# K-SVD Algorithm Using Batch-OMP in Image Denoising

**Qingyi He**
qingyihe@uchicago.edu

**Ruoxi Chen**
ruoxi98@uchicago.edu

## Abstract

Image denoising has been a classic problem for a long time. A great number of algorithms have been crafted and refined to solve this problem. Of all types of solutions, the use of sparse and redundant representations over trained dictionaries, namely, K-SVD algorithm, has been proven to be highly effective and promising. In this project, we focus on K-SVD with Batch-OMP algorithm, try to understand both its mathematics and implementations. Specifically, we wish to understand the algorithm from its optimal range of applications and the impact of its key parameters on the quality of image reconstruction. In order to achieve this goal, we first reviewed image denoising algorithms and the specific kind of K-SVD algorithm using Batch-OMP that attracts us in *Part 1*. Then we walked through the kernel mathematics for both Batch-OMP and the K-SVD algorithms in *Part 2*. Finally, in *Part 3*, we reproduced this algorithm and tested its performance on 6 different images and observed the relationship between ***K*** – the ***target sparsity*** in K-SVD, and the reconstruction quality.

## 1 Introduction

### 1.1 Image denoising algorithms

Image denoising is the process of restoring the best possible quality of an image from its noisy forms, and such task can be quite challenging as significant amounts of information are lost during image degradation. How to get a good estimation of the original image thus becomes the key in image restoration.

According to *Inpainting and Denoising Challenges*, three types of representative denoising approaches have been developed since 1992, namely, ”Not learning-based method”, ”Generative learning-based method”, and ”Discriminative learning-based method”[1]. Filtering-based methods[2] and total variation (TV)-based regularization [3] based on the smoothness assumption are two of the influential image denoising algorithms that belong to the first category. Sparse models such as ***K-SVD*** proposed in 2006 [4] , the one we would like to focus on today, that take advantage of some regularity of the input signals are a large group that belongs to the "Generative learning-based method". And in the very last few years, deep neural networks (DNNs) became the most popular algorithms for image processing, and have proved to be quite effective for image denoising. In fact, many early algorithms such as sparse models were later adjusted to DNNs, such as the sparse denoising auto-encoders proposed in 2012 [5]. Discriminative learning-based methods seem to have taken on the stage for now.

### 1.2 K-SVD algorithms

The sparsity-based K-SVD denoising algorithm was once perceived as “state of the art” shortly after published in 2006. It generalizes the K-means clustering method to dictionary learning. By applying K-means, sample signal data is first cleanly represented by a initial dictionary with K elements, and the size of K here is later transformed into the size of the updated dictionary, known as the ***target sparsity***[6], which serves as a fundamental for sparse coding in the second half of the algorithm. Consequently, the dictionary training efficiency is greatly improved while maintaining an advantageous speed of algorithm convergence[7].

However, this algorithm can be quite computationally demanding, and can only ***improve*** rather than ***optimize*** the global dictionaries. This is where scholars came in trying to use different sparse coding methods and incorporate more computational convenient models to settle these problems. There have been K-SVD variations such as Deep K-SVD denoising [8], hybrid K-SVD based on wavelet transform [9], curvelet transform [10], and SVM classification [11]. And the one we are interested in, K-SVD using Batch-OMP, can be considered as the starting point of K-SVD denoising algorithms in general.

## 2 K-SVD algorithm using Batch-OMP

Generally speaking, K-SVD tries to solve for the problem,

$$min_{\mathbf{D},\underline{\gamma}}\{||\mathbf{X}-\mathbf{D}\underline{\gamma}||^{\mathbf{2}}_{\mathbf{F}}\} \quad \text{Subject To} \quad \forall \mathbf{i}, ||\gamma_{\mathbf{i}}||_{\mathbf{0}} \le \mathbf{K}$$

And it is comprised of three steps: 1) sparse coding step, where sparse approximations of the image is computed by using the initial dictionary; 2) dictionary updating step, where the dictionary is updated so that the quality of the sparse approximations is improved; 3) fit/reconstruction step, where the degraded image is recovered by referring to the updated sparse representation of the true image.

The very first two steps directly incorporates K-means algorithm, where ***K*** iterations are carried out to find the best dictionary to represent the sample image by nearest neighbour so that each input signal in the sample image could be represented by a linear combination of elements in that dictionary. A classic expression is given by the following, where a ***minimum*** of ***K***, which is the non-zero entries in the updated dictionary, again, known as ***target sparsity***, is desired to get the best sparse approximation,

$$\hat{\underline{\gamma}} = Argmin_{\underline{\gamma}}||\underline{x}-\mathbf{D}\underline{\gamma}||^{\mathbf{2}}_{\mathbf{2}} \quad \text{Subject To} \quad ||\underline{\gamma}||_{\mathbf{0}} \le \mathbf{K}$$ [1]

In order to solve for this approximation, *orthogonal matching pursuit* (OMP) is suggested by Prof. Elad in his 2006 paper[4] for its simplicity and efficiency. It rewrites the expression as the inner product between the signal and the dictionary columns, and can effectively identify the index set of atoms participating in the sparse approximation of signal sets.

