

# Image Reconstruction via Bayesian Inference

Zhiwei Gong

## Abstract

*Image reconstruction is a fundamental task for various research and applications. Unfortunately, the robustness of methods such as compressive sensing due to parameter tuning remains an issue even though they have demonstrated high-resolution image recovery in various settings. In this project, we explored the possibility of applying a generalized approach of sparse Bayesian learning to image reconstruction. We focus on the application of sparse Bayesian learning using a Bayesian coordinate descent (BCD) algorithm on image domain and frequency domain with different additive noise. We novelly applied Gaussian Low-pass filter and Gaussian High-pass filter to the task of image reconstruction on frequency domain, and we achieve a better performance when choosing a small variance noise and Gaussian High-Pass filter over qualitative visual checking.*

**Index Terms:** Bayesian inference, image reconstruction, inverse problems.

## 1. Introduction

In this report, we consider the image reconstruction problem [2] of the form

$$\mathbf{y} = F\mathbf{x} + \mathbf{v}, \quad (1)$$

where  $\mathbf{y}$  corresponds to the observed data,  $\mathbf{x}$  denotes the unknown exact image, and  $\mathbf{v}$  represent the noise. For simplicity, we always assume  $\mathbf{v}$  is a zero-mean normal noise with i.i.d. components, that is,

$$\mathbf{v} \sim \mathcal{N}(\mathbf{0}, \alpha^{-1}\mathbf{I}), \quad \alpha > 0. \quad (2)$$

We can think of the linear operator  $F$  in (1), often known as the forward operator, as a discretization of a forward map, which maps the exact image into the observed data. Although, in general, we store the image as a matrix, e.g.,  $X \in \mathbb{R}^{N \times N}$ , it is more convenient for us to consider a vector-representation in (1), that is,

$$\mathbf{x} = \text{Vec}(X) \in \mathbb{R}^{N^2}, \quad (3)$$

where  $\text{Vec}(X)$  denotes the vectorization of  $X$ . We use the following example to help the reader understand the formulation in (1).

**Example:** (noisy Fourier data) We illustrate the image reconstruction problem (1) using a noisy Fourier data

example, which is common in applications such as magnetic resonance imaging (MRI) and synthetic aperture radar (SAR). Here, the observed data is generated by a discrete Fourier transform perturbed by noise. Let  $\mathbf{x}$  denote the vector-representation of the exact image as in (3). To define the corresponding forward operator, we introduce the one-dimensional discrete Fourier matrix

$$D = [d_{jk}]_{1 \leq j, k \leq N} \in \mathbb{C}^{N \times N}, \quad d_{jk} = e^{-2\pi i jk/N}.$$

Then, the two-dimensional discrete Fourier transform of  $X$  is given by  $Y = DXD^\top \in \mathbb{C}^{N \times N}$ . We have

$$\text{Vec}(Y) = (D \otimes D)\mathbf{x},$$

where the matrix tensor product is defined as

$$D \otimes D = [d_{jk}D]_{1 \leq j, k \leq N} \in \mathbb{C}^{N^2 \times N^2}.$$

Thus, the observed data satisfies

$$\mathbf{y} = (D \otimes D)\mathbf{x} + \mathbf{v}. \quad (4)$$

In practice, we rewrite Eq. (4) into a real representation as follows

$$\begin{bmatrix} \text{Re}(\mathbf{y}) \\ \text{Im}(\mathbf{y}) \end{bmatrix} = \begin{bmatrix} \text{Re}(D \otimes D) \\ \text{Im}(D \otimes D) \end{bmatrix} \mathbf{x} + \begin{bmatrix} \text{Re}(\mathbf{v}) \\ \text{Im}(\mathbf{v}) \end{bmatrix}.$$

To solve the image reconstruction problem (1), we interpret it as a linear inverse problem, and formulate the estimator of the exact image  $\mathbf{x}$ , denoted by  $\hat{\mathbf{x}}$ , as the solution to the following  $\ell^1$ -regularized optimization problem [3]

$$\hat{\mathbf{x}} := \arg \min_{\mathbf{x}} \left\{ \|F\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \|R\mathbf{x}\|_1 \right\}, \quad (5)$$

where  $R$  is referred to as the regularization operator and  $\lambda > 0$  as the regularization parameter. The regularization term in (5) balances the data fidelity and the noise. It is clear that the solution  $\hat{\mathbf{x}}$  is affected by both  $\lambda$  and the choice of the regularization operator  $R$ . For example, when  $R\mathbf{x}$  corresponds to the total variation of  $\mathbf{x}$ , that is,  $[R\mathbf{x}]_j = x_{j+1} - x_j$  ( $x_j$  denotes the  $j^{\text{th}}$  component of  $\mathbf{x}$ ), (5) favors the solution of no or little variation. There are many well-developed methods target on solving (5). In this report, we consider statistical approaches that allows for uncertainty in the reconstructed solution.

