Patch-based Bayesian approaches for image restoration

Dai Viet Dai, Françoise Dibos, Sébastien Li-Thiao-Té, Marie Luong

École Doctorale 146

THESE

Pour obtenir le titre de Docteur de l'Université Sorbonne Paris Cité

Spécialité doctorale "Mathématiques appliquées et applications des mathématiques"

Soutenue publiquement à l'Université Paris 13 le 26 juin 2018 par

Dai Viet TRAN

Patch-based Bayesian approaches for image restoration

Devant un jury composé de :

M. Yann GOUSSEAU,	Télécom ParisTech	Rapporteur
M. Aymeric HISTACE,	ENSEA	Rapporteur
Mme Julie DELON,	Université Paris Descartes	
Mme Françoise DIBOS,	Université Paris 13	Directrice de thèse
M. Georges KOEPFLER,	Université Paris Descartes	
M. Sébastien LI-THIAO-TE,	Université Paris 13	Co-encadrant
Mme Marie LUONG,	Université Paris 13	Co-encadrant
M. Canh Duong PHAM,	Vietnam Academy of Science and Technology, Hanoi	

Membre invité :

M. Tien Thuong LE,	Ho Chi Minh City University of Technology	Co-encadrant

Laboratoire Analyse, Géométrie et Application

UMR CNRS 7539, Villetaneusse, France

Université Sorbonne Paris Cité

Résumé

Les travaux présentés dans cette thèse concernent les approches bayésiennes par patchs des problèmes d'amélioration de la qualité d'images. Notre contribution réside en le choix du dictionnaire construit grâce à un ensemble d'images de haute qualité et en la définition et l'utilisation d'un modèle à priori pour la distribution des patchs dans l'espace du dictionnaire.

Nous avons montré qu'un choix attentif du dictionnaire représentant les informations locales des images permettait une amélioration de la qualité des images dégradées. Plus précisément, d'un dictionnaire construit de façon exhaustive sur les images de haute qualité nous avons sélectionné, pour chaque patch de l'image dégradée, un sous dictionnaire fait de ses voisins les plus proches. La similarité entre les patchs a été mesurée grâce à l'utilisation de la distance du cantonnier (Earth Mover's Distance) entre les distributions des intensités de ces patchs. L'algorithme de super résolution présenté a conduit à de meilleurs résultats que les algorithmes les plus connus.

Pour les problèmes de débruitage d'images nous nous sommes intéressés à la distribution à priori des patchs dans l'espace du dictionnaire afin de l'utiliser comme pré requis pour régulariser le problème d'optimisation donné par le Maximum à Postériori. Dans le cas d'un dictionnaire de petite dimension, nous avons proposé une distribution constante par morceaux. Pour les dictionnaires de grande dimension, la distribution à priori a été recherchée comme un mélange de gaussiennes (GMM). Nous avons finalement justifié le nombre de gaussiennes utiles pour une bonne reconstruction apportant ainsi un nouvel éclairage sur l'utilisation des GMM.

Mots-Clés — Débruitage d'images, super-résolution d'images, maximum à posteriori, modèle de mélange gaussien, earth mover's distance, modèle parcimonieux, distribution des patchs, restauration d'images basée sur l'exemple.

Abstract

In this thesis, we investigate the patch-based image denoising and super-resolution under the Bayesian Maximum A Posteriori framework, with the help of a set of high quality images which are known as standard images. Our contributions are to address the construction of the dictionary, which is used to represent image patches, and the prior distribution in dictionary space.

We have demonstrated that the careful selection of dictionary to represent the local information of image can improve the image reconstruction. By establishing an exhaustive dictionary from the standard images, our main attribute is to locally select a sub-dictionary of matched patches to recover each patch in the degraded image. Beside the conventional Euclidean measure, we propose an effective similarity metric based on the Earth Mover's Distance (EMD) for image patch-selection by considering each patch as a distribution of image intensities. Our EMD-based super-resolution algorithm has outperformed comparing to some state-of-the-art super-resolution methods.

To enhance the quality of image denoising, we exploit the distribution of patches in the dictionary space as a an image prior to regularize the optimization problem. We develop a computationally efficient procedure, based on piece-wise constant function estimation, for low dimension dictionaries and then proposed a Gaussian Mixture Model (GMM) for higher complexity dictionary spaces. Finally, we justify the practical number of Gaussian components required for recovering patches. Our researches on multiple datasets with combination of different dictionaries and GMM models have complemented the lack of evidence of using GMM in the literature.

Keywords — Image denoising, image super-resolution, Bayesian Maximum A Posteriori, Gaussian Mixture Model, Earth Mover's Distance, Sparse models, patches distribution, example-based image restoration.

Contents

List of Tables

List of Figures

List of Algorithms

List of Notations

The notations used in this thesis follow the principles below.

i. The bold capital letters are used to represent a two-dimensional (2D) arrays, which can be a matrix (degradation matrix $\mathbf{H}$, downsampling matrix $\mathbf{S}$, etc.), a dictionary of K column vectors (e.g. $\mathbf{D}$) or a 2D image with pixels arranged in rows and columns such as $\mathbf{X}, \mathbf{Y}$.

ii. The bold letter, e.g. $\mathbf{x}$, $\mathbf{y}$, represents a column vector or a vectorization of a 2D image obtained by raster scan the image from top to bottom and left to right.

iii. The subscripts below a symbol stand for the subsets. For instance, $\mathbf{x}_i$ is a sub-column vector (also called image patch) of a vectorization image $\mathbf{x}$, $\mathbf{X}_i$ indicates a 2D array where each of its column is an image patch, etc.

iv. The superscripts express the modes or state of a variable. For example, $\mathbf{x}^s$ is a standard (high quality) image, $\mathbf{D}^l$ denotes a dictionary of low-resolution image patches, $\mathbf{D}^h$ a dictionary of high-resolution patches, etc.

v. The array indexing, which encloses the indices in parentheses, is used to access an element of the array. E.g. $\mathbf{X}(i, j)$ is the pixel at i-th row and j-th column of 2D image $\mathbf{X}$, $\mathbf{x}_i(j)$ is the j-th pixel of the column image patch $\mathbf{x}_i$, etc.

The list of important notations is described as follows.

R^n	A n-dimensional space.
$R^{n \times K}$	A space of two-dimensional arrays of n rows and K columns.
$\mathbf{X}$	An image represented as a two-dimensional (2D) array.
$\mathbf{X}(i, j)$	A pixel in the image $\mathbf{X}$ at i-th row and j-th column.
$\mathbf{x}$	An image represented as a column vector by raster scan the 2D image $\mathbf{X}$ from top to bottom, left to right.
$\mathbf{x}^s$	A vectorization of a good quality image (also called standard image).

$\mathbf{x}_i$	A column vector or an image patch represented as a column.
$\mathbf{x}_i(j)$	The j-th element (pixel) of the (column) image patch $\mathbf{x}_i$.
$\mathbf{D}$	A dictionary (in 2D array) of image patches.
Ω_α	A vector space in R^K generated by K basis vectors.
$\boldsymbol{\alpha}_i$	A vector of representation coefficients of an image patch $\mathbf{x}_i$ in the vector space Ω_α.
$\mathcal{K}(\cdot)$	A kernel function.
$\mathcal{N}(\mu, \sigma^2)$	Gaussian distribution with mean μ and variance σ^2.
PSNR	Peak signal-to-noise ratio.
SSIM	Structural similarity.
EMD	Earth Mover's Distance.
K-SVD	A dictionary learning algorithm for sparse representation which is a generalization of the K-means algorithm and is based on the singular value decomposition.
OMP	Orthogonal matching pursuit algorithm.
NE	Neighbor embedding algorithm.
ScSR	Sparse coding super-resolution algorithm.
SRSW	Super-resolution by sparse weight algorithm [1].

CHAPTER 1

Introduction

Chapter content

1 Introduction

Digital images undoubtedly play a huge part in many applications such as medical imaging, remote sensing, surveillance, and entertainment, etc, because they can convey and store the information that reflect the objects or environment. However, the images acquired from digital imaging systems are the degraded observations of the unknown clean images. The degradation comes from various factors, e.g. noise corruption, blurring, resolution limitation, object movement, or a combination of them. Therefore, image restoration (IR) is a fundamental task that aims to recover a latent high quality image $\mathbf{x} \in R^N$ from its degraded observation $\mathbf{y} \in R^M$. The IR problem can be modeled as follows:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \boldsymbol{\eta} \tag{1.1}$$

where $\mathbf{H}$ denotes the non-invertible degradation matrix and $\boldsymbol{\eta}$ is the Gaussian additive noise of mean zero and standard deviation σ. Depending on different settings of $\mathbf{H}$, the model (1.1) can represent various IR problems. For example, in image denoising, $\mathbf{H}$ is an identity matrix. For image deblurring, the matrix multiplication operation $\mathbf{Hx}$ is equivalent to a two-dimensional convolution $\mathbf{b} \otimes \mathbf{x}$ between a blur kernel $\mathbf{b}$ and the latent image $\mathbf{x}$. By selecting a diagonal matrix of $\mathbf{H}$ with the entries are either 0 or 1 where the zero values denote the missing pixels in the image $\mathbf{x}$, (1.1) becomes the image inpainting issue. Another popular configuration is the super-resolution (SR) problem where $\mathbf{H} = \mathbf{SB}$ is a composition of a blurring $\mathbf{B}$ and a downsampling $\mathbf{S}$ operators.

Image restoration is an ill-posed inverse problem because a lot of informations have been lost during the degradation which prevents a full recovery of the latent image $\mathbf{x}$. Several attempts have been made in the state-of-the-art to deal with the image restoration problem by suppressing the degradation factors and preserving as much as possible the image details. Due to the large space of image content, prior knowledge of image structure is crucial to regularize the inverse solution and obtain a reliable estimation $\hat{\mathbf{x}}$ of $\mathbf{x}$. However, learning priors and optimizing over whole image may lead to tremendous computational challenges. Instead, we select to handle the image restoration problem on small local patterns of image, which are called the image patches, in which the image priors can be learned more efficiently.

In the context of patch-based image restoration, each image is considered to be a set of overlapping image patches, and the reconstruction will be performed on each patch. The image restoration model in (1.1) can be formulated on each patch as follows.

$$\mathbf{y}_i = \mathbf{H}_i \mathbf{x}_i + \boldsymbol{\eta}_i \tag{1.2}$$

where $\mathbf{y}_i \in R^m$ is the i-th patch in the degraded image $\mathbf{y}$, $\mathbf{x}_i \in R^n$ is the high-quality latent version of $\mathbf{y}_i$, $\mathbf{H}_i$ is the degradation matrix on $\mathbf{x}_i$, and $\boldsymbol{\eta}_i$ is the residual noise in $\mathbf{y}_i$.

Without loss of generality, we can represent the latent image patch $\mathbf{x}_i$ by a linear combination of a set of K basis vectors (also called atoms) $\{\mathbf{d}_1, \ldots, \mathbf{d}_j, \ldots, \mathbf{d}_K | \mathbf{d}_j \in R^n\}$ that form a dictionary of patches $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_k] \in R^{n \times K}$. Let $\Omega_\alpha \subset R^K$ denote the vector space generated by the K atoms of the dictionary $\mathbf{D}$. Then, each image patch $\mathbf{x}_i$ can be described by a representation coefficients vector $\boldsymbol{\alpha}_i = [\boldsymbol{\alpha}_i(1), \ldots, \boldsymbol{\alpha}_i(j), \ldots, \boldsymbol{\alpha}_i(K)]^T$ such that $\mathbf{x}_i = \mathbf{D}\boldsymbol{\alpha}_i$. Therefore, the image restoration model in (1.2) becomes:

$$\mathbf{y}_i = \mathbf{H}_i \mathbf{D} \boldsymbol{\alpha}_i + \boldsymbol{\eta}_i \tag{1.3}$$

From the Bayesian framework, the restoration of an image patch $\mathbf{y}_i$ is equivalent

to estimating a representation coefficients vector $\boldsymbol{\alpha}_i$ that maximizes the posteriori probability $p(\boldsymbol{\alpha}_i|\mathbf{y}_i, \mathbf{D})$ as shown in

$$\hat{\boldsymbol{\alpha}}_i = \arg\max_{\boldsymbol{\alpha}_i} p(\boldsymbol{\alpha}_i|\mathbf{y}_i, \mathbf{D}) = \arg\max_{\boldsymbol{\alpha}_i} \{p(\mathbf{y}_i|\mathbf{D}, \boldsymbol{\alpha}_i)p(\boldsymbol{\alpha}_i)\} \tag{1.4}$$

where $p(\boldsymbol{\alpha}_i)$ is called the prior model of $\boldsymbol{\alpha}_i$ since it specifies *a priori* statistical features of image in the vector space Ω_α and is independent of $\mathbf{y}_i$. With the assumption of Gaussian noise corruption $\mathcal{N}(0, \sigma^2)$, the likelihood can be described as $p(\mathbf{y}_i|\mathbf{D}, \boldsymbol{\alpha}_i) \propto \exp\left(-\frac{1}{2}||\mathbf{y}_i - \mathbf{H}_i\mathbf{D}\boldsymbol{\alpha}_i||_2^2\right)$, and the Maximum A Posteriori (MAP) problem in (1.4) can be reformulated as

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\{ \frac{1}{2}||\mathbf{y}_i - \mathbf{H}_i\mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda \log p(\boldsymbol{\alpha}_i) \right\} = \arg\min_{\boldsymbol{\alpha}_i} \left\{ \frac{1}{2}||\mathbf{y}_i - \mathbf{H}_i\mathbf{D}\boldsymbol{\alpha}_i||_2^2 + \lambda\Phi(\boldsymbol{\alpha}_i) \right\} \tag{1.5}$$

where $\Phi(\boldsymbol{\alpha}_i) \propto -\log p(\mathbf{x})$ is called the regularization term and λ is the trade-off parameter which balances the two terms of (1.5). After solving the optimization problem in (1.5), we can obtain an estimation of the latent image patch, denoted by $\hat{\mathbf{x}}_i$, as $\hat{\mathbf{x}}_i = \mathbf{D}\boldsymbol{\alpha}_i$. Finally, the latent image $\mathbf{x}$ can be found by aggregating the overlapped regions between adjacent patches $\hat{\mathbf{x}}_i$.

In this thesis, we investigate the image restoration in the context of image denoising and super-resolution via solving the optimization problem in (1.5). The two main issues involving to find the solution of (1.5) are the determination of the dictionary $\mathbf{D}$ and the formulation of the prior model $p(\boldsymbol{\alpha}_i)$. The dictionary $\mathbf{D}$ contains atoms (patches) that will be directly used to recover the latent patch $\mathbf{x}_i$ and thus has effects on the reconstruction quality. Thus it is preferable to build a dictionary that can hold the similar information in the underlying patch. Additionally, the prior model $p(\boldsymbol{\alpha}_i)$ of the probability distribution of representation coefficients describes the manner that the latent image patch $\mathbf{x}_i$ will be constructed from the K atoms. In this thesis, we will alternatively discuss the influence of both factors on the reconstruction of a degraded image.

The remainder of this chapter is organized as follows. Section 2 presents the motivation and organization of our thesis. In section 3, we briefly describe some image quality assessment metrics used for the evaluation of an image restoration algorithm.

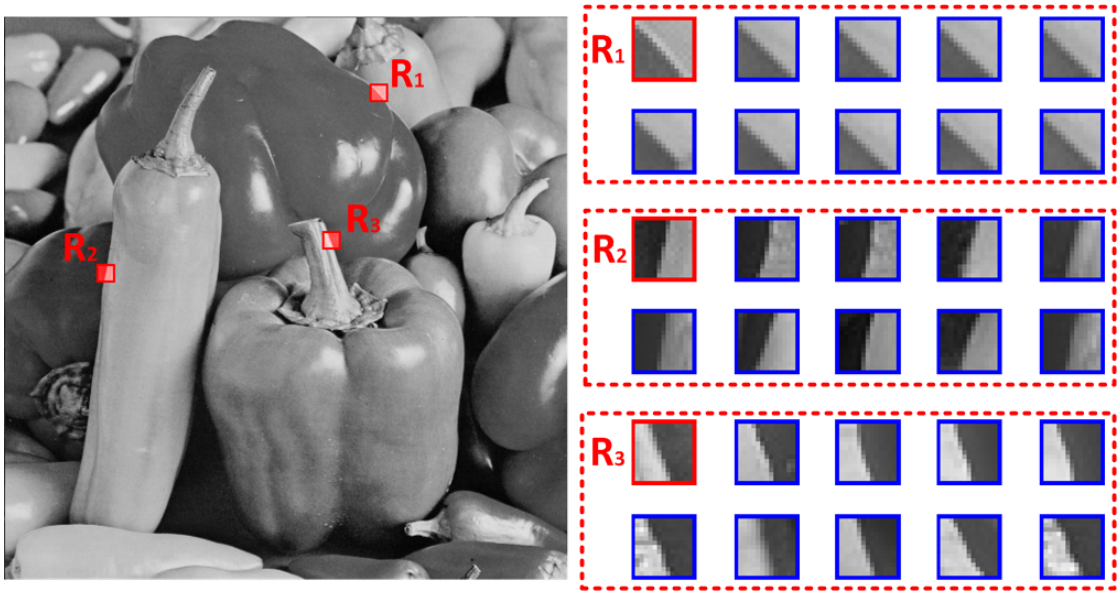
2 Motivation and thesis organization

Patch-based image restoration under the Maximum A Posteriori (MAP) framework has proved successful in the reconstruction of a degraded image [1–5]. The principal concept of this thesis is built upon an interesting property of digital images that relates to the

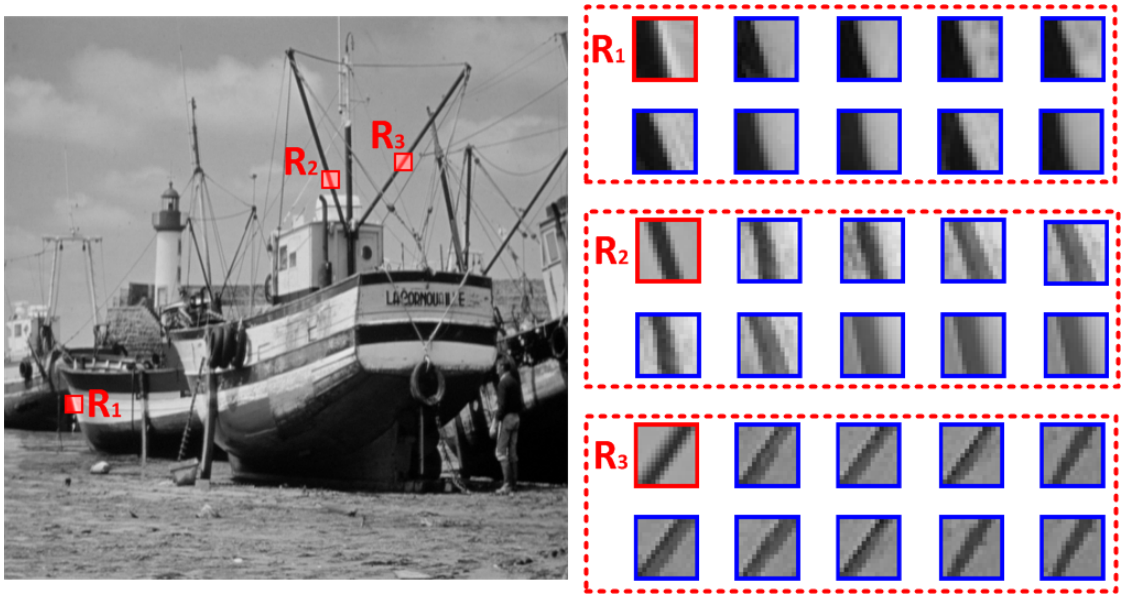
redundancy of information across different images, where the local patterns tend to repeat between multiple images. It is a fact that for a patch in an underlying image $\mathbf{x}$, we can find a list of similar patches in other external good quality (clean, high resolution) images $\{\mathbf{x}^s\}$, which are often used as references and referred to as the standard images or example images.

For a better demonstration, we present on the left of Fig. 1.1 some reference patches, which are marked by rose blocks with red contours, in four example images, including two natural images of Peppers and of Boat, one CT image of Lung and one MRI image of Brain. For standard images, we collected multiple datasets, including the Berkeley Segmentation dataset [6] of natural images, the TCIA dataset [7] of CT image of Lung, and the MRI image of Brain in [8]. For each dataset, we randomly collected 200000 image patches to create a database of patches. For each reference patch in each image, we searched for 9 closest query patches in the database and portrayed them in blue contour blocks as shown on the right of Fig. 1.1. We can observe that the databases established from the external images contain several patches with similar structures to the local patches in underlying images. Therefore, it is expected that we can exploit the useful information in the database of patches in recovering a degraded image patch $\mathbf{y}_i$. In particular, we can learn a dictionary $\mathbf{D}$ or analyze the statistical characteristics of the distribution of image patches $p(\boldsymbol{\alpha}_i)$ in the database and then used them in image restoration.

Our first research is in the context of super-resolution of images affected by Poisson noise and inspired by previous work in the LAGA and L2TI labs on the SRSW method [1]. The patches in the input image are transformed into the Anscombe domain so that the Poisson distribution is converted in approximately additive standard normal noise. Thus we can apply the optimization framework in (1.5) to restore each patch $\mathbf{y}_i$ in the noisy image. In this approach, we construct an exhaustive dictionary $\mathbf{D}$, which is identical with the database of patches, so that we may ensure that the dictionary can cover the local information appearing in the noisy image. However, solving the problem (1.5) with a very large number of atoms in the dictionary requires tremendous computational complexity and makes the implementation challenging. To deal with this issue, a practical solution is to apply a patch-selection step on the dictionary $\mathbf{D}$ in order to choose the most appropriate atoms for the reconstruction. Hence, for each noisy patch $\mathbf{y}_i$, we exploit the Euclidean-based measurement to determine a set of similar pairs of low and high resolution patches in the database to generate a local reproducing kernel Hilbert space. The high resolution patch is recovered based on a linear regression of similar patches in the Hilbert space. We perform experiments on synthetic images

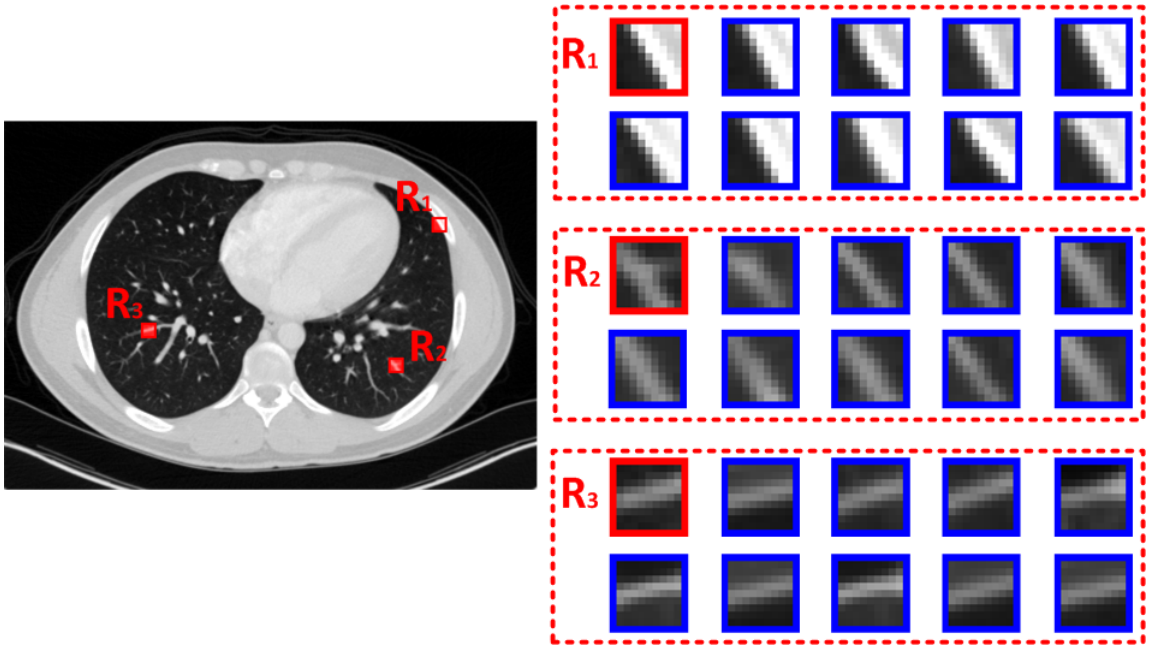


(a) Image of Peppers

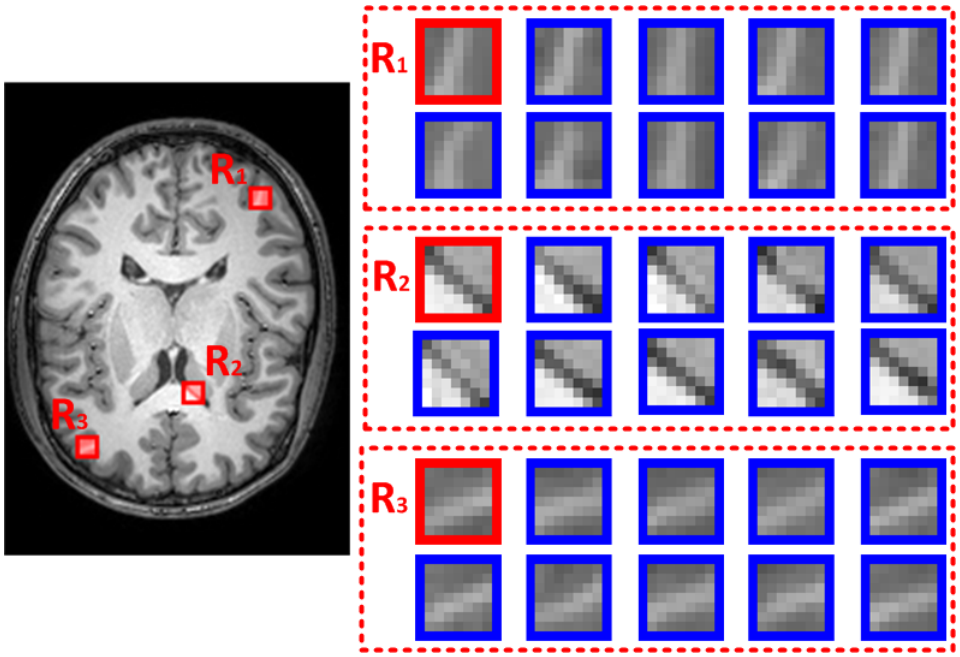


(b) Image of Boat

Figure 1.1: *Illustration of external similarity between multiple images. Referenced patches in images are marked by the rose blocks with red contours, and a few of their matched patches selected from standard images are denoted by blue contour blocks.*



(c) CT image of Lung



(d) MRI image of Brain

Figure 1.1: *Illustration of external similarity between multiple images. Referenced patches in images are marked by the rose blocks with red contours, and a few of their matched patches selected from standard images are denoted by blue contour blocks.*

and medical image dataset and demonstrate the outperforming of the proposed method comparing with some existing super-resolution methods. The details of this work will be reported in chapter 3.

Our second work is an expansion of the first study and the method SRSW [1] for super-resolution of an image corrupted by additive Gaussian noise. Similar to the above approach, we construct a large, exhaustive dictionary to cover as much as possible the local patterns in an input noisy and low-resolution image. However, we study an alternate strategy for selecting the local dictionary (a set of smaller number of atoms) for image super-resolution. Instead of taking into account the classical Euclidean distance as originally designed in SRSW and our previous work, as well as in many super-resolution methods in the literature, we consider an image patch as a distribution of grey levels and propose to exploit the Earth Mover's Distance (EMD) to measure the similarity between image patches. We show that the EMD better describes the perceptual similarity between patches than the Euclidean distance and helps to improve the searching accuracy and efficiency. Moreover, we introduce an ℓ_1-norm-based threshold for each low-resolution noisy patch $\mathbf{y}_i$, which is built on the characteristic of the EMD, to facilitate the searching process and dynamically select the number of similar patches for constructing the local dictionary. Finally, the degraded patch $\mathbf{y}_i$ is restored via solving (1.5) with the obtained local dictionary and the assumption of the sparse model of the probability distribution $p(\boldsymbol{\alpha}_i)$. We carry out several experiments on medical images with different modalities to demonstrate that the super-resolution with EMD is more efficient than the classical Euclidean-based super-resolution methods, in both case of noise-free and noise-corrupted images. We will present this work in further details in chapter 4.

In the two previous works, we have created a dictionary $\mathbf{D}$ by purely extracting a large number of patches in the standard images. Then to solve the restoration problem in (1.5), we first apply a patch-selection strategy to collect a local dictionary of smaller set of similar atoms for each degraded patch $\mathbf{y}_i$. After that, the latent patch $\mathbf{x}_i$ can be recovered using the obtained local dictionary with the assumption of sparsity model for $p(\boldsymbol{\alpha}_i)$. In the conventional sparse representation methods, the distribution of image patches in the vector space generated from the atoms in a local dictionary is assumed to be sparsely distributed and can be characterized in a smaller dimensional subspace. However, the choice of sparse models is too arbitrary and they cannot exactly represent the true distribution of image patches. In our third work in this thesis, we will analyze the distribution of representation coefficients $p(\boldsymbol{\alpha}_i)$ of patches in the vector space generated by atoms of a dictionary. Moreover, to make convenient for studying

the distribution of patches, we train a unique dictionary $\mathbf{D}$ and apply for all noisy image patches rather than using the local dictionary as in the first two researches. By exploiting the redundancy of local information between multiple images, we assume that the distribution of representation coefficients of patches in the database and that of latent patches $\mathbf{x}_i$ in the latent image $\mathbf{x}$ have similar forms. Hence, instead of adopting an arbitrary sparse prior, we propose to estimate the prior probability distribution $p(\boldsymbol{\alpha}_i)$ from the empirical distribution of patches in the database and then apply it to regularize the denoising problem via (1.5). In the scope of this work, we adopt a simple scheme to estimate the prior probability of $p(\boldsymbol{\alpha}_i)$ via a construction of a cell-wise constant histogram. This simple model for $p(\boldsymbol{\alpha}_i)$ allows us to solve the optimization problem efficiently with application to image denoising. To demonstrate the potentiality of the proposed approach, we study a toy problem in three dimensions. By carrying out experiments on synthetic and natural images and making the comparison with existing sparsity models, we justify the applicable capacity of the estimation-based probability in dealing with an image denoising task. The proposed method will be discuss in detail in chapter 5.

In our last contribution in this thesis, we develop the idea introduced in chapter 5 by exploring Gaussian Mixture Models (GMMs) to represent the prior model of distribution of image patches instead of the piecewise constant model. Gaussian mixture models allow us to model the distribution of patches with more details and more regularity. However, solving the denoising problem with the whole GMM prior is inefficient because of high computational complexity. For favorable implementation in practice, the mixture model is trained with all its components, but only one prominent component is used for reconstructing each noisy patch. To our knowledge, justification for this approach is lacking in the literature. Therefore, we attempt to verify this strategy on several image datasets by evaluating the number of Gaussian components required for recovering patches. Our contribution is a comprehensive assessment of the number of useful components in the GMM for patch-based image denoising. We perform extensive simulations for a combination of two dictionary choices, including an identity and a K-SVD-based [2] matrices, and two model complexities of GMM, e.g. a small model and a large one. We conduct several experiments on 8 image datasets, with diversity of modalities and image structures, and study denoising with increasing number of components, which is in range $\{1, 5, 10, 15, 20\}$. We show that all patches in the degraded image can be recovered by only one prominent component with little loss of performance. The interesting outcomes of our studies make strong evidence-based justification for the current practical use of GMM in the literature and drastically reduce computational

cost. The details of this work is introduced in chapter 6.

The structure of this thesis is organized as follows. In chapter 2, we will present the concepts of image restoration methods, especially the existing works that inspired our researches during my thesis. Chapter 3 introduces the super-resolution method for images corrupted by Poisson noise under an exhaustive dictionary of patches and reproducing kernel Hilbert space. In chapter 4, we propose to exploit the Earth Mover's Distance in the pre-filtering step for similar image patches selection in the large dictionary to facilitate the super-resolution process. In chapter 5 we introduce an estimation-based framework for learning the prior distribution of image patches from the standard images and study its applications in image denoising. In chapter 6, we investigate the study of Gaussian mixture model for estimating the image prior in image denoising by analyzing the number of useful Gaussian components in the reconstruction of a noisy image patch. We conclude this thesis and provide some future perspectives in chapter 7.

3 Image quality assessment metrics

Typically, we can not recover the exact unknown original image $\mathbf{X} \in R^{N_1 \times N_2}$ from its degraded version $\mathbf{Y}$ because of the irreversible degradation process. By adopting an image restoration algorithm, we can obtain the reconstructed image $\hat{\mathbf{X}} \in R^{N_1 \times N_2}$ from the observation $\mathbf{Y}$, which is considered as an estimation and a distorted version of the latent image $\mathbf{X}$ and is expected as close as possible to $\mathbf{X}$. In order to evaluate the quality of the reconstructed image $\hat{\mathbf{X}}$, and compare the performances of different competition image restoration methods, we need to use some image quality assessment (IQA) metrics.

Several attempts have been made in the literature to develop good metrics to measure the image quality for different applications such as image denoising, deblurring, super-resolution. Image quality assessment methods can be divided into two categories: subjective assessment and objective assessment. The subjective IQA methods are accurate in reflecting the human perception of the visual quality of an image because they are carried out by human beings. In practice, however, subjective evaluation is usually inconvenient due to the requirement of a large number of observers, time-consuming and can not be an automated process.

On the other hand, the objective validations are computer-based methods that can automatically predict the perceived image quality. Moreover, according to whether the reference image is available or not, the objective assessment methods can be classified

into full-reference metrics or no-reference metrics. In the full-reference approaches, the original good quality image $\mathbf{X}$, which is referred to as reference image, is assumed to be known a priori. After that, we obtain a simulated degraded image $\mathbf{Y}$ through the degradation process and apply a restoration method to get the reconstruction image $\hat{\mathbf{X}}$. In our works, we adopt two objective full-reference metrics, namely the peak signal-to-noise ratio (PSNR) and the structural similarity index (SSIM) [9], to measure the quality of the recovered image $\hat{\mathbf{X}}$ by quantifying the distortion between $\hat{\mathbf{X}}$ and the reference image $\mathbf{X}$. The rest of this section will make a summary on these two metrics.

3.1 Peak signal-to-noise ratio (PSNR)

The PSNR is the most commonly used image quality assessment metric for many image restoration task including denoising and super-resolution. The PSNR are mostly defined via the mean square error (MSE) between the reference image $\mathbf{X}$ and the reconstructed image $\hat{\mathbf{X}}$, which is determined by the ℓ_2-distance:

$$MSE(\mathbf{X}, \hat{\mathbf{X}}) = \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \Big(\mathbf{X}(i,j) - \hat{\mathbf{X}}(i,j)\Big)^2 \tag{1.6}$$

where $\mathbf{X}(i,j)$ denotes the pixel at i-th row and j-th column of the image $\hat{\mathbf{X}}$. The definition of PSNR can be formulated as follows:

$$PSNR(\mathbf{X}, \hat{\mathbf{X}}) = 10 \log_{10} \left(\frac{MAX_X^2}{MSE(\mathbf{X}, \hat{\mathbf{X}})} \right) = 20 \log_{10} \left(\frac{MAX_X}{\sqrt{MSE(\mathbf{X}, \hat{\mathbf{X}})}} \right) \tag{1.7}$$

where MAX_X is the maximum possible pixel value of the image, i.e. $MAX_X = 2^8 - 1 = 255$ for 8-bit images. The PSNR is a simple yet effective metric to describe the gray-level differences between two images. However, its fails to consider the structure distortions of images, which relate to the human visual perception because higher PSNR does not mean higher visual structural similarity between two images. Hence, many researchers have made efforts on finding alternative and better IQA metrics.

3.2 Structural similarity index (SSIM)

In [9], Wang *et al.* proposed the Structural SIMilarity (SSIM) metric based on the assumption that the human visual system is more sensitive to structural information. The SSIM metric is very effective in evaluation the perceptual distortion between two images and has been adopting by the image processing community since the time it was introduced.

The SSIM value between two images is calculated by averaging the similarity index between two local image patches $\mathbf{x}_i, \mathbf{y}_i \in R^n$ taken from the same location of two

images that are being compared. The local SSIM index measures the similarities of three elements of the image patches: luminance, contrast and structure as indicated in (1.8), (1.9) and (1.10).

$$\ell(\mathbf{x}_i, \mathbf{y}_i) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \tag{1.8}$$

$$c(\mathbf{x}_i, \mathbf{y}_i) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \tag{1.9}$$

$$s(\mathbf{x}_i, \mathbf{y}_i) = \frac{2\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3} \tag{1.10}$$

where μ_x, μ_y are the means of the patches $\mathbf{x}_i$ and $\mathbf{y}_i$, respectively; σ_x, σ_y the standard deviations of $\mathbf{x}_i$ and $\mathbf{y}_i$, and $\sigma(\mathbf{x}_i, \mathbf{y}_i) = \frac{1}{n-1}\sum_{j=1}^{n}(\mathbf{x}_i(j) - \mu_x)(\mathbf{y}_i(j) - \mu_y)$ the covariance of the two image patches $\mathbf{x}_i$ and $\mathbf{y}_i$. c_1, c_2, c_3 are positive constant to stabilize the division with weak denominator. Then, the SSIM index between two image patches was defined as a weighted combination of three comparative measures.

$$SSIM(\mathbf{x}_i, \mathbf{y}_i) = \ell(\mathbf{x}_i, \mathbf{y}_i)^\alpha \cdot c(\mathbf{x}_i, \mathbf{y}_i)^\beta \cdot s(\mathbf{x}_i, \mathbf{y}_i)^\gamma \tag{1.11}$$

where α, β, γ are positive constant. In this thesis, we follow the configuration proposed in [9] by setting $\alpha = \beta = \gamma = 1$, $c_1 = (k_1 L)^2$, and $c_2 = 2c_3 = (k_2 L)^2$, with L denote the dynamic range of the pixel values ($L = 255$ for 8-bit images), $k_1 = 0.01$ and $k_2 = 0.03$ are two constants determined by the authors to avoid instability in homogeneous regions. Finally, Wang *et al.* get the specific form of the SSIM between two corresponding patches in two images as:

$$SSIM(\mathbf{x}_i, \mathbf{y}_i) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{1.12}$$

CHAPTER 2

State-of-the-art image restoration

Chapter content

In this chapter, we first make a comprehensive resume of the development of the image restoration methods in the state-of-the-art in section 1. After that, we will present some background information and preliminary knowledge of direct relevance to our researches in this thesis in section 2.

1 State-of-the-art Image Restoration

The history of image restoration began very early when image quality enhancement started to attract much attention in many applications of satellite imagery and remote sensing in the years 1970s. Several contributions have been made in the literature in solving the image restoration problem in (1.1), which aim to estimate the latent image $\mathbf{x}$ from its degraded observation $\mathbf{y}$ such that the original information like edges, image details, etc., can be preserved.

Image restoration techniques can be categorized into three main groups: the local filtering-based methods, the model-based methods and the discriminative learning-based methods. In this chapter, we make a comprehensive overview on the state-of-the-art of image restoration.

1.1 Filtering-based image restoration

An early and simple image restoration approach is known as the local filtering-based methods because they estimate the true value of each pixel in the latent image $\mathbf{x}$ by using a weighted combination of its around neighbor pixels found in a local windows in the degraded image $\mathbf{y}$. The weights are often specified by the coefficients of a spatial filter kernel, whose values are dependent on the spatial distance between two pixels, or by an interpolation function.

In the last few decades, many authors developed various simple local kernels for image restoration applications such as Gaussian filter [10] in image denoising, fixed-function kernels like nearest, bilinear and bicubic interpolation [11, 12] in image super-resolution, etc. Later, several researchers investigated structure adaptive filters with regularized weights and sizes such as bilateral filter [13], anisotropic filtering [14], SUSAN filter [15], least-mean-square adaptive filter [16], steering kernel regression [17], and edge-guided interpolation kernels [18, 19] to better preserve image structures such as edge, corner, etc., and thus enhance the reconstruction results.

In spite of the low complexity and easy implementation, the major drawback of local filtering methods is that they are very sensitive to heavily degraded factors such

as high noise level or large upscaling factor because the correlation between neighbor pixels has been severely corrupted and thus reduce the accuracy of the estimation of pixels in latent image.

1.2 Model-based image restoration

Exploiting only the information of pixels in a local window of the degraded image $\mathbf{y}$, as proposed in the local filtering-based methods, is not enough to produce a reliable estimation of the pixels in the latent image $\mathbf{x}$. In order to achieve a better reconstruction, we need to be provided with some prior knowledge about some properties of the latent image such as the statistical distribution in image space to regularize the restoration process.

Under the Bayesian MAP framework and the assumption of additive Gaussian noise corruption on image, the image restoration problem is described as:

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}||\mathbf{y} - \mathbf{H}\mathbf{x}||_2^2 - \lambda \log\left(p(\mathbf{x})\right) \right\} \tag{2.1}$$

$p(\mathbf{x})$ is called the prior model of $\mathbf{x}$ because it characterizes the distribution of the latent image in a specific space and is independent of the observation $\mathbf{y}$. Several efficient image priors have been proposed in the literature for solving the image restoration problem in (2.1). Many researchers investigated the prior models of the whole image by studying some properties of images in a specific domain.

One characteristic of the images is that their total variations (TV), which measure how much the image intensities change between the adjacent pixels and can be defined as the sum of absolute of the image gradient, are often small with good quality images. When an image is degraded, e.g. corrupted by noise, its TV will dramatically increase. A representation class of image restoration method, named TV-based, exploits the TV of images as a prior model of $p(\mathbf{x})$ to regularize the TV of the degraded image $\mathbf{y}$, with the aim of matching the TV of reconstructed image close to that of latent image $\mathbf{x}$ [20].

Another approach is to study the distribution of images in the wavelet domain which is generated by a set of fixed basis. An image will be represented by the wavelet coefficients in the wavelet domain. Several studies [21–24] have been proposed in the literature to learn a prior model from the distribution of wavelet coefficients of the images.

Learning image priors and performing restoration over the whole image may require tremendous computational effort. Instead, a considerable number of researchers attempted to work with small patterns in the images, which are known as image patches.

By observing the properties and distribution of patches in several images, numerous priors of image patches have been proposed in the state-of-the-art. For example, the nonlocal self-similarity and the low-rank approximation are based on the repetition of patches in different places within an image. The sparse representation and Gaussian mixture model are built to represent the distribution of patches in a vector space generated by a set of fixed atoms. Additionally, there are many endeavors to combine available priors to benefit their advantages and lead to more effective image restoration performances.

1.2.1 Total variation

Total variation (TV) is a well-known, early developed class of methods for image restoration. The TV of an image $\mathbf{x}$ relates to the gradient of image and can be formulated as:

$$TV(\mathbf{x}) = \|\boldsymbol{\mathcal{D}}\mathbf{x}\|_1 \tag{2.2}$$

where $\boldsymbol{\mathcal{D}}$ denotes the gradient operator.

In the pioneering work of TV restoration, Rudin *et al.* [20] exploited the TV as a prior model $p(\mathbf{x})$ of the image to regularize the denoising process and minimize the TV of the reconstructed image. More specifically, the problem in (2.1) was written as

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}||\mathbf{y} - \mathbf{H}\mathbf{x}||_2^2 + \lambda||\boldsymbol{\mathcal{D}}\mathbf{x}||_1 \right\} \tag{2.3}$$

Many algorithms have been developed for solving the problem in (2.3), including dual formation [25], Newton-based method [26], alternating direction method [27], split Bregman algorithm [28], etc.