Specifically, the algorithm [7] we would like to study today followed this suggestion, and further improved the original K-SVD algorithm. It uses a modified dictionary update step rather than the explicit SVD computation to reduce complexity and memory requirements. Also, it employs an optimized OMP implementation which is enhanced for large signal sets and accelerates the sparse-coding step.

To start with, this algorithm adjusts the original OMP to Batch-OMP by incorporating least squares and Cholesky factorization. This update avoids computing the inverse for non-singular dictionaries in the original OMP, allows reconstructing representations of multiple signals at the same time, and thus is more suitable for dealing with large signal sets[2]. Part of the key mathematics can be seen below[3],

$$\text{Cholesky OMP:} \quad \underline{w} = \tilde{L}^{-1}\mathbf{D}^{\mathbf{T}}_{\mathbf{I}}\underline{\mathbf{d}}_{\hat{\mathbf{k}}}, \quad \mathbf{L} = \begin{pmatrix} \tilde{L} & \underline{0} \\ \underline{w}^T & \sqrt{1-\underline{w}^T\underline{w}} \end{pmatrix}, \quad \underline{\gamma}_{\mathbf{I}} = (\mathbf{L}\mathbf{L}^{\mathbf{T}})^{-\mathbf{1}}\underline{\mathbf{a}}^{\mathbf{0}}{}_{\mathbf{I}}$$ [4]

$$\text{Batch OMP:} \quad \mathbf{G} = \mathbf{D}^{\mathbf{T}}\mathbf{D}, \quad \underline{\mathbf{w}} = \mathbf{L}^{-\mathbf{1}}\mathbf{G}_{\mathbf{I},\hat{\mathbf{k}}}, \quad \mathbf{L} = \begin{pmatrix} \tilde{L} & \underline{0} \\ \underline{w}^T & \sqrt{1-\underline{w}^T\underline{w}} \end{pmatrix}, \quad \underline{\gamma}_{\mathbf{I}} = (\mathbf{L}\mathbf{L}^{\mathbf{T}})^{-\mathbf{1}}\underline{\mathbf{a}}^{\mathbf{0}}{}_{\mathbf{I}}$$

After sparse approximation using Batch-OMP, the dictionary update can be performed by optimizing one column (the latest column, *j-th* column in the algorithm) at a time and keeping the rest fixed. The update can be achieved by solving for the optimization problem below,

$$Min_{\mathbf{D},\mathbf{\Gamma}}||\mathbf{X}_{\mathbf{I}}-\mathbf{D}\mathbf{\Gamma}_{\mathbf{I}}||^{\mathbf{2}}_{\mathbf{F}} \quad \text{Subject To} \quad \forall \mathbf{i} \quad ||\underline{\gamma}_{\mathbf{i}}||_{\mathbf{0}} \le \mathbf{K}$$ [5]

[1] $\gamma$ here represents the sparse approximation for the sample image.

[2] For full Batch-OMP algorithms, read: The paper in 2008 and the document for Batch-OMP in Sparse-plex

[3] ***I*** refers to the indices of the selected atoms(ordered sequence); ***k*** refers to the numbers of iterations

[4] $\underline{a}^0 = \mathbf{D}^{\mathbf{T}}\underline{\mathbf{x}}$

[5] $\Gamma$ refers to the entire sparse approximation, specifically, $\Gamma = [\gamma_1, \gamma_2, ..., \gamma_i]$

And thus, the entire algorithm for approximate K-SVD using Batch-OMP is given. ***Line 2*** gives the sparse approximation using Batch-OMP, and ***line 3-13*** give the updated dictionary through iterations.