## 2. Method

In this section, we review the Bayesian learning approach in solving the optimization problem (5) [3]. To implement the Bayesian formulation, we introduce the following density/likelihood functions:

- The likelihood  $p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta})$ , which is the probability density function of  $\mathbf{y}$  given  $\mathbf{x}$  and  $\boldsymbol{\theta}$ . Here,  $\boldsymbol{\theta}$  collects all the involved parameters other than  $\mathbf{x}$  and  $\mathbf{y}$ .
- The prior  $p(\mathbf{x}|\boldsymbol{\theta})$ , which is the probability density function of  $\mathbf{x}$  given  $\boldsymbol{\theta}$ .
- The hyper-prior  $p(\boldsymbol{\theta})$ , which is the probability density function of  $\boldsymbol{\theta}$ .
- The posterior  $p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{y})$ , which is the probability density function of the solution  $\mathbf{x}$  and the parameter  $\boldsymbol{\theta}$  given observation  $\mathbf{y}$ .

We need to point out that the listed density functions may rely only on a subset of  $\boldsymbol{\theta}$ . We abuse the notation for simplicity. The famous Bayes' Rule tells that

$$p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta})p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}).$$

Motivated by the sparsity, we consider the well-known class of conditionally Gaussian priors given by

$$p(\mathbf{x}|\boldsymbol{\beta}) \propto \exp\left(-\frac{1}{2}\mathbf{x}^\top R^\top B R \mathbf{x}\right), \quad (6)$$

where  $B = \text{diag}(\beta_1, \beta_2, \dots)$  is a diagonal inverse covariance matrix.

The likelihood function  $p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta})$  models the connection between the observation  $\mathbf{y}$ , the exact image  $\mathbf{x}$ , and the parameter  $\boldsymbol{\theta}$ . Recall the assumption on the noise vector  $\mathbf{v}$  in (2), which leads to

$$p(\mathbf{y}|\mathbf{x}, \alpha) \propto \exp\left\{-\frac{\alpha}{2}\|\mathbf{F}\mathbf{x} - \mathbf{y}\|_2^2\right\}. \quad (7)$$

As for the hyper-prior, we consider the Gamma distribution with a probability density function

$$\Gamma(z|c, d) = \frac{d^c}{\Gamma(c)} z^{c-1} e^{-dz}, \quad z > 0,$$

where  $c$  and  $d$  are positive shape and rate parameter, respectively. Such choice allows the parameters in the conditionally Gaussian prior (6) to have distinctly different values. In particular, we choose the inverse noise and prior variances,  $\alpha$ , and  $\boldsymbol{\beta}$ , satisfy the Gamma distribution

$$p(\alpha) = \Gamma(\alpha|c, d), \quad p(\beta_i) = \Gamma(\beta_i|c, d). \quad (8)$$

In the theory of Bayesian inference, the conditionally Gaussian prior (6) and the Gamma hyper-prior (8) form a conjugate relationship [4], which implies

$$p(\mathbf{y}|\mathbf{x}, \alpha)p(\mathbf{x}|\boldsymbol{\beta}) \propto \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, C),$$

where the mean vector  $\boldsymbol{\mu}$  and the covariance matrix  $C$  satisfy

$$C = (\alpha F^\top F + R^\top B R)^{-1}, \quad \boldsymbol{\mu} = \alpha C F^\top \mathbf{y}.$$

The Bayesian inference approach in solving the optimization problem (5) can be summarized as Algorithm 1.

---

**Algorithm 1:** Bayesian coordinate descent (BCD) algorithm for the mean [3]

---

```

1 Initialize  $\alpha, \boldsymbol{\beta}$ , and  $\ell = 0$ 
2 repeat
3   Update  $\mathbf{x}$  by  $\mathbf{x}^{\ell+1} = \mathbb{E}_{\mathbf{x}}[\mathbf{x}^\ell, \alpha^\ell, \boldsymbol{\beta}^\ell | \mathbf{y}]$ 
4   Update  $\alpha$  by  $\alpha^{\ell+1} = \mathbb{E}_\alpha[\mathbf{x}^{\ell+1}, \alpha, \boldsymbol{\beta}^\ell | \mathbf{y}]$ 
5   Update  $\boldsymbol{\beta}$  by  $\boldsymbol{\beta}^{\ell+1} = \mathbb{E}_{\boldsymbol{\beta}}[\mathbf{x}^{\ell+1}, \alpha, \boldsymbol{\beta} | \mathbf{y}]$ 
6   Set  $\ell \rightarrow \ell + 1$ 
7 until convergence or maximum number of iterations
      is reached;

```

---

Here, the notation  $\mathbb{E}_{\mathbf{x}}[\cdot]$  stands for taking expectation with respect to  $\mathbf{x}$ . We want to point out that, under all the assumption on the density and likelihood functions, in Algorithm 1 updating  $\mathbf{x}$  reduces to solving the following linear system

$$(\alpha F^\top F + R^\top B R) \mathbf{x}^{\ell+1} = \alpha F^\top \mathbf{y}.$$

## 3. Experiments

Image reconstruction aims to reduce noise and recover resolution loss. Image processing techniques are usually conducted in either of the image (time) domain or in the frequency (Fourier) domain. Deconvolution is the most straightforward and conventional technique for reconstructing images, we apply the BCD algorithm to these two domains to show the performance of image deconvolution.

### 3.1. Experimental Setup

#### 3.1.1 Reference Images

We choose two different reference images from the homework: “image-house” and “image-polygon”, each of size  $256 \times 256$ .