One of the shortcoming of the TV methods is the appearance of staircasing artifacts in slanted regions. Many researchers [29–32] modified the TV model by adding variety of higher-order derivatives of image to reduce the staircasing effects and produced more pleasant reconstruction results. Another drawback of the TV is that it does not consider the orientation of image gradients and thus is not very suitable for images with rich textures where the edges and contours exhibit dominant directions. To overcome this issue, Bayram and Kamasak [33] proposed the directional total-variation (DTV) by introducing the weights in the image gradient coefficients depending on their directions. The strategy of Bayram significantly improved denoising performance on nature texture images with dominant direction. Later, Wang *et al.* [34] exploited the DTV in the image super-resolution and obtained promising results.

The TV regularization showed its effectiveness for restoring of piecewise-smooth

regions while having the capacity of preserving the sharp edges in the image. However, due to the minimization of TV, the small details in degraded image can also be considered as noise and thus will be remove during the reconstruction. In order to better preserve the image details, a combination of the variational approach and another prior model such as the nonlocal self-similarity [35, 36] has been proposed in recent years and achieved promising performances in image restoration.

1.2.2 Wavelet-based image restoration

A critical landmark in the evolution of image restoration was the development of the wavelet theory in the late of 1980s, in which an image is transformed into a wavelet domain generated by a set of orthogonal basis functions, which are built from a mother wavelet with different dilations and translation. Let $\mathbf{W} \in R^{N\times N}$ denote the orthogonal matrix constituted of the wavelet basis. In the wavelet domain, an image $\mathbf{x} \in R^N$ is represented by the wavelet coefficients $\boldsymbol{\alpha}_w$ determined as $\boldsymbol{\alpha}_w = \mathbf{W}\mathbf{x} \in R^N$, which is commonly divided into a low-frequency sub-band (which corresponds to the low-frequency contents of images such as smooth regions, blurred structure) and a set of high-frequency sub-bands (that convey the high-frequency information in image like edges, texture, and noise, which are the most sensitive to human vision). Typically, the image restoration is performed on the high-frequency sub-bands.

Instead of studying the prior model $p(\mathbf{x})$ in the image domain, many researchers investigate in analyzing the distribution of wavelet coefficients and thus introducing prior models in the wavelet domain. A considerable success in application of wavelet transform is in the image denoising. The noising model can be derived from (1.1), with $\mathbf{H}$ is an identity matrix, as $\mathbf{y} = \mathbf{x} + \boldsymbol{\eta}$. Applying the wavelet transform on these images, we have:

$$\mathbf{W}\mathbf{y} = \mathbf{W}\mathbf{x} + \mathbf{W}\boldsymbol{\eta} = \boldsymbol{\alpha}_w + \mathbf{v} \tag{2.4}$$

where $\mathbf{W}\mathbf{y} \in R^N$ is the wavelet coefficients of the noisy image, $\boldsymbol{\alpha}_w$ and $\mathbf{v}$ are the wavelet coefficients of the unknown clean image $\mathbf{x}$ and the residual noise, respectively. The objective of wavelet-based denoising is to estimate $\hat{\boldsymbol{\alpha}}_w$ from the wavelet transform $\mathbf{W}\mathbf{y}$ of the noisy image. Finally, the latent image $\mathbf{x}$ can be recovered by applying the inverse transform $\hat{\mathbf{x}} = \mathbf{W}^{-1}\hat{\boldsymbol{\alpha}}_w$. Under the Bayesian perspective and MAP framework, we have:

$$\hat{\boldsymbol{\alpha}}_w = \arg\max_{\boldsymbol{\alpha}_w}\{p(\boldsymbol{\alpha}_w|\mathbf{W}\mathbf{y}, \mathbf{v})\} = \arg\max_{\boldsymbol{\alpha}_w}\{p(\mathbf{W}\mathbf{y}|\boldsymbol{\alpha}_w, \mathbf{v}) + p(\boldsymbol{\alpha}_w)\} \tag{2.5}$$

Determining a priori model $p(\boldsymbol{\alpha}_w)$ on the statistic properties of wavelet coefficients is crucial for solving the problem in (2.5). Several attempts [21, 23, 24, 30] have been made

in the literature to learn the prior model of $p(\boldsymbol{\alpha}_w)$ based on the observation of wavelet coefficients of multiple standard images. A fascinating characteristic of the wavelet transform is that the wavelet coefficients $\boldsymbol{\alpha}_w$ of a clean image, in the high-frequency sub-bands, often exhibit non-Gaussian behavior with a sharp peak centered around zero and heavy-tailed distribution, which is close to sparse models. That means, only a small portion of wavelet coefficients of image having high-magnitude (at the tails of the distribution) which accounts for most energy of the image, while a large ratio of coefficients (around center of distribution) are very small or close to zero. In practice, the low-magnitude wavelet coefficients relate to noise and small details in image. Therefore, by performing a shrinkage to eliminate small wavelet coefficients, we can remove the noise (and certainly, the small details) from the image.

Many studies have been carried out on exploiting mathematical models to represent the sparse distribution of wavelet coefficients such as the Laplacian model [21, 37], the generalized Gaussian distribution [22], the Gaussian scale mixture model [23] or bivariate distributions [24]. Based on the proposed sparsity models, numerous shrinkage strategies were also introduced for noise reduction, e.g. VisuShrink [21], SureShrink [37], BayesShrink [22], BiShrink [24].

The wavelet transform is also exploited in image super-resolution [38–42]. A common feature of wavelet-based super-resolution reconstruction is the assumption, without loss of generality, that the input low-resolution image $\mathbf{y}$ is considered as the low-frequency sub-band of the wavelet decomposition of the latent high-resolution image $\mathbf{x}$. Therefore, the main purpose is to reconstruct the other missing high-frequency spectra of the latent image $\mathbf{x}$ and then apply the inverse wavelet transform ($\mathbf{W}^{-1}$) to obtain the estimation of $\mathbf{x}$. One representative approach is to deploy the patch-based representation methods such as the sparse representation or external nonlocal self-similarity, which will be discussed in the next sections, to reconstruct the wavelet coefficients in the high-frequency sub-bands.

The wavelet-based methods have shown their efficiency for image restoration, in particularly for denoising and super-resolution. However, when using shrinkage methods to reject the small wavelet coefficients, we not only separate the noise from the image, but also discard the small details in the image. As a result, the reconstructed image can be oversmoothed or smeared out the details.

1.2.3 Sparsity-based learning image restoration

Developing an effective prior over the whole image and then solving the restoration problem is a challenge because of high computation complexity. Another high-regarded

class of method is to study the image restoration problem on image patches, from which the learning of image priors, as well as the optimization process can be more easily performed. In the last decade, the patch-based image restoration methods has been received remarkable research interest. In the context of this thesis, we also concentrate on studying the image restoration problem using the patch-based approaches.

In these methods, an image is partitioned into a set of overlapping image patches, where each patch can be considered as an independent signal or in a correlation with other similar patches in the image. By this way, the reconstruction will be performed on each patch or on collaborative neighborhood patches. Let denote $\{\mathbf{y}_i|\mathbf{y}_i \in R^m; i = 1,\dots,M\}$ is the set of M overlapping patches in the degraded image $\mathbf{y}$. Our aim is to find a latent version $\mathbf{x}_i \in R^n$ of each patch $\mathbf{y}_i$ that satisfies:

$$\mathbf{y}_i = \mathbf{H}_i\mathbf{x}_i + \boldsymbol{\eta}_i \tag{2.6}$$

where $\boldsymbol{\eta}_i$ is the additive noise in the patch $\mathbf{y}_i$, and $\mathbf{H}_i$ denotes the degradation matrix on $\mathbf{x}_i$.

Without loss of generality, the motivation of patch-based methods is to represent each latent image patch $\mathbf{x}_i$ by a linear combination of K basis vectors, which are also called atoms, $\{\mathbf{d}_1,\dots,\mathbf{d}_j\dots,\mathbf{d}_K|\mathbf{d}_j \in R^n\}$ of a dictionary of patches $\mathbf{D} \in R^{n\times K}$. This means $\mathbf{x}_i = \mathbf{D}\boldsymbol{\alpha}_i$, where $\boldsymbol{\alpha}_i \in R^K$ is called the representation coefficients vector of $\mathbf{x}_i$. The problem in (2.6) can be rewritten as

$$\mathbf{y}_i = \mathbf{H}_i\mathbf{D}\boldsymbol{\alpha}_i + \boldsymbol{\eta}_i \tag{2.7}$$

The restoration of a degraded image patch $\mathbf{y}_i$ is equivalent to estimate a coefficients vector $\boldsymbol{\alpha}_i$ that satisfies the degradation model. Under the MAP framework and Gaussian noise assumption, we have:

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i}\{||\mathbf{y}_i - \mathbf{H}_i\mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda\log\left(p(\boldsymbol{\alpha}_i)\right)\} = \arg\min_{\boldsymbol{\alpha}_i}\{||\mathbf{y}_i - \mathbf{H}_i\mathbf{D}\boldsymbol{\alpha}_i||_2^2 + \lambda\Phi(\boldsymbol{\alpha}_i)\} \tag{2.8}$$

In the last ten years, the sparse representation is one of the most highly regarded class of methods for tackling patch-based restoration of images. The principal concept of this approach is based on an assumption that each image patch $\mathbf{x}_i$ can be described as a linear combination of few atoms in the dictionary [2, 43–45]. This leads to the prior model of $\boldsymbol{\alpha}_i$ in (2.8) which can be formulated as $\Phi(\boldsymbol{\alpha}_i) = ||\boldsymbol{\alpha}_i||_0$, where the pseudo norm ℓ_0 counts the number of non-zero elements of the vector. Another well-known approach is to assume that the distribution of representation coefficients $\boldsymbol{\alpha}_i$ in the vector space of dictionary $\mathbf{D}$ exhibits the heavy-tailed forms and thus can be fitted by some

well-known mathematical distributions such as the Laplacian model [1, 3, 46–50], with $\Phi(\boldsymbol{\alpha}_i) \propto \lambda||\boldsymbol{\alpha}_i||_1$, or the hyper-Laplacian model [51–57], which leads to $\Phi(\boldsymbol{\alpha}_i) \propto \lambda||\boldsymbol{\alpha}_i||_p$ with $0 < p < 1$. Likewise, the collaborative representation with ℓ_2-norm regularized least squares, $\Phi(\boldsymbol{\alpha}_i) \propto \lambda||\boldsymbol{\alpha}_i||_2$, has been studied in many image processing task such as recognition [58], super-resolution [59–62] and achieve competitive performances.

Though being useful and popular, ℓ_0 or ℓ_1 based sparsity priors have several limitations, e.g., they fail to capture complex sparse structures. As a result, the elastic net has been proposed [63], in which the regularization constraint can be written as $R(\boldsymbol{\alpha}_i) = \lambda_2||\boldsymbol{\alpha}_i||_2^2 + \lambda_1||\boldsymbol{\alpha}_i||_1$, where the ℓ_1 part ($||\boldsymbol{\alpha}_i||_1$) generates a sparse model, and the ℓ_2 term ($||\boldsymbol{\alpha}_i||_2^2$) encourages group selection. The elastic net is also convex, which can also be efficiently solved by many solvers, e.g., LARS-EN [64].

Despite the impressive performances in image restoration applications, one shortcoming of the sparse models is that they may fail to describe the true distribution of image patch in the vector space generated by atoms of the dictionary $\mathbf{D}$. Therefore, exploring a prior model which can more accurately present the true distribution of image patches can help to improve the quality of image recovering. In chapter 5 and 6, we investigate mathematical models of piece-wise constant function and Gaussian mixture model to precisely characterize the statistical distribution of patches and apply to restore a degraded image.

1.2.4 Nonlocal self-similarity

One of the most remarkable priors for patch-based image restoration is the so-called nonlocal self-similarity (NSS), which is motivated by the rich redundancy of local informations within an image. Based on the fact that for each patch, we can find many similar versions across the whole image, Buades *et al.* [65, 66] proposed a pioneer work named nonlocal means (NLM) to gather repeating structures in a given image and perform a weighted filtering to suppress the noise. The basic idea of NLM method is that each pixel can be restored by a weighted average of other pixels over a large window in the degraded image, where the weights are determined by the similarity between the local patches surrounding these pixels. The NLM filter is able to remove additive white noise while preserving sharp edges and fine texture details and has attracted significant attention in the image processing community at the time of its introduction. Recently, several variants of NLM have been proposed [67–70] to improve the adaptivity of the nonlocal filter.

Image restoration methods with NSS priors have achieved competing performance

in the literature. However, they have some drawbacks. First, the recovering of each degraded patch is built on the weighted graph based on the similarity between image patches, which can be disturbed in many cases, e.g. heavy noise corruption. Thus it can produce inaccurate weights and lead to inexactness in reconstruction. Another shortcoming of this approach is that the performance highly depends on the number of matched patches used for reconstruction. To overcome these weakness, many researchers [1, 4, 61, 62, 71–73] have investigated a combination of nonlocal self-similarity and sparsity priors in the same framework and attained impressive results. We will have a brief discussion on these approaches in section 1.2.7.

A brilliant variant of the nonlocal self-similarity is called the external similarity, which is based on the fact that the similar local information can be found in different images, as demonstrated in chapter 1. Several studies have been made in the literature to exploit the redundancy across images for image restoration. In [74], Chang *et al.* prepared a database of low- and high-resolution image patches from a set of good quality images (called standard images) for image super-resolution. Each patch in the low-resolution image can be approximated by a weighted average of its matched patches retrieved from database. After that, these weights are used to estimate the latent high-resolution patch. The concept of the external similarity is also exploited in recent researches in the image super-resolution [61, 62, 73] where the database of patches are divided into groups and the mappings between low- and high-resolution feature spaces are constructed on each group with the helps of sparse models. Another remarkable approach is to learn a prior model or study a property (e.g. the statistical distribution) of image patches in a database extracted from the standard images. Due to the redundancy of image contents, we can expected that every patch in the reconstructed image is likely followed the prior of patches in the database. Therefore, we can use the learned prior to regularize the image restoration process and recover a degraded image. A glorious contribution in this approach is to take advantage of the mixture model to simulate the probability distribution of image patches, and will be discuss in subsection 1.2.6.

In chapters 3 and 4 of this thesis, inspired from the framework in [1, 74], we exploit the external nonlocal similarity to establish a large dictionary of patches randomly extracted from a set of standard images and then select a subset of similar patches from the dictionary to reconstruct each patch in the degraded image. After that, we investigate the use of some mathematical models such as the piece-wise constant function (chapter 5) or the Gaussian mixture model (chapter 6) to estimate a prior image model from the database of patches and then adopt them to restore a degraded image. The

external nonlocal similarity and sparsity are the core ideas of our work.

1.2.5 Low-rank approximation

Another remarkable image prior derived from the nonlocal self-similarity property of images, is known as the low-rank approximation, which aims to recover an underlying low rank matrix from its degraded observation. The low rank terminology means that the observed data can be represented in a lower dimensional subspace with some different outliers corruption sources.

Recently, patch-based low-rank minimization methods for image/video denoising have achieved great success [75–82]. For each patch $\mathbf{y}_i$ in the underlying image $\mathbf{y}$, a set of its nonlocal similar patches across the image is first retrieved. Then these similar patches (including $\mathbf{y}_i$) are stacking into a matrix $\mathbf{Y}_i$, where each column of $\mathbf{Y}_i$ corresponds to an image patch. The degraded model is described on each group of patches as:

$$\mathbf{Y}_i = \mathbf{X}_i + \mathbf{V}_i \tag{2.9}$$

where $\mathbf{X}_i$ is a matrix containing clean versions of patches (columns) in $\mathbf{Y}_i$ and $\mathbf{N}_i$ is the residual noise in these patches. It is intuitive that the matched patches in $\mathbf{Y}_i$ have close image structures, thus the noiseless version $\mathbf{X}_i$ should lie in a low dimensional subspace. The denoising problem is performed on each of group of patches $\mathbf{Y}_i$ under the constraint of minimization of the rank of $\mathbf{X}_i$ as:

$$\hat{\mathbf{X}}_i = \arg\min_{\mathbf{X}_i}\{rank(\mathbf{X}_i)\} \quad \text{subject to} \quad ||\mathbf{Y}_i - \mathbf{X}_i||_2^2 \leq \epsilon \tag{2.10}$$

where ϵ is a positive constant related to the noise level in the observed image $\mathbf{y}$. Since direct rank minimization in (2.10) is NP hard, non-convex and difficult to solve, the problem is generally relaxed by substitutively minimizing the nuclear norm of the estimated matrix $\mathbf{X}_i$. The nuclear norm of a matrix $\mathbf{X}_i$, denoted by $||\mathbf{X}_i||_*$, is defined as the sum of its singular values, e.g. $||\mathbf{X}_i||_* = \sum_k |\sigma_k(\mathbf{X}_i)|$, where $\sigma_k(\mathbf{X}_i)$ is the k-th singular value of $\mathbf{X}_i$. Using the Lagrange multiplier, the denoising is described as:

$$\hat{\mathbf{X}}_i = \arg\min_{\mathbf{X}_i}\{||\mathbf{Y}_i - \mathbf{X}_i||_2^2 + \lambda||\mathbf{X}_i||_*\} \tag{2.11}$$

The optimization of (2.11) can be obtained by off-the-shell algorithms such as the hard thresholding filter with principle component analysis (PCA) or singular value decomposion (SVD) [75, 76, 83, 84]. The nuclear norm minimization (NNM) approach has been attracting significant attention due to its rapid development in both theory and implementation. In recent years, many researchers investigate to improve the flexibility

of the original nuclear norm, by proposing the weighted nuclear norm [77, 81, 82] or Schatten p-norm minimization [80], and achieve competitive denoising performances in the state-of-the-art.

1.2.6 Gaussian mixture model (GMM)

Under the Bayesian MAP framework (1.5), learning good image priors is of most importance for the success of an image restoration method. In the literature, mixture models have attracted considerable attention due to their adaptability and flexibility in describing the characteristic of signals by assuming that the signals are generated by a mixture of probability distributions. Among various contributions, Gaussian Mixture Models (GMMs) have shown their powerful ability in many applications such as image classification and segmentation [85–88]. Moreover, several studies investigated to exploit GMM as an image prior $p(\mathbf{x})$ to regularize the inverse optimization problem in (1.5). Portilla *et al.* [23] have attained impressive image denoising results by using the Gaussian scale mixture, which is derived from the GMM by assuming different scale factors in the mixture of Gaussians, to model the distribution of wavelet coefficients of an image. Recently, by considering that natural images are non-Gaussian and image patches are regarded as samples of a multivariate random variable, many researchers [5, 89–97] used Gaussian mixture models (GMMs) to characterize the statistical distribution of image patches and obtain state-of-the-art denoising and image restoration results.

In [5], Zoran *et al.* modeled the distribution of patches in the database by a mixture of K Gaussian components as:

$$p(\mathbf{x}_i) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{2.12}$$

where π_k is the mixing weight with $\sum_{k=1}^{K} \pi_k = 1$, $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$ are the mean and covariance matrix of the k-th Gaussian component. The generic GMM prior in (2.12) is then exploited to regularize the denoising problem under the Bayesian MAP framework. In [93], Xu *et al.* combined the nonlocal self-similarity prior and GMM in a unified scheme by considering that a group of similar patches in an image should belong to the same Gaussian component and proposed a collaborative patch-based denoising algorithm called patch-group-based GMM to learn nonlocal self-similarity prior from standard images and then adopted them to recover an underlying image.

Another noticeable GMM-based denoising approach is to learn a GMM from the noisy observation [90, 92] by clustering patches in the degraded image into multiple groups via nonlocal self-similarity and learning a Gaussian distribution for each cluster. An

iterative MAP expectation-maximization algorithm was also proposed to alternatively update the reconstructed image and parameters of the GMM.

Because of the large space of image patches, the generic GMM learned from standard images may not be able to model every image patch well. Some image-specific patches of a given image will be outliers in the view of a generic GMM prior and thus can not be well reconstructed using that GMM. Many attempts have been made in order to deal with this issue [94, 95, 97] by unifying internal and external GMM image patch priors. A generic GMM model is first trained from a collection of patches randomly sampled from standard images and then is adapted to the degraded image by simultaneously adding additional components and refining the component parameters. Although it is very effective for restoring images, a shortcoming of the GMM is the high runtime complexity making them ill-suited for most practical applications. In [98], Parameswaran *et al.* proposed an approximation algorithm to dramatically speed-up the implementation of the GMM in image denoising and deblurring, while incurring a negligible drop in the quality of reconstructed image.

Multiple variants of the Gaussian mixture models have been proposed in the literature for image restoration, especially for image denoising. An important characteristic of the GMM-based restoration is that a complete GMM is learn from the sample of image patches. In the reconstruction process, only one prominent Gaussian component is selected to represent the distribution of each patch in the degraded image. However, to our knowledge, it lacks clear evidences on the use of GMM in the reconstruction of an image. In this thesis, we made a contribution on justification of the number of useful Gaussian components exploited for recovering patches in the degraded image. The details of our proposal will be presented in chapter 6.

1.2.7 Combination of multiple image prior models

Different image priors characterize varying and complementary aspects of natural image statistics, and thus it is possible to combine multiple priors to improve the image restoration performance. In the last few years, numerous studies [1, 4, 61, 71–73, 99–103] have investigated the combination of nonlocal self-similarity (NSS) and sparsity priors in a unified framework for image restoration. For example, Dabov *et al.* [99] proposed a denoising method called BM3D by exploiting the NSS to enhance the sparse representation in transform-domain and achieved state-of-the-art denoising performance. By constructing 3D data from an orthogonal transform of groups of similar image patches and carrying out a 3D collaborative filtering via thresholding and Wiener filter, the residual noise in image patches can be effectively removed. Later, in [4], Dong

et al. proposed an interesting image restoration model named nonlocally centralized sparse representation (NCSR). Based on the NSS prior, Dong suggested a scheme to estimate the sparse coefficients of the original image and defined a sparse coding noise regularization (in ℓ_1-form), which is the difference between those estimates and the sparse coefficients of the observed image, to improve the performance of sparsity-based image restoration.

Many researchers [100, 101] offers a powerful mechanism of combining local sparsity and nonlocal self-similarity of images simultaneously in a unified framework which is the so-called group sparse coding. By partitioning the similar patches in the degraded image into multiple groups and assuming that the patches in the same group are encouraged to be sparsely represented by a similar set of the dictionary atoms, we can ensure that the reconstruction of matches patches should have the same image structure, and thus reduce the artifacts in the recovered image.

Another effective approaches [1, 50] joined the external similarity and sparsity in a same scheme for image restoration. In [1], Trinh *et al.* exploited the external similarity to construct a database of patches from the high quality images (called standard images). After that, for each patch $\mathbf{y}_i$ in the degraded image, the authors retrieved a local dictionary consists of most similar patches in the database, from which the sparse representation was performed to obtain an estimate of the latent patch $\mathbf{x}_i$. A part of our work in this thesis (chapter 4) was inspired from the method [1] in which we proposed a more efficient patch-selection step for searching the similar patches and thus improve the reconstruction performances.

1.3 Discriminative learning-based image restoration

Recently, the discriminative learning methods for image restoration has been attracting considerable attentions due to the blooming of artificial intelligence in computer vision. The concept of these methods is to learn a compact inference or a mapping function from a training set of degraded-latent image pairs, which then is used to reconstruct an underlying image. The general model of learning process can be written as

$$\arg\min_{\boldsymbol{\Theta}} \mathcal{L}(\hat{\mathbf{x}}, \mathbf{x}) \quad \text{s.t.} \quad \hat{\mathbf{x}} = \mathcal{F}(\mathbf{y}, \mathbf{H}; \boldsymbol{\Theta}) \tag{2.13}$$

where $\mathcal{F}(\cdot)$ is the inference or mapping function with parameter set $\boldsymbol{\Theta}$, and $\mathcal{L}(\cdot)$ is the loss function to measure the similarity between the output image $\hat{\mathbf{x}}$ and the ground-truth image $\mathbf{x}$ of the training set. Several attempts have been made in the literature which typically exploit the deep neural networks to represent the mapping function $\mathcal{F}(\cdot)$ such

as the convolutional neural network (CNN) [104–112], multi-layer perceptron [113], stacked sparse denoising autoencoders [114, 115]. After training process and obtaining the parameter $\mathbf{\Theta}$ of the neuron network, we can recover the unknown latent image $\hat{\mathbf{x}}$ from its degraded observation $\mathbf{y}$ by $\hat{\mathbf{x}} = \mathcal{F}(\mathbf{y}, \mathbf{H}; \mathbf{\Theta})$. In the last decade, the discriminative learning methods using the deep convolutional neural network have achieved great success and led the state-of-the-art performances in various image restoration applications.

In a seminal work on deep learning SR, Dong *et al.* [105, 116] first initialized the low-resolution (LR) image to a high-resolution (HR) image using a single filter, commonly bicubic interpolation, and then learns a CNN to predict the residual between the initialized HR image and the ground-truth image. In the super-resolution step, an underlying LR image is upscaled with bicubic interpolation and followed up by adding the residual estimated from the CNN model. Later, Kim *et al.* [106] proposed to increase the number of hidden layer in the convolutional neuron network and cascaded small filters many times in this deep network structure to exploit contextual information over large image regions and noticeably improve the accuracy and visual quality of reconstructed image. Another very effective approach is to establish a deep recursive convolutional network [107, 117], in which the same convolutional layer is repeatedly applied multiple times, to efficiently reuse weight parameters while exploiting a large image context. Another efficient deep learning approach is to directly predict the missing HR pixels from the LR image, as proposed in [108], in which the handcrafted bicubic filter in the SR pipeline is replaced by more complex upscaling filters specifically trained on each feature map in the LR space via a sub-pixel convolution layer.

Recently, neuron network has also been successfully used in image denoising. In [104], a convolutional neuron network is learned from a training set of noisy-clean image pairs to map a degraded image to the reconstructed image. Chen and Pock [118] proposed a Trainable Nonlinear Reaction Diffusion model to train filters and influence functions in feed-forward deep network by unfolding a fixed number of gradient descent inference steps. Instead of considering the whole image, mapping functions between noisy patches and their noise-free versions extracted from standard images can be learned via multi layer perceptrons [113] or Stacked Sparse Denoising Auto-encoder model [114] and applied to image denoising.

Despite the very competitive performance of the discriminative model learning for image restoration, its shortcoming is that the parameters Θ and mapping functions $\mathcal{F}(\ldots)$ are trained for each specific degradation process (e.g. a determined noise level). Thus, this can restrict the flexibility of application capacity of the trained models for

different image restoration tasks.

2 Methods related to our work

In this section, we present significant preliminary knowledges and background information of some related works and directly relevant approaches that will be utilized or examined in the thesis. We commence with a quick description of the super-resolution method developed by our colleagues in the LAGA and L2TI labs named SRSW [1], which inspires our researches proposed in chapter 3 and chapter 4. Afterwards, we briefly summarize the principal concepts of two well-known competing super-resolution methods called Neighbor Embedding (NE [74]) and Super-Resolution Via Sparse Representation (ScSR [3]), which are used to evaluate the performance and effectiveness of our proposed algorithms in chapter 3 and 4. Then, we present a résumé of the image restoration method using Expected Patch Log Likelihood (EPLL [5]), which coincides with one of our specific test cases proposed in chapter 6. After that, we will follow up with a presentation of the celebrated K-SVD algorithm [2] using for training a dictionary of patches, as well as a comprehensive comparison for dictionary selection mentioned in our proposed methods discussed in chapter 5 and chapter 6. We finish this section with the description of the Earth Mover's Distance, which is used as a metric for measuring the similarity between two image patches in chapter 4 of this thesis.

2.1 Image Super-Resolution by Sparse Weight (SRSW)

In [1], Trinh *et al.* proposed a patch-based super-resolution method, named Super-Resolution by Sparse Weight (SRSW), for recovering each high-resolution image patch $\mathbf{x}_i \in R^n$ from its degraded low-resolution version $\mathbf{y}_i \in R^m$, which is assumed to be generated by the following model (derived from (1.2)):

$$\mathbf{y}_i = \mathbf{S}_i\mathbf{B}_i\mathbf{x}_i + \boldsymbol{\eta}_i. \tag{2.14}$$

where $\mathbf{B}_i$ and $\mathbf{S}_i$ are the blurring and downsampling operations on image patch, respectively. $\boldsymbol{\eta}_i \sim \mathbf{N}(0,\sigma^2)$ is the residual noise in each patch $\mathbf{y}_i$. The SRSW method consists of two main phases: Building of dictionaries and Super-resolution reconstruction.

Dictionary construction

The authors constructed a couple of exhaustive high-resolution dictionary $\mathbf{D}^h$ and low-resolution dictionary $\mathbf{D}^l$ from a list of available standard (clean, high resolution) images $\{\mathbf{x}^s\}$ as:

i. Create a high-resolution database $\{\mathbf{x}_k^s | \mathbf{x}_k^s \in R^n; k = 1, \ldots, P\}$ by randomly extract a set of P high-resolution patches from standard images.

ii. Generate the low-resolution database of patches $\{\mathbf{y}_k^s | \mathbf{y}_k^s \in R^m; k = 1, \ldots, P\}$ by downsampling the high-resolution database with the degrading factors: $\mathbf{y}_k^s = \mathbf{S}_i \mathbf{B}_i \mathbf{x}_k^s$.

iii. Obtain the couple of high- and low-resolution dictionaries by normalizing the databases of patches:

$$\begin{aligned} \mathbf{D}^h &= \{\mathbf{d}_j^h | \mathbf{d}_j^h = \frac{\mathbf{x}_j^s}{\left\|\mathbf{x}_j^s\right\|_2}, j = 1, \ldots, P\} \\ \mathbf{D}^l &= \{\mathbf{d}_j^l | \mathbf{d}_j^l = \frac{\mathbf{y}_j^s}{\left\|\mathbf{y}_j^s\right\|_2}, j = 1, \ldots, P\} \end{aligned} \tag{2.15}$$

Super-resolution

The SRSW method is developed based on a core idea that the distributions of high-resolution patches and their corresponding low-resolution version in two vector spaces generated by high-resolution dictionary $\mathbf{D}^h$ and low-resolution dictionary $\mathbf{D}^l$ are identical. That means, a high-resolution patch and its corresponding low-resolution version share the same representation coefficients vector. Thus, for each low-resolution patch $\mathbf{y}_i$, the authors found the representation coefficients vector $\hat{\boldsymbol{\alpha}}_i$ under the Bayesian MAP framework

$$\hat{\boldsymbol{\alpha}}_i = \arg\max_{\boldsymbol{\alpha}_i} p(\mathbf{y}_i | \mathbf{D}^l, \boldsymbol{\alpha}_i) p(\boldsymbol{\alpha}_i) \tag{2.16}$$

and then used it as to decode the high-resolution dictionary $\mathbf{D}^h$ to get an estimation of the latent high-resolution image patch as $\hat{\mathbf{x}}_i = \mathbf{D}^h \hat{\boldsymbol{\alpha}}_i$.

In [1], Trinh *et al.* adopt the sparse Laplacian distribution as a prior model of $p(\boldsymbol{\alpha}_i)$. Moreover, they proposed a weighting vector $\mathbf{w}_i$, which measured to the similarity between $\mathbf{y}_i$ and the atoms $\mathbf{d}_j^l$ of the dictionary, to weight the sparse code $\boldsymbol{\alpha}_i$. Hence, the MAP problem in (2.16) is formulated as:

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\{ \left\|\mathbf{y}_i - \mathbf{D}^l \boldsymbol{\alpha}_i\right\|_2^2 + \left\|\mathbf{w}_i^T \boldsymbol{\alpha}_i\right\|_1 \right\} \tag{2.17}$$

However, solving the problem (2.17) with the whole dictionary $\mathbf{D}^l$ with very large number of P atoms is a really computational challenge. The authors introduced a patch-filtering step to select a smaller set of most similar atoms for recovering the underlying image patch $\mathbf{y}_i$. Hence, for each low-resolution patch $\mathbf{y}_i$, a couple of local

dictionaries $\mathbf{D}_i^h$ and $\mathbf{D}_i^l$ were collected from the large dictionaries $\mathbf{D}^h$ and $\mathbf{D}^l$:

$$\begin{aligned}\mathbf{D}_i^l &= \{\mathbf{d}_j^l \in \mathbf{D}^l | d(\mathbf{d}_j^l, \mathbf{y}_i) < r_i\} \\ \mathbf{D}_i^h &= \{\mathbf{d}_j^h \in \mathbf{D}^h | d(\mathbf{d}_j^l, \mathbf{y}_i) < r_i\}\end{aligned} \tag{2.18}$$

where r_i is a positive threshold chosen for each patch $\mathbf{y}_i$ in such a way that the first K atoms with smallest distances $d(\mathbf{d}_j^l, \mathbf{y}_i)$ are selected ($\mathbf{D}_i^l \in R^{m\times K}, \mathbf{D}_i^h \in R^{n\times K}$). $d(\mathbf{d}_j^l, \mathbf{y}_i)$ is the distance between the atom $\mathbf{d}_j^l$ and $\mathbf{y}_i$, and is defined in equation (26) in the paper [1] as:

$$d(\mathbf{d}_j^l, \mathbf{y}_i) = \left\|\mathbf{y}_i - \mu_{il}\mathbf{d}_j^l\right\|_2^2 + \left|E(\mathbf{y}_i - \mu_{il}\mathbf{d}_j^l)\right| + \left|\mathrm{Var}(\mathbf{y}_i - \mu_{il}\mathbf{d}_j^l) - \sigma^2\right| \tag{2.19}$$

with $\mu_{il} = E(\mathbf{y}_i)/E(\mathbf{d}_j^l)$. Thus, the problem (2.17) is rewritten on local dictionary $\mathbf{D}_i^l$ as:

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\{ \left\|\mathbf{y}_i - \mathbf{D}_i^l\boldsymbol{\alpha}_i\right\|_2^2 + \left\|\mathbf{w}_i^T\boldsymbol{\alpha}_i\right\|_1 \right\} \tag{2.20}$$

In addition, the weighting vector $\mathbf{w}_i$ in (2.20) is now determined on the local dictionary $\mathbf{D}_i^l$, $\mathbf{w}_i = [\mathbf{w}_i(1), \ldots, \mathbf{w}_i(j), \ldots, \mathbf{w}_i(K)]^T$, in term of distances $d(\mathbf{d}_j^l, \mathbf{y}_i)$ as:

$$\mathbf{w}_i(j) = \begin{cases} e^{d(\mathbf{d}_j^l, \mathbf{y}_i)} & \text{if } d(\mathbf{d}_j^l, \mathbf{y}_i) > \gamma m\sigma^2 \\ d(\mathbf{d}_j^l, \mathbf{y}_i) & \text{otherwise} \end{cases} \tag{2.21}$$

where γ is a positive constant.

After solving (2.20) with multiplicative updates algorithm to achieve the optimal value of $\hat{\boldsymbol{\alpha}}_i$, the latent high-resolution image patch can be estimated by $\hat{\mathbf{x}}_i = \mathbf{D}_i^h\hat{\boldsymbol{\alpha}}_i$.

The above restoration process is repeated for every patches $\mathbf{y}_i$ in the degraded image $\mathbf{y}$. After that, the super-resolved patches $\hat{\mathbf{x}}_i$ are aggregated to obtain a coarse estimate of the high-resolution image $\mathbf{x}$. The authors finally apply a back-projection process to enhance the quality of the reconstructed image.

2.2 Image Super-Resolution through Neighbor Embedding (NE)

Chang *et al.* proposed in [74] a patch-based super-resolution method called Neighbor Embedding (NE) to recover each patch $\mathbf{y}_i$ in the low-resolution image $\mathbf{y}$ with a support of standard images $\{\mathbf{x}^s\}$. Generally, this method can be divided into two phases: construction of dictionaries and super-resolution.

Construction of dictionaries

From the list of standard high-resolution images $\{\mathbf{x}^s\}$, a couple of exhaustive dictionaries, which is identical to the database of patches, is created as follows:

i. Generate a low-resolution image $\mathbf{y}^s$ for each standard image $\mathbf{x}^s$ by blurring (with operator $\mathbf{B}$) and then downsampling (using operator $\mathbf{S}$) as $\mathbf{y}^s = \mathbf{SBx}^s$.

ii. Randomly collect P couples of low-resolution patch $\mathbf{y}_k^s$ and its high-resolution version $\mathbf{x}_k^s$ from the set of low- and high-resolution images $\{\mathbf{y}^s, \mathbf{x}^s\}$ to form the couple of low-resolution dictionary $\mathbf{D}^l$ and high-resolution dictionary $\mathbf{D}^h$.

$$\begin{aligned} \mathbf{D}^l &= \{\mathbf{d}_j^l | \mathbf{d}_j^l \equiv \mathbf{y}_j^s, j = 1, \ldots, P\} \\ \mathbf{D}^h &= \{\mathbf{d}_j^h | \mathbf{d}_j^h \equiv \mathbf{x}_j^s, j = 1, \ldots, P\} \end{aligned} \tag{2.22}$$

Super-reconstruction

Performing the patch super-resolution on very large dictionaries is difficult and time-consuming. Thus Chang *et al.* proposed to retrieve a couple of local dictionaries $\mathbf{D}_i^l \subset \mathbf{D}^l$ and $\mathbf{D}_i^h \subset \mathbf{D}^h$ for recovering each low-resolution patch $\mathbf{y}_i$ in the degraded image $\mathbf{y}$:

$$\begin{aligned} \mathbf{D}_i^l &= \{\mathbf{d}_j^l \in \mathbf{D}^l | \left\| \mathbf{d}_j^l - \mathbf{y}_i \right\|_2 < r_i\} \\ \mathbf{D}_i^h &= \{\mathbf{d}_j^h \in \mathbf{D}^h | \left\| \mathbf{d}_j^l - \mathbf{y}_i \right\|_2 < r_i\} \end{aligned} \tag{2.23}$$

The positive constant r_i was set for each patch $\mathbf{y}_i$ such that only first K atoms with smallest Euclidean distance to $\mathbf{y}_i$ is selected.

The NE method was developed based on two assumptions:

- The image patches form a manifold, and a set of similar patches lie on or close to a locally linear part of the manifold. Thus, each image patch $\mathbf{y}_i$ (or $\mathbf{x}_i$) can be express by a linear combination of its neighbors in the local dictionary $\mathbf{D}_i^l$ (or $\mathbf{D}_i^h$), e.g. $\mathbf{y}_i = \mathbf{D}_i^l \boldsymbol{\alpha}_i$ with $\boldsymbol{\alpha}_i \in R^K$ is a weighting vector.
- The corresponding patches in low- and high-resolution manifolds share the same weighting vector $\boldsymbol{\alpha}_i$. Thus the authors can find $\boldsymbol{\alpha}_i$ from the relationship between $\mathbf{y}_i$ and $\mathbf{D}_i^l$ and then apply it to estimate the latent high-resolution patch $\mathbf{x}_i = \mathbf{D}_i^h \boldsymbol{\alpha}_i$.

For each low-resolution patch $\mathbf{y}_i$, Chang found the weighting vector $\boldsymbol{\alpha}_i = [\boldsymbol{\alpha}_i(1), \ldots, \boldsymbol{\alpha}_i(j), \ldots, \boldsymbol{\alpha}_i(K)]^T$ by solving the least mean square reconstruction error:

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\| \mathbf{y}_i - \mathbf{D}_i^l \boldsymbol{\alpha}_i \right\|_2^2 \quad \text{s.t.} \quad \sum_{j=1}^{K} \boldsymbol{\alpha}_i(j) = 1 \tag{2.24}$$

Hence, the latent high-resolution image patch can be estimated as $\hat{\mathbf{x}}_i = \mathbf{D}_i^h \hat{\boldsymbol{\alpha}}_i$. The super-resolution was performed for every patch $\mathbf{y}_i$ in the degraded image, then Chang obtained the super-resolved image $\hat{\mathbf{x}}$ by aggregating the overlapping regions of the attained patches $\hat{\mathbf{x}}_i$.

2.3 Image Super-Resolution Via Sparse Representation (ScSR)

In [3], Yang *et al.* proposed a patch-based dictionary learning method, named ScSR, for enhancing the resolution of each low-resolution patch $\mathbf{y}_i$. This method is composed of two main phases: dictionary learning and super-resolution.

Dictionary learning

Instead of constructing a large couple of dictionaries from the database of patches, in which we have to perform a patch-filtering step to select a couple of local dictionaries for each low-resolution patch $\mathbf{y}_i$ as in the NE or SRSW method, Yang *et al.* proposed to jointly train two smaller low- and high-resolution dictionaries with the support of sparse model.

From the list of standard high-resolution images $\{\mathbf{x}^s\}$, Yang *et al.* generated the corresponding low-resolution images $\{\mathbf{y}^s\}$ by blurring and downsampling as $\mathbf{y}^s = \mathbf{SBx}^s$. After that, a set of P couples of low- and high-resolution patches $\{\mathbf{y}_k^s \in R^m, \mathbf{x}_k^s \in R^n | k = 1,\ldots,P\}$ were randomly collected from the training images $\{\mathbf{y}^s, \mathbf{x}^s\}$. From the database of patches, the authors learned a couple of low-resolution dictionary $\mathbf{D}^l \in R^{m\times K}$ and high-resolution dictionary $\mathbf{D}^h \in R^{n\times K}$ in the Bayesian MAP perspective and the Laplacian prior model of the probability distribution of image patches. The fundamental of the ScSR method was to assume that the representation of each low-resolution patch $\mathbf{y}_k^s$ and its corresponding high-resolution patch $\mathbf{x}_k^s$ under the low-resolution and high-resolution dictionaries share the same sparse code $\boldsymbol{\alpha}_k^s$. That means, they can formulate the problem of determining the dictionaries as

$$\begin{aligned}\mathbf{D}^l &= \underset{\mathbf{D}^l,\{\boldsymbol{\alpha}_k^s\}}{\arg\min}\left\{\sum_{k=1}^{P}\left\|\mathbf{y}_k^s - \mathbf{D}^l\boldsymbol{\alpha}_k^s\right\|_2^2 + \lambda\sum_{k=1}^{P}\|\boldsymbol{\alpha}_k^s\|_1\right\}\\ \mathbf{D}^h &= \underset{\mathbf{D}^h,\{\boldsymbol{\alpha}_k^s\}}{\arg\min}\left\{\sum_{k=1}^{P}\left\|\mathbf{x}_k^s - \mathbf{D}^h\boldsymbol{\alpha}_k^s\right\|_2^2 + \lambda\sum_{k=1}^{P}\|\boldsymbol{\alpha}_k^s\|_1\right\}\end{aligned} \tag{2.25}$$

Let $\mathbf{Y}^s = [\mathbf{y}_1^s,\ldots,\mathbf{y}_k^s,\ldots,\mathbf{y}_P^s] \in R^{m\times P}$ denote the database of P low-resolution patches, $\mathbf{X}^s = [\mathbf{x}_1^s,\ldots,\mathbf{x}_k^s,\ldots,\mathbf{x}_P^s] \in R^{n\times P}$ refer to the database of P high-resolution patches and $\mathbf{A}^s = [\boldsymbol{\alpha}_1^s,\ldots,\boldsymbol{\alpha}_k^s,\ldots,\boldsymbol{\alpha}_P^s] \in R^{K\times P}$ be the matrix of P representation coefficients vectors. Yang combined the two objective functions in (2.25), forcing the high-resolution and low-resolution representations to share the same codes, and obtained:

$$\{\mathbf{D}^l,\mathbf{D}^h\} = \underset{\mathbf{D}^l,\mathbf{D}^h,\mathbf{A}^s}{\arg\min}\left\{\frac{1}{m}\left\|\mathbf{Y}^s - \mathbf{D}^l\mathbf{A}^s\right\|_2^2 + \frac{1}{n}\left\|\mathbf{X}^s - \mathbf{D}^h\mathbf{A}^s\right\|_2^2 + \lambda\left(\frac{1}{m}+\frac{1}{n}\right)\|\mathbf{A}^s\|_1\right\} \tag{2.26}$$

By defining

$$\mathbf{X}^c = \begin{bmatrix} \frac{1}{\sqrt{m}}\mathbf{Y}^s \\ \frac{1}{\sqrt{n}}\mathbf{X}^s \end{bmatrix}, \quad \mathbf{D}^c = \begin{bmatrix} \frac{1}{\sqrt{m}}\mathbf{D}^l \\ \frac{1}{\sqrt{n}}\mathbf{D}^h \end{bmatrix}, \quad \hat{\lambda} = \lambda\left(\frac{1}{m} + \frac{1}{n}\right) \tag{2.27}$$

then, the problem in (2.26) can be rewritten as:

$$\mathbf{D}^c = \underset{\mathbf{D}^c, \mathbf{A}^s}{\arg\min} \left\{ \|\mathbf{X}^c - \mathbf{D}^c\mathbf{A}^s\|_2^2 + \hat{\lambda}\|\mathbf{A}^s\|_1 \right\} \tag{2.28}$$

Therefore, we can use the same learning strategy in the single dictionary case for training the two dictionaries $\mathbf{D}^l$ and $\mathbf{D}^h$. In [3], the authors alternatively updated the "pseudo" dictionary $\mathbf{D}^c$ and sparse code $\mathbf{A}^s$ using the sparse coding algorithm introduced in [119].

