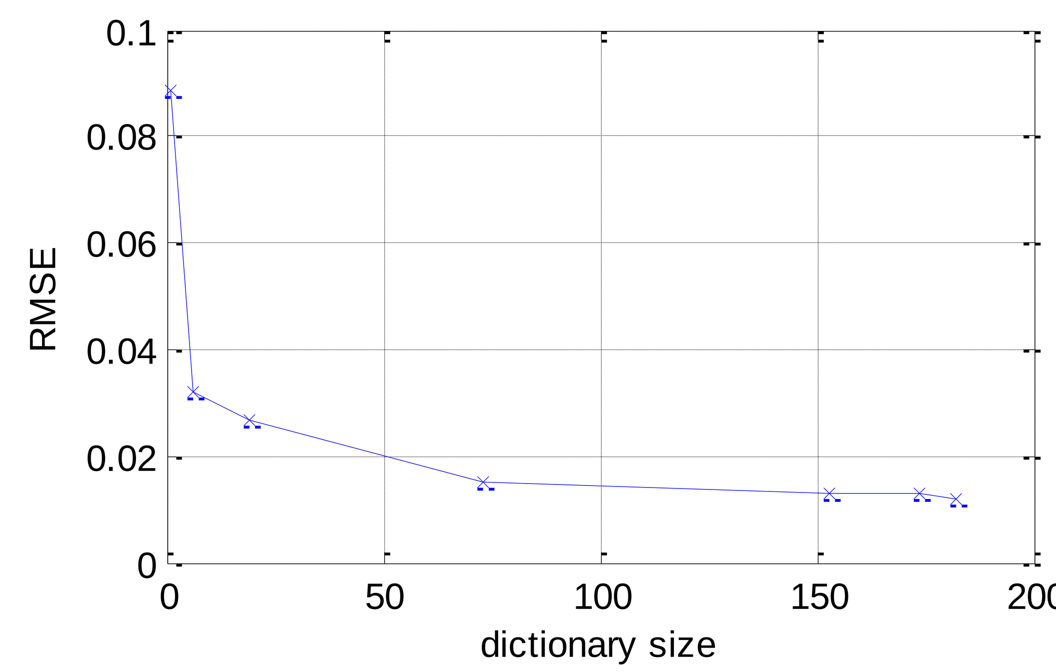


Sub clustering K-SVD

K-SVD

Sub Clustering K-SVD Iterations
Learning dictionary from a public face images database





Conclusion

In this paper, we present a so-called sub clustering K-SVD algorithm to provide a useful tool for estimating the proper size of dictionary in sparse signal representation. As compared with K-SVD method, a rather smaller dictionary is needed to satisfy the given error bound.