#### 3.1.2 Evaluation Metrics

The deconvolution process is a post-processing step that emphasizes finer details of an image and makes it appear

sharper. So measure the image deconvolution performance by:

(1) Qualitative visual checking. Visually check reference image, noisy blurred image and reconstructions to see if the BCD algorithm provides a sharper image, e.g., reduce noise and recover resolution.

(2) Quantitatively measure the level of the image to the level of the background noise by signal-to-noise ratio (SNR), defined as  $\mathbb{E}[x^2]/\sigma^2$ , where  $\mathbb{E}[x^2] \approx (x_1^2 + \dots + x_n^2)/n$ .

### 3.1.3 Implementation Details

We choose the additive noise to be i.i.d. complex Gaussian noise, and set the different variance ( $\sigma^2$ ) to 0.001, 0.01 and 0.1. The Regularization operator ( $R$ ) is second-order Total Variation (TV) operator [1].

We perform the BCD algorithm on both of time domain and Fourier domain. For image deconvolution in frequency (Fourier) domain, the filters we apply are Gaussian Low-Pass filter and Gaussian High-Pass filter with different filter sizes ( $D_0$ ). The experiment is implemented in MATLAB.

## 3.2. Image Deconvolution

### 3.2.1 Time Domain

We first consider deconvolution of the images on time domain illustrated in Fig.1 and Fig.2. The corresponding model and second-order TV regularization operator are respectively shown as:

$$\mathbf{y} = F\mathbf{x} + \boldsymbol{\nu}, \quad R = \begin{bmatrix} -1 & 2 & -1 \\ \ddots & \ddots & \ddots \\ -1 & 2 & -1 \end{bmatrix}, \quad (9)$$

where  $\boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, \sigma^2 I)$ ,  $R \in \mathbb{R}^{(n-2) \times n}$  (in our experiment,  $n = 256$ ),  $F$  is the linear forward operator:

$$[F]_{ij} = h k(h[i-j]), \quad i, j = 1, \dots, n, \quad (10)$$

where  $h = \frac{1}{n}$  is the distance between consecutive grid points, and  $k$  is the Gaussian convolution kernel with  $\gamma = 0.01$ :

$$k(s) = \frac{1}{2\pi\gamma^2} e^{-\frac{s^2}{2\gamma^2}}$$

As Fig.1a, Fig.1b and Fig.1c show, with the variance of additive noise increases, it is harder to recover the noisy blurred image, and the SNR values are 309.8165, 30.9817 and 3.0982 respectively. Similarly, Fig.2a, Fig.2b and Fig.2c show the same trend as  $\sigma^2$  increases, and the SNR values are 541.2291, 54.1229 and 5.4123, respectively.

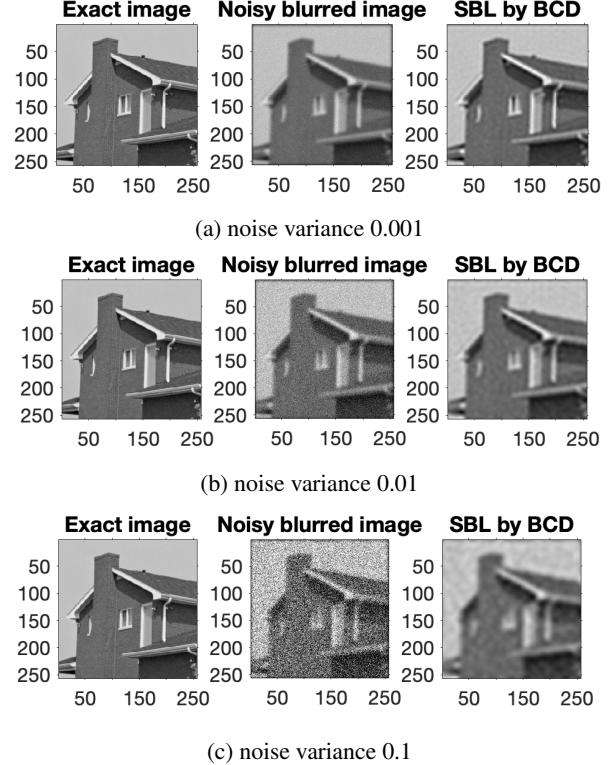


Figure 1: House image deconvolution on time domain with different noise variances

### 3.2.2 Fourier Domain

In this section, we apply the BCD algorithm on Fourier domain with different filters. The corresponding model form is same as Eq.9, while  $F$  is different from Eq.10 in time domain, for Fourier domain, here we define  $F = GF_1$ , where  $G$  is the filter we choose to apply, e.g., Gaussian Low-Pass filter or High-pass filter,  $F_1$  is the Discrete Fourier Transform (DFT). In contrast of implementation on time domain, for Fourier domain, we add the Gaussian noise on frequency space instead of image space.

The Gaussian Low-Pass filter we use is defined as:

$$G_{low}[k_1, k_2] = e^{-\frac{1}{2}(\frac{D[k_1, k_2]}{D_0})^2}, \quad (11)$$

where  $D[k_1, k_2] = \sqrt{k_1^2 + k_2^2}$  is the distance function from origin,  $D_0$  is the filter size.