Super-resolution

The super-resolution on degraded image $\mathbf{y}$ was performed on each low-resolution patch $\mathbf{y}_i$ using the couple of dictionaries $\mathbf{D}^l$ and $\mathbf{D}^h$ jointly trained from the database of patches. Firstly, Yang found the sparse code $\boldsymbol{\alpha}_i$ of $\mathbf{y}_i$ in the vector space generated by low-resolution dictionary $\mathbf{D}^l$ by solving the Bayesian MAP problem:

$$\hat{\boldsymbol{\alpha}}_i = \underset{\boldsymbol{\alpha}_i}{\arg\min} \left\{ \left\|\mathbf{y}_i - \mathbf{D}^l\boldsymbol{\alpha}_i\right\| + \lambda\|\boldsymbol{\alpha}_i\|_1 \right\} \tag{2.29}$$

Then the estimation of the latent high-resolution image patch $\mathbf{x}_i$ was obtained as $\hat{\mathbf{x}}_i = \mathbf{D}^h\hat{\boldsymbol{\alpha}}_i$. The high-resolution patches $\hat{\mathbf{x}}_i$ were put into the proper locations in the high-resolution grid and Yang calculated the average of overlapping regions to get the super-resolved image. An iterative back-projection process was applied on the initial super-resolution image to enhance the reconstruction quality.

2.4 Image restoration using Expected Patch Log Likelihood (EPLL)

Zoran and Weiss [5] proposed an effective method called Expected Patch Log Likelihood (EPLL), which learned prior models from standard images to regularize the image restoration problem that modeled in (1.1). The key idea of this method is to develop an algorithm such that the patches in reconstructed image are likely to follow the designed prior, while keeping the reconstructed image still close to the corrupted image.

The proposed method EPLL can be interpreted in the point view of Bayesian MAP, which maximizes the posteriori probability of the latent image $\mathbf{x} \in R^N$ given the observation $\mathbf{y}$:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\arg\max}\, p(\mathbf{x}|\mathbf{y}) = \underset{\mathbf{x}}{\arg\max}\{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})\} \tag{2.30}$$

With the assumption of Gaussian noise corruption, the likelihood is described as $p(\mathbf{y}|\mathbf{x}) \propto \exp\left(-\|\mathbf{y}-\mathbf{H}\mathbf{x}\|_2^2/2\sigma^2\right)$. Moreover, Zoran and Weiss construct prior on image patches $\{\mathbf{x}_i \in R^n | i = 1,\ldots,N\}$ of the image. For more convenience, they denote an image patch as $\mathbf{R}_i\mathbf{x}$, where $\mathbf{R}_i$ is a matrix which extracts the i-th patch from the image. The prior statistical model of the whole image can be inferred by assuming that each image patch is independently draw from a prior model p. Thus $p(\mathbf{x}) = \prod_{i=1}^{N} p(\mathbf{x}_i) = \prod_{i=1}^{N} p(\mathbf{R}_i\mathbf{x})$. The problem in (2.30) can be rewritten in the logarithm form as:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ \frac{\lambda}{2}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|_2^2 - \sum_{i=1}^{N} \log(p(\mathbf{R}_i\mathbf{x})) \right\} \tag{2.31}$$

Direct optimization of the cost function in (2.31) may be very hard, depending on the prior used. Zoran and Weiss presented an alternative optimization method called "Half Quadratic Splitting", in which a set of image patches $\{\mathbf{z}_i \in R^n | i = 1,\ldots,N\}$ are introduced for the overlapping patches $\mathbf{R}_i\mathbf{x}$ in the latent image $\mathbf{x}$.

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x},\{\mathbf{z}_i\}} \left\{ \frac{\lambda}{2}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|_2^2 - \sum_{i=1}^{N} \left[\frac{\beta}{2}\|\mathbf{z}_i - \mathbf{R}_i\mathbf{x}\|_2^2 - \log(p(\mathbf{z}_i)) \right] \right\} \tag{2.32}$$

where β is a positive fixed value. The authors adopted an iterative manner for solving (2.32) by alternatively updating the variables as the following framework.

1. Fix $\mathbf{x}$, update the new value for each image patch $\mathbf{z}_i$ by solving

$$\hat{\mathbf{z}}_i = \arg\min_{\mathbf{z}_i} \left\{ \frac{\beta}{2}\|\mathbf{z}_i - \mathbf{R}_i\mathbf{x}\|_2^2 - \log(p(\mathbf{z}_i)) \right\} \tag{2.33}$$

2. Fix $\{\mathbf{z}_i\}$, update $\mathbf{x}$ by

$$\hat{\mathbf{x}} = \left(\lambda\mathbf{H}^T\mathbf{H} + \beta\sum_{j=1}^{N} \mathbf{R}_j^T\mathbf{R}_j \right)^{-1} \left(\lambda\mathbf{H}^T\mathbf{y} + \beta\sum_{j=1}^{N} \mathbf{R}_j^T\mathbf{z}_j \right) \tag{2.34}$$

3. Repeat step (1) and (2) until reaching the stopping condition.

At an iteration, the current reconstructed image $\hat{\mathbf{x}}$ is fixed and each patch i in the image $\hat{\mathbf{x}}$ is updated via the optimizing $\mathbf{z}_i$ in (2.33). After achieving new values of image patches, the current reconstructed image $\hat{\mathbf{x}}$ will be updated by aggregating the overlapping regions on image patches and then matching with the degraded image $\mathbf{y}$ using (2.34).

Finding the solution of (2.33) depends on the determination of the prior model of distribution of image patches in the image. In [5], Zoran and Weiss exploited the

Gaussian Mixture Model (GMM) to characterize the probability distribution of patches in a database and expected that the patches in the underlying image $\mathbf{x}$ will likely follow the same distribution. More specifically, they randomly collected a set of P patches $\{\mathbf{x}_k^s | k = 1, \ldots, P\}$ from a list of standard images $\{\mathbf{x}^s\}$ and assumed that each patch $\mathbf{x}_k^s$ was drawn from a mixture of finite M Gaussian components $\{\pi_m, \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m\}$ as $p(\boldsymbol{\alpha}_k^s) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{\alpha}_k^s | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$. After learning the parameters of the GMM model, Zoran and Weiss supposed the prior model of image patches in the reconstructed image $\mathbf{x}$ can be described as $p(\mathbf{z}_i) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\mathbf{z}_i | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$. However, solving the problem in (2.33) is difficult. To tackle this issue, the authors proposed to select only one Gaussian component from the GMM to model the prior probability of $p(\mathbf{z}_i)$ and introduced a scheme for optimizing (2.33) as follows.

i. Given each patch $\mathbf{R}_i\mathbf{x}$ in the current reconstructed image $\mathbf{x}$, calculate the conditional mixing weight γ_{im}

$$\gamma_{im} = \frac{\pi_m \mathcal{N}(\mathbf{R}_i\mathbf{x} | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m + \sigma_i^2 \mathbf{I})}{\sum_{l=1}^{M} \pi_m \mathcal{N}(\mathbf{R}_i\mathbf{x} | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m + \sigma_i^2 \mathbf{I})} \tag{2.35}$$

with σ_i is the standard deviation of residual noise in i-th patch in the current reconstructed image $\mathbf{x}$, $\mathbf{I}$ is an identity matrix.

ii. Select one Gaussian component which has the highest conditional mixing $m_{max} = \max_m \gamma_{im}$. Thus the probability distribution of $p(\mathbf{z}_i)$ in (2.33) is described as $p(\mathbf{z}_i) \propto \mathcal{N}(\mathbf{z}_i | \boldsymbol{\mu}_{m_{max}}, \boldsymbol{\Sigma}_{m_{max}})$.

iii. Solve the problem(2.33) to obtain the new value of $\mathbf{z}_i$:

$$\hat{\mathbf{z}}_i = (\beta \boldsymbol{\Sigma}_{m_{max}} + \mathbf{I})^{-1} (\beta \boldsymbol{\Sigma}_{m_{max}} \mathbf{R}_i\mathbf{x} + \boldsymbol{\mu}_{m_{max}}) \tag{2.36}$$

In chapter 6 of this thesis, we will analyze the use of GMM as a prior model of the distribution of representation coefficients of image patches, to regularize the denoising process. We inherit the same iteration optimization process presented in (2.33) and (2.34) the EPLL method. At each iteration, we first update each patch in the current reconstructed image and then aggregate the obtained patches and match the reconstructed image to the corrupted image to ensure the coherent of image contents between the recovering image and the degraded image.

2.5 K-SVD: An over-complete dictionary learning for sparse representation

In [2], Aharon *et al.* proposed a brilliant method, known as K-SVD, for learning a dictionary $\mathbf{D}$ from a training set of P image patches, denoted by $\mathbf{X}^s = \{\mathbf{x}_k^s | \mathbf{x}_k^s \in$

$R^n, k = 1, \ldots, P\}$, in which the dictionary can well express the structure appearing in the patches, as well as promote the sparse representation of each image patch $\mathbf{x}_k^s$. The fundamental principle of the K-SVD method is to construct the dictionary $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_j, \ldots, \mathbf{d}_K] \in R^{n \times K}$ from the training set $\mathbf{X}^s$ such that each image patch $\mathbf{x}_k^s \in \mathbf{X}^s$ can be approximately represented by only a few number of atoms in the dictionary. In [2], Aharon *et al.* study the following objective function:

$$\mathbf{D} = \underset{\mathbf{D}, \{\boldsymbol{\alpha}_k^s\}}{\arg\min} \left\{ \sum_{k=1}^{P} \|\mathbf{x}_k^s - \mathbf{D}\boldsymbol{\alpha}_k^s\| \right\} \quad \text{subject to} \quad \forall k, \|\boldsymbol{\alpha}_k^s\|_0 \leq T \tag{2.37}$$

where T is a positive constant that constrains the sparsity (number of nonzero entries) of the representation coefficient vector $\boldsymbol{\alpha}_k^s$.

The authors use an iterative process to alternatively optimize the expression in (2.37). First, they keep $\mathbf{D}$ unchanged and find the sparse code $\boldsymbol{\alpha}_k^s$ for each patch $\mathbf{x}_k^s$ in the training set using the Orthogonal Matching Pursuit (OMP) algorithm [120] as present in algorithm 2.1. After that, the sparse representation coefficients vectors $\{\boldsymbol{\alpha}_k^s\}$ are fixed and the dictionary is updated atom by atom, with the support of the singular value decomposition (SVD) on the representation error corresponding to each atom. The summary of K-SVD algorithm is presented as follows:

i. **Initialization**: Set a initial value of dictionary $\mathbf{D}^{(0)} \in R^{n \times K}$ with ℓ_2 normalized columns. Set iterator $J = 1$.

ii. **Repeat** until convergence:

- *Sparse coding stage*: Fix $\mathbf{D}^{(J-1)}$, compute sparse representation coefficients vector $\boldsymbol{\alpha}_k^s$ for each patch $\mathbf{x}_k^s$ in the training set by solving (2.38) using the OMP introduced in algorithm 2.1.

$$\hat{\boldsymbol{\alpha}}_k^s = \underset{\boldsymbol{\alpha}_k^s}{\arg\min} \{ \left\| \mathbf{x}_k^s - \mathbf{D}^{(J-1)}\boldsymbol{\alpha}_k^s \right\|_2^2 \} \quad \text{s.t.} \quad \|\boldsymbol{\alpha}_k^s\| \leq T_0 \tag{2.38}$$

- *Dictionary update stage*: Update each column (atom) $\mathbf{d}_i, i = 1, \ldots, K$ in $\mathbf{D}^{(J-1)}$ by
  - Define a group of patches that use $\mathbf{d}_i$ in representation: $\omega_i = \{j | 1 \leq j \leq N, \boldsymbol{\alpha}_k^s(j) \neq 0\}$.
  - Compute the representation error for all patches in ω_i when removing the $\mathbf{d}_i$ from the dictionary:

$$\mathbf{E}_{\omega_i} = (\mathbf{X}_i^s)_{\omega_i} - \sum_{j \neq i} \boldsymbol{\alpha}_k^s(j)\mathbf{d}_j \tag{2.39}$$

– Apply SVD decomposition $\mathbf{E}_{\omega_i} = \mathbf{U}\boldsymbol{\Delta}\mathbf{V}^T$. Update the atom $\mathbf{d}_i$ to be the first column of $\mathbf{U}$ ($\mathbf{d}_i = \mathbf{U}(:,1)$). Similarity, update the corresponding representation coefficients vector $\boldsymbol{\alpha}_i^s = \boldsymbol{\Delta}(1,1) * \mathbf{V}(:,1)$

- Increase the iterator $J = J + 1$.

Algorithm 2.1: The orthogonal matching pursuit (OMP)

Input : Image patch $\mathbf{x}_k^s$, dictionary $\mathbf{D}$, sparse constraint T_0, representation error ϵ

Let ω be the index set of the selected dictionary atoms and initialize as $\omega = \emptyset$

while ***not*** *converged* **do**

  Compute the correlation of $\mathbf{x}_k^s$ to each dictionary atom $\mathbf{d}_j$ which is not in ω, as $\frac{(\mathbf{x}_k^s)^T \mathbf{d}_j}{\|\mathbf{x}_k^s\|_2 \|\mathbf{d}_j\|_2}$.

  Pick the atom $\mathbf{d}_j$ which has largest correlation and include in into the set $\omega = \omega \cup \{i\}$.

  Update the representation coefficient $\boldsymbol{\alpha}_{k,\omega}^s = (\mathbf{D}_\omega^T \mathbf{D}_\omega)^{-1} (\mathbf{D}_\omega^T \mathbf{x}_k^s)$, where $\mathbf{D}_\omega \subset \mathbf{D}$ contains all the atoms belonging to the set ω, $\boldsymbol{\alpha}_{k,\omega}^s$ consists of non-zero entries of $\boldsymbol{\alpha}_k^s$.

  Compute the signal residual by subtracting the selected atoms $\mathbf{x}_k^s = \mathbf{x}_k^s - \mathbf{D}_\omega \boldsymbol{\alpha}_{k,\omega}$.

  Check stop criterion ($|\omega| > T_0$ or $\|\mathbf{x}_k^s\|_2 < \epsilon$)

end

Output : Sparse representation coefficient vector $\boldsymbol{\alpha}_k^s$

2.6 Dictionary selection

In chapter 5 and 6, we demonstrate that the sparse models may fail to characterize the true distribution of representation coefficients of image patches in a vector space generated by atoms in a dictionary. Therefore, we investigate to estimate the probability distribution of patches in a database extracted from standard images as a prior to enhance the quality of a noisy image. However, analyzing the distribution of image patches in different vector space created by local dictionary for each patch $\mathbf{y}$ is difficult. To this end, we construct a unique vector space from one global dictionary $\mathbf{D}$ learned from a training set of patches to recover all degraded patches in an underlying image. In this subsection, we will discuss on our selection of a global dictionary for image restoration.

Determining a dictionary $\mathbf{D}$ which can convey the local information of the training set of patches is crucial for the success of an image restoration method. In the literature,

there are two main categories of the dictionary: the analytical dictionary and the data-adaptive dictionary.

(a) Airfield **(b)** Airplane **(c)** Baboon **(d)** Baby **(e)** Barbara

(f) Boat **(g)** Bridge **(h)** Cameraman **(i)** Couple **(j)** Fruits

(k) Boy **(l)** Hill **(m)** House **(n)** Jellybeans **(o)** Leaves

(p) Lena **(q)** Man **(r)** Monarch **(s)** Peppers **(t)** Zelda

Figure 2.1: *20 natural test images.*

One simple and representative approach was the use of analytic dictionary where the fixed transform bases such as wavelet [121], discrete cosine transform (DCT) [122], etc, are taken into account to build the dictionary $\mathbf{D}$. However, off-the-shelf bases dictionaries have limitation in their ability to represent different types of image structures. Elad and Aharon [44] indicated that the dictionary learned from the image patches produced superior performances for image denoising than the analytic dictionary such as the DCT.

On the other hand, several efforts have been made in the literature to design

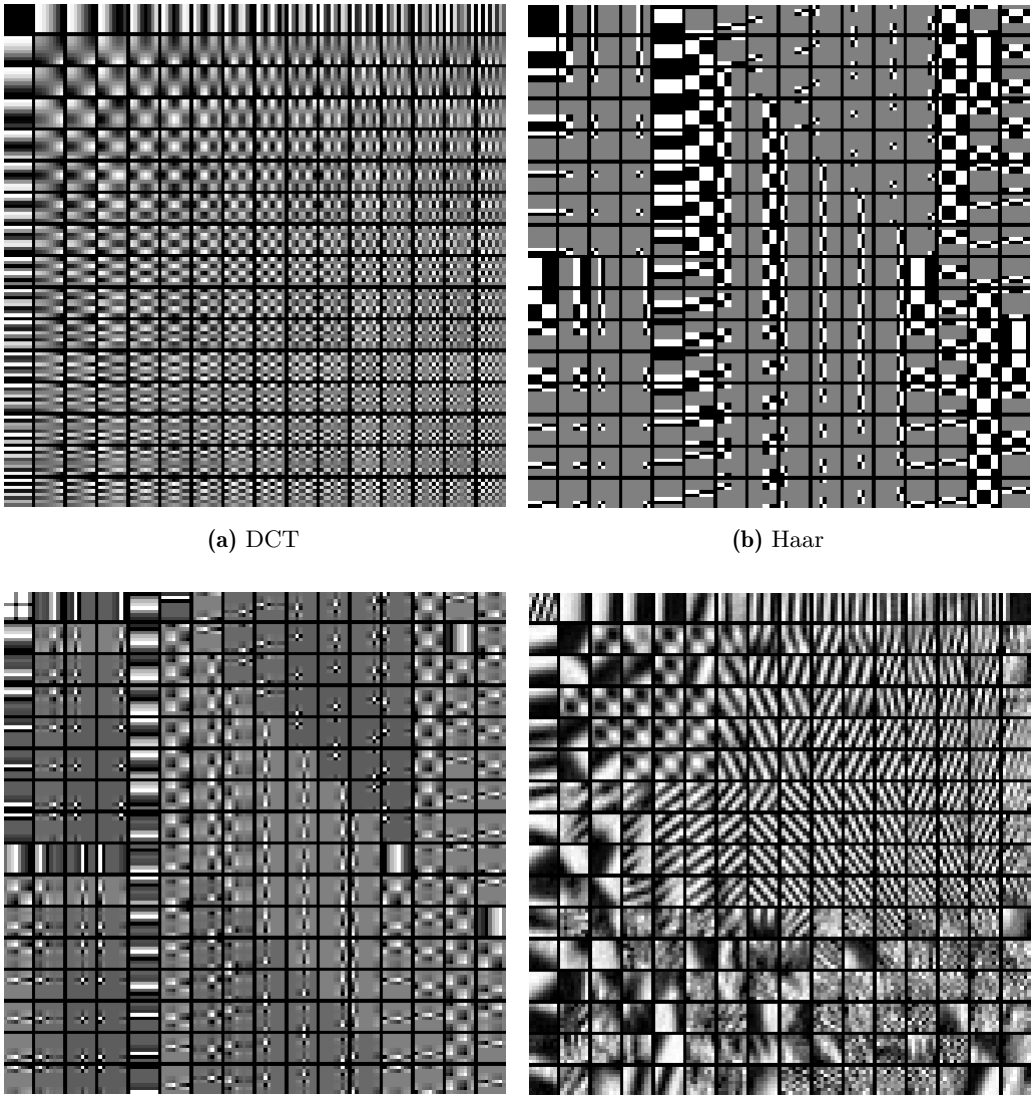
data-adaptive dictionaries including principal component analysis (PCA) [123–126], sparsity learning [2, 3, 45, 47, 127–129], or PCA-sparsity combination [4, 71, 72]. In the PCA-based dictionary learning, the patches in the training set $\mathbf{X}$ (which may be collected from standard images or in the degraded image itself) is first partition into multiple groups $\{\mathbf{X}_k\}$ and each dictionary is trained for each group via performing the principal component analysis on the covariance matrices. In the stage of restoration, each degraded patch $\mathbf{y}_i$ in the underlying image $\mathbf{y}$ will be assigned to one group and thus the corresponding dictionary is exploited to restore $\mathbf{y}_i$. However, in our researches in chapter 5 and 6, we need to prepare a global dictionary for every patches in an image to study the distribution of patches in a unique vector space. Therefore, we do not examine the choice of PCA-based dictionary here. Instead, we investigate the sparsity-based methods for the dictionary learning. In the literature, there are no full comprehensive comparison between different sparse learning dictionary methods, and the choice of a sparse model to learn the dictionary is an arbitrary decision. In our studies, we adopt the seminal work of Aharon *et al.* [2] known as the K-SVD algorithm, for training a data-adaptive dictionary. In the rest of this subsection, we make a comparison of image reconstruction performances between the K-SVD dictionary and the other analytic dictionaries.

For demonstration, we conducted the denoising on 20 widely used natural test images as shown in Fig. 2.1. For simplicity, we only consider the luminance channel. Each image was degraded by adding Gaussian noise of zero mean and standard deviation $\sigma = 10$ and $\sigma = 30$, respectively. The noisy image $\mathbf{y}$ is first partitioned into a set of overlapping patches of size 8×8 ($m = 64$) and the denoising was performed on each patch using the ℓ_0 sparse model as:

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \|\boldsymbol{\alpha}_i\|_0 \quad \text{s.t.} \quad \|\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 \leq \epsilon \tag{2.40}$$

where ϵ is a positive constant relates to the noise level. The optimization of (2.40) can be found using the orthogonal matching pursuit method (OMP) in algorithm 2.1. We prepared three types of analytic dictionaries of K atoms, with $K \in \{64, 128, 256, 512\}$, including the DCT dictionary, the Haar wavelet dictionary and the Daubechies D4 (Db4) dictionary. Note that the analytic dictionaries were fixed for all images and different noise levels. In addition, when $K > 64$, we constructed overcomplete dictionaries from the orthogonal bases by shifting each basis (column) along vertical direction. For the data-adaptive dictionary, we randomly collected $P = 100000$ patches from the noisy image to train a dictionary using the K-SVD algorithm.

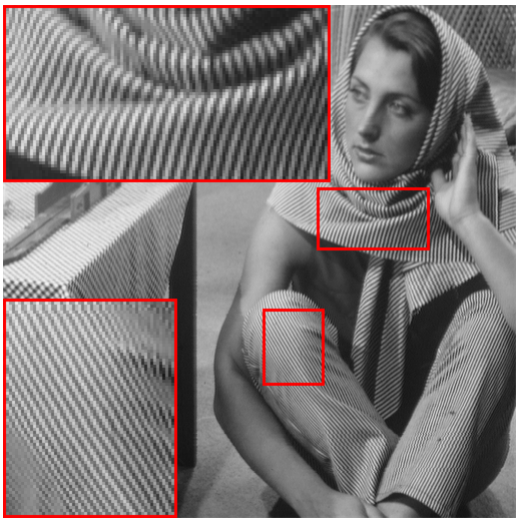
Fig. 2.2 displays these three overcomplete dictionaries of $K = 256$ atoms and the



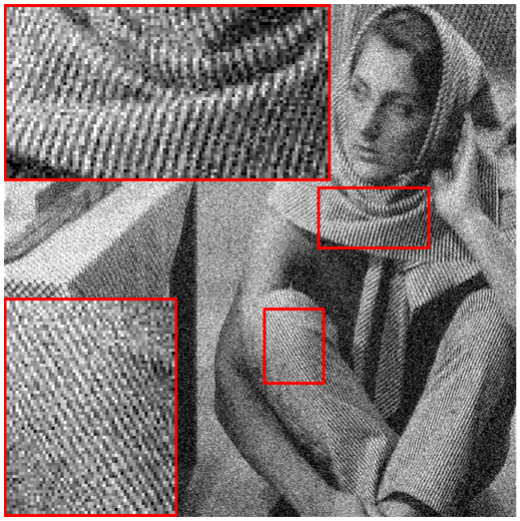
(a) DCT **(b)** Haar

(c) DB4 **(d)** K-SVD (Barbara with $\sigma = 30$)

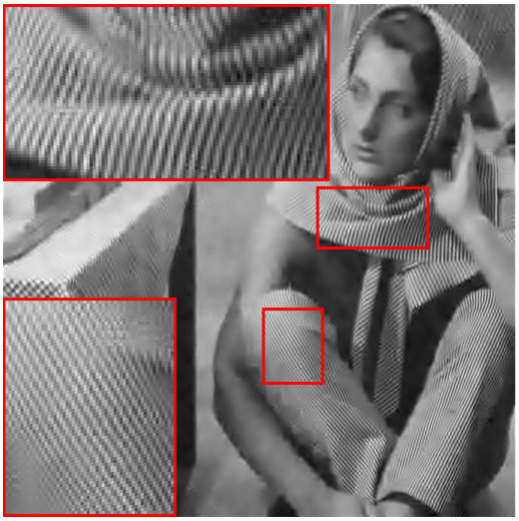
Figure 2.2: *Different types of overcomplete dictionaries of 256 atoms. (a)-(c) Fixed overcomplete dictionaries constructed from Discrete Cosine Transform (DCT) bases, Haar Wavelet Transform bases, and Daubechies D4 Wavelet Transform (Db4) bases. (d) Data-adaptive dictionary trained from noisy image of Barbara with* $\sigma = 30$ *using K-SVD algorithm [2].*



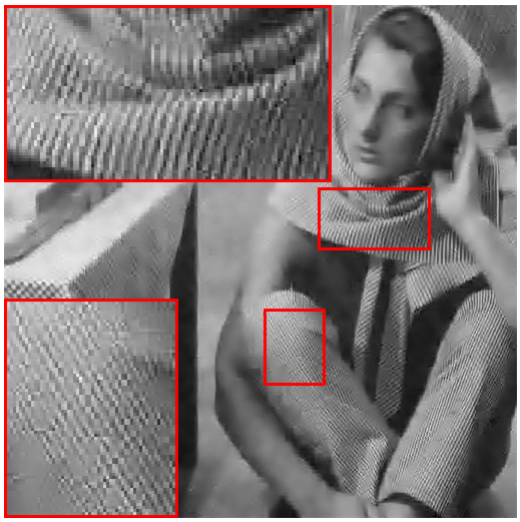
(a) Original image



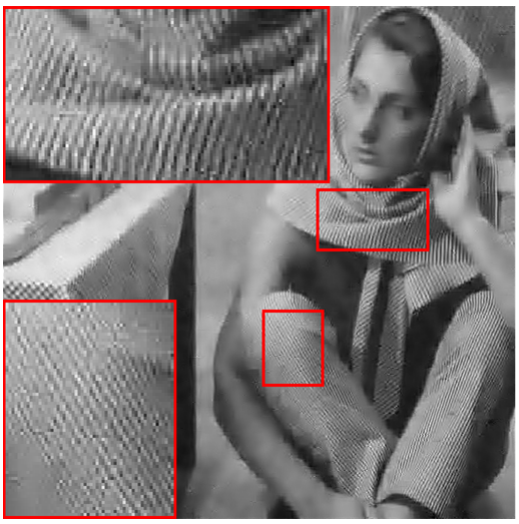
(b) Noisy image at $\sigma = 30$ (18.59 - 0.342)



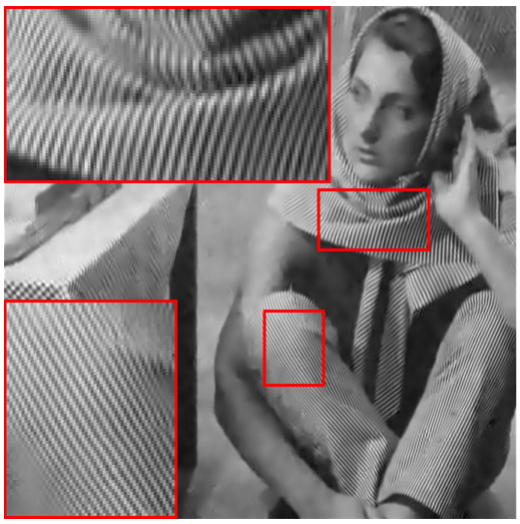
(c) Denoising by DCT dictionary (27.61 - 0.800)



(d) Denoising by Haar dictionary (26.19 - 0.762)

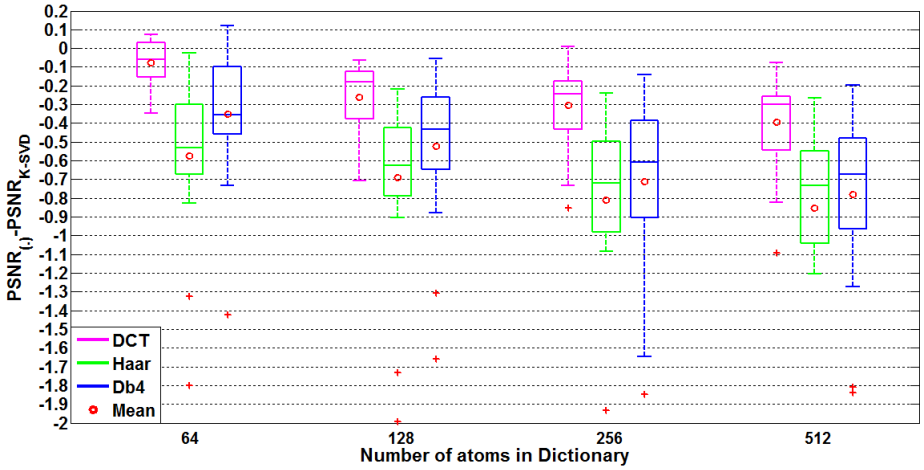


(e) Denoising by Db4 dictionary (26.33 - 0.763)

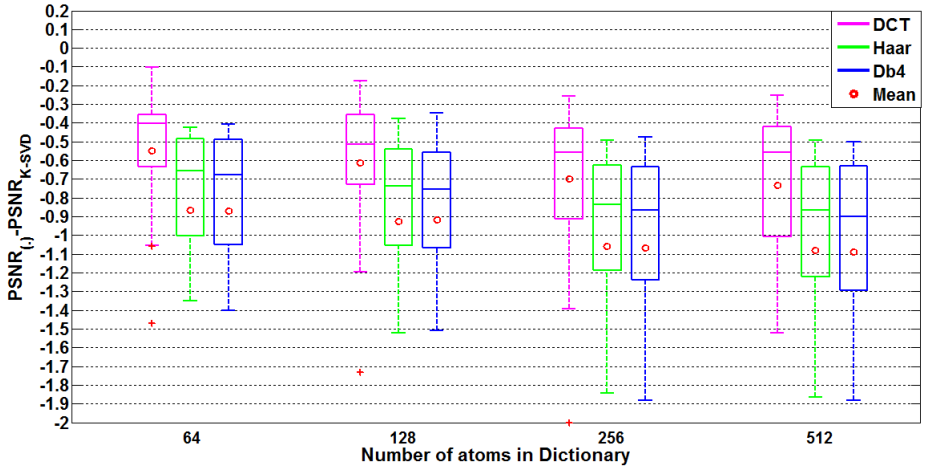


(f) Denoising by K-SVD dictionary (28.65 - 0.824)

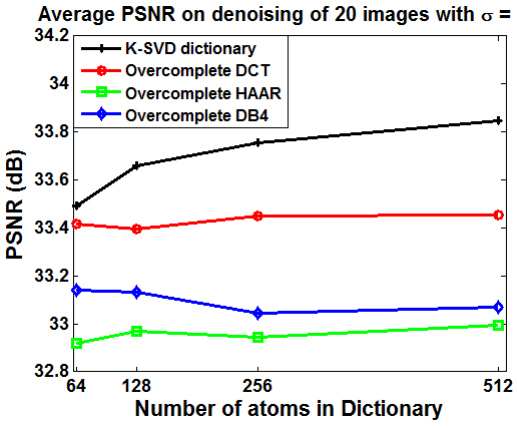
Figure 2.3: *Denoising results on image of Barbara at noise level $\sigma = 30$ using 4 types of dictionaries of 256 atoms shown in Fig. 2.2.*



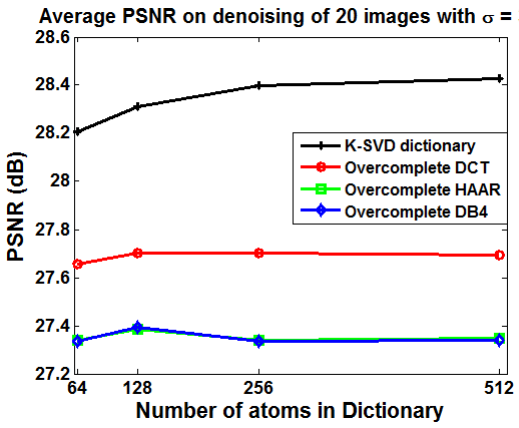
(a) $\sigma = 10$



(b) $\sigma = 30$



(c) $\sigma = 10$



(d) $\sigma = 30$

Figure 2.4: *Denoising performance on 20 natural test images shown in Fig. 2.1 with 4 different types of dictionaries: DCT, Haar, Db4 and K-SVD. Distribution of differences in term of PSNR between denoising with DCT, Haar, Db4 dictionaries and K-SVD dictionary at two noise levels* $\sigma = 10$ *(a) and* $\sigma = 30$ *(b). Average PSNR results on 20 test images with respect to variation of dictionaries' sizes at* $\sigma = 10$ *(c) and* $\sigma = 30$ *(d).*

K-SVD dictionary trained from the image of Barbara in case of noise level $\sigma = 30$. We can observe that the K-SVD dictionary consists of multiple local structures appearing in the test image (Barbara) like diagonal textures, which can not be found in the analytic dictionaries. Fig. 2.3 presents the denoising results with 4 different dictionaries, which demonstrates that the reconstruction using K-SVD dictionary can better preserve the details of the image, as well as produce less artifacts than the others.

Moreover, Fig. 2.4(a)-(b) show the distribution of difference (in term of PSNR) between the reconstruction of 3 analytic dictionaries DCT, Haar, Db4 and the result of K-SVD dictionary on 20 images. It can be seen that for complete dictionary ($K = 64$) and low noise $\sigma = 10$, a large portion of images can be better restored by K-SVD dictionary. Furthermore, the K-SVD surpasses the other dictionaries for the case of overcomplete dictionaries ($K > 64$). In addition, Fig 2.4(c)-(d) show the average PSNR results of 20 images with respect to the number of atoms in the dictionaries which infers that the use of an overcomplete K-SVD dictionary can improve the performance of a denoising method.

We have indicated that a dictionary trained with K-SVD algorithm can promote the reconstruction of a degraded image, comparing to the analytic dictionaries. Therefore, in our studies in chapter 5 and 6, we exploit the K-SVD algorithm to learn a global dictionary for image restoration.

2.7 The Earth Mover's Distance (EMD)

In chapter 3 of this thesis, we present a super-resolution method that take advantage of the redundancy of local information between different images of the same modality to generate a large, exhaustive dictionary that is identical to the database of patches, to recover a degraded low-resolution image $\mathbf{y}$. Hence, for each patch $\mathbf{y}_i$ in the underlying image $\mathbf{y}$, we need to apply a patch-filtering step to create a local dictionary that contains only most similar patches in the database and use it in reconstruction of $\mathbf{y}_i$. Selecting a good local dictionary is the key issue of the image reconstruction algorithm.

In chapter 4, rather than using the conventional Euclidean distance to measure the similarity between two image patches, we consider an image patch $\mathbf{y}_i \in R^m$ as a distribution of gray levels at the pixels' locations and thus we can explore a metric that compares two histograms for image patches comparison. The commonly used metrics in practice such as Minkowski (ℓ_1, ℓ_p, ℓ_2 distances), χ^2, etc., are the bin-to-bin distances which assume that two histograms, e.g. $\mathbf{h}_1$ and $\mathbf{h}_2$, are already aligned and compute the bin-wise differences between them. However this assumption can be violated through

degradation of image, shape deformation, or image translation, etc. Thus, a cross-bin distance is preferred to address the alignment problem.

Among several contribution on cross-bin histogram comparison, the Earth Mover's Distance (EMD) has shown its effectiveness in perceptual evaluation the similarity between two distributions in the literature. The remainder of this subsection will briefly describe the concepts of the EMD, and in particularly one of its improvement version in both accuracy and computational complexity called FastEMD and denoted as $\widehat{EMD}$.

The Earth Mover's Distance is based on the minimal cost that must to be paid to transform one distribution into the other. It has been used in many applications such as image retrieval [130–135], texture and color classification [136], object recognition and matching [137, 138] and image segmentation [139, 140]. The basic concept of EMD was first introduced by Peleg *et al.* in [141] to measure perceptual similarity between two gray images. Each L-gray-level image is considered as a distribution of "pebbles" corresponding to its intensity, placed on the plane only at pixels' locations. Peleg measured the distance between two gray images by proposed a minimization of a linear cost matching function to transfer the pebbles from one image to another. Later, Rubner *et al.* [130] introduced the original EMD to measure the similarity between two normalized histograms. The EMD is a natural and intuitive metric between two histograms if we think one of them as piles of masses and the other as holes sitting on the ground. Each pile of masses or a hole is a bin of the histogram. To quantify the difference between two histograms, we can measure how many masses we should get from one pile and how far to move them so that the holes are exactly filled in by the masses from the piles. The moving distance from a pile to a hole is called the ground distance, and the amount of masses is named the flow. Hence, EMD is the minimal total ground distance traveled weighted by the amount of masses moved.

For more convenience, we denote $\mathbf{h}_1 = [\mathbf{h}_1(1),\ldots,\mathbf{h}_1(i),\ldots,\mathbf{h}_1(m)]^T \in R^m$ to be the first histogram of m bins and $\mathbf{h}_2 = [\mathbf{h}_2(1),\ldots,\mathbf{h}_2(j),\ldots,\mathbf{h}_2(n)]^T \in R^m$ to be the second histogram of n bins. Then the EMD distance between two histograms $\mathbf{h}_1$ and $\mathbf{h}_2$ is defined as:

$$EMD(\mathbf{h}_1,\mathbf{h}_2) = \min_{f_{ij}} \frac{\sum_{i,j} f_{ij} d_{ij}}{\sum_{i,j} f_{ij}} \quad \text{subject to} \quad (2.42) \tag{2.41}$$

$$f_{il} \geq 0, \sum_j f_{ij} \leq \mathbf{h}_1(i), \sum_i f_{ij} \leq \mathbf{h}_2(j), \sum_{i,j} f_{ij} = \min\left(\sum_i \mathbf{h}_1(i), \sum_j \mathbf{h}_2(j)\right) \tag{2.42}$$

where $1 \leq i \leq m, 1 \leq j \leq n$, $\{f_{ij}\}$ represents the amount masses moved from the bin i-th

of $\mathbf{h}_1$ to the bin j-th of $\mathbf{h}_2$. d_{ij} denotes the ground distance between bin i and bin j of two histograms. The constraints in(2.42) can be intuitively understood as the amount of moving masses from a pile to a hole must be a positive value, and can not exceed the available total masses of the pile, or can not be larger than the capacity of the hole. There are two main problems with the original EMD of Rubner. First, it is only designed for normalized histograms and can not be used to evaluate two histograms with different masses ($\sum_i \mathbf{h}_1(i) \neq \sum_j \mathbf{h}_2(j)$). Second, for a general ground distance (e.g. Euclidean distance), it has a high computational time cost.

Based on the proposed model of Rubner [130], many authors developed different EMD distance for image retrieval [134, 142, 143] such as interest point matching using feature descriptors [142], contour matching by embedding the EMD into a normal space [137] or in a Wavelet domain [143]. Their objective is to improve the performances and reduce the time consumption of the EMD in many different ways, such as constructing a tree-based graph [131, 142] or proposed different type of ground distance [134]. However, like the original EMD of Rubner, all of these models are only used for distributions with equal total mass.

In [132], Pele *et al.* proposed a variant of the original EMD model, called $\widehat{EMD}$ to deal with unequal total mass distributions by adding into 2.41 a term of the total mass difference between them, as described in

$$\widehat{EMD}_\alpha(\mathbf{h}_1, \mathbf{h}_2) = \left(\min_{f_{ij}} \sum_{i,j} f_{ij} d_{ij} \right) + \left| \sum_i \mathbf{h}_1(i) - \sum_j \mathbf{h}_2(j) \right| \times \alpha \max_{i,j} d_{ij} \quad \text{subject to} \quad (2.42) \tag{2.43}$$

Pele *et al.* have demonstrated that with $\alpha > 0.5$ the ground distance d_{ij} is a metric, the $\widehat{EMD}$ is a metric distance.

Thereafter, Pele [133] presented a robust EMD algorithm, called $FastEMD$, expanded the model from [132] by using the thresholded ground distances. That means

$$d_{ij} = \begin{cases} d_{ij} & \text{if} \quad d_{ij} \leq d_{max} \\ d_{max} & \text{if} \quad d_{ij} > d_{max} \end{cases} \tag{2.44}$$

As demonstrated in [133], adopting thresholded distances into $\widehat{EMD}$ helps to reduce the number of edges in the flow-network of the $\widehat{EMD}$ and thus it can accelerate the speed of $\widehat{EMD}$. Compare to the original EMD distance of Rubner [130], the metric $\widehat{EMD}$ in [133] improves both accuracy and speed and could be capable to apply on large histograms and databases. The implementation of the $\widehat{EMD}$ can be found at `http://www.ariel.ac.il/sites/ofirpele/fastemd/code/`.

In the context of this thesis, we exploit the $\widehat{EMD}$ as a metric to measure the similarity between image patches and apply it to select a local dictionary for recovering each degraded patch $\mathbf{y}_i$. The details of this process will be described in chapter 4.