**Algorithm 1** K-SVD using Batch-OMP

**Input:** Signal set $\mathbf{X}$, initial dictionary $\mathbf{D_0}$, target sparsity $\mathbf{K}$, number of iterations k.
**Output:** Dictionary $\mathbf{D}$, sparse matrix $\mathbf{\Gamma}$, such that $\mathbf{X} \approx \mathbf{D\Gamma}$ Init: Set $\mathbf{D} := \mathbf{D_0}$

```
1:  for n = 1...k do
2:      ∀i Γ_i := Argmin_γ ||x_i − Dγ||_2^2   Subject To   ||γ||_0 ≤ K
3:      for j = 1...L do
4:          D_j := 0
5:          I := {indices of the signals in X whose representations use d_j}
6:          g := Γ_{j,I}^T
7:          r := X_I − DΓ_I
8:          d := r^T g
9:          d := ||d||_2
10:         g := rd
11:         D_j = d
12:         Γ_{j,I} := g^T
13:     end for
14: end for
```

## 3 K-SVD Application and Test Results

### 3.1 Test on different types of images

**Data source** After rewriting Python syntax by referring to the algorithm we discussed above, we choose 6 different types of images to test its performance. There are black and white, colored, 3-d, informative CT photo, black and white mosaic and colored mosaic images. The goal is to test the performance of K-SVD using Batch-OMP on as much types of signals as possible to find its optimal applications.

**Pre-Processing** We first tried to uniform the pixel sizes of those images by setting them all to be 600x400 after we read them into the environment. And next, we manually added Gaussian noise with fixed seed = 1 to make sure choices of noise type and standard deviations won't affect test results.

**Denoising and observations** This K-SVD algorithm turns out to work on all types of images we chose. And the performance can be (though vaguely) seen from below. Also, with larger pixel sizes, the cost of running the algorithm can be considerably higher. We tried one black and white image with pixel size 2800 x 2800, it cost us roughly an hour to denoise. How will other variations of K-SVD perform on such large images would be another interesting experiment for us to carry out in the future.

### 3.2 Observe the impact of different K(s) on reconstruction quality measured by PSNR

**PSNR** Peak signal-to-noise ratio (PSNR) stands for the ratio between the maximum representative power of the signal matrix and the power of noise that affects the accuracy of representation by the signal matrix. It is one of the most widely used metrics to quantify the reconstruction quality for degraded images. With ***higher*** scores, the reconstruction quality of the noised image will be ***better***. This is the central measurement we used to examine the impact of K on the reconstruction quality[6].

**Experiments and findings** To observe the change, we varied ***K*** from 10 to 120 in steps of 10 units at a time, and we expected to see a close-to-linear relationship between the choice of ***K*** and the reconstruction accuracy. And the result can be seen from the graph below. With higher ***K***, the

[6]PSNR $= 10 \cdot \log_{10}\left(\frac{\mathrm{MAX}_I^2}{\mathrm{MSE}}\right) = 20 \cdot \log_{10}(\mathrm{MAX}_I) - 10 \cdot \log_{10}(\mathrm{MSE})$, where *I* represents a noise-free image and *K* is its noisy approximation.

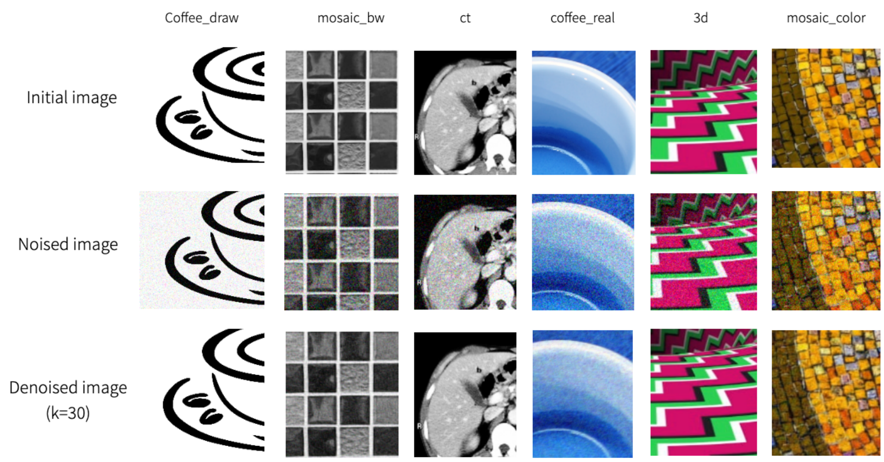


Figure 1: 6 initial, noised, and denoised images

reconstruction accuracy tends to drop eventually for 3 out of 6 images we chose. And for those images, **K = 10 or 20** turned out to be the most accurate condition for image reconstruction. It would be helpful for us to focus on ***K*** smaller than 20 to dive deeper into this topic.

However, there are indeed three exceptions, which are the black and white coffee, 3-d and the colored mosaic image, where accuracy surprisingly increased as ***K*** grew. Also, the reconstruction errors vary greatly across these 6 images, with that of the black and white coffee image surprisingly being the lowest, namely, the worst of all experiments. Our guess at the moment for the unexpectedness is mainly from the perspective of color scales and numbers of pixels with related to each large color category. More experiments that better control over factors other than color, for instance, noise type and level, number of iterations, and number of training signals would be necessary to make more responsible assumptions regarding which types of images might be the most suitable for K-SVD denoising algorithm.