Correspondingly, the Gaussian High-Pass filter we use is defined as:

$$G_{high}[k_1, k_2] = 1 - e^{-\frac{1}{2}(\frac{D[k_1, k_2]}{D_0})^2} \quad (12)$$

**Gaussian Low-Pass filter** First, we apply the Gaussian Low-Pass filter with  $D_0 = 15$  in Eq.11 shown in Fig.3. The results are illustrated in Fig.4 and Fig.5.

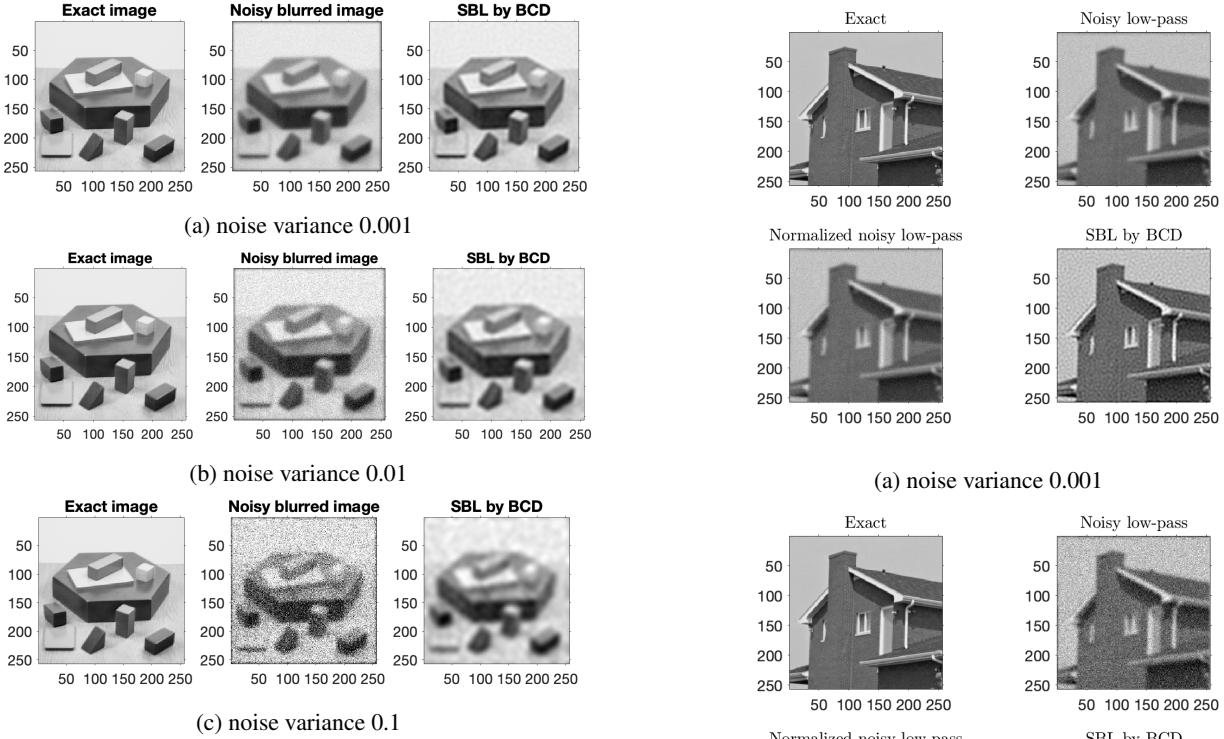


Figure 2: Polygon image deconvolution on time domain with different noise variances

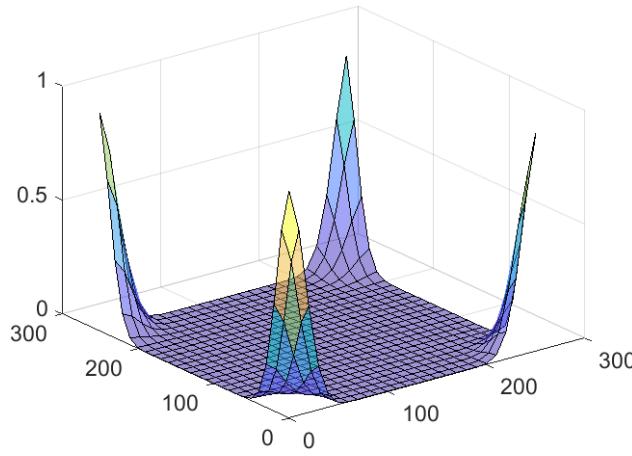


Figure 3: Gaussian Low-Pass filter with  $D_0 = 15$

As Fig.4a, Fig.4b and Fig.4c show, with the variance of additive noise increases, it is harder to recover the noisy blurred image. Similarly, Fig.5a, Fig.5b and Fig.5c show the same trend as  $\sigma^2$  increases.