CHAPTER 3

Patch-based super-resolution for medical images corrupted by Poisson noise

Abstract

This chapter investigates the patch-based super-resolution method for medical images corrupted by Poisson noise, with the help of a given set of standard images. The signal-dependent noise in the image can be stabilized to a constant variance using the Anscombe transform. After that, for each patch in the noisy low-resolution image, our aim is to find a couple of local dictionaries that contains the nearest neighbor patches from the database and use them to estimate the latent high-resolution patch by computing a regression function based on the construction of a reproducing kernel Hilbert space. To obtain the corresponding local dictionaries, a coarse search using the conventional Euclidean distance is first performed, followed by a refinement using a statistical criterion. The proposed method shows its effectiveness in enhancing the resolution and removing the Poisson noise in a degraded image, comparing to some existing super-resolution methods. Our work was published in the proceedings of the 2015 SPIE Medical Imaging conference[1].

[1] Dai-Viet Tran, Marie Luong, Sébastien Li-Thao-Té, Jean-Marie Rocchisani, Françoise Dibos, Thuong Le-Tien, "Super-resolution for medical images corrupted by heavy noise", Proc. SPIE 9413, Medical Imaging 2015: Image Processing

Chapter content

1 Introduction

The work presented in this chapter is inspired apart from my master's internship at the L2TI and LAGA labs, where I studies the performance of an available super-resolution method[2] developed by the colleagues of the laboratories. This method is the SRSW, which was proposed for recovering a degraded low-resolution image corrupted by additive Gaussian noise, with the support of a set of standard images (same modality, high-resolution, good quality images). In this chapter, we investigate a novel super-resolution method based on regression while inspiring the similar idea of using the standard images for enhancing the spatial resolution of a degraded image but for the case of Poisson noise. Our aim is to estimate a latent high-resolution image $\mathbf{x} \in R^N$ from its degraded observation $\mathbf{y} \in R^M$, which is assumed to be generated by the model:

$$\mathbf{y} = \xi \mathcal{P}(\mathbf{SBx}/\xi) \tag{3.1}$$

where $\mathcal{P}(\cdot)$ is the Poisson distribution law, $\mathbf{S}$ is the downsampling operator with the magnification factor s, $\mathbf{B}$ is the blur operator and ξ is a constant using to control the Poisson noise level (the higher value of ξ, the heavier the noise affects the image).

In this thesis, we investigate the patch-based approach for recovering a degraded image $\mathbf{y}$, in which the image is considered as a set of overlapping image patches $\{\mathbf{y}_i | \mathbf{y}_i \in R^m; i = 1, \ldots, M\}$. The model in (3.1) can be formulated on each image patch as:

$$\mathbf{y}_i = \xi \mathcal{P}(\mathbf{S}_i \mathbf{B}_i \mathbf{x}_i / \xi) \tag{3.2}$$

where $\mathbf{S}_i$ and $\mathbf{B}_i$ denote the downsampling and blurring operators on each image patch, $\mathbf{x}_i$ is the high-resolution version of $\mathbf{y}_i$. The fundamental proposals of our work are based on two significant characteristics. First, because Poisson noise in (3.2) is a signal dependent noise model, it is difficult to separate the noise from the true image. Fortunately, in [144], Anscombe demonstrated that we can stabilize the noise variance in the Poisson distribution by applying a nonlinear transform, which is then named as Anscombe transform, on the signal as:

$$\mathcal{T}(\mathbf{y}_i) = \sqrt{\mathbf{y}_i + \frac{3}{8}} \tag{3.3}$$

As a result, the Poisson distribution of image patch is converted into the approximate additive standard normal and thus we can more easily reduce the noise in image patch by using a framework for Gaussian noise removal. Second, as mentioned in chapter 1,

[2] Super-Resolution by Sparse Weights (SRSW [1]). Please refer to section 2.1 of chapter 2 for more details.

we exploit the redundancy of local information across different images of the same modality to collect a large couple of high- and low-resolution dictionaries $\mathbf{D}^h \in R^{n\times P}$ and $\mathbf{D}^l \in R^{m\times P}$ of P image patches from the list of standard images. However, recovering an image patch $\mathbf{y}_i$ with given large dictionaries is a high time-consuming process. Therefore, similar to the existing super-resolution methods [1, 74, 145], we only select a smaller couple of local dictionaries $\{\mathbf{D}_i^h \in R^{n\times K}, \mathbf{D}_i^l \in R^{m\times K}, K \ll P\}$ that match the image patch $\mathbf{y}_i$ for reconstruction. In this work, we propose to learn a regression function (mapping) between the low-resolution patches and high-resolution patches in a reproducing kernel Hilbert space (RKHS) generated from the atoms of local dictionaries, and then use it to map the degraded low-resolution patch $\mathbf{y}_i$ into the high-resolution space to estimate the latent high-resolution patch $\mathbf{x}_i$. The proposed method is preferred to as SRRH which stands for Super-Resolution by Regression function in a reproducing kernel Hilbert space.

The rest of this chapter is organized as follows. Section 2 presents our proposed method for super-resolution of image corrupted by Poisson noise. In section 3, we conduct some experimental tests to evaluate the performance of the proposed method in comparison with other existing methods. The conclusion and future works are presented in section 5.

2 Proposed SRRH method for image super-resolution

The proposed method consists of two phases: construction of dictionaries and super-resolution.

2.1 Construction of the dictionaries

In the first stage, we construct a couple of high- and low-resolution dictionaries $\{\mathbf{D}^h, \mathbf{D}^l\}$ from the available standard images $\{\mathbf{x}^s\}$. We randomly extract a set of large P high-resolution patches $\{\mathbf{x}_k^s | \mathbf{x}_k^s \in R^n; k = 1, \ldots, P\}$ from the standard images. After that, these high-resolution patches are blurred and downsampled to obtain their corresponding low-resolution versions $\{\mathbf{y}_k^s | \mathbf{y}_k^s \in R^m; k = 1, \ldots, P\}$. The couple of high- and low-resolution dictionaries $\mathbf{D}^h$ is determined to be identical to the database of high- and low-resolution patches.

$$\begin{aligned} \mathbf{D}^h &= \left\{ \mathbf{d}_k^h | \mathbf{d}_k^h \equiv \mathbf{x}_k^s; k = 1, \ldots, P \right\} \\ \mathbf{D}^l &= \left\{ \mathbf{d}_k^l | \mathbf{d}_k^l \equiv \mathbf{y}_k^s; k = 1, \ldots, P \right\} \end{aligned} \tag{3.4}$$

2.2 Image super-resolution

For each patch $\mathbf{y}$ in the degraded low-resolution image $\mathbf{y}$, we first select a couple of local dictionaries $\{\mathbf{D}_i^h \in R^{n\times K}, \mathbf{D}_i^l \in R^{m\times K}, K \ll P\}$ of K most similar atoms from the large dictionaries $\{\mathbf{D}^h, \mathbf{D}^l\}$ using the following steps [50].

i. Coarse search: We find a subset of L couples of closest patches from the dictionaries with the help of the Euclidean distance.

$$\begin{aligned}\mathbf{D}_*^l &= \left\{\mathbf{d}_j^l \in \mathbf{D}^l \middle| \left\|\mathbf{y}_i - \mathbf{d}_j^l\right\|_2 < r_i\right\} \\ \mathbf{D}_*^h &= \left\{\mathbf{d}_j^h \in \mathbf{D}^h \middle| \left\|\mathbf{y}_i - \mathbf{d}_j^l\right\|_2 < r_i\right\}\end{aligned} \tag{3.5}$$

where r_i is a positive constant is chosen for each patch $\mathbf{y}_i$ such that only the first L most matched atoms $\mathbf{d}_j$ with smallest Euclidean distance to $\mathbf{y}_i$ are acquired.

ii. Refined search: We apply the Anscombe transform as (3.3) on both low-resolution patches in $\mathbf{D}_*^l$ and $\mathbf{y}_i$ to convert the Poisson noise in $\mathbf{y}_i$ into approximately the additive standard normal noise. After that, we keep only $K(K < L)$ couples of patches in $\{\mathbf{D}_*^l, \mathbf{D}_*^h\}$ which have the smallest values of the statistical criterion $\mathcal{V}_j = \left|E\left(\mathcal{T}(\mathbf{y}_i) - \mathcal{T}(\mathbf{d}_j^l)\right)\right| + \left|Var\left(\mathcal{T}(\mathbf{y}_i) - \mathcal{T}(\mathbf{d}_j^l)\right) - 1\right|$ to create the local dictionaries.

$$\begin{aligned}\mathbf{D}_i^l &= \left\{\mathbf{d}_j^l \in \mathbf{D}_*^l | \mathcal{V}_j < t_i\right\} \\ \mathbf{D}_i^h &= \left\{\mathbf{d}_j^h \in \mathbf{D}_*^h | \mathcal{V}_j < t_i\right\}\end{aligned} \tag{3.6}$$

where t_i is a set such that only the first K most matched atoms are selected.

In the next stage, we construct a linear regression mapping function between the low-resolution space of image patch and their corresponding high-resolution space, denoted as $f_{SR}(\cdot)$, from the low- and high-resolution atoms in the local dictionaries. Hence, the estimation of the high-resolution latent patch $\mathbf{x}_i$ can be found from its degraded version $\mathbf{y}_i$ by using the mapping function as $\mathbf{x}_i = f_{SR}(\mathbf{y}_i)$.

The linear regression function $\mathbf{f}_{SR}(\cdot)$ is built in a reproducing kernel Hilbert space H_K generated from the vector space of low-resolution atoms in local dictionary $\mathbf{D}_i^l$, which is defined in [146] as:

$$H_K = \left\{ f_{SR}(\mathbf{z_i}) \mid f_{SR}(\mathbf{z}_i) = \sum_{j=1}^{K} \kappa\left(\mathbf{z}_i, \mathbf{d}_j^l\right)\boldsymbol{\alpha}_j; \mathbf{z}_i \in R^m; \boldsymbol{\alpha}_j \in R^n; j = 1, \ldots, K \right\} \tag{3.7}$$

where $\kappa(\mathbf{z}_i, \mathbf{d}_j^l) = \exp\left(-\left\|\mathbf{z}_i - \mathbf{d}_j^l\right\|_2^2 / 2h^2\right)$ is the Gaussian kernel function and h is a decay parameter. The regression function $f_{SR}(\cdot)$ is learned from the couple of local

dictionaries $\{\mathbf{D}_i^h, \mathbf{D}_i^l\}$ by minimizing Tikhonov regularization as:

$$\hat{f}_{sr} = \underset{f_{SR} \in H_K}{\arg\min} \left\{ \sum_{j=1}^{K} \left\| \mathbf{d}_j^h - f_{SR}(\mathbf{d}_j^l) \right\|_2^2 + \lambda \left\| f_{SR} \right\|_{H_K}^2 \right\} \tag{3.8}$$

where λ is a regularization parameter.

Replacing f_{SR} in(3.8) by its definition from (3.7), the problem in (3.8) is equivalent to finding a set of basic vector $\{\boldsymbol{\alpha}_j \in R^n; j = 1, \ldots, K\}$ such that:

$$\{\hat{\boldsymbol{\alpha}}_j\} = \underset{\boldsymbol{\alpha}_j}{\arg\min} \left\{ \sum_{j=1}^{K} \left\| \mathbf{d}_j^h - \sum_{k=1}^{K} \kappa\left(\mathbf{d}_j^l, \mathbf{d}_k^l\right) \boldsymbol{\alpha}_k \right\|_2^2 + \lambda \sum_{j=1}^{K} \left\| \boldsymbol{\alpha}_j \right\|_2^2 \right\} \tag{3.9}$$

If we denote $\mathbf{R} = \{\mathbf{R}(j,k) | \mathbf{R}(j,k) = \kappa(\mathbf{d}_j^l, \mathbf{d}_k^l); 1 \leq j,k \leq K; \mathbf{d}_j^l, \mathbf{d}_k^l \in \mathbf{D}_i^l\}$ to be a $K \times K$ matrix of regression coefficients, and $\mathbf{A}^i = \{\boldsymbol{\alpha}_j | \boldsymbol{\alpha}_j \in R^n; j = 1, \ldots, K\} \in R^{n \times K}$ to be the matrix of basic vectors $\{\boldsymbol{\alpha}_j\}$. The optimization problem in (3.9) can be written as:

$$\hat{\mathbf{A}}^i = \underset{\mathbf{A}^i}{\arg\min} \left\{ \left\| \mathbf{D}_i^h - \mathbf{A}^i \mathbf{R} \right\|_2^2 + \lambda \left\| \mathbf{A}^i \right\|_2^2 \right\} \tag{3.10}$$

The problem (3.10) has a close-form solution described in (3.11)

$$\hat{\mathbf{A}}^i = \mathbf{D}_i^h \mathbf{R}^T \left(\mathbf{R}\mathbf{R}^T + \lambda \mathbf{I} \right)^{-1} \tag{3.11}$$

After obtaining the optimal values of basic vectors $\{\hat{\boldsymbol{\alpha}}_j \in \hat{\mathbf{A}}^i\}$, we can achieve an estimation $\hat{\mathbf{x}}_i$ of the latent high-resolution image patch $\mathbf{x}_i$ from its low-resolution degraded version $\mathbf{y}_i$ by the linear regression mapping:

$$\hat{\mathbf{x}}_i = f_{SR}(\mathbf{y}_i) = \sum_{j=1}^{K} \kappa(\mathbf{y}_i, \mathbf{d}_j^l) \hat{\boldsymbol{\alpha}}_j \tag{3.12}$$

After performing the super-resolution on every low-resolution patches $\mathbf{y}_i$, the entire latent high-resolution image $\mathbf{x}$ can be found by putting the super-resolved patches $\hat{\mathbf{x}}_i$ on their proper locations in the image and averaging the overlapping regions between adjacent patches.

3 Performance evaluation

In this section, we compare the performance of the proposed method with some other existing super-resolution methods, including the Bicubic interpolation, the Neighbor Embedding super-resolution (NE [74]) and the sparsity super-resolution method (ScSR [3]).

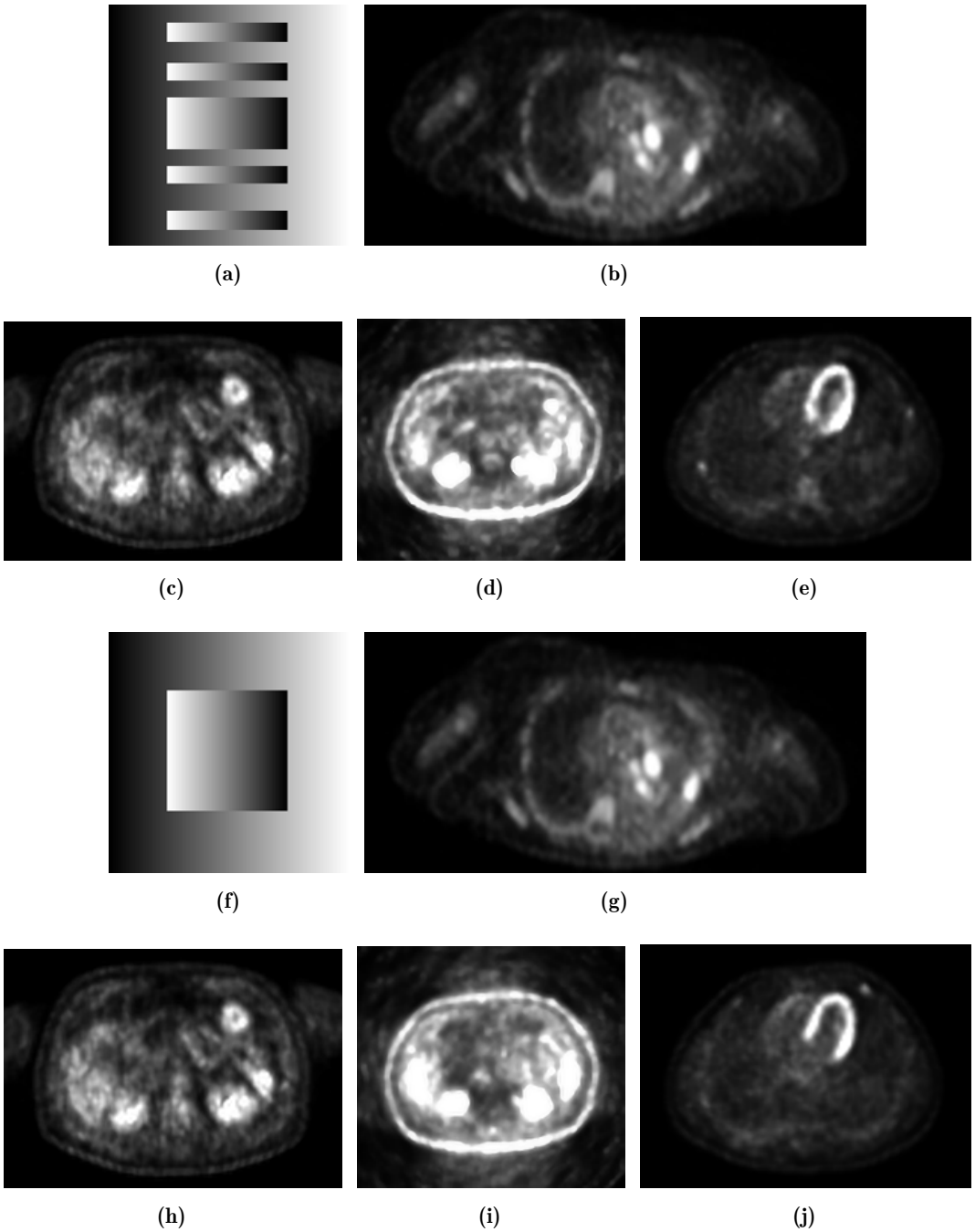


Figure 3.1: *(a)-(e) Standard image and (f)-(j) test images. (a) & (f) synthetic images, (b) & (g) PET images of thorax (pet1), (c) & (h) PET images of abdomen in the area of the kidney (pet2), (d) & (i) PET images of abdomen in the area of the kidney (pet3), (e) & (j) PET images of thorax through the lungs and heart (pet4).*

Image	Noise scale ξ	PSNR				SSIM			
		Bicubic	NE	ScSR	SRRH	Bicubic	NE	ScSR	SRRH
synthetic	1	28.39	29.73	**36.08**	33.03	0.613	0.934	**0.948**	0.933
	3	24.47	28.89	**31.54**	30.22	0.411	0.884	**0.891**	0.859
	5	22.51	28.02	**29.34**	28.72	0.327	0.844	**0.845**	0.799
pet1	1	35.29	34.77	37.53	**41.55**	0.901	0.947	0.955	**0.979**
	3	30.68	33.66	35.20	**37.06**	0.782	0.920	0.933	**0.946**
	5	28.39	32.67	33.98	**35.05**	0.704	0.906	0.919	**0.920**
pet2	1	33.25	29.50	33.47	**37.64**	0.915	0.876	0.921	**0.971**
	3	28.83	28.83	30.60	**34.98**	0.811	0.858	0.873	**0.943**
	5	26.79	28.31	29.69	**32.42**	0.742	0.836	0.855	**0.908**
pet3	1	30.35	26.83	**31.12**	30.86	0.871	0.891	0.922	**0.932**
	3	25.87	26.15	27.72	**29.05**	0.718	0.863	0.876	**0.891**
	5	23.89	25.55	26.58	**27.79**	0.625	0.836	0.853	**0.855**
pet4	1	36.42	35.81	40.29	**41.23**	0.901	0.964	0.977	**0.978**
	3	31.87	34.80	37.45	**38.22**	0.790	0.946	**0.963**	0.954
	5	29.65	33.92	35.90	**36.65**	0.718	0.933	**0.952**	0.939

Table 3.1: *PSNR and SSIM comparison for SR with magnification factor* $s = 2$

The summaries of NE and ScSR methods can be found in section 2 of chapter 2. The two image quality assessment metrics PSNR and SSIM are used for objective evaluation.

We carry out experiments on one synthetic image and four Positron Emission Tomography (PET) images as shown in Fig. 3.1(f)-(j) with different levels of Poisson noise $\xi = 1, 3, 5$. We consider them as the high-resolution images and create the corresponding low-resolution degraded images using the model in (3.1), with the 7×7 Gaussian kernel blur of standard deviation $\sigma = 1$, and the downsampling with magnigication factor $s = 2$. For each test image, we use corresponding standard image displayed in Fig. 4.4(a)-(e), to construct the couple of high- and low-resolution dictionaries $\{\mathbf{D}^h, \mathbf{D}^l\}$ of $P = 30000$ patches according to section 2.1. The size of high-resolution and low-resolution patches are set to 9×9 and 5×5 pixels, respectively. The number of atoms in local dictionaries is set to $K = 5$, and the value of L in the coarse search step is chosen as $L = 5 \times K$. The regularization parameter in (3.8) is set to $\lambda = 10^{-5}$, and the decay parameter is chosen as $h = 1000$.

For the ScSR method, the value of regularization parameter λ is set to 0.8, and the size of dictionary is 1024. The number of nearest neighbor of the NE method is $K = 5$.

Table 3.1 presents the results of competing super-resolution methods, where the best

values of PSNR and SSIM are in bold red numbers. We can observe that the proposed method achieve superior performances for the PET images.

Fig. 3.2 introduces the super-resolution results on the synthetic image with noise level $\xi = 3$. For the proposed method, we shows two results with the low-resolution patch size is 5×5 and 7×7 in Fig. 3.2(f) and (g). We can see the competing method (bicubic, NE, ScSR) recover the image with many artifacts. With the default patch size 5×5, our method produces some artifacts at the central of the image. When we increase the patch size to 7×7, the proposed method achieves better super-resolution performance and generates very pleasant result.

We also show the super-resolution results on PET images in Fig. 3.3 - Fig. 3.5. It can be seen that the proposed method yields better reconstructed image with much preserved details and closer to the original images that the competing methods.

4 Empirical Study on Parameters

In this section, we will discover the effects of some parameters on the performance of the proposed method, including the number of atoms K in the couple of local dictionaries (which is also the nearest neighbors of patches in the dictionary for each noisy image $\mathbf{y}_i$) and the size of patches. Note that the magnification factor is always set to $s = 2$.

4.1 Effect of number of atoms K in local dictionaries

We will study the variation of super-resolution performance in term of PSNR of five test images in Fig. 3.1(f)-(j) with respect to the number nearest neighbors K. We present in Fig. 3.6 experiments on these test images for several values of K in [1, 5, 10, 15, 20, 30, 50, 100], and at two noise levels $\xi = 1$ and $\xi = 5$. We can see that the choice of K depends on the image content and noise level. In overall, the range of K yielding good quality results of super-resolution is around 5 to 10, for both cases of noise.

4.2 Effect of patch size

We also examine the super-resolution performance of the proposed method in the variation of size of low-resolution patch $\sqrt{m} \times \sqrt{m}$. Fig. 3.7 presents the simulated results in term of PSNR with respect to various sizes of patch, $\sqrt{m} \in [3, 5, 7, 9, 11, 19]$ at three noise levels $\xi = 1, 3, 5$. We can observe show that the choice of m highly depends on the image content. There exists a range of patch sizes $\sqrt{m} \in [5, 11]$ providing optimal PSNR indices. In particularly, for the synthetic image, the LR patch size of the synthetic

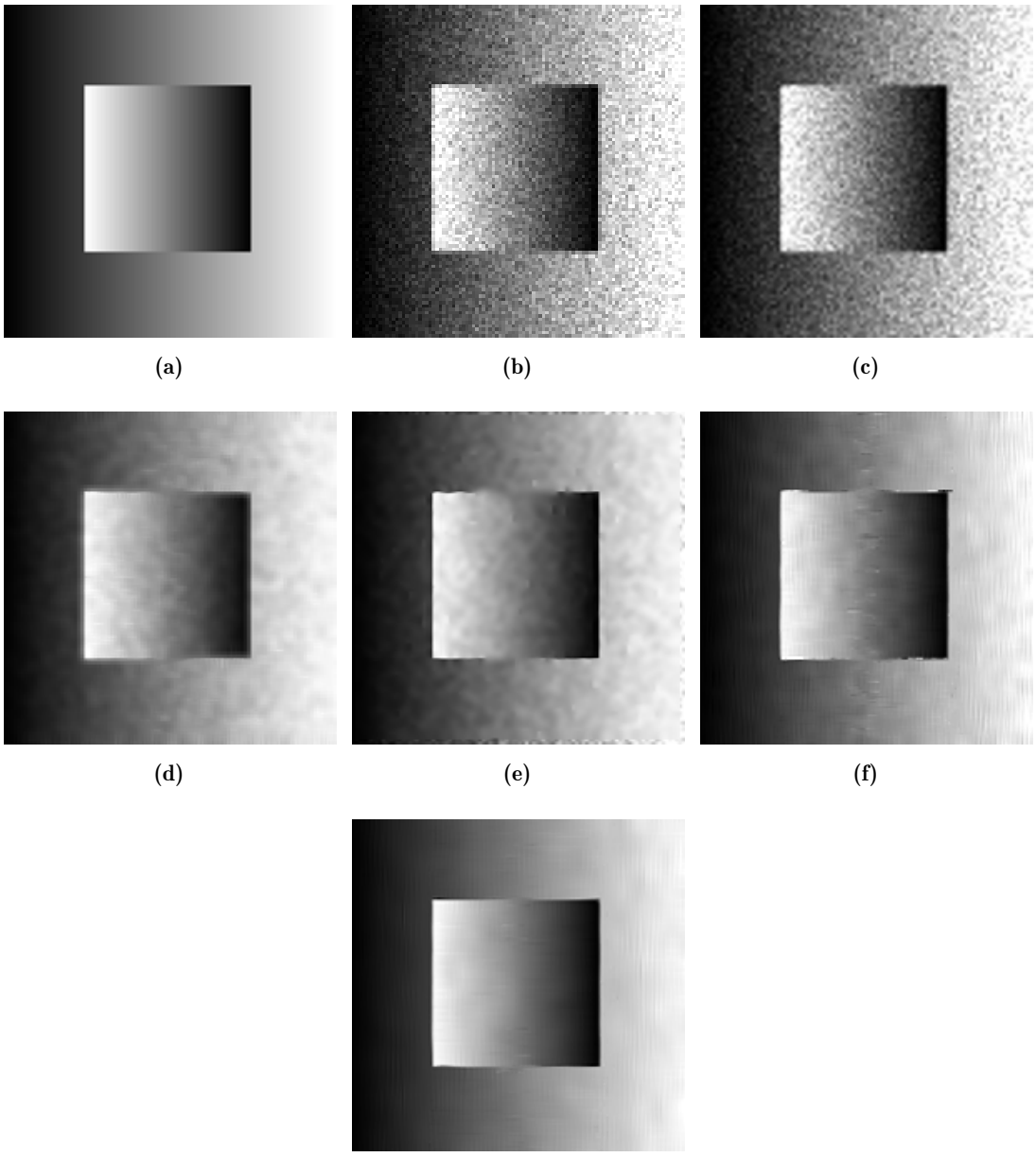


Figure 3.2: *Super-resolution results on synthetic image with noise level $\xi = 3$. (a) Original high-resolution image. (b) The low-resolution noisy image (shown with nearest neighbor interpolation. (c) Result of bicubic interpolation (PSNR = 24.47, SSIM = 0.411). (d) Result of NE method (PSNR = 28.89, SSIM = 0.884). (e) Result of ScSR method (PSNR = 31.54, SSIM = 0.891). (f) Result of the proposed method with low-resolution patch size 5×5 (PSNR = 30.22, SSIM = 0.859). (g) Result of the proposed method with low-resolution patch size 7×7 (PSNR = 32.30, SSIM = 0.937).*

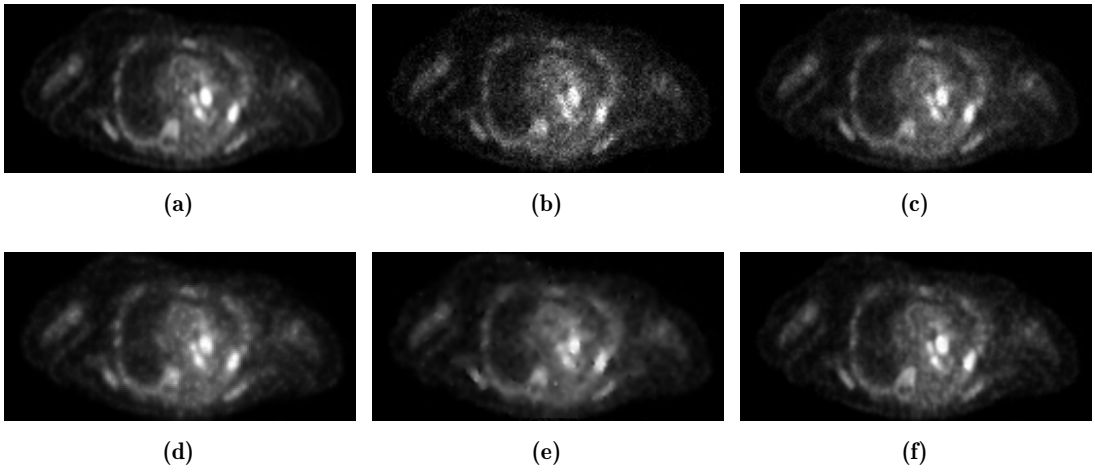


Figure 3.3: *Super-resolution results on PET image of thorax with noise level $\xi = 5$. (a) Original high-resolution image in Fig. 3.1(g). (b) The low-resolution noisy image (shown with nearest neighbor interpolation. (c) Result of bicubic interpolation (PSNR = 28.39, SSIM = 0.704). (d) Result of NE method (PSNR = 32.67, SSIM = 0.906). (e) Result of ScSR method (PSNR = 33.98, SSIM = 0.919). (f) Result of the proposed method (PSNR = 35.05, SSIM = 0.920).*

image yielding good SR results is 7×7 pixels. With this patch size, the proposed method obtains higher PSRN and SSIM values (PSNR = 34.58, SSIM = 0.972 for $\xi = 1$; PSNR = 32.30, SSIM = 0.937 for $\xi = 3$ and PSNR = 30.77, SSIM = 0.918 for $\xi = 5$) than the results of 5×5 patch size in Table 3.1.

5 Conclusion

We present in this chapter an effective patch-based method that takes into account the repetition of image patches across multiple images of the same modality to improve the spatial resolution of a given noisy and low-resolution image. By selecting only a small set of K nearest neighbors from the large dictionary, which is identical to the database of patches, the super-resolution is performed via learning a regression function in the reproducing kernel Hilbert space. Comparing to other state-of-the-art super-resolution algorithms, the proposed method achieves better results, especially in case of heavy Poisson noise.

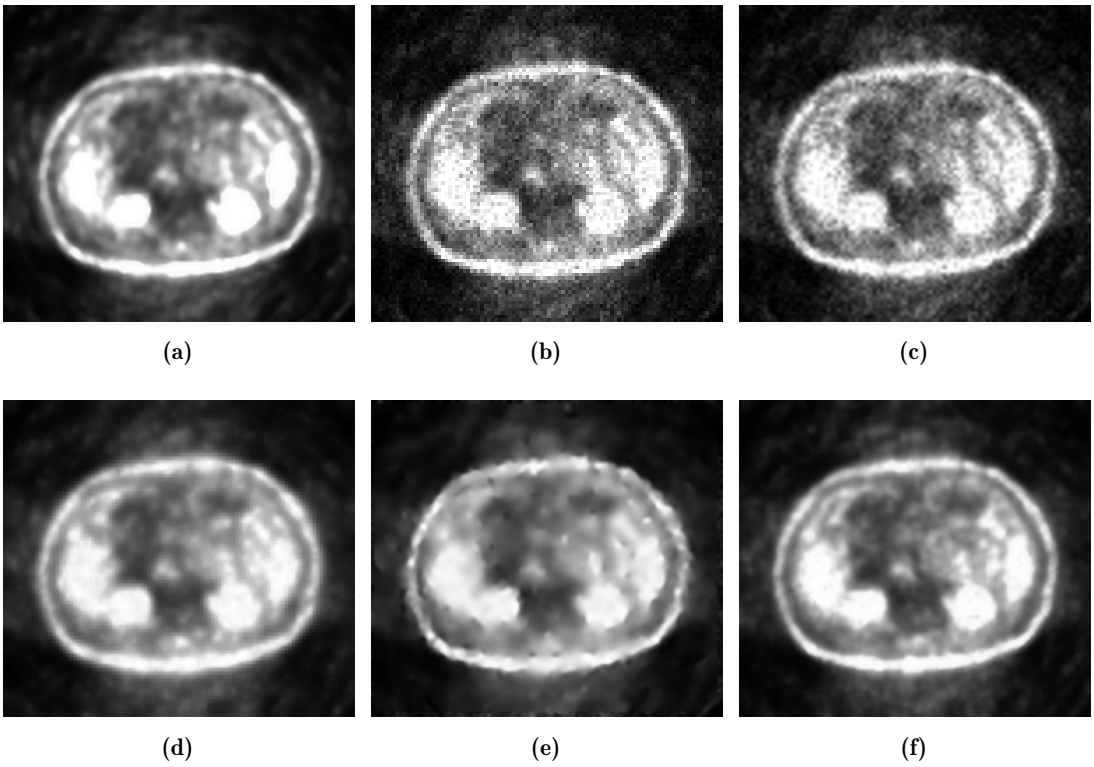
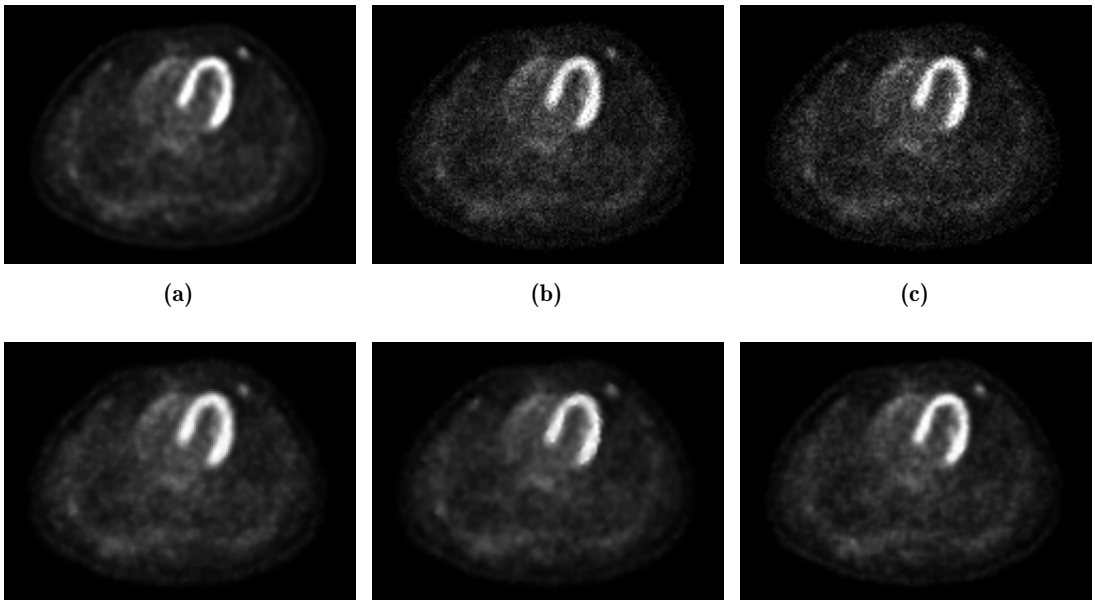


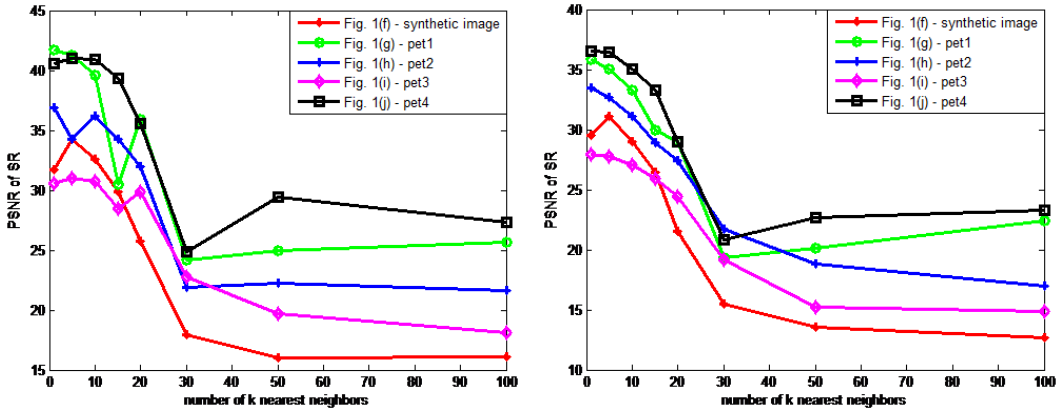
Figure 3.4: *Super-resolution results on PET image of abdomen with noise level $\xi = 3$. (a) Original high-resolution image in Fig. 3.1(i). (b) The low-resolution noisy image (shown with nearest neighbor interpolation. (c) Result of bicubic interpolation (PSNR = 25.87, SSIM = 0.718). (d) Result of NE method (PSNR = 26.15, SSIM = 0.863). (e) Result of ScSR method (PSNR = 27.72, SSIM = 0.876). (f) Result of the proposed method (PSNR = 29.05, SSIM = 0.891).*



(a) (b) (c)

(d) (e) (f)

Figure 3.5: *Super-resolution results on PET image of thorax with noise level $\xi = 3$. (a) Original high-resolution image in Fig. 3.1(j). (b) The low-resolution noisy image (shown with nearest neighbor interpolation. (c) Result of bicubic interpolation (PSNR = 29.65, SSIM = 0.718). (d) Result of NE method (PSNR = 33.92, SSIM = 0.933). (e) Result of ScSR method (PSNR = 35.90, SSIM = 0.952). (f) Result of the proposed method (PSNR = 36.65, SSIM = 0.939).*



(a) (b)

Figure 3.6: *Effect of number of atoms K of the local dictionaries with noise levels $\xi = 1$ (a) and $\xi = 5$ (b).*

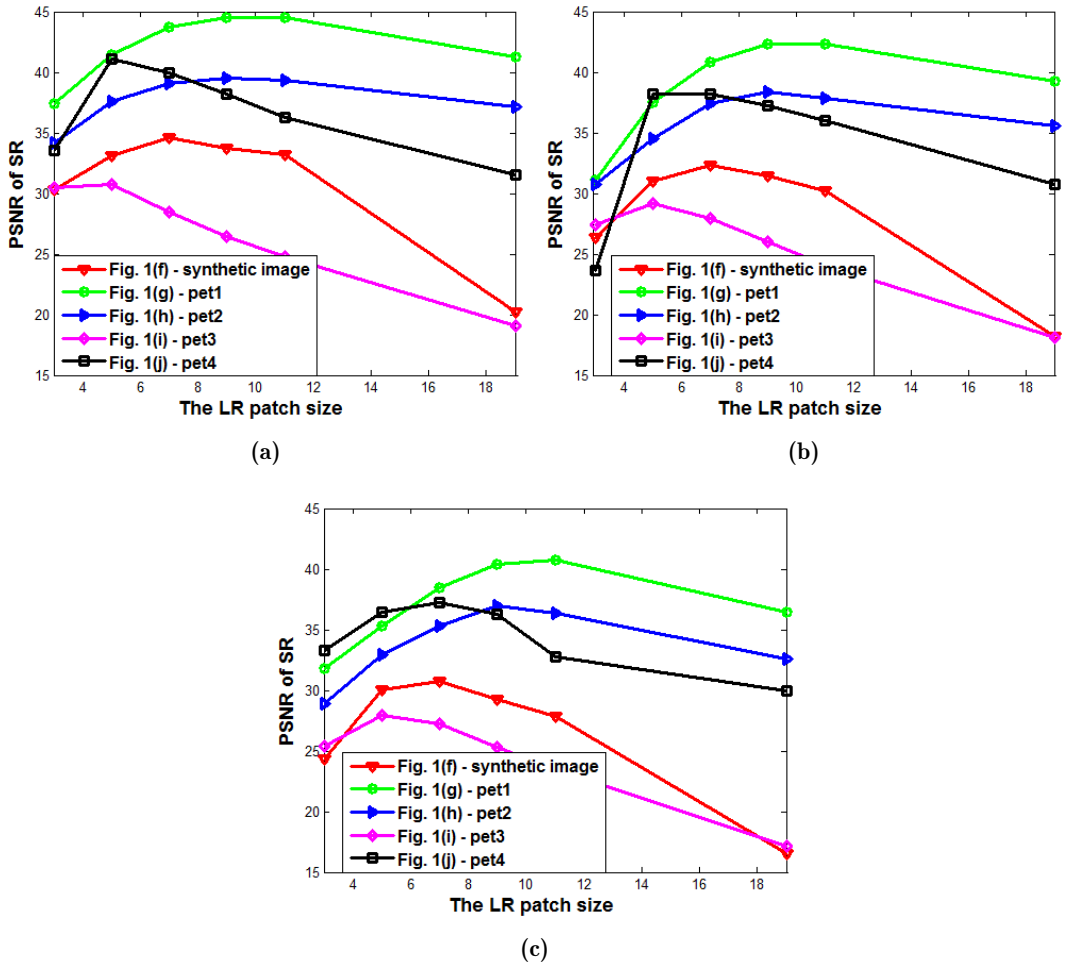


Figure 3.7: *Effect of the size of low-resolution patch* ($\sqrt{m} \times \sqrt{m}$) *with noise levels* $\xi = 1$ *(a),* $\xi = 3$ *(b) and* $\xi = 5$ *(c).*

CHAPTER 4

Example-based super-resolution for enhancing spatial resolution of medical images

Abstract

This chapter investigates the selection of a set of similar patches (called local dictionary) from a large dictionary, which is identical to the database of patches, to recover a degraded image patch under the Bayesian MAP and sparsity framework. In the literature, the bin-to-bin metrics, e.g. Euclidean distance, are conventionally used to measure the similarity between two image patches. However, these metrics assume that two image patches are well aligned and thus fail to deal with the distortion between two patches such as the translation. In this work, we proposed to adopt an efficient cross-bin metric named the Earth Mover's Distance (EMD) to evaluate the similarity between two image patches, by considering each patch as a distribution of image intensities. We introduce an ℓ_1-based threshold on the EMD to dynamically select the number of similar patches in the local dictionary for reconstructing each degraded image patch. Experiments on different modalities of medical images have demonstrated the efficiency of the EMD as a patch-selection metric for image restoration. A detailed description of our proposed method was published in the 2016 International Conference of the IEEE Engineering in Medicine and Biology Society[1]

[1]D. V. Tran, S. Li-Thiao-Té, M. Luong, T. Le-Tien, F. Dibos and J. M. Rocchisani, "Example-based super-resolution for enhancing spatial resolution of medical images", 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2016, pp. 457-460.

Chapter content

1 Introduction

This chapter handles the super-resolution problem to reconstruct a latent high-resolution image $\mathbf{x} \in R^N$ from its degraded observation $\mathbf{y} \in R^M$, which is assumed to be generated by the following model (derived from (1.1)).

$$\mathbf{y} = \mathbf{SBx} + \boldsymbol{\eta} \tag{4.1}$$

where $\mathbf{S}$ is the downsampling operator with magnification factor s, $\mathbf{B}$ is the blur operator and $\boldsymbol{\eta} \sim \mathcal{N}(0, \sigma^2)$ is the additive Gaussian noise of zero mean and standard deviation σ.