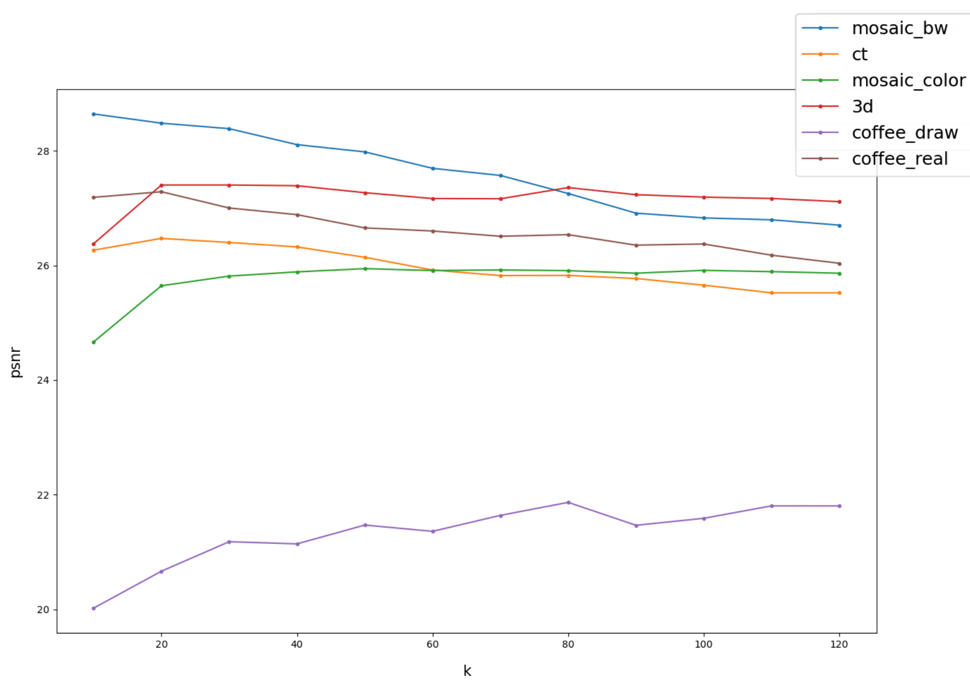


Figure 2: Change of reconstruction quality as K varies measured by PSNR

## References


[1] S. Gu, K.-K. Ma, and L.-H. Chen, "A Brief Review of Image Denoising Algorithms and Beyond | SpringerLink," 2019.

[2] T. Chen, K.-K. Ma, and L.-H. Chen, "Tri-state median filter for image denoising," *IEEE Transactions on Image Processing*, vol. 8, pp. 1834–1838, Dec. 1999. Conference Name: IEEE Transactions on Image Processing.

[3] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, pp. 259–268, Nov. 1992.

[4] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, pp. 4311–4322, Nov. 2006. Conference Name: IEEE Transactions on Signal Processing.

[5] J. Xie, L. Xu, and E. Chen, "Image Denoising and Inpainting with Deep Neural Networks," in *Advances in Neural Information Processing Systems*, vol. 25, Curran Associates, Inc., 2012.

[6] M. Elad and M. Aharon, "Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries," *IEEE Transactions on Image Processing*, vol. 15, pp. 3736–3745, Dec. 2006. Conference Name: IEEE Transactions on Image Processing.

[7] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient Implementation of the K-SVD Algorithm using Batch Orthogonal Matching Pursuit," p. 15, 2008.

[8] M. Scetbon, M. Elad, and P. Milanfar, "Deep K-SVD Denoising," *IEEE Transactions on Image Processing*, vol. 30, pp. 5944–5955, 2021. Conference Name: IEEE Transactions on Image Processing.

[9] H. Li and F. Liu, "Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries in Wavelet Domain," in *2009 Fifth International Conference on Image and Graphics*, pp. 754–758, Sept. 2009.

[10] S. Routray, A. K. Ray, and C. Mishra, "Improving performance of K-SVD based image denoising using curvelet transform," in *2015 International Conference on Microwave, Optical and Communication Engineering (ICMOCE)*, pp. 381–384, Dec. 2015.

[11] S. Routray, A. K. Ray, C. Mishra, and G. Palai, "Efficient hybrid image denoising scheme based on SVM classification," *Optik*, vol. 157, pp. 503–511, Mar. 2018.