**Gaussian Lower-Pass filter** We change the size of Gaussian Low-Pass filter  $D_0$  to 25 to get a Gaussian “lower-pass” filter shown in Fig.6. The results are illustrated in Fig.7 and Fig.8. Since for large noise variance, we did

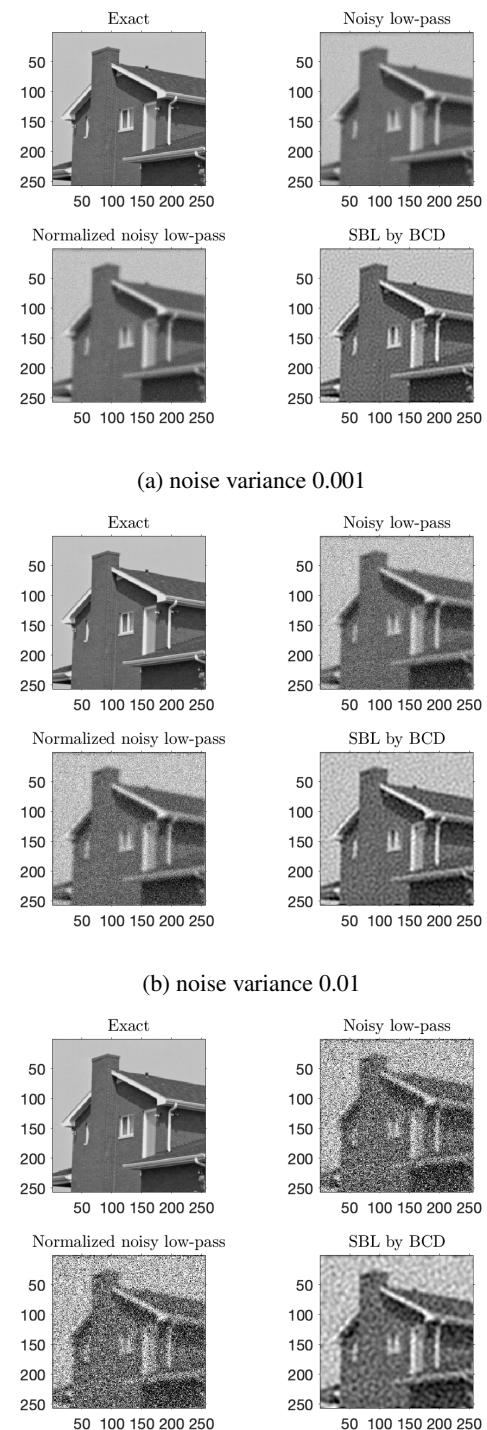


Figure 4: House image deconvolution on Fourier domain with Gaussian Low-Pass filter ( $D_0 = 15$ ) and different noise variances

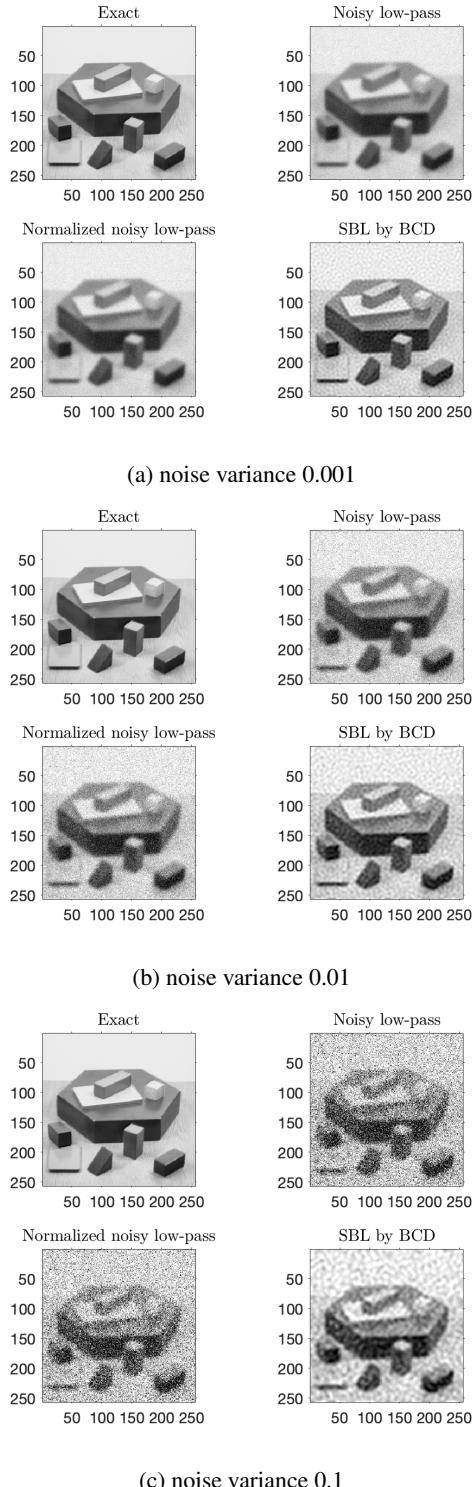


Figure 5: Polygon image deconvolution on Fourier domain with Gaussian Low-Pass filter ( $D_0 = 15$ ) and different noise variances

not get a good performance, so we just show the case in small noise variance. We compare the result of Fig.7 and Fig.8 with Fig.4a and Fig.5a respectively, the low-pass filter ( $D_0 = 15$ ) performs better than the lower-pass filter ( $D_0 = 25$ ).