In the context of our thesis, we investigate the patch-based approach in which an image $\mathbf{y}$ is considered as a set of overlapping patches $\{\mathbf{y}_i | \mathbf{y}_i \in R^m, i = 1, \ldots, M\}$ and the super-resolution is performed on each patch $\mathbf{y}_i$ by finding its latent high-resolution version $\mathbf{x}_i$. Without loss of generality, we can represent $\mathbf{x}_i$ in a vector space created by the atoms of a dictionary of patches $\mathbf{D}^h$, denoted as $\mathbf{x}_i = \mathbf{D}^h \boldsymbol{\alpha}_i$, with $\boldsymbol{\alpha}_i$ is called the representation coefficients vector. Under the Bayesian MAP perspective as shown in (1.5), the restoration of $\mathbf{y}_i$ is corresponding to find the representation vector $\boldsymbol{\alpha}_i$ which satisfies:

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\{ \frac{1}{2} ||\mathbf{y}_i - \mathbf{S}_i \mathbf{B}_i \mathbf{D}^h \boldsymbol{\alpha}_i||_2^2 - \lambda \log p(\boldsymbol{\alpha}_i) \right\} \tag{4.2}$$

where $\mathbf{S}_i$, $\mathbf{B}_i$ denote the degradation factors on image patch $\mathbf{y}_i$.

In order to find the solution of (4.2), we need to know the prior model of the probability distribution $p(\boldsymbol{\alpha}_i)$. In this work, we adopt the sparse model where $p(\boldsymbol{\alpha}_i)$ is assumed to obey the Laplacian distribution, described as $p(\boldsymbol{\alpha}_i) \propto \exp(-\tau \|\boldsymbol{\alpha}_i\|_1)$. In addition, we define $\mathbf{D}^l = \mathbf{S}_i \mathbf{B}_i \mathbf{D}^h$ to be the dictionary of low-resolution patches, the problem in (4.2) is formulated as:

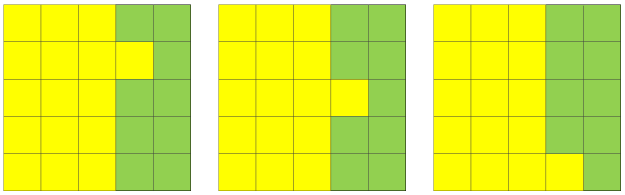
$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\{ \frac{1}{2} ||\mathbf{y}_i - \mathbf{D}^l \boldsymbol{\alpha}_i||_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1 \right\} \tag{4.3}$$

An important issue in our approach is to determine the couple of high- and low-resolution dictionaries $\{\mathbf{D}^h, \mathbf{D}^l\}$. In this chapter, we adopt the similar idea of chapter 3 by exploiting the redundancy of information in multiple images to construct a couple of of high- and low-resolution dictionaries identical to the large database of patches collected from a list of standard images. In addition, we acquire a large number of atoms (patches) in dictionaries to ensure that they can cover all the local structure of the degraded image $\mathbf{y}$. However, one shortcoming of this approach is the computational challenge when solving (4.3) with a huge dictionary $\mathbf{D}^l$. To overcome this issue, the conventional solution is to apply a patch-filtering step to select a couple of local dictionaries $\{\mathbf{D}_i^h, \mathbf{D}_i^l\}$

for each degraded patch $\mathbf{y}_i$, on which we can perform the patch recovering. Therefore, retrieving the local dictionaries is crucial in the success of our super-resolution method. In the existing methods [1, 74], the common Euclidean distance is adopt for image patch selection.

As indicated in section 2.7 of chapter 2, the Earth Mover's Distance (EMD) developed by Pele *et al.* [133], denoted as $\widehat{EMD}$, is an efficient cross-bin metric for measure the distance between two distributions or histograms. Hence, we consider that each image patch to be a histogram of gray levels placed at pixels' locations, where each pixel can be interpreted as a bin, and its gray level is the mass of this bin in the histogram. Therefore, in the statistical perspective, comparison between two image patches is equivalent to quantify the distance between two distributions and thus, can be calculated with the $\widehat{EMD}$ metric.

To demonstrate the effectiveness of the $\widehat{EMD}$ for patch selection, we present in Fig. 4.1 a simple example of synthetic image patches, where (a) is a reference patch, (b) and (c) are two query patches. The intensity of yellow and green pixels of these patches are 200 and 0, respectively. It is easily to compute the Euclidean distances between these patches as $\ell_2(a,b) = \|a-b\|_2 = 282.8$ and $\ell_2(a,c) = \|a-c\|_2 = 282.8$. In addition, with the EMD, we have $\widehat{EMD}(a,b) = 200$ and $\widehat{EMD}(a,c) = 600$. Therefore, using the Euclidean assessment, we can not distinguish the two image patches (b) and (c). On the other hand, by adopting the $\widehat{EMD}$, we can determine that the patch in (b) is closer to the reference patch (a) than the other in (c). In addition, it can be observed that the $\widehat{EMD}$ correctly describes the perceptual similarity of patches in Fig 4.1(a) and Fig 4.1(b), while the Euclidean distance falsely state that patches (b) and (c) are similar.



(a) **(b)** **(c)**

Figure 4.1: *Comparison of the FastEMD and the Euclidean distance for patch similarity. The Euclidean distance (ℓ_2) does not match perceptual similarity. The intensity of yellow and green pixels are 200 and 0, respectively. The Euclidean distance $\ell_2(a,b) = \ell_2(a,c) = 282.8$, $\widehat{EMD}(a,b) = 200$, $\widehat{EMD}(a,c) = 600$.*

For further illustration, we report in Fig 4.2 another example with natural image

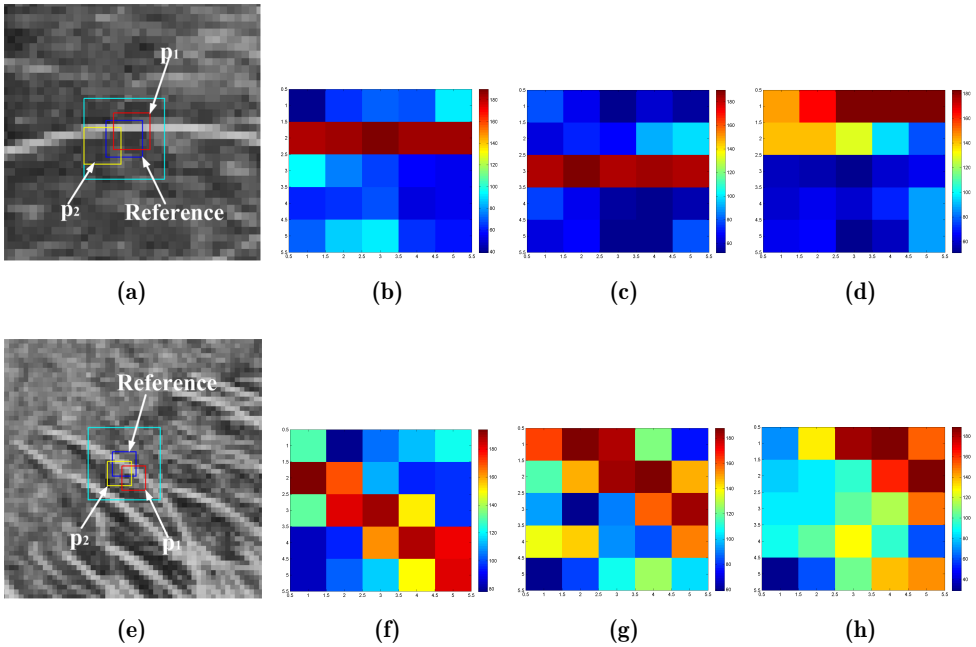


Figure 4.2: *Comparison of the FastEMD and the Euclidean distance (ℓ_2) for patch similarity. (a) and (e) The zoom-in selected regions with one reference patch and two candidate patches. (b) Reference image patch (blue square in (a)). (c) First candidate patch p_1 (red square in (a)) with $\ell_2 = 361.1$, $\widehat{EMD}$ = 812. (d) Second candidate p_2 (yellow square in (a)) with $\ell_2 = 296.8$, $\widehat{EMD}$ = 1027. (f) Reference image patch in (e). (g) First candidate patch in (e) with $\ell_2 = 326.8$, $\widehat{EMD}$ = 1282. (h) Second candidate in (e) with $\ell_2 = 326.6$, $\widehat{EMD}$ = 2207.*

patches. In each row, the left image shows a crop of a region in image of Baboon, with the reference patch is marked by red contour and the two query patches are drawn with blue contours. On the right, we show the zooming of the reference patch and the two query, respectively. On the first row of Fig 4.2, the Euclidean distance selects the patch (d) as a more similar candidate to represent the reference patch (a), while the $\widehat{EMD}$ indicates that the query patch in (b) is more suitable to approximate the reference in (a). The same conclusion can be drawn in the second row. In summary, the $\widehat{EMD}$ metric may helps us to collect a set of patches with higher perceptual similarity for a reference patch than the conventional Euclidean distance.

The rest of this paper is organized as follows. Section 2 describes details of the proposed super-resolution method, referred to as SREMD, which makes use of the $\widehat{EMD}$ for patch similarity measurement. The experimental results and comparison with some existing methods are reported in section 3. The conclusion and discussion are presented in section 4.

2 Super-Resolution using Earth Movers Distance (SREMD)

The proposed super-resolution method (SREMD) consists of two main stages, including the construction of dictionary and the super-resolution.

2.1 Dictionary construction

In the first stage, we randomly extract a set of P high-resolution patches $\{\mathbf{x}_k^s | \mathbf{x}_k^s \in R^n, k = 1, \ldots, P\}$ from a list of available standard images $\{\mathbf{x}^s\}$. The high-resolution dictionary is determined to be identical with the database of high-resolution patches as $\mathbf{D}^h = \{\mathbf{d}_k^h | \mathbf{d}_k^h \equiv \mathbf{x}_k^s, k = 1, \ldots, P\} \in R^{n \times P}$. After that, we degrade each atom in the high-resolution dictionary $\mathbf{D}^h$ to obtain the corresponding low-resolution dictionary $\mathbf{D}^l = \mathbf{S}\mathbf{B}\mathbf{D}^h = \{\mathbf{d}_k^l | \mathbf{d}_k^l \in R^m, k = 1, \ldots, P\} \in R^{m \times P}$.

2.2 Super-resolution

In the super-resolution stage, the degraded low-resolution image $\mathbf{y} \in R^M$ is partitioned into a set of overlapping patches $\{\mathbf{y}_i | \mathbf{y}_i \in R^m, i = 1, \ldots, M\}$ and the super-resolution is performed on each patch $\mathbf{y}_i$ using (4.3). However, solving the sparse representation of the whole dictionary of large atoms is challenged. Therefore, we collect a couple of local dictionaries $\{\mathbf{D}_i^h \in R^{n \times K}, \mathbf{D}_i^l \in R^{m \times K}\}$ of K most similar atoms ($K \ll P$) retrieved form the large dictionaries for recovering each degraded patch $\mathbf{y}_i$. Hence, the problem (4.3) is rewritten as:

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\{ \frac{1}{2} ||\mathbf{y}_i - \mathbf{D}_i^l \boldsymbol{\alpha}_i||_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1 \right\} \tag{4.4}$$

With the solution of $\hat{\boldsymbol{\alpha}}_i$ obtained from (4.4), we can estimate the latent high-resolution image patches as:

$$\hat{\mathbf{x}}_i = \mathbf{D}_i^h \hat{\boldsymbol{\alpha}}_i \tag{4.5}$$

A common approach for patch-filtering is to make use of the conventional bin-to-bin distances such as the Euclidean metric (ℓ_2) as proposed in many methods [1, 74] and in chapter 3 of this thesis. As indicated in previous section, the $\widehat{EMD}$ suggests a remarkable solution in dealing with image patch selection, which can return query patches with higher perceptual similarity to the referenced patch than using the Euclidean measurement. Moreover, because the $\widehat{EMD}$ is a cross-bin metric, it can address to the alignment and translation between image patches better than the ℓ_2 distance by providing the candidates with less translation to the referenced patch, and thus leads to improve the reconstruction of degraded image patch.

In this work, instead of fixing the number of most similar patches K in the local dictionaries $\{\mathbf{D}_i^h, \mathbf{D}_i^l\}$ as in chapter 3, we want to construct a couple of local dictionaries with dynamic number of atoms, such that the value of K depends on the local information of each patch $\mathbf{y}_i$, as well as the redundancy of $\mathbf{y}_i$ in the large dictionary $\mathbf{D}^l$. Therefore, in the rest of this section, we introduce a threshold of the $\widehat{EMD}$ between the underlying image patch $\mathbf{y}_i$ and the atoms $\mathbf{d}_k^l$ in the dictionary $\mathbf{D}^l$, for automatically selecting the closest patches.

For each given low-resolution patch $\mathbf{y}_i$, we may not find atoms in the dictionary $\mathbf{D}^l$ that exactly match to it, but we may retrieve its translated or rotated version. In the scope of this work, we just consider the case of subpixel shift. That means for each given patch $\mathbf{y}_i$ in the observed image $\mathbf{y}$, we expect to select the local dictionary $\mathbf{D}_i^l$ that contains the subpixel shifted versions of $\mathbf{y}_i$ from the dictionary $\mathbf{D}^l$ with the help of the $\widehat{EMD}$ metric.

Let $\mathbf{y}_i = [\mathbf{y}_i(1), \ldots, \mathbf{y}_i(j), \ldots, \mathbf{y}_i(m)]^T$ be the low-resolution patch of m pixels, and $\mathbf{y}_i^{sh} = \left[\mathbf{y}_i^{sh}(1), \ldots, \mathbf{y}_i^{sh}(l), \ldots, \mathbf{y}_i^{sh}(m)\right]^T$ be the subpixel shifted version of $\mathbf{y}_i$. Inspired from the definition of $\widehat{EMD}$ [133] presented in (2.43), (2.44) in chapter 2, the $\widehat{EMD}$ between two patches $\mathbf{y}_i$ and $\mathbf{y}_i^{sh}$ is formulated as:

$$\widehat{EMD}\left(\mathbf{y}_i, \mathbf{y}_i^{sh}\right) = \left(\min_{\{f_{jl}\}} \sum_{j=1}^{m}\sum_{l=1}^{m} f_{jl} d_{jl}\right) + \left|\sum_{j=1}^{m} \mathbf{y}_i(j) - \sum_{l=1}^{m} \mathbf{y}_i^{sh}(l)\right| \times \max_{j,l} d_{jl} \quad \text{s. t.} \quad (4.7) \tag{4.6}$$

$$\begin{cases} \sum_{l=1}^{m} f_{jl} \leq \mathbf{y}_i(j), \sum_{j=1}^{m} f_{jl} \leq \mathbf{y}_i^{sh}(l), \\ \sum_{j=1}^{m}\sum_{l=1}^{m} f_{jl} = \min\left(\sum_{j=1}^{m} \mathbf{y}_i(j), \sum_{l=1}^{m} \mathbf{y}_i^{sh}(l)\right), f_{jl} \geq 0 \end{cases} \tag{4.7}$$

where $d_{jl} = \min\left(||j-l||_2, d_{max}\right)$ is the thresholded ground distance in ℓ_2 norm between the j-th pixel of $\mathbf{y}_i$ and the l-th pixel of $\mathbf{y}_i^{sh}$, and d_{max} is a threshold of ground distance.

In order to dynamically select a couple of local dictionaries $\{\mathbf{D}_i^h, \mathbf{D}_i^l\}$ for the degraded patch $\mathbf{y}_i$, we compute an upper bound (denoted as δ_{sup}^i) of the $\widehat{EMD}$ between $\mathbf{y}_i$ and $\mathbf{y}_i^{sh}$ in (4.6). Thereby, any low-resolution atoms $\mathbf{d}_k^l \in R^m$ in the low-resolution dictionary $\mathbf{D}^l$ such that $\widehat{EMD}\left(\mathbf{y}_i, \mathbf{d}_k^l\right) \leq \delta_{sup}^i$ is acquired.

An upper bound of the first term in (4.6) can be inferred in the perspective of transportation problem and is demonstrated in Fig. 4.3. For simplicity, we assume that the intensity $\mathbf{y}_i(j)$ of the j-th pixel of $\mathbf{y}_i$ corresponds to $\mathbf{y}_i(j)$ units of masses. For a pixel j in the patch $\mathbf{y}_i$, suppose that $\mathbf{y}_i(j) \geq \mathbf{y}_i^{sh}(j)$, we will move $\mathbf{y}_i^{sh}(j)$ units from

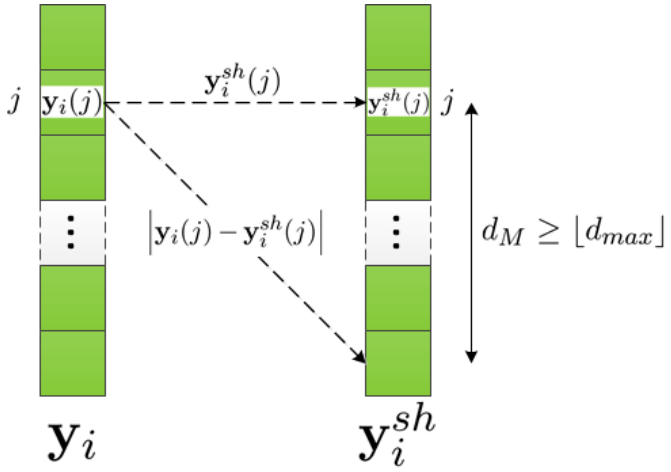


Figure 4.3: *Illustration of the transportation of units of masses between two corresponding pixels in image patches.*

pixel j of $\mathbf{y}_i$ to the same location in $\mathbf{y}_i^{sh}$, and the remaining $(\mathbf{y}_i(j) - \mathbf{y}_i^{sh}(j))$ units will be moved by d_M (with $d_M \geq \lfloor d_{max} \rfloor$) pixels away from j. The cost of this work is $\left|\mathbf{y}_i(j) - \mathbf{y}_i^{sh}(j)\right| \times d_{max}$. As a result, the first term in (4.6) is bounded by:

$$\min_{\{f_{jl}\}} \sum_{j=1}^{m} \sum_{l=1}^{m} f_{jl} d_{jl} \leq \sum_{j=1}^{m} \left|\mathbf{y}_i(j) - \mathbf{y}_i^{sh}(j)\right| \times d_{max} \tag{4.8}$$

Now it is easy to get a margin for value of the $\widehat{EMD}$ in (4.6):

$$\widehat{EMD}\left(\mathbf{y}_i, \mathbf{y}_i^{sh}\right) \leq 2 \times \sum_{j=1}^{m} \left|\mathbf{y}_i(j) - \mathbf{y}_i^{sh}(j)\right| \times d_{max} \tag{4.9}$$

The upper bound δ_{sup}^i of the $\widehat{EMD}$ is determined in the right side of (4.9) as:

$$\delta_{sup}^i = 2 \times \sum_{j=1}^{m} \left|\mathbf{y}_i(j) - \mathbf{y}_i^{sh}(j)\right| \times d_{max} \tag{4.10}$$

Consequently, we apply δ_{sup}^i as a threshold of the $\widehat{EMD}$ to retrieve the local dictionaries $\{\mathbf{D}_i^h, \mathbf{D}_i^l\}$ for each patch $\mathbf{y}_i$ from the large couple of dictionaries $\{\mathbf{D}^h, \mathbf{D}^l\}$ as

$$\begin{aligned} \mathbf{D}_i^l &= \left\{\mathbf{d}_k^l | \mathbf{d}_k^l \in \mathbf{D}^l; k = 1, \ldots, P; \widehat{EMD}(\mathbf{y}_i, \mathbf{d}_k^l) \leq \delta_{sup}^i\right\} \\ \mathbf{D}_i^h &= \left\{\mathbf{d}_k^h | \mathbf{d}_k^h \in \mathbf{D}^h; k = 1, \ldots, P; \widehat{EMD}(\mathbf{y}_i, \mathbf{d}_k^l) \leq \delta_{sup}^i\right\} \end{aligned} \tag{4.11}$$

The couple of local dictionaries is now exploited to recover the degraded image patch $\mathbf{y}_i$ by first solving the sparse representation problem in (4.4). In the scope of this work, we adopt the multiplicative updates algorithm proposed by Sha *et al.* [147] to optimize (4.4). With the optimal value of the representation coefficient vector $\hat{\boldsymbol{\alpha}}_i$, we

can estimate the high-resolution latent patch of $\mathbf{y}_i$, as in (4.5), $\hat{\mathbf{x}}_i = \mathbf{D}_i^h \hat{\boldsymbol{\alpha}}_i$. Moreover, the denoised version of $\mathbf{y}_i$ can be obtained by $\hat{\mathbf{y}}_i = \mathbf{D}_i^l \hat{\boldsymbol{\alpha}}_i$.

Algorithm 4.1: Image Super-Resolution with Earth Mover's Distance (SREMD)

Input : Couple of high- and low-resolution dictionaries $\{\mathbf{D}^h \in R^{n\times P}, \mathbf{D}^l \in R^{m\times P}\}$, low-resolution observation $\mathbf{y}$, size of low-resolution patch $\sqrt{m}\times\sqrt{m}$, regularization λ in (4.4), number of shifted pixel p_s, thresholded ground distance d_{max}, number of iteration in back-projection step T

1 Partition the low-resolution image into a set of low-resolution patches $\{\mathbf{y}_i | \mathbf{y}_i \in R^m, i = 1,\ldots,M\}$.
2 **foreach** $\mathbf{y}_i$ *in image* $\mathbf{y}$ **do**
3 Determine its shifted versions $\{\mathbf{y}_i^{sh}\}$ of p_s pixel in 8 main directions.
4 Calculate the threshold δ_{sup}^i using (4.10) for each of 8 shifted patches $\mathbf{y}_i^{sh}$ and choose the maximum value of δ_{sup}^i.
5 Calculate the distance $\widehat{EMD}(\mathbf{y}_i, \mathbf{d}_k^l)$ $(k = 1,\ldots,P)$ to all P atoms of $\mathbf{D}^l$ and select the couple of local dictionaries $\{\mathbf{D}_i^h, \mathbf{D}_i^l\}$ according to (4.11).
6 Optimize (4.4) to get the representation coefficients vector $\hat{\boldsymbol{\alpha}}_i$.
7 Estimate the high-resolution patch $\hat{\mathbf{x}}_i = \mathbf{D}_i^h \hat{\boldsymbol{\alpha}}_i$ and the denoised low-resolution patch $\hat{\mathbf{y}}_i = \mathbf{D}_i^l \hat{\boldsymbol{\alpha}}_i$
8 **end**
9 Aggregate the patches to obtain the initial high-resolution image $\mathbf{X}^0$ and denoised low-resolution image $\hat{\mathbf{Y}}$.
10 Apply iterative back-projection as (4.13) to enhance the reconstruction

Output : The estimate of the latent high-resolution image $\hat{\mathbf{X}}$

Enforcing the Reconstruction of entire image

For achieving the whole high-resolution image, the super-resolved patch $\hat{\mathbf{x}}_i$ and the denoised patch $\hat{\mathbf{y}}_i$ are put on their proper locations in the corresponding grids and averaged in overlapping regions to produce an initial estimation of the high-resolution image $\mathbf{X}^0$, as well as a denoised low-resolution image $\hat{\mathbf{Y}}$. The final high-resolution image $\hat{\mathbf{X}}$ is determine by project $\mathbf{X}^0$ onto the solution space of $\mathbf{SBX} = \hat{\mathbf{Y}}$ to enforce the constraint of the degradation model. Similar to [1], we have:

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \left\| \mathbf{X} - \mathbf{X}^0 \right\|_2^2 \quad \text{subject to} \quad \mathbf{SBX} = \hat{\mathbf{Y}} \tag{4.12}$$

We use the iterative back-projection method [148] to optimize (4.12) as

$$\mathbf{X}^{t+1} = \mathbf{X}^t + \left(\left(\hat{\mathbf{Y}} - \mathbf{SBX}^t \right) \uparrow_s \right) * \mathbf{b} \tag{4.13}$$

where $\mathbf{X}^t$ is the estimate of the latent high-resolution image at t-iteration ($t \geq 0$), $\uparrow_s$ is the bicubic upscaling operation with magnification factor s, $\mathbf{b}$ is a Gaussian blur kernel of size 5×5 and standard deviation 1. We note that in case of noise-free low-resolution image $\mathbf{y}$ (degradation model in (4.1) without the presence of noise $\boldsymbol{\eta}$), we use the original observation image $\mathbf{Y}$ to replace the denoised image $\hat{\mathbf{Y}}$ in the back-projection in (4.12). The summary of our proposed super-resolution method is described in algorithm 4.1

3 Performance evaluation

In this section, the proposed method (SREMD) is compared with some super-resolution algorithms in the literature, including bicubic interpolation, Neighbor Embedding-based method (NE[2]) of Chang *et al.* [74], Sparse coding-based method (ScSR[3]) of Yang *et al.* [3] and Super-Resolution by Sparse Weight (SRSW[4]) of Trinh *et al.* [1]. The summaries of these methods can be found in section 2 of chapter 2 of this thesis. To evaluate the objective performance of the super-resolved images, we adopt two image quality assessment metrics called PSNR and SSIM. Please consult the section 3 presented in chapter 1 for further details.

We first carried out the experiments on five images of organs as shown in Fig. 4.4(f)-(j), which are considered as the original high-resolution images $\mathbf{x}$. From these high-resolution images, the corresponding low-resolution test images are generated by blurring and downsampling with factor $s = 2$, and then added the zero-mean Gaussian noise at different values of standard deviation ($\sigma = 0, 10, 20$). For each test image, we use the corresponding standard image as shown in Fig. 4.4(a)-(e), to construct the couple of high- and low-resolution dictionaries $\{\mathbf{D}^h, \mathbf{D}^l\}$ of $P = 50000$ patches according to section 2.1. The default sizes of low-resolution and high-resolution patches are 5×5 and 9×9 pixels, respectively. The parameter λ in (4.4) is set to 1000. The subpixel shift of each image patch is chosen to 0.5 pixel. The threshold of ground distance in (4.9) is set to $d_{max} = 3$.

For the ScSR method, the value of regularization parameter λ is set to 0.8, and the size of dictionary is 1024. The parameters of the SRSW method are set to $\lambda = 0.0001$ and $\gamma = 64$.

Table 4.1 provides the results of SR methods, where the best values of PSNR and SSIM are in bold red numbers. We can observe that in most of cases the SREMD yields

[2] http://www.jdl.ac.cn/user/hchang/publication.htm
[3] http://www.ifp.illinois.edu/~jyang29/index.html
[4] http://www-l2ti.univ-paris13.fr/~luong/SRSW_IEEE_TIP/SRSW.html

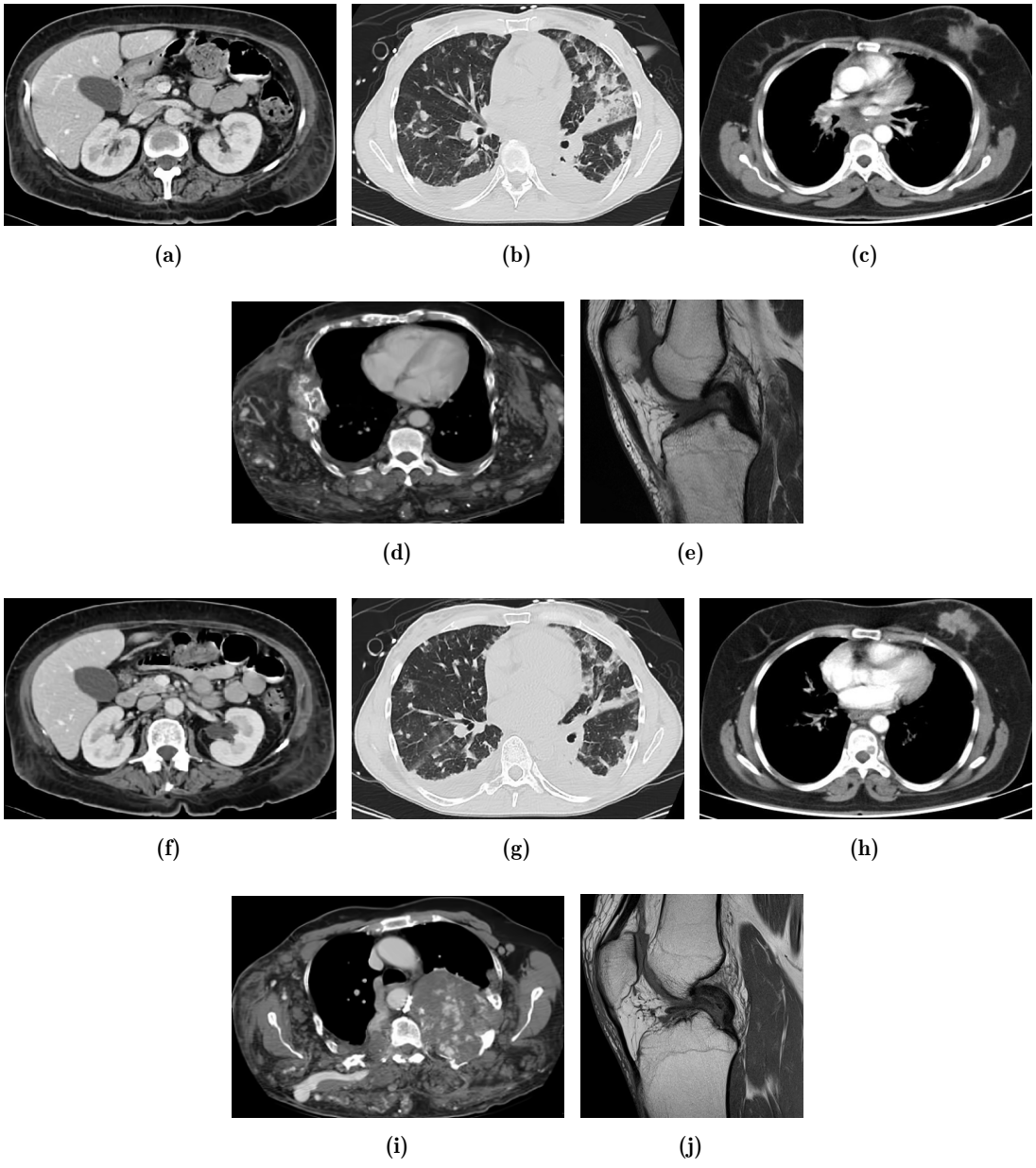


Figure 4.4: *(a)-(e): Standard images used to establish the database. (a) CT image of abdomen, (b) CT image of thorax, (c) and (d) CT images of chest, (e) MRI image of knee. (f)-(j): the corresponding test images.*

better performance than the other SR methods, especially for the SSIM index.

Image	σ	PSNR					SSIM				
		Bb	NE	ScSR	SRSW	SREMD	Bicubic	NE	ScSR	SRSW	SREMD
(f)	0	34.28	25.92	38.31	38.37	**39.85**	0.98	0.86	**0.99**	**0.99**	**0.99**
	10	28.91	25.38	30.88	30.54	**30.99**	0.72	0.74	0.82	0.81	**0.89**
	20	24.10	24.09	26.90	27.12	**27.78**	0.51	0.66	0.70	0.70	**0.76**
(g)	0	30.99	24.09	35.70	34.96	**36.58**	0.94	0.81	0.97	0.97	**0.98**
	10	27.45	23.76	**29.26**	28.67	28.79	0.72	0.76	**0.82**	0.81	**0.82**
	20	23.30	22.84	25.89	26.35	**26.62**	0.50	0.65	0.69	0.70	**0.75**
(h)	0	32.96	24.82	38.08	37.11	**40.20**	0.97	0.85	**0.98**	**0.98**	**0.98**
	10	28.66	24.29	30.96	30.38	**31.32**	0.65	0.64	0.72	0.71	**0.88**
	20	24.23	23.20	26.94	26.85	**27.70**	0.45	0.55	0.59	0.59	**0.68**
(i)	0	29.78	23.83	**33.01**	32.19	32.87	0.92	0.73	**0.96**	0.95	**0.96**
	10	27.08	23.43	**27.91**	27.19	27.61	0.68	0.58	0.70	0.69	**0.83**
	20	23.53	22.43	25.04	24.80	**26.07**	0.50	0.51	0.57	0.57	**0.78**
(j)	0	31.87	25.90	34.07	33.95	**34.17**	0.89	0.71	**0.93**	**0.93**	**0.93**
	10	27.91	25.48	**28.79**	28.47	28.53	0.67	0.63	0.70	0.70	**0.75**
	20	23.50	24.35	25.93	**26.28**	26.22	0.45	0.56	0.59	0.59	**0.64**

Table 4.1: *Comparison of the SR performance of different methods*

For subjective comparison, we present in Fig. 4.5 the super-resolution results of a region of interest (ROI) of the low-resolution image of thorax with noise level $\sigma = 20$ (created from Fig.4.4(g)). As can be seen in the zoom-in images in Fig. 4.5(b)-Fig. 4.5(d), the results of bicubic interpolation is blur and noisy, the NE method gives a very blur image, while the ScSR method brings an acceptable result with noise suppression but with many artifacts. For high noise level ($\sigma = 20$), the SRSW and the SREMD methods visually preserve the most details while achieving better noise reduction. But the background in SRSW method has a little more artifacts than the SREMD.

We also conducted experiments on the publicly available image datasets of human brain (MIDAS[5]), abdomen, chest and lung (NBIA[6]). The MIDAS contains MR images of the brain of 100 healthy subjects acquired from 20 patients over 18 years old. We randomly chosen 3 sequences of $T1$ MR images for the simulation, from which 11 images are used for the test and 5 other images are collected as standard images. For the NBIA dataset, which includes various sequences of CT images of abdomen, chest and lung, we randomly selected 2 CT images of abdomen, 8 CT images of lung and 9 CT image of chest for additional super-resolution evaluation. With each type of CT images, we also

[5] http://insight-journal.org/midas/community/view/21
[6] https://imaging.nci.nih.gov

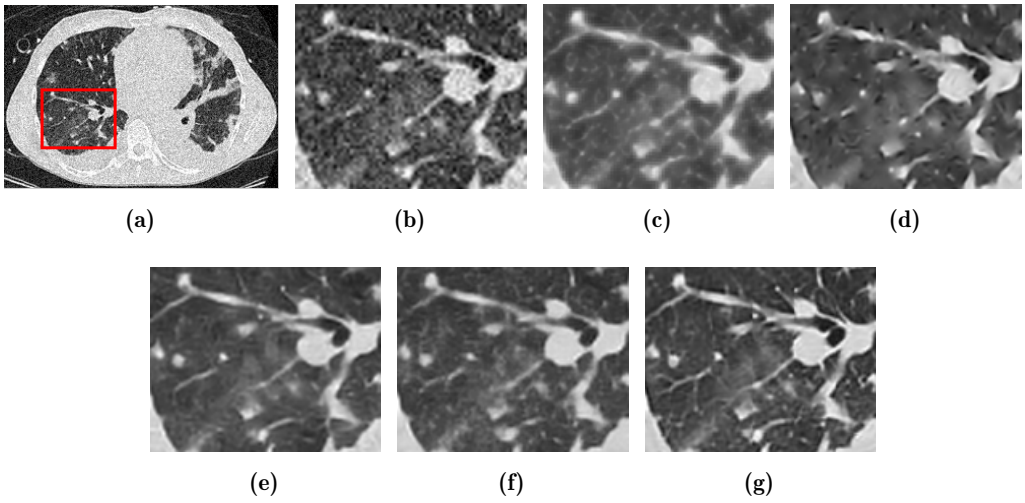


Figure 4.5: *The super-resolution results of a Region Of Interest (ROI) of the CT image of thorax with magnification $s = 2$ and noise level $\sigma = 20$. (a) LR image (size 270×180) created from Fig. 4.4(g) with a ROI (the red rectangle). (b)-(f) The ROI up-scaled by the Bicubic interpolation, the NE method ($K = 4$), the ScSR method ($\lambda = 0.8$), the SRSW method and the proposed SREMD method. (g) The ROI in the original test image (Fig. 4.4(g)).*

acquired 5 other images to generate the standard images. The SREMD is compared to above competing methods without the presence of Gaussian noise (σ is set to 0 in all test). For three methods NE, ScSR and SRSW, we use the same standard images as the proposed SREMD method to create the database of patches. Fig. 4.6 shows the evolution of PSNR with respect to the number of test images which proves the out-performance of our proposed method over the competing algorithms for the noiseless images.

3.1 Effect of the regularization parameter λ

To demonstrate the effects of the parameter λ on the super-resolution performance of the proposed method, we conducted the experiments on CT image of chest (Fig. 4.4(i)) with magnification factor $s = 2$ for different noise levels ($\sigma = 0, 10, 20$). The values of λ is set in $[0, 10, 50, 100, 250, 500, 1000, 1500, 2000, 5000, 10^4, 5 \cdot 10^4, 10^5]$. Fig. 4.7 presents the behavior of PSNR and SSIM on the variation of λ. We can observe that the range of λ that brings the best performances of the SREMD methods is around 1000, with all different noise levels. Therefore, the default value of λ is chosen as 1000 in the experiments in this paper.

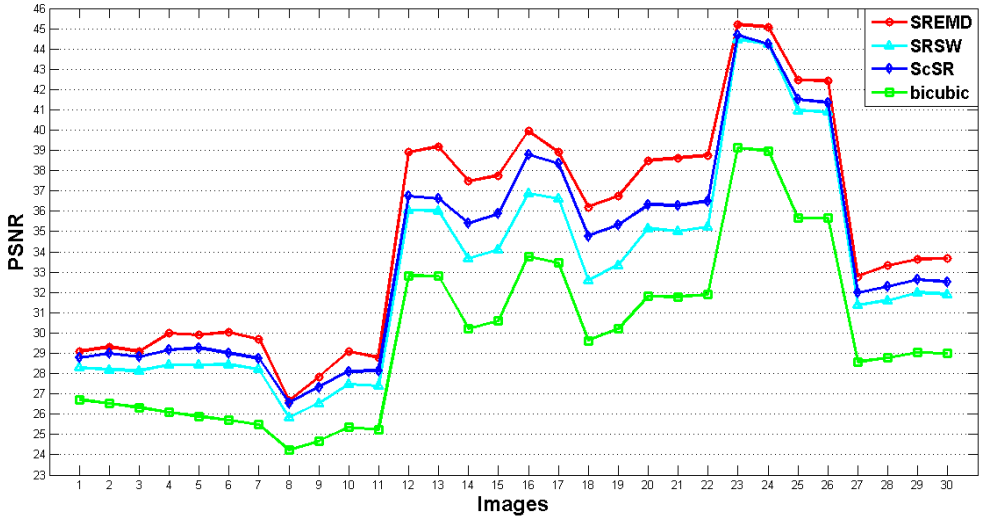


Figure 4.6: *PSNR values of different SR methods on MRI images of brain (1-11), CT images of abdomen (12-13), CT images of chest (14-22) and CT images of lung (23-30).*

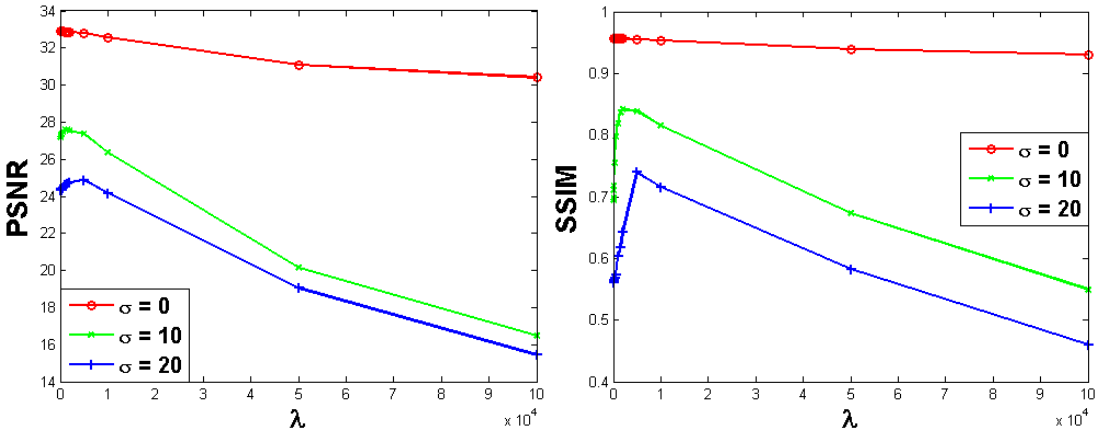


Figure 4.7: *Effect of the parameter λ on the SR performance of CT image of chest (Fig. 4.4(i)) through PSNR and SSIM values.*

4 Conclusion

In this chapter, we present an effective example-based super-resolution method which enhances the spatial resolution while robustly reducing the noise in a degraded low-resolution image. By exploiting the redundancy of local contents in multiple images, we construct a large couple of high- and low-resolution dictionaries which are identical to the database of patches collected from the standard images.

We have demonstrated that the cross-bin $\widehat{EMD}$ metric is more efficient than the conventional Euclidean distance for patch-selection and apply it to search a couple of local dictionaries for recovering each patch in the degraded image. We proposed a simple

ℓ_1-based threshold to dynamically restrict the number of atoms in local dictionaries.

The experimental results proves the effectiveness of our method over some Euclidean-based super-resolution methods in the literature either for noisy or noiseless LR images.

CHAPTER 5

Patch-based Image Denoising: Probability Distribution Estimation vs. Sparsity Prior

Abstract

Determining a good prior model that can well characterize the statistical distribution of image in the vector space generated by the atoms of a dictionary is essential in recovering a degraded patch under the Bayesian maximum a posteriori perspective. In the sparsity approach, the prior model is often assumed to obey an arbitrarily chosen distribution and sometimes may fail to describe the true distribution of image patches. In this chapter, we motivate from the redundancy of local information between multiple images to assume that, in a vector space, the latent patches of an unknown image $\mathbf{x}$ share the same distribution with the patches in a database extracted from a list of standard images. Therefore, our aim is to justify that we can estimate a probability distribution function from the empirical distribution of image patches in the database and then use it as a prior to regularize the image restoration process. We introduce a simple histogram estimation scheme to represent the distribution of image patches in a low-dimensional vector space and apply it for image denoising. We demonstrate that using the estimated probability distribution as an image prior is more efficient than the arbitrary sparsity models for noise removal. The results of our work were published at the 25-th European Signal Processing Conference in 2017[1].

[1] D. V. Tran, S. Li-Thiao-Té, M. Luong, T. Le-Tien and F. Dibos, "Patch-based image denoising: Probability distribution estimation vs. sparsity prior", 25th European Signal Processing Conference (EUSIPCO), Kos, 2017, pp. 1490-1494.

Chapter content

1 Introduction

In two previous chapters, we have exploited the external similarity and redundancy between multiple images in solving the restoration of a degraded image $\mathbf{y}$ in the patch-based framework. By building an exhaustive dictionary of image patches which is identical to the large database of patches, the reconstruction was performed on each patch $\mathbf{y}_i$ of the underlying image $\mathbf{y}$ by firstly applying a patch-filtering step on the dictionary for selecting a smaller set of similar candidates (also called a local dictionary). In the next stage, the latent image patch $\mathbf{x}_i$ (high quality version of $\mathbf{y}_i$) was recovered in the hypothesis of sparse representation in which the distribution of image patches is assumed to follow a prior mathematical model such as the Laplacian distribution.

There are some shortcomings in the aforementioned approach. Firstly, utilizing a large dictionary requires a pre-filtering step for selecting a local dictionary that can slow down the reconstruction process. Secondly, the choice of sparse model in restoring an image patch under the given local dictionary is an arbitrary decision and may not reflect the true distribution of image patches.

In this chapter, we propose to use a unique global dictionary $\mathbf{D}$ for recovering all patches $\mathbf{y}_i$ in the degraded image $\mathbf{y}$, which can be learned from the image itself or from the database of patches (collected from the standard images). Additionally, we concentrate on investigating the distribution of image patches in the vector space generated by the dictionary $\mathbf{D}$. In the next section, we will demonstrate that the sparsity models may not exactly represent the true distribution of image patches. Additionally, the main objective of this chapter is to justify that the probability distribution function learned from the database of patches can be employed as a prior to regularize the optimization process, leading to improvements in image reconstruction comparing to an arbitrary selection of sparse models. For persuasive verification, especially in the scope of this chapter, we study the application of estimation of probability distribution for solving the image denoising problem by conducting several experiments on different image modalities

The remainder of this chapter is organized as follows. Section 2 briefly describes the context of image denoising and introduces our motivation of estimation of probability distribution from the database of patches. In section 3, we expose how to employ the estimated probability function in solving the image denoising problem. After that, we present some denoising applications of the proposed method, which is referred to as ProbaEst, including the case of synthetic images, binary images, and popular natural test images. This chapter will be ended with some discussion and perspective presented

in section 5.

2 Problem Statement and Motivation

Image denoising is a fundamental task in low level vision. In this work, we tackle a simple and widely used degradation model (derived from (1.1) with $\mathbf{H}$ is the identity matrix), where an image $\mathbf{x} \in R^N$ is assumed to be corrupted by additive Gaussian noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \sigma^2)$ of mean zero and standard deviation σ as shown in (5.1)

$$\mathbf{y} = \mathbf{x} + \boldsymbol{\eta} \tag{5.1}$$

In the context of this thesis, we focus on the patch-based image reconstruction in which each image is considered as a set of overlapping patches, and the denoising will be performed on each patch. We can formulate the problem in (5.1) as:

$$\mathbf{y}_i = \mathbf{x}_i + \boldsymbol{\eta}_i \tag{5.2}$$

where $\mathbf{y}_i \in R^n$ is the i-th $(1 \leq i \leq N)$ patch in the noisy image $\mathbf{y}$, $\mathbf{x}_i \in R^n$ is the latent clean version of $\mathbf{y}_i$ and $\boldsymbol{\eta}_i$ is the residual noise in $\mathbf{y}_i$.

Among various contribution on image patch-based restoration, the dictionary learning approaches [2, 3, 43–45] have been receiving noticeable attention in the last decade. The aim of these methods is to seek a set of K basis vectors (also referred to as atoms) $\{\mathbf{d}_1, \ldots, \mathbf{d}_j, \ldots, \mathbf{d}_K | \mathbf{d}_j \in R^n, 1 \leq j \leq K\}$ to form a dictionary $\mathbf{D} \in R^{n \times K}$ such that the clean image patch $\mathbf{x}_i$ can be described by a linear combination of atoms $\{\mathbf{d}_j\}$ in the dictionary. That means, $\mathbf{x}_i = \mathbf{D}\boldsymbol{\alpha}_i$, where $\boldsymbol{\alpha}_i \in R^K$ is called the representation coefficients vector. Let $\Omega_\alpha \subset R^K$ be a vector space generated by K atoms of the dictionary $\mathbf{D}$. Then each image patch $\mathbf{x}_i$ can be represented by the vector of coefficients $\boldsymbol{\alpha}_i$ in Ω_α. Therefore, rather than modeling the distribution of image patches, we can analyze the distribution of representation coefficients in the vector space Ω_α to extract useful prior information to regularize the image denoising problem.

Under the Bayesian MAP perspective, the denoising task is equivalent to find a representation vector $\hat{\boldsymbol{\alpha}}_i$ which maximizes the posterior conditional probability:

$$\begin{aligned} \hat{\boldsymbol{\alpha}}_i &= \arg\max_{\boldsymbol{\alpha}_i} p(\boldsymbol{\alpha}_i | \mathbf{D}, \mathbf{y}_i) = \arg\max_{\boldsymbol{\alpha}_i} \{p(\mathbf{y}_i | \mathbf{D}, \boldsymbol{\alpha}_i) p(\boldsymbol{\alpha}_i)\} \\ &= \arg\min_{\boldsymbol{\alpha}_i} \left\{ ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda \log p(\boldsymbol{\alpha}_i) \right\} = \arg\min_{\boldsymbol{\alpha}_i} \left\{ ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i||_2^2 + \lambda \Phi(\boldsymbol{\alpha}_i) \right\} \end{aligned} \tag{5.3}$$

Where $p(\boldsymbol{\alpha}_i)$ is called the prior model of image patches since it specifies a priori statistical features of representation coefficients of patches in the vector space Ω_α and is

independent of observation $\mathbf{y}_i$. The two main issues in solving the problem in (5.3) are how to determine a dictionary $\mathbf{D} \in R^{n \times K}$ and what is the prior model of the distribution $p(\boldsymbol{\alpha}_i)$ in Ω_α.

As mentioned in section 2.6 in chapter 2, a data-adaptive dictionary, e.g. the K-SVD, has demonstrated its out-performances comparing to fixed analytic dictionaries in restoring a noisy image. In this chapter, we construct a dictionary $\mathbf{D}$ such that it contains similar local patterns in the images by observing the structure of images or adopting an existing dictionary training method such as the K-SVD.

With a given dictionary $\mathbf{D}$, determining a good prior model of $p(\boldsymbol{\alpha}_i)$ in the vector space Ω_α, which is expected to be identical to the real distribution of representation coefficients of patches $\{\mathbf{x}_i\}$ in the latent image $\mathbf{x}$, is the key issue in the success of an image denoising algorithm. A conventional approach in the literature is the sparsity model [2, 44, 149, 150]. The principal idea is to assume that the representation coefficients of image patches in the vector space Ω_α to be sparsely distributed and can be characterized in a smaller dimensional subspace, e.g. $p(\boldsymbol{\alpha}_i) \propto \exp(-\lambda||\boldsymbol{\alpha}_i||_0)$, with the ℓ_0 pseudo-norm counts the number of non-zero elements in $\boldsymbol{\alpha}_i$. Many authors [1, 43, 51, 55] observed that the distribution of image patches in the vector space Ω_α exhibit heavy-tailed forms and proposed some mathematical models such as Laplacian ($\Phi(\boldsymbol{\alpha}_i) \propto ||\boldsymbol{\alpha}_i||_1$), hyper-Laplacian ($\Phi(\boldsymbol{\alpha}_i) \propto ||\boldsymbol{\alpha}_i||_p$, with $0 < p < 1$) to approximately describe the distributions of patches.

There are two shortcomings of the sparsity priors. Firstly, the choice of a sparse model in a denoising algorithm is an arbitrary decision, which can be either non-convex models such as ℓ_0-norm, ℓ_p-norm ($0 < p < 1$) or a convex relaxation ℓ_1 model. Secondly, the selected sparse model may not well portray the true distribution of image patches in the vector space Ω_α generated by K atoms of the dictionary $\mathbf{D}$. It is straightforwardly recognized that a prior model which can accurately express the true distribution $p(\boldsymbol{\alpha}_i)$ of image patches may enhance the reconstruction of the degraded image. However, the true distribution $p(\boldsymbol{\alpha}_i)$ of representation coefficients of patches in the latent clean image $\mathbf{x}$ is underdetermined in practice. Therefore, introducing a prior model that can accurately approximate the true distribution of $p(\boldsymbol{\alpha}_i)$ is essential to improve the denoising performance.

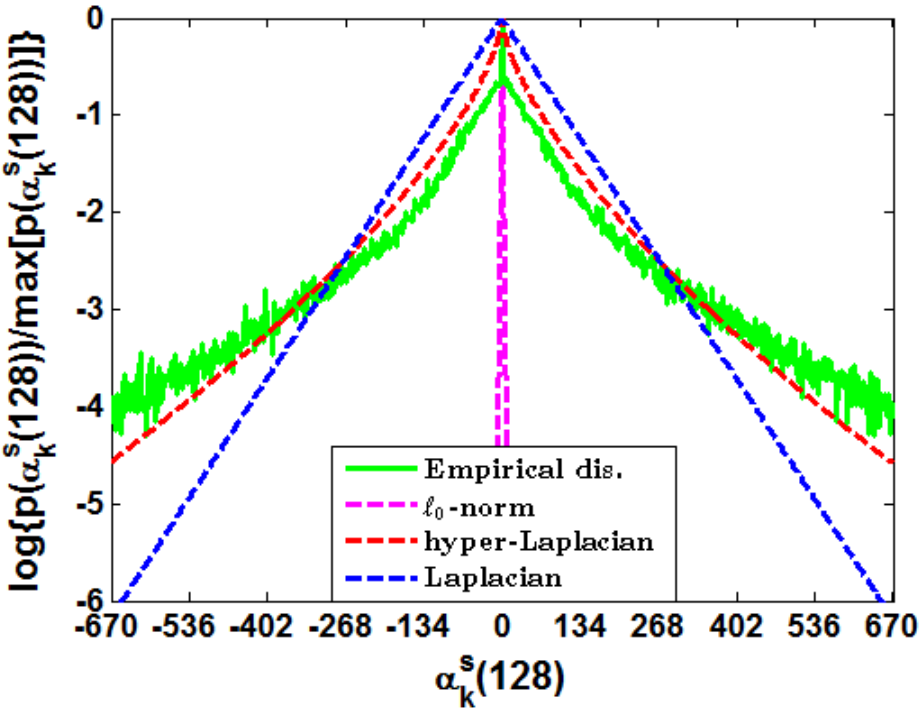
On the purpose of studying a more consistent and reliable prior model involving to the distribution of representation coefficients of image patches in the vector space Ω_α, we explored the external similarity property of images to learn the model of the probability distribution $p(\boldsymbol{\alpha}_i)$ from the standard clean images. More specifically, due to

the repetition of image contents across multiple images, as illustrated in section 2 of chapter 1, we can expect that the latent image patches $\{\mathbf{x}_i\}$ in the unknown image $\mathbf{x}$ and the patches in the database (extracted from the standard images $\{\mathbf{x}^s\}$) share the same distribution of representation coefficients. Therefore, instead of adopting an available sparse prior for representing the distribution of patches in the latent image $\mathbf{x}$, we can estimate the distribution $p(\boldsymbol{\alpha}_i)$ from the empirical distribution $p(\boldsymbol{\alpha}_k^s)$ of representation coefficients of patches in the database and then use it to reconstruct the noisy image patch $\mathbf{y}_i$ in (5.3).

Figure 5.1: *Some of standard images in Kodak PhotoCD Dataset (http://r0k.us/graphics/kodak/)*

For better comprehensive understanding the statistical property of distribution of representation coefficients of patches in the vector space Ω_α, we carry out the following experiment on image patches of the database extracted from standard images $\{\mathbf{x}^s\}$. The dictionary $\mathbf{D}$ is prepared as follows. We use the image of Boat shown in Fig. 2.1(f) as an example. First, we add the Gaussian noise of mean zero and standard deviration $\sigma = 20$ to the image. Then, we randomly extract a set of 100000 image patches of size 8×8 pixels ($n = 64$) from the noisy image and adopt the K-SVD algorithm, as described in subsection 2.5 of chapter 2, to learn an overcomplete dictionary $\mathbf{D} \in R^{64\times 256}$ of $K = 256$ atoms.

To create the list of standard images $\{\mathbf{x}^s\}$, we collected 24 high-quality noise free images from the Kodak photoCD dataset (see Fig. 5.1) which contains diversity of real-life scenes such as human subjects, building, flowers, etc. From 24 standard images of the Kodak dataset, we randomly extract a set of $P = 200000$ image patches $\{\mathbf{x}_k^s | \mathbf{x}_k^s \in R^{64}; k = 1, \ldots, 200000\}$ and calculate their corresponding representation coefficients vectors $\{\boldsymbol{\alpha}_k^s\}$ in the vector space $\Omega_\alpha \subset R^{256}$ by using the least mean square error



(a) Distribution $p(\boldsymbol{\alpha}_k^s(128))$ of representation coefficients on 128-th atom

Figure 5.2: *Empirical distribution $p(\boldsymbol{\alpha}_k^s(128))$ of representation coefficients of image patches in the database with respect to the 128-th atoms of the dictionary $\mathbf{D}$ are plotted (in log domain) in green curve. The fitting sparse models for the real distribution of patches are also sketched, including the ℓ_0-norm in magenta dash curve, Laplacian ℓ_1-norm in red dash curve and hyper-Laplacian ℓ_p-norm (with $p = 2/3$) in blue dash curve*

estimation

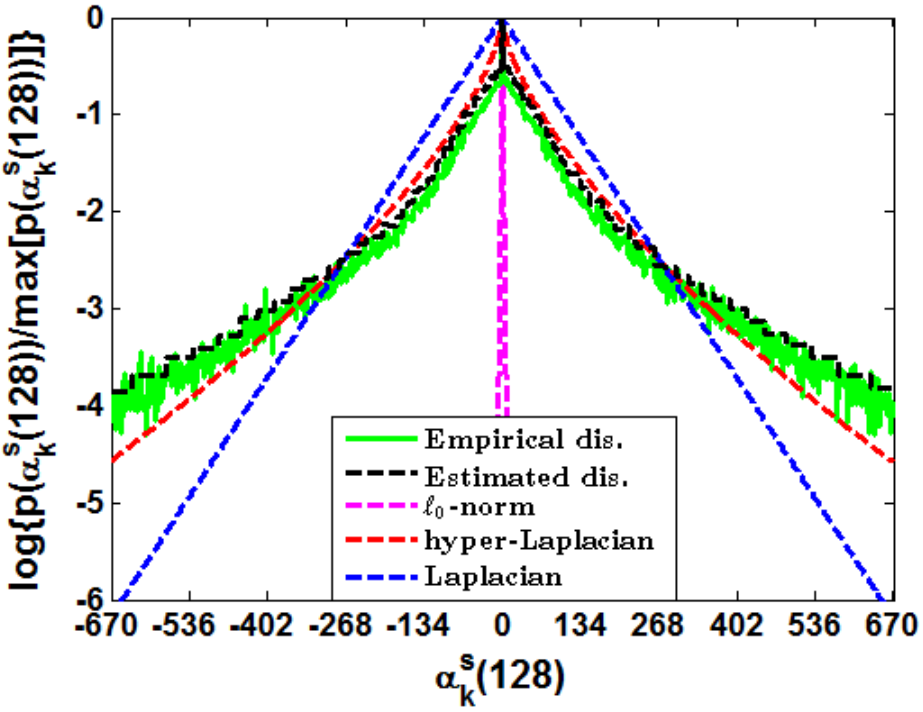
$$\boldsymbol{\alpha}_k^s = \arg\min_{\boldsymbol{\alpha}_k^s} ||\mathbf{x}_k^s - \mathbf{D}\boldsymbol{\alpha}_k^s||_2^2 = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{x}_k^s = \mathbf{P}_D\mathbf{x}_k^s \tag{5.4}$$

where $\mathbf{P}_D = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T$ is the projection matrix. Hence, with a database of patches in the image domain, we obtain a distribution $p(\boldsymbol{\alpha}_k^s)$ of representation coefficients vectors in Ω_α.

Let $\mathbf{A} = [\boldsymbol{\alpha}_1^s, \ldots, \boldsymbol{\alpha}_k^s, \ldots, \boldsymbol{\alpha}_{200000}^s] \in R^{256\times 200000}$ denote the ensemble of all representation vectors. Thus, the j-th row of $\mathbf{A}$ coincides with the representation coefficients $\{\boldsymbol{\alpha}_k^s(j)\}$ of image patches in the database on the j-th atom $\mathbf{d}_j$ of the dictionary $\mathbf{D}$. In order to facilitate the observation of statistical property of the real distribution $p(\boldsymbol{\alpha}_k^s)$ of image patches in the database, we plot (in log domain) the distribution of the 128-th row of matrix $\mathbf{A}$ in green curve in Fig. 5.2, which corresponds to the representation coefficients $\{\boldsymbol{\alpha}_k^s(128)\}$. Moreover, we also portray the fittings of true distribution (the green curves) with three sparse models, including the ℓ_0-norm ($p(\boldsymbol{\alpha}_k^s) \propto \exp(-\gamma||\boldsymbol{\alpha}_k^s||_0)$) in magenta dash line, Laplacian ($p(\boldsymbol{\alpha}_k^s) \propto \exp(-\gamma||\boldsymbol{\alpha}_k^s||_1)$) in blue dash line, and the

hyper-Laplacian ($p(\boldsymbol{\alpha}_k^s) \propto \exp(-\gamma||\boldsymbol{\alpha}_k^s||_p)$ with $p = 2/3$) in red dash curve.

An important remark can be drawn from Fig. 5.2. The sparse models such as Laplacian and hyper-Laplacian are imperfect, but satisfactory in approximately representing the distribution of patches in the vector space Ω_α. However, they fail to describe the true distributions of image patches with the fitting errors between the green curves and the simulated lines of sparse models. Therefore, in this chapter, instead of selecting a sparse hypothesis as the prior model of the distribution $p(\boldsymbol{\alpha}_i)$ of latent patches, we propose to estimate it from the real distribution $p(\boldsymbol{\alpha}_k^s)$ (e.g. shown in green curve) of representation coefficients of patches in the database. The purpose of our research is to demonstrate that an accurate estimation of the distribution $p(\boldsymbol{\alpha}_k^s)$ can be well employed as a prior information of $p(\boldsymbol{\alpha}_i)$ to regularize the image patch denoising, and leads to outperforming reconstructions than an arbitrary choice of a sparse model.



(a) Distribution $p(\boldsymbol{\alpha}_k^s(128))$ of representation coefficients on 128-th atom

Figure 5.3: *Empirical distribution $p(\boldsymbol{\alpha}_k^s(128))$ of representation coefficients of image patches in the database (green curve). The fitting sparse models, including the ℓ_0-norm (in magenta dash curve), Laplacian (in red dash curve) and hyper-Laplacian (in blue dash curve). The black dash line is the histogram estimation of the real distribution of patches.*

Histogram is a simple yet effective tool that gives a rough sense of the density of an underlying distribution of the data and can be used to estimate its probability distribution function. In our work, we propose a median-based binning approach to

discretize the distribution $p(\boldsymbol{\alpha}_k^s)$ of patches, which will be introduced in more details in the next section. For a quick demonstration, we show in Fig. 5.3 the empirical distribution $p(\boldsymbol{\alpha}_k^s(128))$ of representation coefficients of image patches in the database corresponding to the 128-th atoms in the dictionary, as well as its estimation versions in dashed black lines and the fitting simulated sparse models. We can observe that the estimated histogram can well represent the empirical distribution of representation coefficients of patches with smaller fitting errors comparing to the sparse models.

In the next section, we will present the idea of estimation of probability distribution function $p(\boldsymbol{\alpha}_i)$ of patches in the latent image $\mathbf{x}$ from the empirical distribution $p(\boldsymbol{\alpha}_k^s)$ of representation coefficients of patches in the database.

3 Denoising by probability distribution estimation

In this section, we consider the distribution $p(\boldsymbol{\alpha}_i)$ of representation coefficients of latent patches as a histogram estimated from the distribution $p(\boldsymbol{\alpha}_k^s)$ of representation coefficients of patches in the database. We will discover the concept of our work on estimation of $p(\boldsymbol{\alpha}_i)$ from the empirical distribution $p(\boldsymbol{\alpha}_k^s)$ via constructing a histogram of piecewise constant functions. After that, we propose a framework that exploits the obtained probability functions in solving the denoising optimization problem.

3.1 Estimation of probability distribution $p(\alpha_i)$ from the database

Building histograms on one- or two-dimensional data is easy and visualizing. However, for high dimensional data, we are likely to run into the curse of dimensionality, where we have an exponential number of hypercube bins and nearly all of them may probably be empty. As estimation is a difficult problem in high dimensional spaces, we consider the vector space with dimensions up to three ($\Omega_\alpha \subset R^3$). This is equivalent to choosing a dictionary with only three atoms ($K = 3$) $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3] \in R^{n\times 3}$. After that, we randomly extract a set of P image patches $\{\mathbf{x}_k^s \in R^n, k = 1, 2, \ldots, P\}$ from the standard images $\mathbf{x}^s$ to create a database of patches. Each image patch $\mathbf{x}_k^s$ is encoded in the vector space Ω_α by a corresponding representation coefficient vector $\boldsymbol{\alpha}_k^s$, determined by the projection as described in (5.4), $\boldsymbol{\alpha}_k^s = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{x}_k^s$.

In the scope of this work, rather than arbitrarily selecting an available sparse model to represent the probability distribution $p(\boldsymbol{\alpha}_i)$, we prefer to approximately calculate it from the real distribution $p(\boldsymbol{\alpha}_k^s)$ in Ω_α of representation coefficients $\boldsymbol{\alpha}_k^s$ of patches in the database. We introduced a simple yet effective method, by constructing a

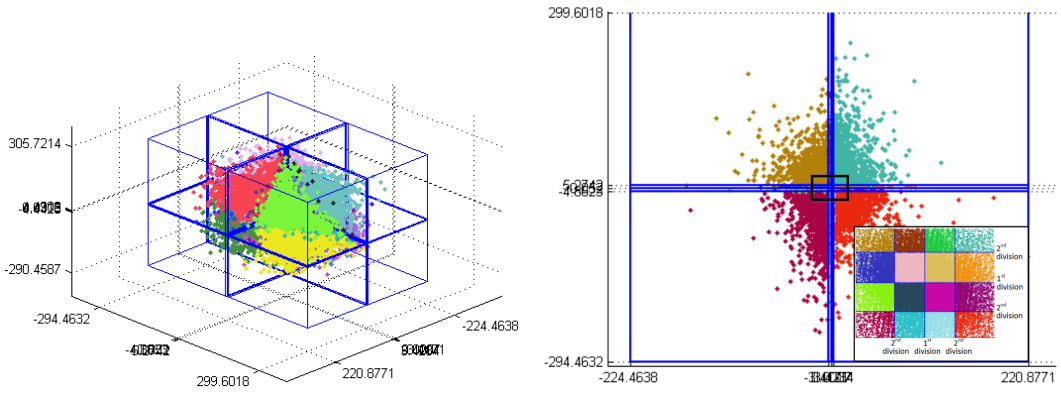
three-dimensional (3D) histogram for estimating the empirical distribution of patches. However, building a histogram for the distribution $p(\boldsymbol{\alpha}_k^s)$ in the whole vector space Ω_α is a complex problem and can be broken down into smaller and simpler steps by determining 3 one-dimensional (1D) histograms according to 3 atoms in the dictionary. Hence, the distribution of patches along the j-th coordinate, denoted as $p(\boldsymbol{\alpha}_k^s(j))$, of the vector space Ω_α is roughly approximated by a piecewise constant function of a 1D histogram. Finally, the estimated 3D histogram $p(\boldsymbol{\alpha}_i)$ of the real probability distribution $p(\boldsymbol{\alpha}_k^s)$ becomes a cube-wise constant function whose values are generated from the three 1D histograms by computing the density of patches simultaneously falling into each cube made of three 1D bins. As a results, the empirical distribution $p(\boldsymbol{\alpha}_k^s)$ of representation coefficients of patches in the database is approximated by a grid of cubes $\{\Omega_\alpha^c\}$ such that every patches $\boldsymbol{\alpha}_i$ stored in the same cube Ω_α^c share an identical probability of occurrence, as defined in (5.5).

$$p(\boldsymbol{\alpha}_i) = \begin{cases} \dfrac{1}{\text{Volume of } \Omega_\alpha^c} \times \dfrac{\text{Number of patches in } \Omega_\alpha^c}{\text{Number of patches in } \Omega_\alpha} = \dfrac{1}{V_\alpha^c} \times \dfrac{P_c}{P}, & \text{if } \boldsymbol{\alpha}_i \in \Omega_\alpha^c \\ 0, & \text{otherwise} \end{cases} \tag{5.5}$$

where P_c is the number of patches falls inside the cube Ω_α^c which contains $\boldsymbol{\alpha}_i$, V_α^c is the volume of Ω_α^c (as Ω_α is the 3D vector space), P is the total number of patches in the database.

A major question in estimating the histogram $p(\boldsymbol{\alpha}_i)$ is how to determine the size of each cube Ω_α^c such that $p(\boldsymbol{\alpha}_i)$ can accurately represent the empirical distribution $p(\boldsymbol{\alpha}_k^s)$. To this end, we consider to establish a 1D histogram for each dimension j of the space Ω_α, which coincides with the representation coefficients $\boldsymbol{\alpha}_k^s(j)$ on the j-th atom of the dictionary, by dividing the entire range of values of $\boldsymbol{\alpha}_k^s(j)$ into a series of B non-overlapping intervals (also known as bins) and counting how many values falling into each bin. Practically, the choice of bin width, which is commonly inversely proportional to the number of bin B over the range, is often a little arbitrary. In a simply conventional proposal, the range limited by the minimum and maximum values of $\boldsymbol{\alpha}_k^s(j)$ in the j-th dimension is split into B equal-width intervals. Despite the straightforwardness, a shortcoming of equal-size bin division is the inflexibility in describing the density of a distribution. Moreover, to precisely characterize a true distribution, we need to divide the range into a large number of bins B, which causes an increase in the complexity of estimation algorithm.

To cope with this issue, we proposed a median-based division strategy in which each axis of Ω_α is split in such a way that each interval contains the same number of patches. Hence, the width of each bin can adapt to the density of the values of



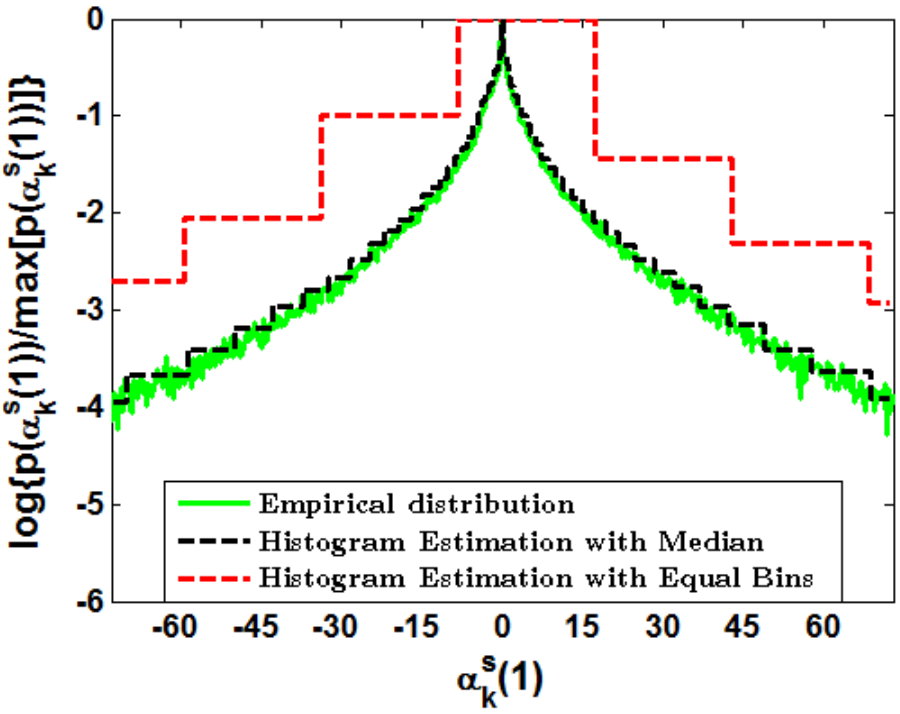
(a) (b)

Figure 5.4: *Partition of $\boldsymbol{\alpha}$-space into grid of $4 \times 4 \times 4$ rectangular cells. (a) View in 3D. (b) A x-y slice at position $z = 0$ with a zoom in of center.*

representation coefficients $\boldsymbol{\alpha}_k^s(j)$. That means, for an interval of the j-th coordinate of Ω_α with high density of patches, its length will be set to small, and so on. In the experiments, each dimension of the vector space Ω_α is separated into B intervals by $B-1$ median points using a recursive division, so that each bin contains P/B points. As a result, we obtain a grid of $(B-1)^3$ quantiles $\mathcal{G}_\alpha = \{\boldsymbol{\alpha}_l^q | l = 1, \ldots, (B-1)^3\}$ in the R^3 space Ω_α. In Fig. 5.4, we demonstrate an example of distribution of points in Ω_α which is partitioned in $4 \times 4 \times 4$ (with $B = 4$, $K = 3$) cubes.

Using the proposed approach, we can produce a good estimation of the histogram $p(\boldsymbol{\alpha}_i)$ to portray the true empirical distribution $p(\boldsymbol{\alpha}_k^s)$ of representation coefficients of patches in the database with only a small number of bins in each dimension of the vector space Ω_α. To illustration, we consider an example on the image of Boat. We generate a noisy image by adding Gaussian noise of mean zero and standard deviation of 20. After that, we randomly extract a set of 100000 patches of size 3×3 pixels to train a dictionary of $K = 3$ atoms, thanks to the K-SVD algorithm. We also collect $P = 200000$ patches in the 24 standard images of the Kodak dataset and project them into the vector space Ω_α of the dictionary. Fig. 5.5 shows (in green curve) the empirical distribution $p(\boldsymbol{\alpha}_k^s(1))$ of the representation coefficients of patches in the database corresponding to the first atom of the dictionary, as well as two estimated histograms using equal-width bin division (in red dash lines) and median-based split (in black dash lines). The number of intervals (bins) in each dimension of Ω_α is set to $B = 32$ for two methods. It can be observed that in the median-based bin division, we only need to partition the coordinate into $B = 32$ bins to get a histogram that is very approximate to the true distribution of patches. On the other hand, the histogram obtained by the conventional equal-bin division method with the same number of bin B fails to match the distribution of patches. Therefore, in

all experimental test, we adopt the median-based partition with the number of bin is set to $B = 32$ for each dimension of the vector space Ω_α.



(a) Distribution $p(\boldsymbol{\alpha}_k^s(1))$ of representation coefficients on 128-th atom

Figure 5.5: *The empirical distribution $p(\boldsymbol{\alpha}_k^s(1))$ of representation coefficients of image patches in the database corresponding to the first atom in the dictionary (green curve) and the 32-bins estimated histograms according to median-based division (black dash line) and equal-bin division (red dash curve).*

3.2 Solving the optimization problem

In this section, we will present how to recover a clean image patch $\mathbf{x}_i$ from its noisy version $\mathbf{y}_i$, given the same dictionary $\mathbf{D} \in R^{n\times 3}$ in section 3.1 and the prior knowledge of probability distribution $p(\boldsymbol{\alpha}_i)$ estimated from the database of patches.

The denoised patch can be generated as a linear combination of atoms in the dictionary, $\hat{\mathbf{x}}_i = \mathbf{D}\hat{\boldsymbol{\alpha}}_i$. Our objective is equivalent to find the value of $\hat{\boldsymbol{\alpha}}_i$ which satisfies the MAP optimization problem in (5.3):

$$\hat{\boldsymbol{\alpha}}_i = \underset{\boldsymbol{\alpha}_i \in \Omega_\alpha}{\arg\min} \left\{ ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda \log p(\boldsymbol{\alpha}_i) \right\} = \underset{\boldsymbol{\alpha}_i \in \Omega_\alpha}{\arg\min}\, J(\boldsymbol{\alpha}_i) \tag{5.6}$$

Where $J(\boldsymbol{\alpha}_i) = ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda \log p(\boldsymbol{\alpha}_i)$ is called the cost function, $\lambda > 0$ is a regularization parameter.

As indicated in section 3.1, we have divided the vector space Ω_α into disjoint cubes

$\{\Omega_\alpha^c\}$, and the value of $p(\boldsymbol{\alpha}_i)$ is constant in each cube Ω_α^c and is determined as in (5.5). Consequently, the cost function $J(\boldsymbol{\alpha}_i)$ in (5.6) becomes a convex piecewise quadratic function. A simple way to handle the problem (5.6) is the exhaustive solution that minimizes $J(\boldsymbol{\alpha}_i)$ on each 3D cube Ω_α^c and then select the best value $\hat{\boldsymbol{\alpha}}_i$ from which we achieve the smallest cost. Therefore, we can rewrite (5.6) as:

$$\hat{\boldsymbol{\alpha}}_i = \underset{\Omega_\alpha^c \subset \Omega_\alpha}{\arg\min} \left\{ \underset{\boldsymbol{\alpha}_i \in \Omega_\alpha^c}{\arg\min}\, J(\boldsymbol{\alpha}_i) \right\} = \underset{\Omega_\alpha^c \subset \Omega_\alpha}{\arg\min}\, J_c(\boldsymbol{\alpha}_i; \Omega_\alpha^c) \tag{5.7}$$

As defined in (5.5), the value of $p(\boldsymbol{\alpha}_i)$ is equal to zero when $\boldsymbol{\alpha}_i$ is outside a cube Ω_α^c, and receive a constant value for every point $\boldsymbol{\alpha}_i \in \Omega_\alpha^c$. Thus, the minimum cost function on Ω_α^c, denoted as $J_c(\boldsymbol{\alpha}_i; \Omega_\alpha^c)$, becomes:

$$J_c(\boldsymbol{\alpha}_i; \Omega_\alpha^c) = \underset{\boldsymbol{\alpha}_i \in \Omega_\alpha^c}{\arg\min}\, J(\boldsymbol{\alpha}_i) = \underset{\boldsymbol{\alpha}_i \in \Omega_\alpha^c}{\arg\min}\, ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda \log\left(\frac{P_c}{V_\alpha^c P}\right) \tag{5.8}$$

$J_c(\boldsymbol{\alpha}_i; \Omega_\alpha^c)$ turns into the minimization of a convex quadratic function $J(\boldsymbol{\alpha}_i)$ on a subspace Ω_α^c. As a result, the optimized value of $J_c(\boldsymbol{\alpha}_i; \Omega_\alpha^c)$ occurs at either a point inside or one of the eight corner of Ω_α^c. We will prove as follows.

Suppose that there exists a cube Ω_α^d such that $J_c(\boldsymbol{\alpha}_i; \Omega_\alpha^d) = \arg\min_{\boldsymbol{\alpha}_i \in \Omega_\alpha^d} ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda \log\left(\frac{N_d}{V_\alpha^d N}\right)$ gets the minimum at $\boldsymbol{\alpha}_i^* = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{y}_i$, and $\boldsymbol{\alpha}_i^* \in \Omega_\alpha^d$. Hence, for all other cubes $\Omega_\alpha^c \neq \Omega_\alpha^d$, the minimum of cost function on Ω_α^c is determined as:

$$J_c(\boldsymbol{\alpha}_i; \Omega_\alpha^c) = \underset{\boldsymbol{\alpha}_i \in \Omega_\alpha^c, \boldsymbol{\alpha}_i \notin \Omega_\alpha^d}{\arg\min} ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda \log\left(\frac{P_c}{V_\alpha^c P}\right) \tag{5.9}$$

Because the root of the first derivative $\frac{\partial J_c(\boldsymbol{\alpha}_i; \Omega_\alpha^c)}{\partial \boldsymbol{\alpha}_i}$ only occurs at $\boldsymbol{\alpha}_i^* = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{y}_i \notin \Omega_\alpha^c$, we always have:

$$\begin{cases} \dfrac{\partial J_c(\boldsymbol{\alpha}_i; \Omega_\alpha^c)}{\partial \boldsymbol{\alpha}_i} \neq \overrightarrow{0} \quad \forall \boldsymbol{\alpha}_i \in \Omega_\alpha^c \\ \dfrac{\partial^2 J_c(\boldsymbol{\alpha}_i; \Omega_\alpha^c)}{\partial \boldsymbol{\alpha}_i^2} = \mathbf{D}^T\mathbf{D} > 0 : \text{ is a positive definite matrix} \end{cases} \tag{5.10}$$

Therefore, the value of the cost function $J(\boldsymbol{\alpha}_i)$ in a cube Ω_α^c always increase as we go from one corner to another corner of Ω_α^c. Consequently, the optimized value of (5.9) is only obtained at one of the corners of Ω_α^c. As a result, the restoration of each noisy image patch $\mathbf{y}_i$ is attained via minimizing the cost function $J(\boldsymbol{\alpha}_i)$ in (5.6) under the constraint of cube-wise constant function of the probability distribution $p(\boldsymbol{\alpha}_i)$.

Moreover, we have partitioned the vector space Ω_α into B^3 rectangular bins by grid of $(B-1)^3$ quantiles $\mathcal{G}_\alpha$. Therefore, these points becomes the corners of cubes Ω_α^c, and

Algorithm 5.1: Image denoising with estimated probability

Input : Set of standard images $\{\mathbf{x}^s\}$, noisy image $\mathbf{y}$, dictionary $\mathbf{D}$, noise level σ, regularized parameter λ, patch size $\sqrt{n} \times \sqrt{n}$, overlap between two patches, number of bin B

1 Randomly extract P patches $\{\mathbf{x}_k^s, k = 1, \ldots, N\}$ from standard images $\{\mathbf{x}^s\}$, then subtract the mean value from each patch $\mathbf{x}_k^s = \mathbf{x}_k^s - \text{mean}(\mathbf{x}_k^s)$.

2 Project P patches into vector space Ω_α generated by K atoms of the dictionary $\mathbf{D}$ using (5.4) to obtain the distribution of representation coefficients $p(\boldsymbol{\alpha}_k^s)$.

3 Estimate the histogram $p(\boldsymbol{\alpha}_i)$ from the distribution $p(\boldsymbol{\alpha}_k^s)$ using (5.5)

3.1 Determine $(B-1)^K$ quantile points $\mathcal{G}_\alpha = \{\boldsymbol{\alpha}_l^q | l = 1, \ldots, (B-1)^K\}$, based on recursive median splitting, which divide the entire range of values in each dimension of Ω_α into B bins.

3.2 Determine B^K cubes $\{\Omega_\alpha^c\}$ from $(B-1)^K$ quantiles $\mathcal{G}_\alpha$ and compute the density of patches falling in each cube Ω_α^c.

4 Partition the noisy image $\mathbf{y}$ into overlapping patches $\mathbf{y}_i$.

5 **for** *each image patch* $\mathbf{y}_i \in \mathbf{y}$ **do**

6 Subtract its mean value: $\mathbf{y}_i = \mathbf{y}_i - \mu_y$.

7 Calculate $\boldsymbol{\alpha}_i^* = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{y}_i$.

8 Define a set of anchor points: $\mathcal{L}_\alpha = \{\boldsymbol{\alpha}_i^*, \mathcal{G}_\alpha\} = \{\boldsymbol{\alpha}_l^a | l = 1, \ldots, (B-1)^K + 1\}$.

9 Solve problem (5.11) by calculating values of the cost function $J(\boldsymbol{\alpha}_i)$ in (5.6) at all anchor points $\boldsymbol{\alpha}_l^a \in \mathcal{L}_\alpha$ and choosing the optimized value $\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_l^a \in \mathcal{L}_\alpha} J(\boldsymbol{\alpha}_l^a)$.

10 Estimate the denoised image patch: $\hat{\mathbf{x}}_i = \mathbf{D}\hat{\boldsymbol{\alpha}}_i + \mu_y$

11 **end**

12 Average the overlapped regions of denoised patches $\hat{\mathbf{x}}_i$ to obtain the entire denoised image $\hat{\mathbf{x}}$.

Output : The denoised image $\hat{\mathbf{x}}$.

the optimal value of $J(\boldsymbol{\alpha}_i)$ in (5.6) will take place either at $\boldsymbol{\alpha}_i^* = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{y}_i$ or at one point in $\mathcal{G}_\alpha$. Let $\mathcal{L}_\alpha = \{\boldsymbol{\alpha}_i^*, \mathcal{G}_\alpha\} = \{\boldsymbol{\alpha}_l^a | l = 1, \ldots, (B-1)^K + 1\}$ denote the set of anchor points. For a noisy image patch $\mathbf{y}_i$, to minimize (5.6), we can easily compute the value of cost function $J(\boldsymbol{\alpha}_i)$ at anchor points and choose the optimized value $\hat{\boldsymbol{\alpha}}_i$ among $\mathcal{L}_\alpha$ such that $J(\hat{\boldsymbol{\alpha}}_i)$ is smallest.

$$\hat{\boldsymbol{\alpha}}_i = \underset{\boldsymbol{\alpha}_l^a \in \mathcal{L}_\alpha}{\arg\min} \left\{ ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_l^a||_2^2 - \lambda \log p(\boldsymbol{\alpha}_l^a) \right\} \tag{5.11}$$

The value of $p(\boldsymbol{\alpha}_l^a)$ in (5.11) and the production $\mathbf{D}\boldsymbol{\alpha}_l^a$ at each anchor point $\boldsymbol{\alpha}_l^a \in \mathcal{L}_\alpha$ can be computed offline to accelerate the speed of the optimization of (5.11). When we obtain the solution of optimal representation coefficients $\hat{\boldsymbol{\alpha}}_i$, the clean version of the noisy image patch $\mathbf{y}_i$ can be estimated by the weighted combination of atoms in the dictionary:

$$\hat{\mathbf{x}}_i = \mathbf{D}\hat{\boldsymbol{\alpha}}_i \tag{5.12}$$

The denoising process is repeatedly applied to all patches in the noisy image $\mathbf{y}_i$ to get the estimations of the latent clean patches. After that, we aggregate the overlapping regions between adjacent patches to achieve the final denoised image $\hat{\mathbf{x}}_i$. The summary of the image denoising framework, including the estimation of probability distribution function procedure, is presented in algorithm 5.1.

3.3 Complexity analysis

In the stage of estimation of probability distribution function (step 1 to 3 in the algorithm 5.1), we firstly extract P patches $\{\mathbf{x}_k^s\}$ of size $\sqrt{n} \times \sqrt{n}$ from the standard images and project them into the vector space $\Omega_\alpha \subset R^K$ generated from the dictionary $\mathbf{D} \in R^{n \times K}$. The complexity of this process is $O(PK^3 + PK^2 n)$. After that, the distribution of P representation coefficients vectors $\{\boldsymbol{\alpha}_k^s\}$ is estimated via constructing a histogram in R^K space. Each dimension of Ω_α space is split into B bins using $\log(B)$ times of recursive median finding algorithm. The complexity of searching $(B-1)^K$ median values of P points in K-dimension is $O(P^K(\log(B))^K)$. The calculation of cube-wise histogram in (5.5) takes $O(B^K)$ operations. Thus the overall complexity of the probability estimation is $O(P^K(\log(B))^K + B^K + PK^3 + PK^2 n)$.

In the denoising stage (steps 5 to 11 in the algorithm 5.1), the most expensive computational complexity is to solve the optimization problem in (5.11), which takes $O(N(Kn+n^2)B^K)$ operations, with N is the number of noisy patches $\mathbf{y}_i$ in the degraded image $\mathbf{y}$.

As we can see, the computational complexity of the denoising algorithm is highly depend on the number of atoms K of the dictionary $\mathbf{D}$ and the number of bins B in each coordinate of the vector space Ω_α. With a fixed value of B, we have an algorithm with exponential complexity in term of K. If K is set to a large value (high dimensional vector space Ω_α), the complexity will dramatically increase and become a big challenge in implementation of the algorithm. Because of the curse of high dimensionality and expensive computational cost, we choose a small dictionary with $K = 3$ atoms to demonstrate the idea of using estimated histogram of the distribution of external patches in the denoising of an image patch.