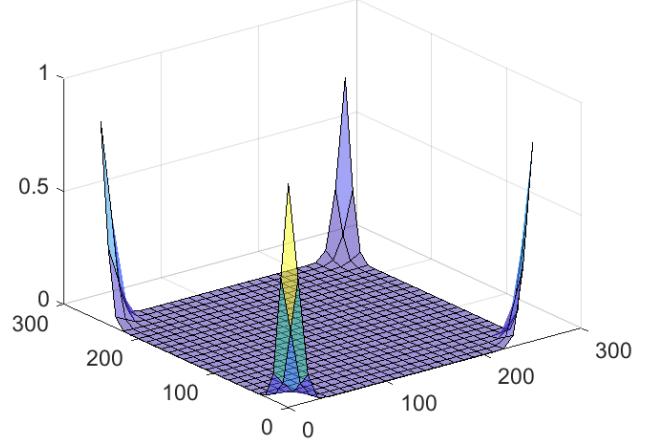


Figure 6: Gaussian Lower-Pass filter with  $D_0 = 25$

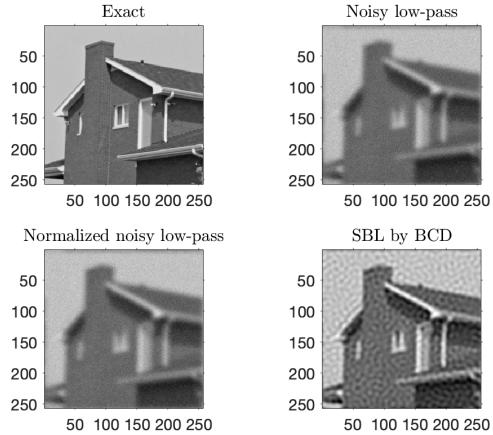


Figure 7: House image deconvolution on Fourier domain with Gaussian Lower-Pass filter ( $D_0 = 25$ ) and noise variance 0.001

**Gaussian High-Pass filter** In this section, we firstly apply the Gaussian High-Pass filter with  $D_0 = 25$  in Eq.12 shown in Fig.10. The results are illustrated in Fig.11 and Fig.12.

Since when consider the high-pass filter, the filtered noisy image is dim due to the fact that most of the energy are stored in low-frequency modes (Fig.9). To improve the visualization, we normalize the filtered image according to

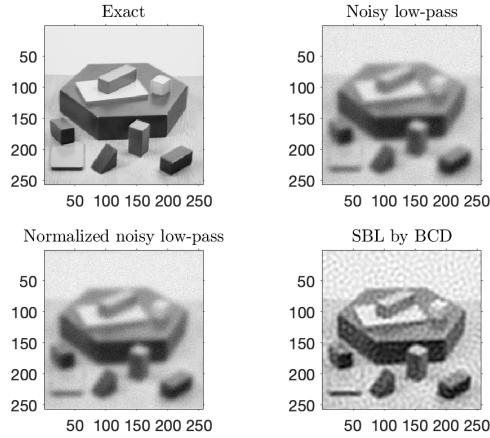


Figure 8: Polygon image deconvolution on Fourier domain with Gaussian Lower-Pass filter ( $D_0 = 25$ ) and noise variance 0.001

the total energy ( $l_1$  norm of the matrix shown in Eq.13).

$$\hat{y} = y + (\bar{x} - \bar{y}), \quad (13)$$

where  $y$  is the Normalized image,  $\hat{y}$  is Noisy high-pass image and  $x$  is the Exact image. This action does not affect the total amount of information available.

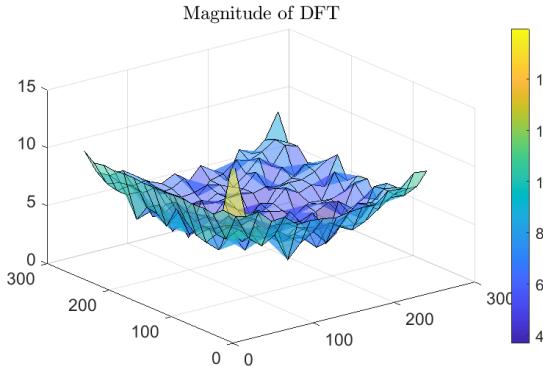


Figure 9: Magnitude of DFT

**Gaussian Higher-Pass filter** Similarly, we change the size of Gaussian High-Pass filter  $D_0$  to 15 to get a Gaussian “higher-pass” filter shown in Fig.13. The results are illustrated in Fig.7 and Fig.8. We just show the case in small noise variance. We compare the result of Fig.14 and Fig.15 with Fig.11a and Fig.12a respectively, the high-pass filter ( $D_0 = 25$ ) performs better than the higher-pass filter ( $D_0 = 15$ ).

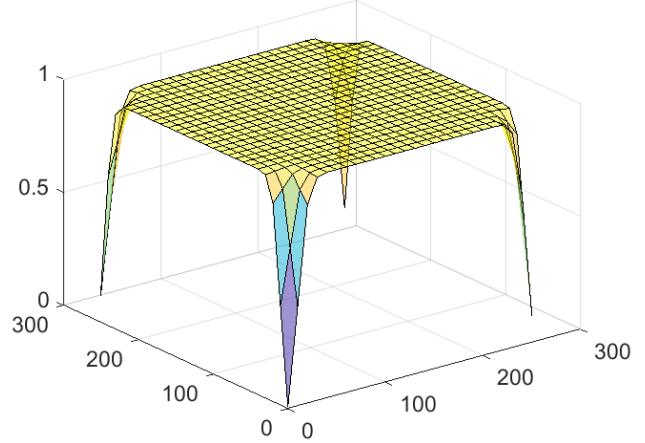


Figure 10: Gaussian High-Pass filter with  $D_0 = 25$

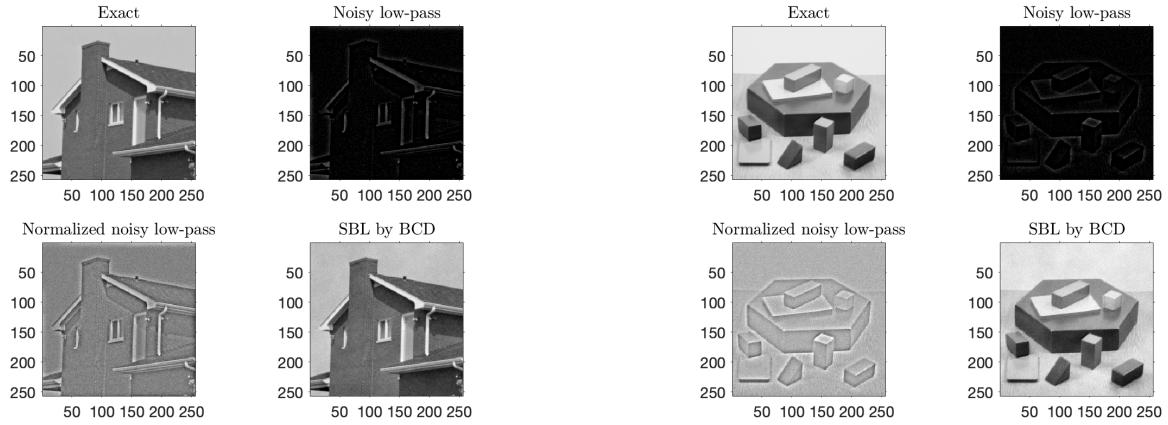
#### 4. Discussion and Conclusion

We frame the problem of image reconstruction as a linear inverse problem. Inspired by the sparse Bayesian learning, we adopted the Bayesian coordinate descent (BCD) algorithm on image domain and a different frequency domain to perform the image deconvolution task. We achieved better performance when noise variance is small (0.001), the larger the variance of the noise the more difficult the problem is. In comparison with the high-pass filter, low-pass filter, although does not change the brightness significantly, makes the filtered image harder to recover. From the image reconstruction point of view, we can tell that for this specific image high-frequency data contains more information about the details of the image while low-freq includes global information such as brightness.

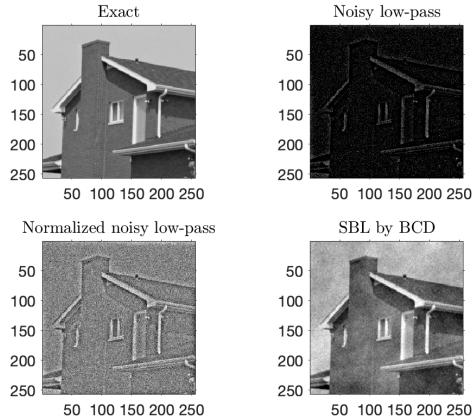
Despite the preliminary success in image reconstruction using the BCD algorithm, the performance of image with large noise variance and low-pass filter is not quite effective. Adjusting different hyper-parameters and low-pass filters on frequency domain to get an excellent performance for large noise variance case could be a promising future direction.

#### References

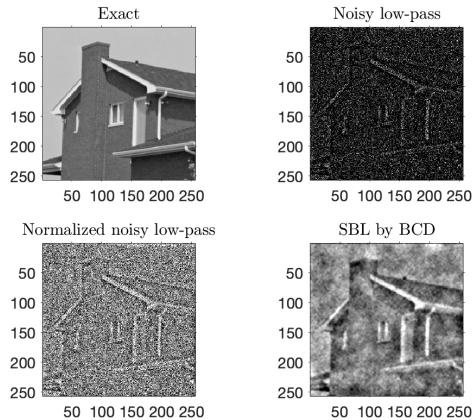
- [1] S. D. Babacan, R. Molina, and A. K. Katsaggelos. Parameter estimation in tv image restoration using variational distribution approximation. *IEEE transactions on image processing*, 17(3):326–339, 2008.
- [2] J. M. Bardsley. Mcmc-based image reconstruction with uncertainty quantification. *SIAM journal on scientific computing*, 34(3):A1316–A1332, 2012.
- [3] J. Glaubitz, A. Gelb, and G. Song. Generalized sparse bayesian learning and application to image reconstruction. *arXiv:2201.07061*, 2022.