Additionally, with a given dictionary (K is fixed), the complexity is described in polynomial expression of the number of bins B in each coordinate of vector space Ω_α. It can be observed that if we increase the number of bins by 2 times, the denoising stage (especially in step 9) needs 2^K times longer to handle the reconstruction of the same image. As indicated in section 3.1, we only need to divide each dimension of Ω_α into $B = 32$ bins using the median-based splitting approach to get a good estimation of the distribution of external patches. Therefore, in our experiments, the value of B is set to $B = 32$ to ensure the balance between time complexity and the restoration performance.

4 Denoising performance and evaluation

In this section, our aim is to demonstrate that for a given dictionary $\mathbf{D}$, using the estimated probability distribution from the patches in the database as an image prior is more efficient than the sparsity models for noise removal in the Bayesian MAP framework (5.3). Comparison with other denoising methods such as nonlocal self-similarity [65, 99], deep-learning [114] is out of the scope of our work on this chapter.

Given a dictionary, our method differs from the existing sparsity models in the way we determine the probability distribution function $p(\boldsymbol{\alpha}_i)$. While the sparsity methods assume that the probability of representation coefficients $p(\boldsymbol{\alpha}_i)$ in Ω_α obeys a specific model such as the Laplacian $p(\boldsymbol{\alpha}_i \propto \exp(-\lambda||\boldsymbol{\alpha}_i||_1))$, we propose to estimate $p(\boldsymbol{\alpha}_i)$ via constructing a histogram of the empirical distribution of $\boldsymbol{\alpha}_k^s$ of external patches in Ω_α.

In the remaining of this section, we make comparisons on the denoising performance of our proposed method with two famous sparse models of ℓ_0-norm (5.13) and ℓ_1-norm (5.14).

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\{ \| \mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i \|_2^2 + \lambda ||\boldsymbol{\alpha}_i||_0 \right\} \tag{5.13}$$

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\{ \| \mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i \|_2^2 + \lambda ||\boldsymbol{\alpha}_i||_1 \right\} \tag{5.14}$$

The problem in (5.13) can be efficiently solved by the orthogonal matching pursuit (OMP) algorithm [120]. In our implementation, we use the LARS algorithm developed by Zou *et al.* [63] to find the solution of (5.14). For more convenient, we refer to our proposed method as ProbaEst, and the others in (5.13) and (5.14) as OMP and LARS, respectively. To have a nondiscriminatory assessment of these methods, we adopt the same dictionary for all methods and interpret the results in terms of both quantitative measures and visual quality.

Our proposed estimation method is limited to a dictionary with three atoms ($K = 3$). For this reason, the method is a priori best suited for low complexity images. To explore its performance, we have constructed a test benchmark on synthetic images in section 4.2. We also discuss the results obtained on binary and natural images in sections 4.3 and 4.4.

4.1 Parameter setting

In all experimental tests, the size of an image patch is set to 3×3 and the overlap between two adjacent patches is 2 pixels. In cases of binary and natural images, the Kodak PhotoCD Dataset (shown in Fig. 5.1) is used as standard images. For the stage of estimation of $p(\boldsymbol{\alpha}_i)$ as described in section 3.1, we use $P = 500000$ patches randomly extracted from standard images to form the database of patches.

In our empirical work, the noisy images are generated from the corresponding noise-free versions by adding Gaussian noise with different levels $\sigma = 10, 20, 30$. In regard to binary tests, all images are converted into binary versions using a threshold method. For objective quality assessment of denoising of vertical structure and natural images, we employ two widely used metrics named peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM). In case of binary images, the PSNR is not sufficient for subjective assessment, since it is a point-based measurement, and mutual relations between pixels are not taken into account. Instead, we adopt two metrics called the Dice ratio and the distance-reciprocal distortion measure (DRDM [151]) for evaluation. The DRDM, which exploits the correlation distance between black and white pixels within image, is a objective distortion measure to judge the similarity between two binary images. As demonstrated in [151], the DRDM metric matches well to subjective assessment by human visual perception. An essential notice is that the small value of DRDM indicate that two binary images are close.

4.2 Denoising of vertical structure images

We first evaluate the denoising performance of our proposed method ProbaEst with two sparse models OPM and LARS on simple vertical structure images. We generate 10 vertical structure test images made of constant-gray-value stripes placed on a dark background as show in Fig. 5.6(b)-(e), where both the width and values of stripes are chosen randomly. We also need an adaptive standard image with similar vertical structures to exploit the redundancy of informations between images. Therefore, we create an image with stripes of gray levels gradually increasing from 1 to 255 as shown on Fig. 5.6(a) and use it as standard image for all comparing methods.

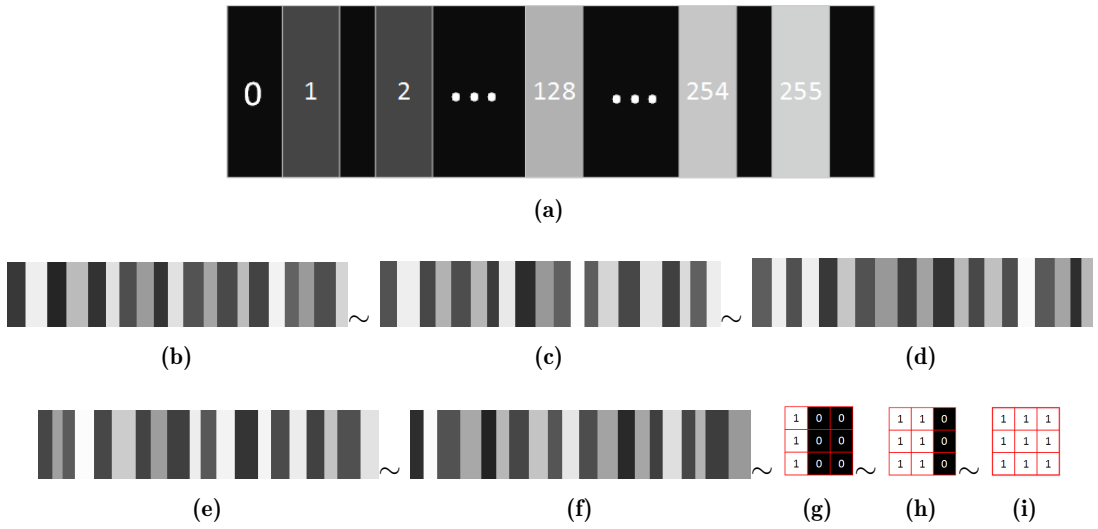


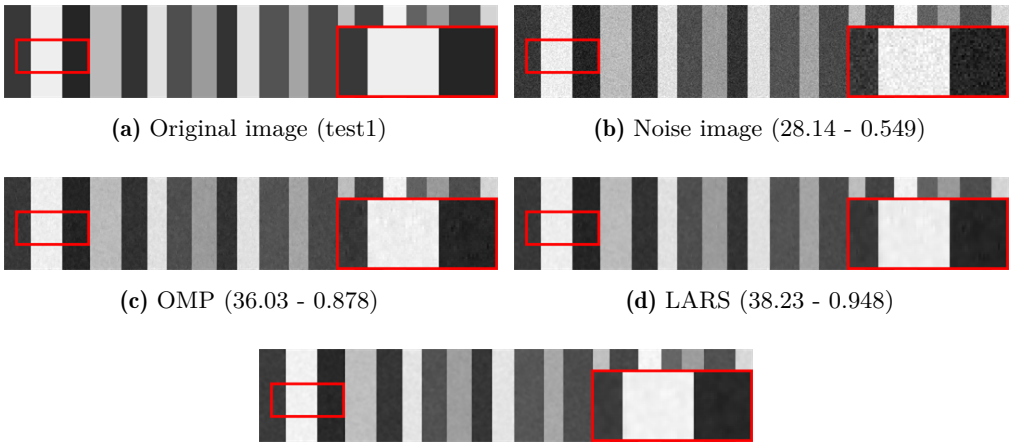
Figure 5.6: *(a) Vertical structure standard images with gradually increasing values of stripes. (b)-(f) Some of vertical structure test images with random width and values of stripes (from test1 to test5). (g)-(i) The patch-form of the three elements (atoms) of the dictionary* $\mathbf{D}$

In our experiments, we observe that the results of denoising are highly dependent on the choice of dictionary $\mathbf{D}$. In regard to our task, where all images are composed of vertical stripes, a dictionary with vertical structural atoms can be more adapted to the image structure. Therefore, we select a dictionary with the vertical structure as exposed in Fig. 5.6(f)-(h). Tables 5.1 and 5.2 provide the results of denoising in terms of PSNR and SSIM, where the best values are in bold red numbers. We can observe that for small noise corruption ($\sigma = 10$), the LARS algorithm (5.14) yields better values of PSNR, but our proposed method ProbaEst achieves higher SSIM performances. For qualitative illustration, we present in Fig. 5.7 the denoising results of image test with noise level $\sigma = 10$. Although the LARS algorithm attains the highest PSNR value, the proposed method restores a more pleasant visual image with less vertical artifacts than the LARS and OMP method.

Images	$\sigma = 10$			$\sigma = 20$			$\sigma = 30$		
	OMP	**LARS**	**ProbaEst**	**OMP**	**LARS**	**ProbaEst**	**OMP**	**LARS**	**ProbaEst**
test1	36.03	**38.23**	35.82	31.20	32.11	**32.30**	28.14	28.72	**29.38**
test2	36.03	**38.21**	35.77	31.31	32.20	**32.43**	28.25	28.75	**29.60**
test3	35.98	**38.14**	35.72	31.24	32.08	**32.47**	28.11	28.52	**29.32**
test4	35.97	**38.08**	35.56	31.23	32.02	**32.54**	28.16	28.67	**29.50**
test5	36.00	**38.06**	35.81	31.34	32.04	**32.58**	28.03	28.46	**29.25**
test6	36.08	**38.23**	35.83	31.28	32.08	**32.53**	28.15	28.60	**29.57**
test7	35.95	**38.08**	35.74	31.16	31.89	**32.51**	28.05	28.41	**29.29**
test8	36.06	**38.10**	35.77	31.26	32.21	**32.58**	28.14	28.58	**29.59**
test9	35.95	**38.08**	35.88	31.19	32.06	**32.74**	28.21	28.68	**29.51**
test10	36.03	**38.11**	35.74	31.42	32.25	**32.66**	28.14	28.43	**29.49**
test11	36.00	**38.12**	35.77	31.29	32.13	**32.65**	28.18	28.61	**29.57**
test12	36.09	**38.16**	35.70	31.32	32.14	**32.51**	28.32	28.73	**29.77**
test13	35.90	**38.18**	35.71	31.49	32.36	**32.67**	28.20	28.76	**29.62**
test14	35.96	**37.93**	35.75	31.31	31.94	**32.63**	28.16	28.63	**29.66**
test15	35.97	**38.08**	35.83	31.31	32.06	**32.79**	28.04	28.43	**29.67**
test16	36.09	**38.19**	35.79	31.31	32.18	**32.61**	28.08	28.62	**29.44**
test17	35.95	**38.11**	35.72	31.18	32.02	**32.55**	27.99	28.44	**29.22**
test18	36.03	**38.18**	35.87	31.32	32.12	**32.62**	28.28	28.66	**29.70**
test19	35.94	**38.01**	35.81	31.19	31.99	**32.46**	28.16	28.72	**29.54**
test20	35.93	**37.88**	35.68	31.18	31.99	**32.57**	28.04	28.37	**29.50**
Average	36.00	**38.11**	35.76	31.28	32.09	**32.57**	28.14	28.59	**29.51**

Table 5.1: *The PSNR measure of denoising of vertical stripes images in Fig. 5.6 with different methods.*

Moreover, it can be seen that with heavier noise levels ($\sigma = 20, 30$), the proposed method is superior to the sparse models, with the improvements are $0.4 \sim 1.4$ dB of PSNR and $0.05 \sim 0.12$ of SSIM. For visual assessment, Figures 5.8, 5.9, 5.10 show the denoising results of image test9, test13 and test15, respectively. It is evident that the OMP and LARS are likely to generate much more artifacts in the denoised images than the proposed method. More precisely, referring to the results from Fig. 5.8(c)-(e), Fig. 5.9(c)-(e), Fig. 5.10(c)-(e), the proposed method ProbaEst is more robust against the vertical artifacts in reconstruction of noisy images and generates more preferable outputs.

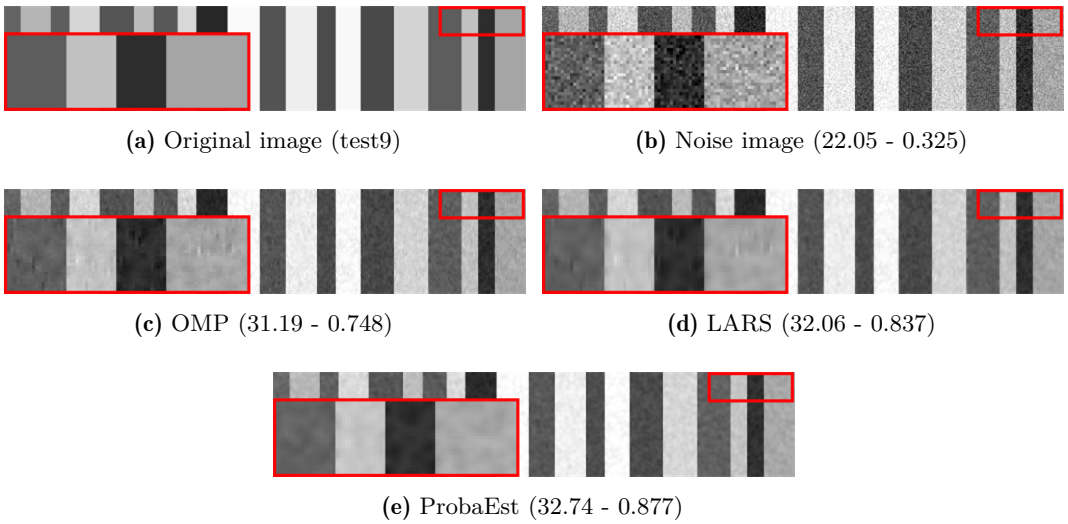


(a) Original image (test1) **(b)** Noise image (28.14 - 0.549)

(c) OMP (36.03 - 0.878) **(d)** LARS (38.23 - 0.948)

(e) ProbaEst (35.82 - 0.961)

Figure 5.7: *Results of denoising on image test1 with* $\sigma = 10$. *(a)-(e) are the original image, noisy image, result of OMP, LARS and ProbaEst, respectively, with the zoom-in of region-of-interest (R.O.I).*

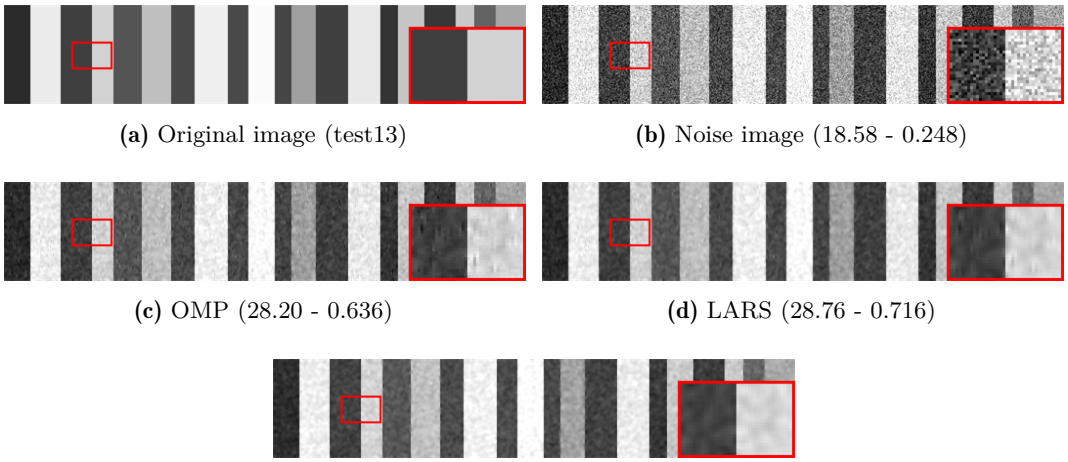


(a) Original image (test9) **(b)** Noise image (22.05 - 0.325)

(c) OMP (31.19 - 0.748) **(d)** LARS (32.06 - 0.837)

(e) ProbaEst (32.74 - 0.877)

Figure 5.8: *Results of denoising on image test9 with* $\sigma = 20$. *(a)-(e) are the original image, noisy image, result of OMP, LARS and ProbaEst, respectively, with the zoom-in of region-of-interest (R.O.I).*

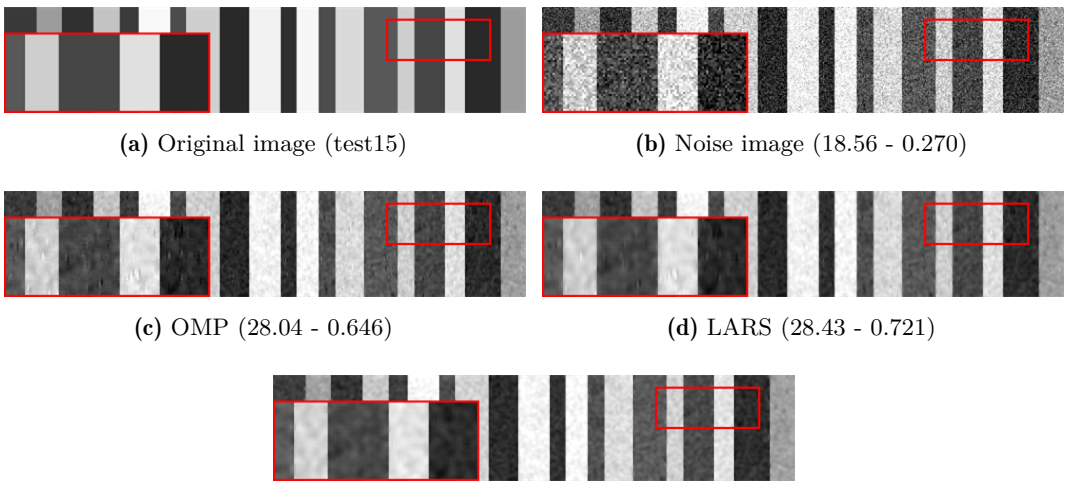


(a) Original image (test13) **(b)** Noise image (18.58 - 0.248)

(c) OMP (28.20 - 0.636) **(d)** LARS (28.76 - 0.716)

(e) ProbaEst (29.62 - 0.768)

Figure 5.9: *Results of denoising on image test13 with $\sigma = 30$. (a)-(e) are the original image, noisy image, result of OMP, LARS and ProbaEst, respectively, with the zoom-in of region-of-interest (R.O.I).*



(a) Original image (test15) **(b)** Noise image (18.56 - 0.270)

(c) OMP (28.04 - 0.646) **(d)** LARS (28.43 - 0.721)

(e) ProbaEst (29.67 - 0.773)

Figure 5.10: *Results of denoising on image test15 with $\sigma = 30$. (a)-(e) are the original image, noisy image, result of OMP, LARS and ProbaEst, respectively, with the zoom-in of region-of-interest (R.O.I).*

Images	$\sigma = 10$			$\sigma = 20$			$\sigma = 30$		
	OMP	**LARS**	**ProbaEst**	**OMP**	**LARS**	**ProbaEst**	**OMP**	**LARS**	**ProbaEst**
test1	0.878	0.948	**0.961**	0.742	0.830	**0.868**	0.632	0.712	**0.758**
test2	0.879	0.950	**0.962**	0.751	0.837	**0.872**	0.640	0.718	**0.768**
test3	0.879	0.949	**0.963**	0.752	0.836	**0.874**	0.642	0.716	**0.767**
test4	0.882	0.950	**0.962**	0.759	0.842	**0.876**	0.645	0.720	**0.769**
test5	0.882	0.950	**0.963**	0.757	0.841	**0.877**	0.640	0.715	**0.764**
test6	0.884	0.951	**0.963**	0.760	0.843	**0.878**	0.645	0.721	**0.770**
test7	0.880	0.949	**0.962**	0.752	0.835	**0.871**	0.640	0.716	**0.767**
test8	0.885	0.952	**0.964**	0.758	0.842	**0.877**	0.646	0.722	**0.772**
test9	0.881	0.950	**0.963**	0.748	0.837	**0.877**	0.649	0.726	**0.774**
test10	0.885	0.952	**0.964**	0.756	0.839	**0.876**	0.646	0.721	**0.770**
test11	0.881	0.950	**0.962**	0.754	0.839	**0.877**	0.645	0.721	**0.770**
test12	0.882	0.951	**0.963**	0.754	0.839	**0.875**	0.651	0.728	**0.773**
test13	0.875	0.947	**0.961**	0.757	0.842	**0.876**	0.636	0.716	**0.768**
test14	0.884	0.951	**0.963**	0.766	0.845	**0.879**	0.656	0.731	**0.779**
test15	0.881	0.950	**0.963**	0.760	0.843	**0.879**	0.646	0.721	**0.773**
test16	0.881	0.951	**0.963**	0.747	0.834	**0.872**	0.631	0.710	**0.763**
test17	0.881	0.950	**0.963**	0.751	0.835	**0.872**	0.642	0.716	**0.766**
test18	0.879	0.950	**0.963**	0.750	0.835	**0.870**	0.642	0.719	**0.771**
test19	0.879	0.949	**0.962**	0.754	0.837	**0.873**	0.646	0.724	**0.771**
test20	0.886	0.952	**0.964**	0.770	0.851	**0.884**	0.657	0.731	**0.782**
Average	0.881	0.950	**0.963**	0.755	0.839	**0.875**	0.644	0.720	**0.770**

Table 5.2: *The SSIM measure of denoising of vertical stripes images in Fig. 5.6 with different methods.*

4.3 Denoising of binary images

In this part, we will not only compare the performance of the proposed method with the sparse patch-based algorithms OMP and LARS, but also with others denoising methods developed for binary images, called Iterated Conditional Mode (ICM) [152] and Graph Cuts [153]. The ICM method is based on the maximization of local conditional probabilities, which begins from some initial assignment of states to pixels (0 or 1), then cycles through the pixels, greedily maximizes the potential of each pixel given its neighbors until no further local improvements are possible. While in the latter method, we construct a graph of nodes (sources and sinks) from input image and optimize the energy function by maximizing amount of flow passing from the source to the sink.

In the experiments, all images are converted into binary versions using a simple

thresholding method as a pre-processing step. We used all the 24 images in the Kodak PhotoCD Dataset (Fig. 5.1) as standard images and 5 natural images of Baby, House, Lena, Monarch and Peppers shown in Fig. 2.1 are adopted for the evaluation. The dictionary is chosen as the same as Fig. 5.6(f)-(h) for the ProbaEst, OMP and LARS methods.

Images	$\sigma = 20$					$\sigma = 30$					$\sigma = 40$				
	OMP	**LARS**	**ICM**	**Gr.Cuts**	**ProbaEst**	**OMP**	**LARS**	**ICM**	**Gr.Cuts**	**ProbaEst**	**OMP**	**LARS**	**ICM**	**Gr.Cuts**	**ProbaEst**
House	0.875	0.877	**0.902**	0.819	0.900	0.856	0.856	**0.883**	0.797	**0.883**	0.838	0.838	0.871	0.788	**0.875**
Lena	0.908	0.917	0.940	0.916	**0.941**	0.883	0.884	0.921	0.889	**0.923**	0.847	0.848	**0.902**	0.871	0.900
Monarch	0.923	0.930	0.930	0.876	**0.948**	0.894	0.894	0.894	0.808	**0.931**	0.863	0.863	0.861	0.759	**0.923**
Peppers	0.941	0.948	0.955	0.946	**0.963**	0.918	0.918	0.930	0.919	**0.951**	0.888	0.888	0.907	0.885	**0.939**
Baby	0.971	0.974	0.980	0.976	**0.980**	0.961	0.961	0.974	0.969	**0.976**	0.945	0.945	0.967	0.962	**0.970**
Average	0.923	0.929	0.941	0.906	**0.946**	0.902	0.902	0.920	0.876	**0.932**	0.876	0.876	0.901	0.853	**0.921**

Table 5.3: *Comparison of denoising methods for binary images using Dice ratio.*

Images	$\sigma = 20$					$\sigma = 30$					$\sigma = 40$				
	OMP	**LARS**	**ICM**	**Gr.Cuts**	**ProbaEst**	**OMP**	**LARS**	**ICM**	**Gr.Cuts**	**ProbaEst**	**OMP**	**LARS**	**ICM**	**Gr.Cuts**	**ProbaEst**
House	10.56	10.00	6.44	19.90	**6.24**	12.73	12.72	8.91	23.90	**8.48**	14.93	14.96	10.32	25.21	**9.51**
Lena	9.43	8.24	4.96	8.57	**4.61**	12.23	12.21	7.04	12.52	**6.57**	16.58	16.53	9.43	15.50	**9.29**
Monarch	4.49	3.93	4.25	8.84	**2.23**	6.51	6.50	7.13	15.99	**3.26**	9.00	9.01	10.27	21.97	**3.85**
Peppers	6.19	5.22	4.31	5.17	**2.90**	8.99	8.99	7.73	9.11	**4.16**	12.93	12.94	10.98	14.71	**5.69**
Baby	4.08	3.44	1.90	2.43	**1.72**	5.77	5.78	2.77	3.53	**2.37**	9.05	9.07	4.17	5.04	**3.40**
Average	6.95	5.96	4.37	8.98	**3.54**	9.24	9.24	6.71	13.01	**4.96**	12.49	12.50	9.03	16.48	**6.34**

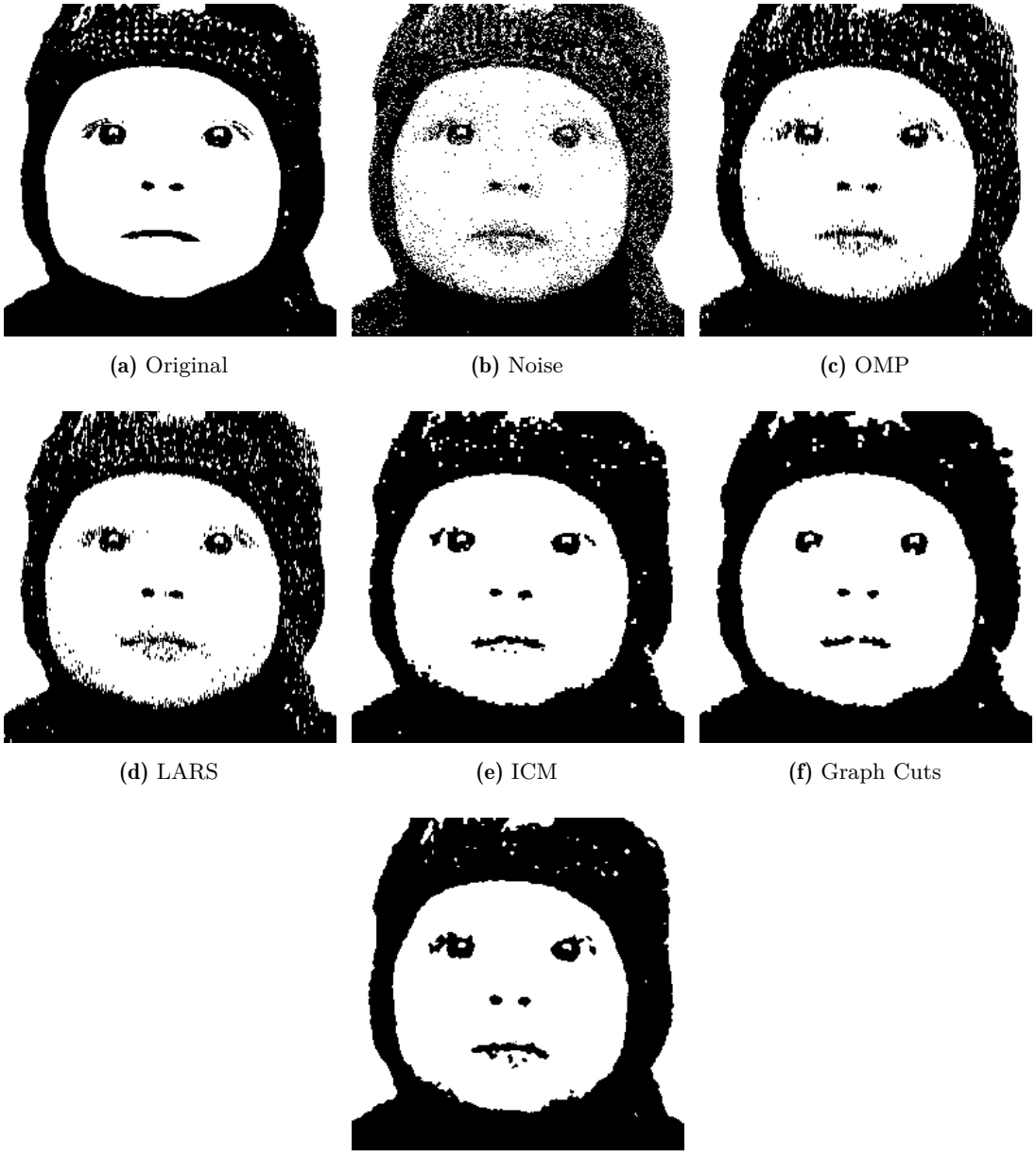
Table 5.4: *Comparison of denoising methods for binary images using DRDM measurement.*

Table 5.3 and 5.4 present the results of denoising of binary images in terms of Dice Ratio and DRDM measurement, respectively, where the best values are shown in bold red numbers. We can observe that the proposed method achieves better assessment metrics than the other methods, which proves the effectiveness of our method for binary image denoising.

For further illustration, Fig. 5.11, Fig. 5.12 and Fig. 5.13 show the denoising results of competitive methods. In Fig. 5.11, the ICM, Graph Cuts and proposed method ProbaEst achieve higher noise-reduction than the sparsity ones, but we have lost some small details on the beanie. The results of ICM and ProbaEst are very close and similar with the original image, while the Graph Cuts discards some black regions around the eyes in the image. In the contrary, the OMP and LARS visually conserve more details on the beanie, but they produce much more noise on the face, as well as the vertical artifacts within the image.

In addition, Fig. 5.12 and Fig. 5.13 display the denoising results of binary image of monarch and peppers, respectively. As can be observed, proposed method is more efficient in noise reduction than the sparsity (OMP, LARS) and the ICM. More particularly, the OMP and LARS generate images with vertical artifacts, while the Graph

Cuts tends to produce an over-expansion foreground image. Visually, proposed method achieves very competitive denoising performance, where its results are more similar compare to the original images.

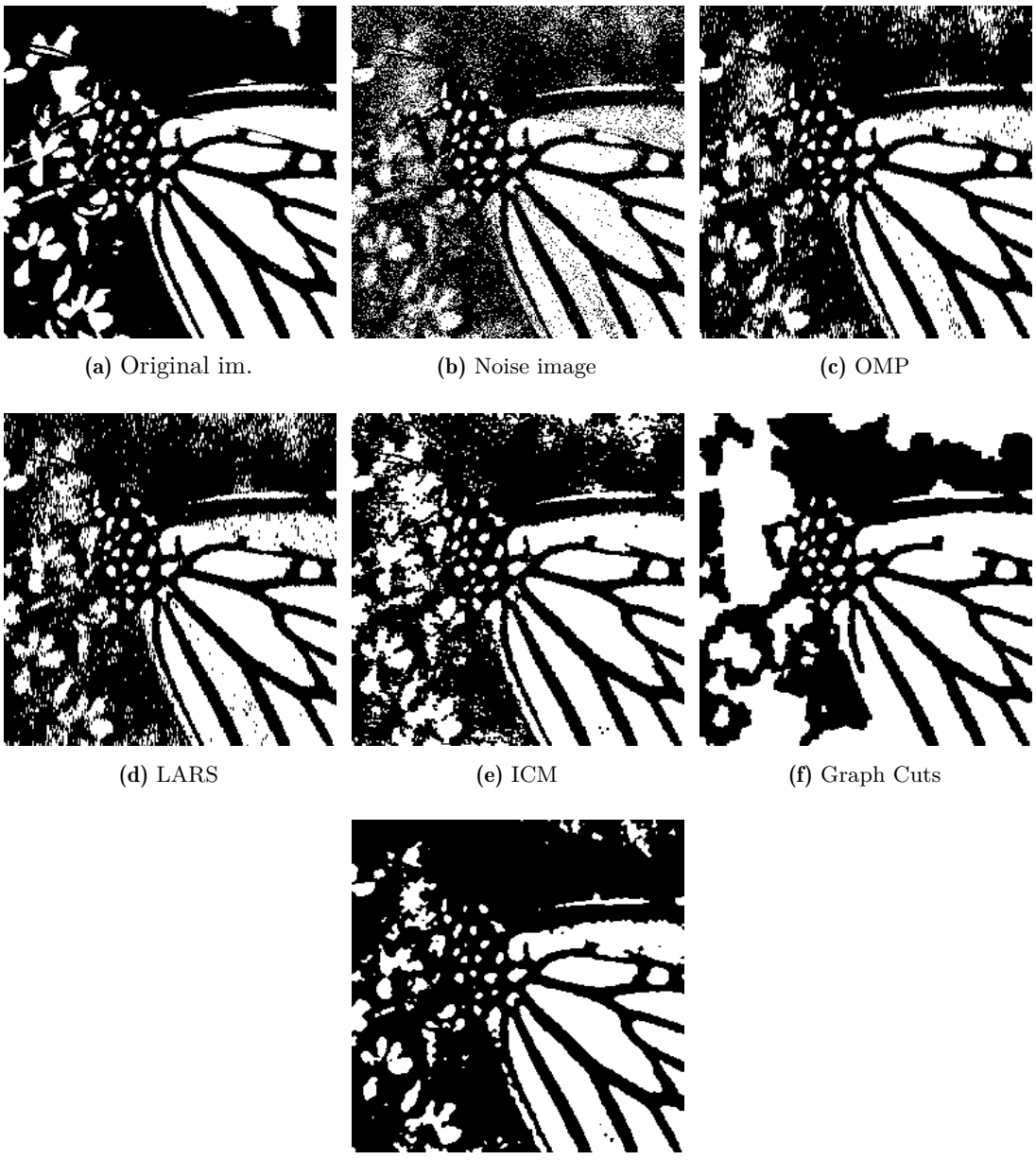


(a) Original **(b)** Noise **(c)** OMP

(d) LARS **(e)** ICM **(f)** Graph Cuts

(g) ProbaEst

Figure 5.11: *Results of denoising on binary image of Baby with* $\sigma = 30$.



(a) Original im. **(b)** Noise image **(c)** OMP

(d) LARS **(e)** ICM **(f)** Graph Cuts

(g) ProbaEst

Figure 5.12: *Results of denoising on binary image of Monarch with* $\sigma = 30$.

(a) Original **(b)** Noise **(c)** OMP

(d) LARS **(e)** ICM **(f)** Graph Cuts

(g) ProbaEst

Figure 5.13: *Results of denoising on binary image of Peppers with* $\sigma = 30$.

4.4 Denoising of natural images

In the rest of our work in this chapter, we compare the performance of the proposed method with two sparsity-based algorithms OMP and LARS for denoising on 20 widely used natural images as shown in Fig. 2.1. We use all 24 images in the Kodak dataset (Fig. 5.1) as standard images $\{\mathbf{x}^s\}$. Moreover, we carry out the experiments by adopting the same dictionary for all methods, which is directly trained from each noisy image, using the K-SVD algorithm [2].

Images	$\sigma = 10$			$\sigma = 20$			$\sigma = 30$		
	OMP	**LARS**	**ProbaEst**	**OMP**	**LARS**	**ProbaEst**	**OMP**	**LARS**	**ProbaEst**
Airfield	**32.19**	32.04	31.74	28.48	**29.03**	29.05	26.15	26.99	**27.19**
Airplane	33.57	**34.40**	33.96	29.37	30.55	**30.60**	26.74	28.03	**28.30**
Baboon	**28.45**	27.05	26.28	25.42	**24.81**	24.73	23.35	23.11	**23.20**
Baby	33.66	**34.69**	34.64	29.57	**31.12**	30.98	27.07	28.75	**28.91**
Barbara	**31.32**	30.66	29.75	**27.54**	27.31	27.31	25.06	25.15	**25.35**
Boat	33.56	**34.19**	33.96	29.20	30.21	**30.34**	26.54	27.83	**28.07**
Bridge	**30.28**	29.29	28.82	**26.71**	26.56	26.64	24.67	24.94	**25.01**
Cameraman	**31.22**	30.53	30.07	27.80	28.04	**28.07**	25.63	26.17	**26.53**
Couple	**32.49**	32.35	31.93	28.42	28.84	**28.96**	26.02	26.82	**27.06**
Fruits	33.37	**34.00**	33.68	29.42	**30.61**	30.54	26.81	28.13	**28.44**
Boy	31.83	**31.59**	31.53	28.48	**29.29**	29.27	26.29	27.51	**27.68**
Hill	**32.19**	32.01	31.85	28.49	29.14	**29.22**	26.16	27.20	**27.41**
House	33.16	**33.63**	33.35	29.22	30.19	**30.28**	26.60	27.79	**28.15**
Jellybeans	34.40	**36.35**	35.94	29.89	**31.62**	31.57	27.12	28.75	**29.01**
Leaves	**31.44**	31.05	30.19	27.52	27.41	**27.69**	24.86	24.80	**25.49**
Lena	33.09	**33.59**	33.37	29.21	**30.36**	30.34	26.69	28.09	**28.31**
Man	32.44	**32.47**	32.20	28.61	29.39	**29.45**	26.22	27.34	**27.57**
Monarch	32.27	**32.39**	31.89	28.44	28.95	**29.15**	25.88	26.57	**27.05**
Peppers	32.54	**32.71**	32.36	28.92	**29.95**	29.89	26.52	27.78	**28.09**
Zelda	32.81	**33.20**	33.02	29.06	**30.14**	30.13	26.59	27.96	**28.19**
Average	32.31	**32.41**	32.03	28.49	29.17	**29.21**	26.05	26.99	**27.25**

Table 5.5: *Denoising performance on 20 natural images with respect to PSNR.*

In reality, with a simple dictionary of three atoms, the denoising results will not be good enough for visual quality requirement. But we have to emphasize again that the objective of our research is to demonstrate that using the estimated probability distribution as the image prior is more efficient than a sparsity model for noise removal. As illustrated in Tables 5.5 and 5.6, the proposed method yields better values of PSNR and SSIM than the sparsity algorithms in most of cases of high noise levels ($\sigma = 20, 30$).

Images	$\sigma = 10$			$\sigma = 20$			$\sigma = 30$		
	OMP	**LARS**	**ProbaEst**	**OMP**	**LARS**	**ProbaEst**	**OMP**	**LARS**	**ProbaEst**
Airfield	**0.845**	0.839	0.840	0.713	0.743	**0.744**	0.607	0.655	**0.665**
Airplane	0.848	0.899	**0.901**	0.702	0.790	**0.793**	0.584	0.682	**0.704**
Baboon	**0.879**	0.841	0.821	**0.768**	0.736	0.735	0.668	0.641	**0.644**
Baby	0.856	0.902	**0.904**	0.711	**0.795**	0.792	0.596	0.692	**0.707**
Barbara	0.867	**0.874**	0.862	0.740	**0.763**	0.762	0.625	0.653	**0.666**
Boat	0.869	0.905	**0.910**	0.732	0.797	**0.801**	0.618	0.695	**0.714**
Bridge	**0.892**	0.867	0.862	**0.775**	0.761	0.767	0.678	0.681	**0.683**
Cameraman	0.849	**0.882**	0.881	0.701	0.765	**0.768**	0.592	0.662	**0.682**
Couple	0.874	**0.879**	**0.879**	0.745	0.775	**0.782**	0.640	0.686	**0.702**
Fruits	0.839	**0.876**	**0.876**	0.690	**0.768**	0.765	0.568	0.658	**0.678**
Boy	0.807	**0.791**	0.789	0.647	**0.675**	0.674	0.525	0.575	**0.583**
Hill	**0.852**	0.844	0.847	0.717	0.745	**0.750**	0.608	0.658	**0.670**
House	0.833	0.855	**0.856**	0.689	0.752	**0.754**	0.577	0.653	**0.674**
Jellybeans	0.861	**0.937**	**0.937**	0.719	**0.827**	0.823	0.601	0.711	**0.731**
Leaves	0.944	**0.961**	0.957	0.882	0.906	**0.908**	0.818	0.842	**0.858**
Lena	0.834	**0.857**	0.855	0.694	**0.757**	0.755	0.578	0.660	**0.675**
Man	0.863	**0.881**	**0.881**	0.729	0.779	**0.782**	0.618	0.685	**0.699**
Monarch	0.908	**0.936**	0.933	0.813	**0.862**	**0.862**	0.721	0.780	**0.796**
Peppers	0.813	**0.819**	0.816	0.666	**0.720**	**0.720**	0.552	0.625	**0.643**
Zelda	0.838	**0.866**	0.865	0.694	**0.759**	0.757	0.576	0.655	**0.670**
Average	0.859	**0.875**	0.874	0.726	0.774	**0.775**	0.617	0.677	**0.692**

Table 5.6: *Denoising performance on 20 natural images in term of SSIM.*

Particularly, with $\sigma = 30$, the improvements are around 0.3 to 1.2 dB on average in term of PSNR.

For visual assessment, Fig. 5.14, 5.15, 5.16 show the denoising results of competing methods on image of Peppers, Airplane and Baby, respectively. Globally, the three methods produce acceptably visual results. For better evaluation, we display the zooming of some regions in the images. It can be observed that the proposed method ProbaEst demonstrates its effectiveness in reconstruction of degraded images with less artifacts than the sparsity models.

5 Conclusion

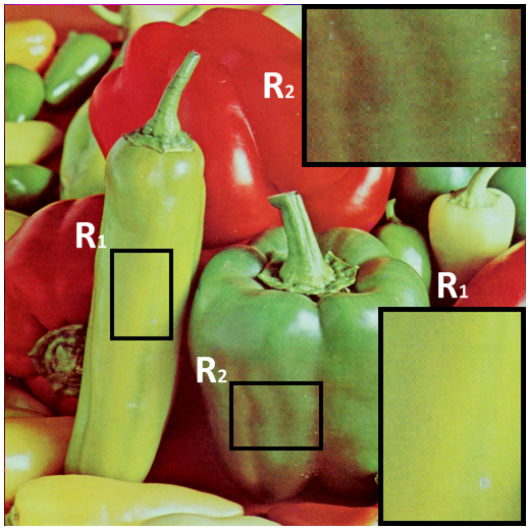
This chapter gives some significant remarks on the use of sparse model to describe the priori knowledge of distribution of representation coefficients of latent image patches, in the perspective of Bayesian MAP. First, the choice of an available sparse model in a

image restoration algorithm is an arbitrary decision. Second, the selected model may be insufficient to characterize the true distribution of image patches.

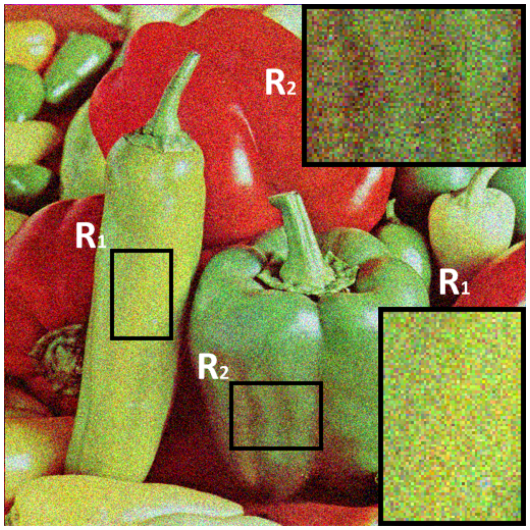
In this chapter, we investigate the distribution of representation coefficients of image patches in the vector space generated from the atoms in an available dictionary to study a prior model for patch-based image denoising. The principal idea of our research is built upon the interesting property of redundancy of local patterns across images, where local patches tend to repeatedly appear in an underlying image and the standard images. Therefore, the patches in a latent image and in the database are considered to share the same distribution of representation coefficients. Thus we proposed to estimate a prior model from the empirical distribution of representation coefficients of patches in the database and use it to restore a degraded image.