(a) noise variance 0.001

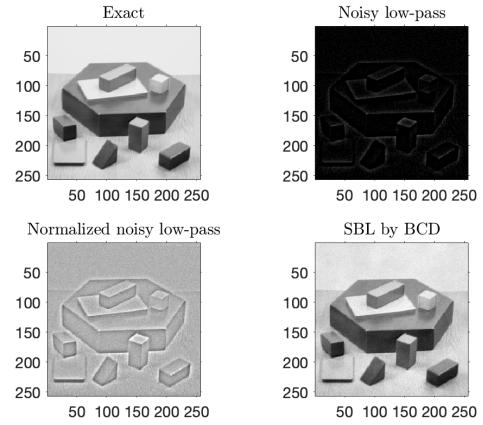


(b) noise variance 0.01

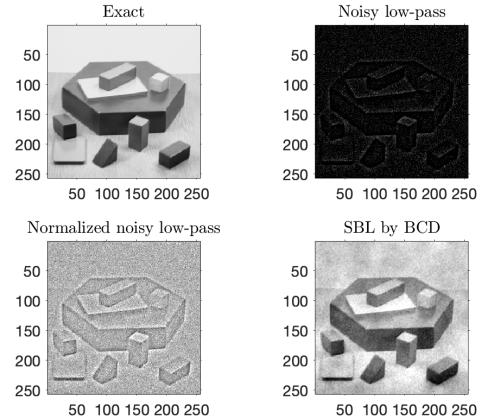


(c) noise variance 0.1

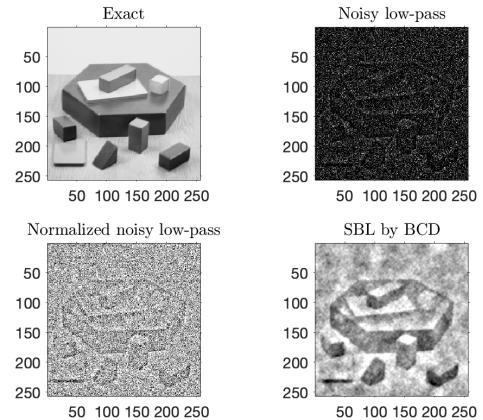
Figure 11: House image deconvolution on Fourier domain with Gaussian High-Pass filter ( $D_0 = 25$ ) and different noise variances



(a) noise variance 0.001



(b) noise variance 0.01



(c) noise variance 0.1

Figure 12: Polygon image deconvolution on Fourier domain with Gaussian High-Pass filter ( $D_0 = 25$ ) and different noise variances

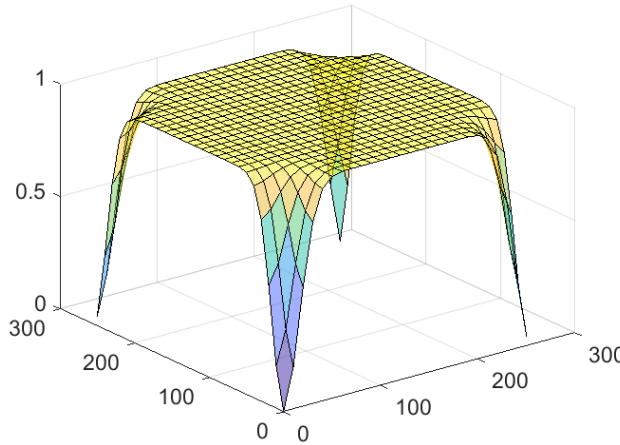


Figure 13: Gaussian Higher-Pass filter with  $D_0 = 15$

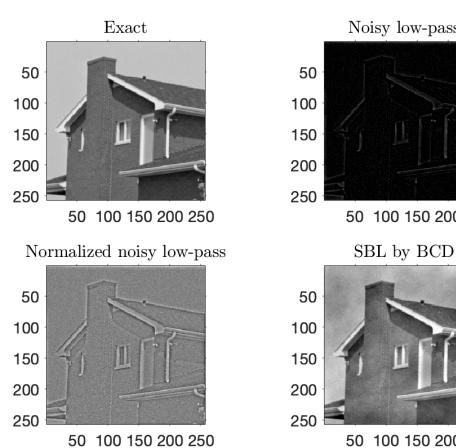


Figure 14: House image deconvolution on Fourier domain with Gaussian Higher-Pass filter ( $D_0 = 15$ ) and noise variance 0.001

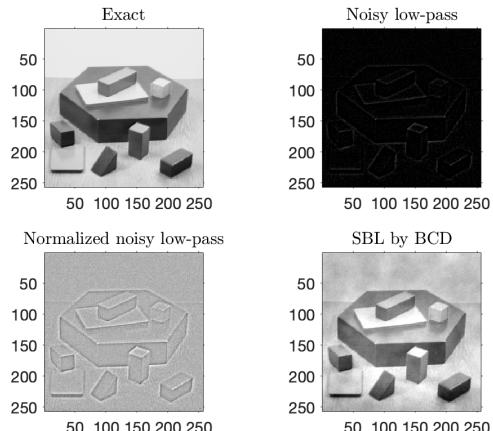


Figure 15: Polygon image deconvolution on Fourier domain with Gaussian Higher-Pass filter ( $D_0 = 15$ ) and noise variance 0.001

[4] L. Wasserman and S. O. service). *All of Statistics: A Concise Course in Statistical Inference*. Springer, New York, NY, 2013;2004;2011;2010;.