We have demonstrated that with a small dimensional vector space (or a dictionary contains up to 3 atoms), we can produce an accurate estimation of the true distribution of representation coefficients of image patches in the database by constructing a histogram with piecewise constant functions. An efficient median-based division approach has been proposed to precisely model the distribution of image patches with small number of bins of histogram in each dimension of the vector space. We also introduce a framework that takes into account the estimation-based probability distribution in reconstruction of a noisy image. By exploring experiments on multiple type of images, we have proved that using estimated prior from the distribution of patches in the database can improve the performance of a denoising algorithm, rather than adopting an arbitrary sparse model.

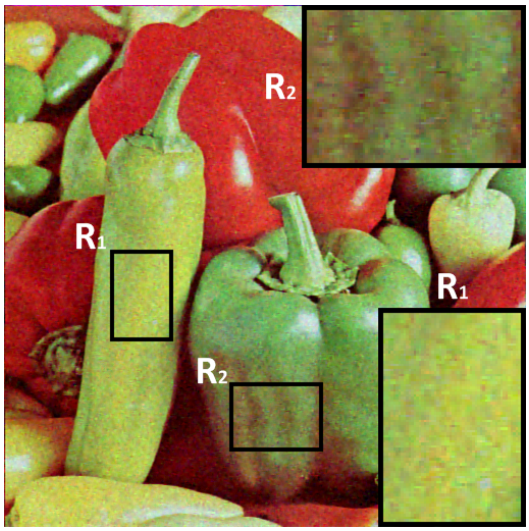
However, the proposed approach with piecewise constant estimation of probability distribution is inefficient for high dimensional vector space, which corresponds to a large dictionary in practice, due to the curse of dimensionality and the high computational complexity. To deal with this problem, we can use a kernel density estimator such as the Gaussian Mixture Model (GMM) to learn the prior model of probability distribution function from the distribution of external patches in high-dimensional vector space. We will expand the proposed method, as well as discuss the use of GMM in solving the denoising optimization in the next chapter.



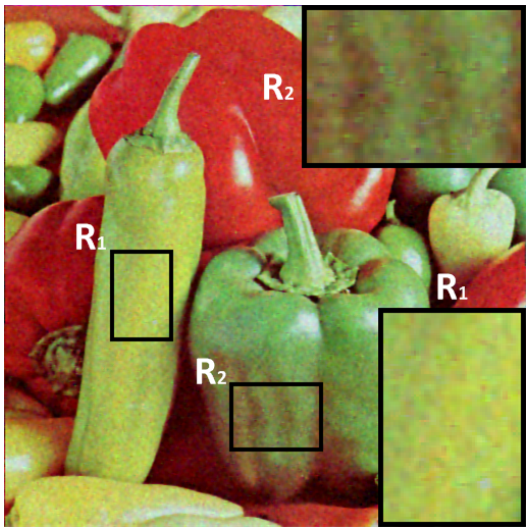
(a) Original



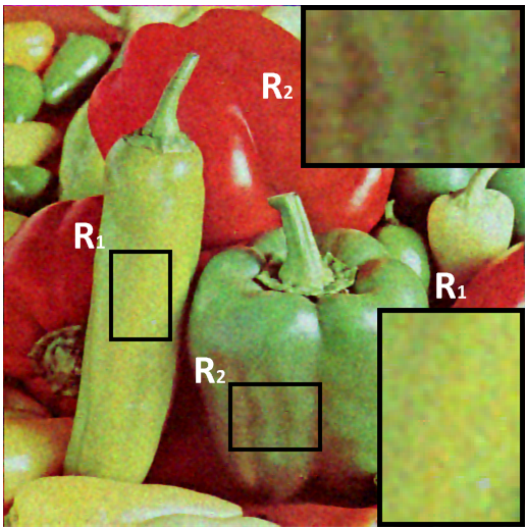
(b) Noise (18.59 - 0.219)



(c) OMP (26.52 - 0.552)

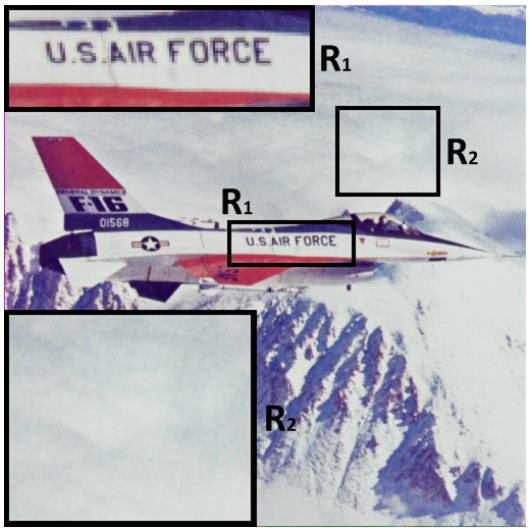


(d) LARS (27.78 - 0.625)

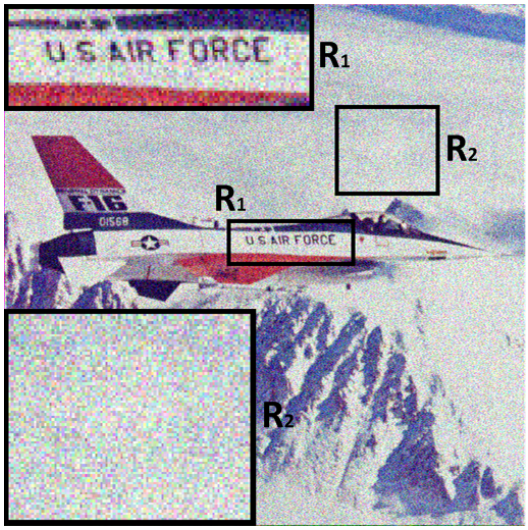


(e) ProbaEst (28.09 - 0.643)

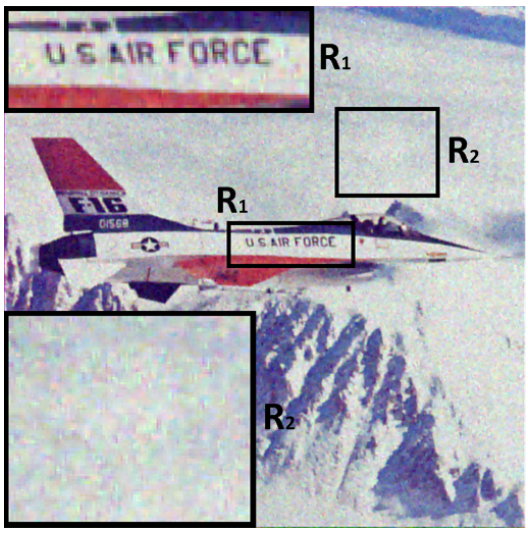
Figure 5.14: *Results of denoising on image of peppers with* $\sigma = 30$.



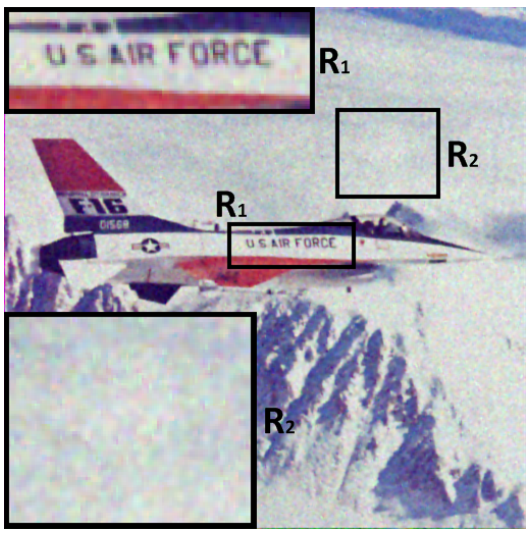
(a) Original



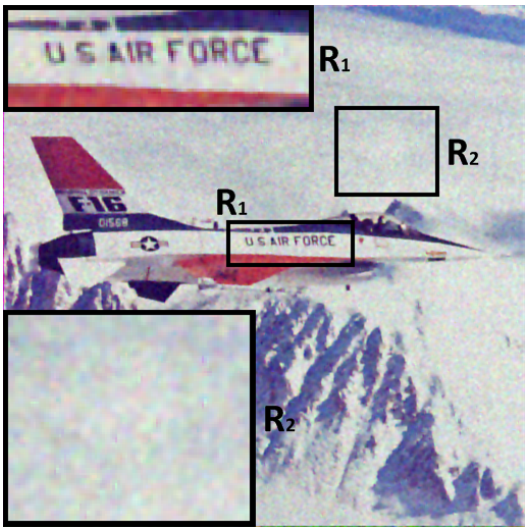
(b) Noise (18.58 - 0.22)



(c) OMP (26.74 - 0.584)

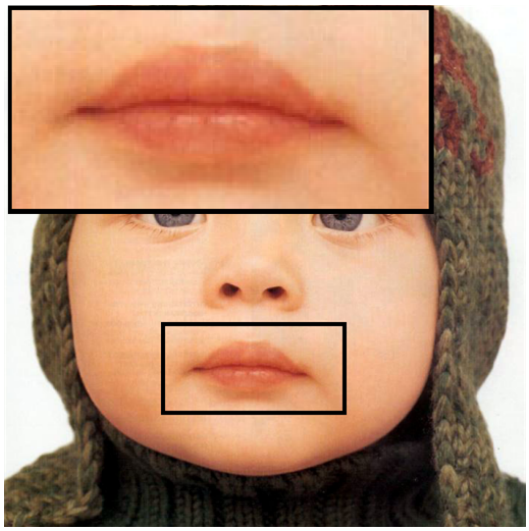


(d) LARS (28.03 - 0.682)

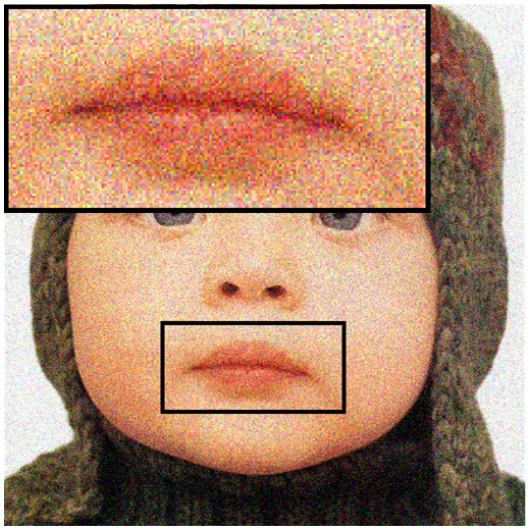


(e) ProbaEst (28.30 - 0.704)

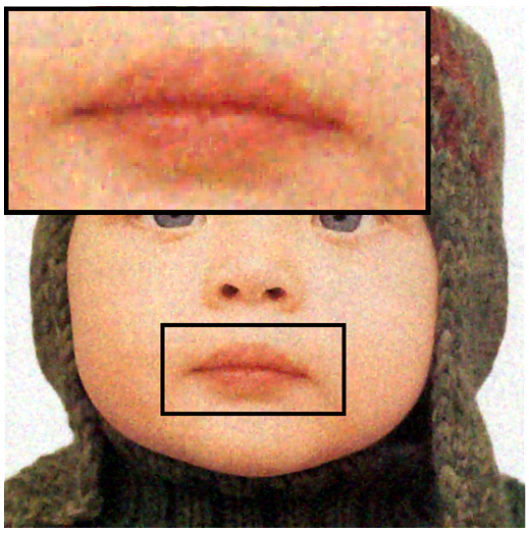
Figure 5.15: *Results of denoising on image of airplane with* $\sigma = 30$.



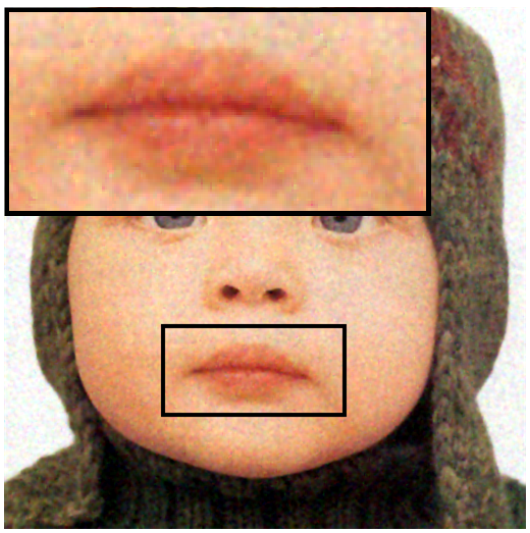
(a) Original



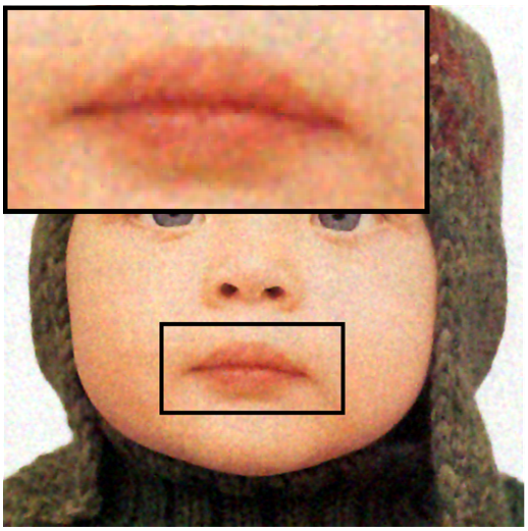
(b) Noise (18.60 - 0.209)



(c) OMP (27.07 - 0.596)



(d) LARS (28.75 - 0.692)



(e) ProbaEst (28.91 - 0.707)

Figure 5.16: *Results of denoising on image of airplane with* $\sigma = 30$.

CHAPTER 6

Number of Useful Components in Gaussian Mixture Models for Patch-based Image Denoising

Abstract

In the previous work, we showed that estimating the actual probability distribution of image patches improves image reconstruction. However our estimation method is limited to low dimensional dictionaries, so we now consider Gaussian Mixture Models (GMMs) for better representing the prior. However, when using GMMs as a prior for image denoising under the Bayesian maximum a posteriori (MAP) perspective, only a single prominent Gaussian component is usually selected to recover a noisy image patch, which leads to computationally efficient implementations. In this chapter, we attempt to justify this on several image datasets by evaluating the number of Gaussian components required for recovering patches. We show that even patches without a prominent component in the prior can be recovered with little loss of performance. Comparisons between two dictionary choices and between small and large models suggest that large gains are attainable, but only one component is required for reconstruction. A summary of our work has been submitted in the 2018 International Conference on Image and Signal Processing (ICISP[1]).

[1]Dai-Viet Tran, Sébastien Li-Thiao-Té, Marie Luong, Françoise Dibos, "Number of Useful Components in Gaussian Mixture Models for Patch-based Image Denoising", accepted to ICISP, July 2-4, 2018.

Chapter content

1 Introduction

The initial results of the proposed method in chapter 5 have validated the successfulness of using the estimation of empirical distribution of patches in the database to recover a degraded image, rather than adopting an arbitrarily available sparse model. However, the developed framework using histogram to estimate the distribution $p(\boldsymbol{\alpha}_i)$ of patches has limitation in dealing with high-dimensional vector space $\Omega_\alpha \subset R^K$, due to the curse of dimensionality where a large portion of number of hypercube bins may probably be empty. Moreover, with a large dictionary $K \gg 3$, the complexity of histogram estimation algorithm dramatically increases and makes the implementation impossible. Therefore, in denoising applications presented in chapter 5, the number of atoms in the dictionary $\mathbf{D}$ is chosen really small with $K = 3$, which corresponds to a three-dimensional vector space Ω_α. However, in practical applications, a complete dictionary ($K = n$) or an over-complete dictionary ($K > n$) has demonstrated its out-performance for image restoration comparing to a small dictionary.

One representative solution for dealing with estimation of probability in high-dimensional vector space Ω_α is the use of parametric kernel density estimators. Among various contributions, the Gaussian Mixture Model (GMM) has been commonly exploited in representing the distribution of data in a high-dimensional space and successfully employed in image restoration [5, 89, 90, 92, 93, 97]. The fundamental of these approaches is to estimate the empirical distribution of patches in the database by a linear combination (often known as a mixture) of finite number of M Gaussian distributions. However, when employing the GMM as a prior model for solving image denoising under the Bayesian MAP perspective in (5.3), only a single prominent Gaussian component is usually selected to represent the probability distribution $p(\boldsymbol{\alpha}_i)$, which leads to computationally efficient implementations.

In this chapter, we explore the Gaussian mixture model to estimate the distribution $p(\boldsymbol{\alpha}_k^s)$ of the representation coefficients of patches in the database and use it as an image prior to represent the probability distribution $p(\boldsymbol{\alpha}_i)$ of patches in the latent image. A questionable issue is how many useful components in GMM should be used to describe the probability distribution $p(\boldsymbol{\alpha}_i)$ and thus reconstruct a noisy patch. Can we apply all M Gaussian components or only one prominent component is enough? To our knowledge, justification for this approach is lacking in the literature. Therefore, earlier than making comparison the denoising performance of the proposed method with other competing sparsity models as in chapter 5, we concentrate in this chapter to verify this scientific question on several image datasets by evaluating the number of

Gaussian components required for recovering patches.

The remainder of this chapter is organized as follows. Section 2 briefly introduces the patch-based image denoising problem with the GMM prior, as well as our motivation of exploiting GMM to estimate the probability distribution $p(\boldsymbol{\alpha}_i)$ from the empirical distribution of representation coefficients of patches in the database. Section 3 gives a quickly description of the datasets used for evaluation. The details of the Gaussian mixture model as Prior for Image Denoising (GPID) method are presented in section 4. The experimental results on a combination of two dictionary choices and two model complexities of the GMM are shown in section 5. This chapter will be ended with some discussion and perspective in section 6.

2 Problem Statement and Motivation

In this work, we consider the Gaussian Mixture Model (GMM) for distribution estimation. The fundamental concept is to suppose that each representation coefficients vector $\boldsymbol{\alpha}_k^s$ of patches in the database is drawn from a mixture of M Gaussian distributions of unknown parameters $\{\boldsymbol{\mu}_m, \Sigma_m\}$ as described in (6.1)

$$p(\boldsymbol{\alpha}_k^s) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{\alpha}_k^s | \boldsymbol{\mu}_m, \Sigma_m) \tag{6.1}$$

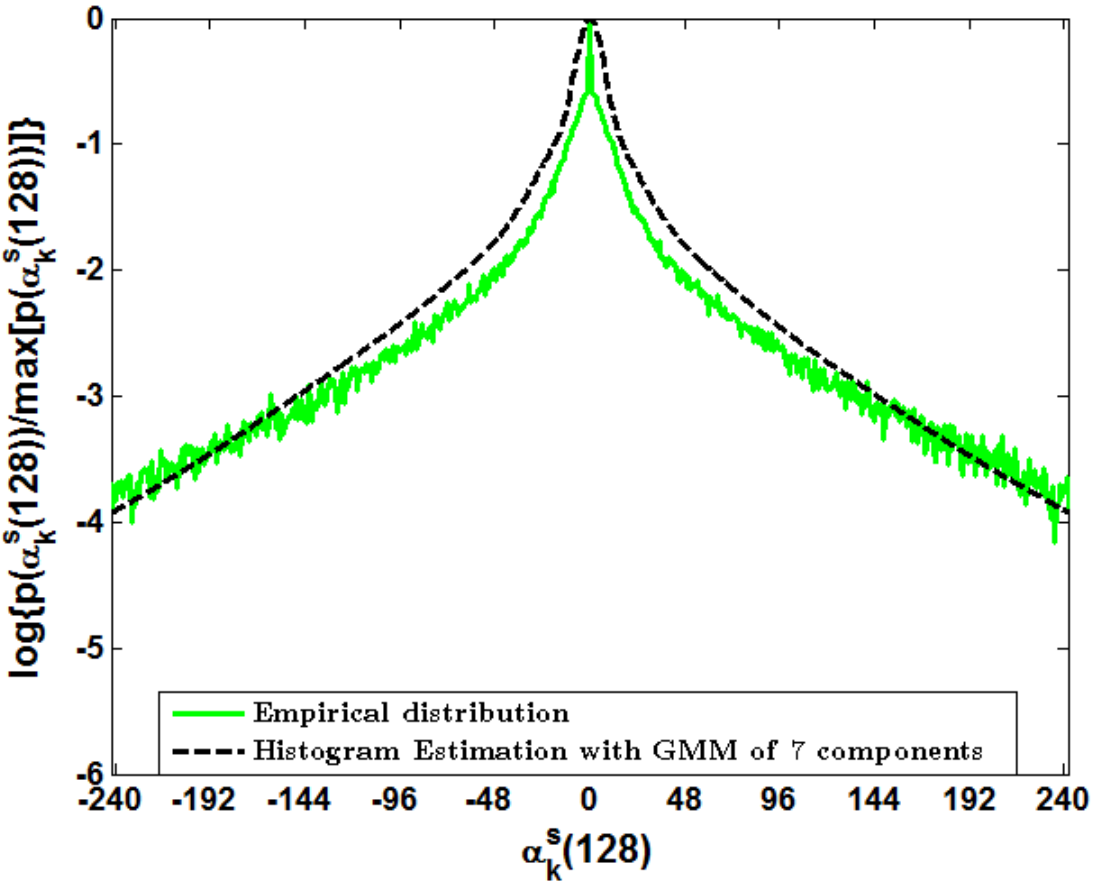
Where $\{\pi_m | m = 1, \ldots, M\}$ are the mixing weights with $\sum_{m=1}^{M} \pi_m = 1$, $\boldsymbol{\mu}_m \in R^K$ and $\Sigma_m \in R^{K \times K}$ are the mean and covariance matrix of the m-th Gaussian component, respectively. $\mathcal{N}(\boldsymbol{\alpha}_k^s | \boldsymbol{\mu}_m, \Sigma_m)$ is the Gaussian distribution defined in (6.2)

$$\mathcal{N}(\boldsymbol{\alpha}_k^s | \boldsymbol{\mu}_m, \Sigma_m) = \frac{1}{\sqrt{2\pi |\Sigma_m|}} \exp\left(-\frac{1}{2} (\boldsymbol{\alpha}_k^s - \boldsymbol{\mu}_m)^T \Sigma_m^{-1} (\boldsymbol{\alpha}_k^s - \boldsymbol{\mu}_m) \right) \tag{6.2}$$

where $|\Sigma_m|$ is the determinant of Σ_m.

In order to demonstrate the benefit of using GMM in approximately represent the distribution of patches, we study some simulations on the image of Boat shown in Fig. 2.1(f) as follows. We added Gaussian noise with $\sigma = 20$ to the image, and randomly extracted 100000 patches of size 8×8 ($n = 64$) to train a dictionary of $K = 256$ atoms ($\mathbf{D} \in R^{64 \times 256}$) using the K-SVD algorithm (please refer to section 2.5 of chapter 2 for detailed description).

We collected 200 training images from the Berkeley segmentation dataset [6] as standard images $\{\mathbf{x}^s\}$. A database of $P = 200000$ clean patches $\{\mathbf{x}_k^s | \mathbf{x}_k^s \in R^{64}; k = 1, \ldots, 200000\}$ are randomly extracted from standard images and we compute their corresponding representation coefficients vectors $\boldsymbol{\alpha}_k^s$ using the least mean square error



(a) Distribution $p(\boldsymbol{\alpha}_k^s(128))$ of representation coefficients on 128-th atom

Figure 6.1: *The empirical distribution* $p(\boldsymbol{\alpha}_k^s(128))$ *of representation coefficients of image patches in the database corresponding to the* 128*-th atom in the dictionary (green curve) and the estimated distribution using GMM with* $M = 7$ *components (black dash line).*

estimation as in (5.4). We examined the distribution of representation coefficients $\boldsymbol{\alpha}_k^s(128)$ along the 128-th dimension of the vector space Ω_α, which corresponds to the weight of the 128-th atom of the dictionary. From 200000 values of $\{\boldsymbol{\alpha}_k^s(128)|k = 1,\ldots,200000\}$, we learned a GMM model that approximates the distribution $p(\boldsymbol{\alpha}_k^s(128))$. Fig. 6.1 shows the empirical distribution $p(\boldsymbol{\alpha}_k^s(128))$ of representation coefficients (in green curve) of 200000 patches in the database. We also plot the estimated distribution of GMM model in black dash line. It can be observed that, although there are some fitting errors, but using a mixture of $M = 7$ Gaussian components can acceptably characterize the true distribution $p(\boldsymbol{\alpha}_k^s(128))$ of representation coefficients of patches in the database, especially for the tails of the distribution. Therefore, the GMM model can be exploited to estimate the distribution of image patches.

Our aim is to estimate the parameter $\{\pi_m, \boldsymbol{\mu}_m, \Sigma_m|m = 1,\ldots,M\}$ from the empirical distribution $p(\boldsymbol{\alpha}_k^s)$ of P patches in the database. After that, the obtained GMM is used

to model the prior distribution of representation coefficients $p(\boldsymbol{\alpha}_i)$ of latent patches $\mathbf{x}$.

$$p(\boldsymbol{\alpha}_i) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{\alpha}_i | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \tag{6.3}$$

Hence, the denoising problem in (5.3) becomes:

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\{ ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda \log \left(\sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{\alpha}_i | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \right) \right\} \tag{6.4}$$

Optimization of (6.4) is a very time-consuming process and pose a huge challenge in implementation of denoising algorithm.

Using GMM is not a new approach in image restoration with several publications have been reported in the literature [5, 89–94]. Typically, a GMM model was learned from the distribution of image patches in the database. However, a really difficult obstacle that the researchers have to face is how to solve the optimization with the given GMM model under the assumption of Bayesian MAP. A conventional solution proposed in the existing works [5, 89–91, 93] is to select only one prominent Gaussian among M distribution of the GMM to represent the prior model of $p(\boldsymbol{\alpha}_i)$ of each noisy patch $\mathbf{y}_i$. Thus, the optimization turns into a convex quadratic problem and can be easily solved by a close-form formula. In [92], the author first divided the degraded image $\mathbf{y}$ into M groups of similar patches, and for each group, a Gaussian distribution was estimated to model the distribution of image patches.

To our knowledge, existing GMM-based denoising methods only used one prominent component of the GMM for representing the probability for each image patch, which leads to computationally efficient implementations. However, a questionable issue is using only one dominant component from M components of the GMM is enough to produce a good estimation of the latent image patch $\mathbf{x}_i$? In other words, may we improve the reconstruction performance when exploiting more Gaussian components to represent the prior distribution of latent patches? Additionally, what is the useful number of Gaussian components should be used for denoising an image patch $\mathbf{y}_i$? Unfortunately, the justification for these issues is lacking in the state-of-the-art.

The proposed denoising model in this work is the development of our framework introduced in chapter 5 by investigating the GMM for estimating distribution of representation coefficients of image patches in case of high-dimensional vector space Ω_α (which corresponds to a large dictionary $K \gg 3$). Optimization of denoising problem (6.4) with GMM prior is an important step in our proposed method. In this chapter, our motivation is to evaluate the useful number of Gaussian components using to recover

a noisy image patch. To this end, we divide the patches $\mathbf{y}_i$ in an input image into a set N_1 of simple patches with a prominent component and a set N_2 of the remaining patches. We focus on the set N_2 and conduct multiple experiments to show that only marginal gains can be obtained by considering the full GMM in denoising. We explore different choices of dictionary (identity matrix and K-SVD based) and two choices of GMM complexity on PSNR and reconstruction error and discuss the type of images that are difficult to reconstruct.

3 Datasets

We conduct the experiments on denoising on 8 different datasets with different image types and structures. Fig. 6.2 shows some example images of these datasets. For each dataset, we collect two samples of images, one is used for training the GMM models, and the other is adopted for the validation. We briefly present these datasets as follows:

Cartoon [154] contains 590 images of popular cartoon characters. We choose 45 images to train a GMM and 80 images for evaluation.

Urban [155] contains 100 images of urban scenes with high self-similarity and many repeated patterns. We use 25 images for training and 25 images for denoising.

Nature We use 200 training images in the Berkeley Segmentation datasets [6] to learn a GMM model and 20 popular natural test images shown in Fig. 2.1 for testing.

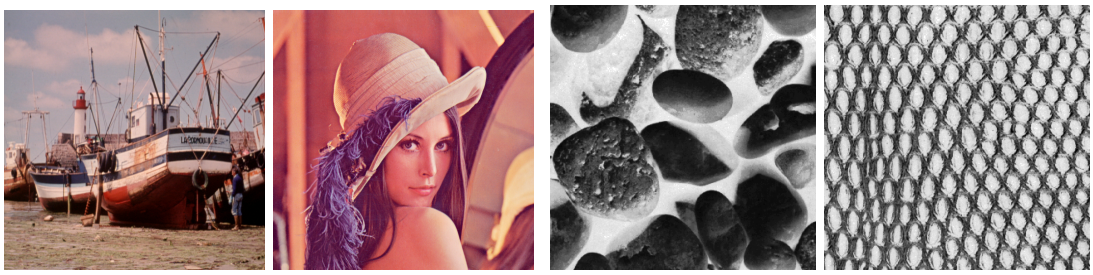
Brodatz [156] contains 112 grayscale images of natural textures. We select 30 good quality and content-rich images and split each of them into 4 non-overlapping sub-images. 90 sub-images are used for training the GMM and 30 sub-images for denoising validation.

Dtd [157] contains textural images in the wild such as band, braid, spiral, grid, etc. We choose 55 images for training and 40 images for denoising.

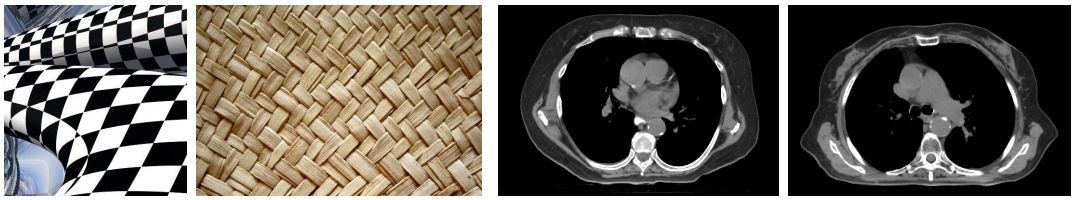
CT of Thorax and CT of Lung We download 7 sequences of CT lung images and 12 sequences of CT thorax images from [7]. 40 thorax images are used for training the GMM and 40 images for testing. The numbers of images for training and testing of CT images of Lung are 40 and 60.

MRI Brain We download 16 sequences of MRI brain images from [8]. 80 images are selected from 7 sequences for training and 60 images are chosen in other sequences for denoising.

(a) Cartoon **(b)** Urban



(c) Nature **(d)** Brodatz



(e) DTD **(f)** CT of Thorax

(g) CT of Lung **(h)** MRI of Brain

Figure 6.2: *Some images in 8 datasets.*

4 Image denoising with a Gaussian mixture model

In this section, we present the GMM-based image denoising method (referred to as GPID) that takes into account the GMM model learned from the representation coefficients of external patches as an image prior to regularize the denoising problem. We will demonstrate how to learn the GMM from a set of standard images $\{\mathbf{x}^s\}$ in section 4.1. After that, we introduce a framework that evaluates the number of Gaussian components used in solving the optimization problem in section 4.2.

4.1 Training the GMM on a patch database

From the training set of good quality noise-free images $\mathbf{x}^s$, we randomly extract P patches $\{\mathbf{x}_k^s \in R^n | k = 1, \ldots, P\}$ of size $\sqrt{n} \times \sqrt{n}$. After mean subtraction, each patch $\mathbf{x}_k^s$ is encoded in the vector space Ω_α, generated by K atoms of the dictionary $\mathbf{D}$, as in (5.4), with $\boldsymbol{\alpha}_k^s = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{x}_k^s$.

The probability distribution of representation coefficients vectors $p(\boldsymbol{\alpha}_k^s)$ of P patches in the database can be modeled by a GMM of M components as indicated in (6.1), with $p(\boldsymbol{\alpha}_k^s) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{\alpha}_k^s | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$. By assuming that each representation coefficients vector $\boldsymbol{\alpha}_k^s$ is independently sampled from the GMM distribution, the overall likelihood objective function is determined as

$$\mathcal{L}(\Theta) = \prod_{k=1}^{P} p(\boldsymbol{\alpha}_k^s) = \prod_{k=1}^{P} \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{\alpha}_k^s | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \tag{6.5}$$

Where $\Theta = \{\pi_m, \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m | m = 1 \ldots, M\}$ is the set of parameters of the GMM model, which can be found by maximizing the likelihood $\mathcal{L}$ (or log-likelihood $\ln(\mathcal{L})$ in equivalence).

$$\Theta = \arg\max_{\Theta} \{\ln \mathcal{L}(\Theta)\} = \arg\max_{\Theta} \left\{ \sum_{k=1}^{P} \ln \left(\sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{\alpha}_k^s | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \right) \right\} \tag{6.6}$$

The problem in (6.6) can be solved using the iterative Expectation-Maximization (EM) algorithm [158] via two alternative steps. In the E-step, with given values of Θ, a hidden variable γ_{km} (also called the "membership probability") which relates to the probability of $\boldsymbol{\alpha}_k^s$ belonging to the m-th ($1 \leq m \leq M$) Gaussian components is computed as in (6.7).

$$\gamma_{km} = \frac{\pi_m \mathcal{N}(\boldsymbol{\alpha}_k^s | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)}{\sum_{l=1}^{M} \pi_l \mathcal{N}(\boldsymbol{\alpha}_k^s | \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)} \tag{6.7}$$

In the M-step, the parameters Θ of the GMM are updated using (6.8) - (6.10)

$$\boldsymbol{\mu}_m = \frac{\sum_{k=1}^{P} \gamma_{km} \boldsymbol{\alpha}_k^s}{\sum_{k=1}^{P} \gamma_{km}} \tag{6.8}$$

$$\boldsymbol{\Sigma}_m = \frac{\sum_{k=1}^{P} \gamma_{km} (\boldsymbol{\alpha}_k^s - \boldsymbol{\mu}_m)^T (\boldsymbol{\alpha}_k^s - \boldsymbol{\mu}_m)}{\sum_{k=1}^{P} \gamma_{km}} \tag{6.9}$$

$$\pi_m = \frac{\sum_{k=1}^{P} \gamma_{km}}{P} \tag{6.10}$$

4.2 Denoising algorithm

As indicated in section 2, we can adopt the GMM model learned from distribution of external images as an image prior to characterize the probability distribution of representation coefficients $p(\boldsymbol{\alpha}_i)$ of a latent clean image $\mathbf{x}_i$. That means $p(\boldsymbol{\alpha}_i = \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{\alpha}_i | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m))$. After mean substraction, denoising a patch $\mathbf{y}_i$ is equivalent to finding the optimal representation coefficients vector $\hat{\boldsymbol{\alpha}}_i$ in (6.4) such that the clean latent patch can be estimated $\hat{\mathbf{x}}_i = \mathbf{D}\hat{\boldsymbol{\alpha}}_i$.

However, solving problem (6.4) with the whole GMM of $p(\boldsymbol{\alpha}_i)$ is a very time-consuming process. To overcome this issue, existing studies [5, 89, 90, 92, 93] proposed to assign the noisy patch $\mathbf{y}_i$ to a single Gaussian component. The selection can be done according to maximum of posterior probability γ_{im} that $\mathbf{y}_i$ belongs to the m-th Gaussian component. To determine the probability γ_{im}, we recall that the noise model is formulated on each image patch as $\mathbf{y}_i = \mathbf{D}\boldsymbol{\alpha}_i + \boldsymbol{\eta}_i$, with $\boldsymbol{\eta}_i \sim \mathcal{N}(0, \sigma^2)$. Thus if we suppose that $\boldsymbol{\alpha}_i$ is drawn from the m-th Gaussian distribution with mean $\boldsymbol{\mu}_m$ and covariance matrix $\boldsymbol{\Sigma}_m$, as well as $\boldsymbol{\alpha}_i$ and $\boldsymbol{\eta}_i$ are two independent variables, then $\mathbf{y}_i$ will belong to another Gaussian distribution with the corresponding mean $\mathbf{D}\boldsymbol{\mu}_m$ and covariance matrix $\mathbf{D}\boldsymbol{\Sigma}_m\mathbf{D}^T + \sigma^2\mathbf{I}$, with $\mathbf{I} \in R^{K \times K}$ is an identity matrix. Therefore, the posteriori probability γ_{im} is:

$$\gamma_{im} = p(\pi_m, \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m | \mathbf{y}_i) = \frac{\pi_m \mathcal{N}(\mathbf{y}_i | \mathbf{D}\boldsymbol{\mu}_m, \mathbf{D}\boldsymbol{\Sigma}_m\mathbf{D}^T + \sigma^2\mathbf{I})}{\sum_{l=1}^{M} \pi_l \mathcal{N}(\mathbf{y}_i | \mathbf{D}\boldsymbol{\mu}_l, \mathbf{D}\boldsymbol{\Sigma}_l\mathbf{D}^T + \sigma^2\mathbf{I})} \tag{6.11}$$

where $0 \leq \gamma_{im} \leq 1$ and $\sum_{m=1}^{M} \gamma_{im} = 1$. For convenience, we assume that $\gamma_{i1} \geq \gamma_{i2} \geq \ldots \geq \gamma_{iK}$. When only the first Gaussian component with maximum membership probability γ_{i1} is used to represent the probability distribution of $p(\boldsymbol{\alpha}_i)$, the problem (6.4) becomes:

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left\{ ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda(\boldsymbol{\alpha}_i - \boldsymbol{\mu}_1)^T \boldsymbol{\Sigma}_1^{-1} (\boldsymbol{\alpha}_i - \boldsymbol{\mu}_1) \right\} \tag{6.12}$$

(6.12) is a convex quadratic problem and has a closed form solution as:

$$\hat{\boldsymbol{\alpha}}_i = \left(\mathbf{D}^T\mathbf{D} + \lambda\boldsymbol{\Sigma}_1^{-1} \right)^{-1} \left(\mathbf{D}^T\mathbf{y}_i + \lambda\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 \right) \tag{6.13}$$

Typically, this approach is acceptable when the first component is dominant and the other components do not contribute much to the optimization. In practice, we define the set of dominant patches $N_1 = \{\mathbf{y}_i | \gamma_{i1} \geq 0.9\}$ and we call N_2 the set of the remaining patches. N_1 patches are restored via (6.13) whereas the patches in N_2 are restored by considering the first L components of the GMM with largest posteriori probabilities corresponding to $\{\gamma_{i1}, \gamma_{i2}, \ldots, \gamma_{iL}\}$. Consequently, for all N_2 patches in the noisy image,

we consider the simplification of problem 6.4, as described in (6.14), which can be solved using a gradient descent algorithm. The denoising method is presented in Algorithm 6.1.

$$\boldsymbol{\alpha}_i = \arg\min_{\boldsymbol{\alpha}_i} ||\mathbf{y}_i - \mathbf{D}\boldsymbol{\alpha}_i||_2^2 - \lambda \log \left\{ \sum_{l=1}^{L} \pi_l \mathcal{N}(\boldsymbol{\alpha}_i | \boldsymbol{\mu}_l, \Sigma_l) \right\} \tag{6.14}$$

Algorithm 6.1: GMM as Prior for Image Denoising (GPID)

Initialization: $\mathbf{X}^{(0)} = \mathbf{Y}$.
for $t = 1$ *to* T **do**
  for *each patch* $\mathbf{y}_i \in \mathbf{Y}$ **do**
    - Subtract its mean value (μ_y): $\mathbf{y}_i = \mathbf{y}_i - \mu_y$.
    - Calculate γ_{im} $(1 \le m \le M)$ via (6.11) and arrange in descending order.
    - **If** $\gamma_{i1} \ge 0.9$ **then** $\hat{\boldsymbol{\alpha}}_i = \left(\mathbf{D}^T\mathbf{D} + \lambda\Sigma_1^{-1}\right)^{-1}\left(\mathbf{D}^T\mathbf{y}_i + \lambda\Sigma_1^{-1}\boldsymbol{\mu}_1\right)$
    - **Else** select L components with largest value of γ_{im}, **then** solve for $\hat{\boldsymbol{\alpha}}_i$ in (6.14) using gradient descent.
    - Estimate the latent clean patch: $\mathbf{x}_i = \mathbf{D}\hat{\boldsymbol{\alpha}}_i + \mu_y$.
  end
  Aggregate the denoised patches $\mathbf{x}_i$ to recover the entire denoised image $\mathbf{X}^t$
  Regularize the denoised image: $\mathbf{X}^{(t)} = (\eta\mathbf{Y}^{(t-1)} + \beta\mathbf{X}^t)/(\eta + \beta)$.
end

4.3 Complexity Analysis

The denoising method GPID consists of two parts: off-line training and denoising.

In the training phase, a GMM of M components are learned from a set of P representation coefficients vector $\{\boldsymbol{\alpha}_k^s | \boldsymbol{\alpha}_k^s \in R^K, k = 1, \ldots, P\}$. In the E-step, computing the "membership probability" in (6.7) requires $O(K^3PM)$ operators. In the M-step, the total complexity of updating the parameters of the GMM is $O(K^2PM)$. Therefore, the overall complexity of training the GMM is $O(K^3PM)$.

In the denoising process, N_1 patches in the noisy image are restored via (6.13) that needs $O(K^2N_1)$ operations, and N_2 patches ($N_2 \approx 10\%N - 20\%N$) are recovered using gradient descent with the complexity $O(LT_{gd}K^3N_2)$, where T_{gd} is the number of iterations of the gradient descent algorithm. The computation of the membership probabilities requires $O(n^3NM)$ operations. The denoising step is repeated T times and therefore totally takes $O(n^3NMT + K^2N_1T + LT_{gd}K^3N_2T)$ complexity.

5 Experimental results

To show the effect of the restriction to the dominant component, we examine the performance of the GPID method on N_2 patches with a varying number of components $L \in \{1,5,10,15,20\}$ in step 7 of the optimization algorithm. We study the differences in peak signal-to-noise ratio (PSNR) and mean gray-level reconstruction error for the 8 datasets presented in section 3, for the identity dictionary $\mathbf{D} = \mathbf{I}$ and a K-SVD dictionary, and for small ($M = 20$) and large ($M = 200$) numbers of Gaussian components.

In all experiments, we degrade the images from the database with white Gaussian noise with standard deviation $\sigma = 30$. We train the two GMMs for each dataset on $P = 2.10^6$ randomly extracted patches of size $n = 8 \times 8$. In the GPID denoising method, we use $T = 5$ regularizing iterations with $\eta = n/\sigma^2$, $\beta = [1,4,8,16,32]/\sigma^2$. We set $\lambda = 0.9\sigma^2$ and 1000 maximum iterations in gradient descent optimization. All the experiments are implemented in the Matlab 2013a environment on a machine with Intel Core i7-4770K CPU of 3.5 GHz and 16 GB of RAM.

From the examples in Figure 6.3, we notice that N_2 patches can usually be found close to the edges or contours. We also compute the PSNR values obtained for the GPID method as a function of L. On these examples, only modest gains can be obtained by considering several components in the reconstruction. These properties are explored further by computing the distributions of PSNR gains and reconstruction error.

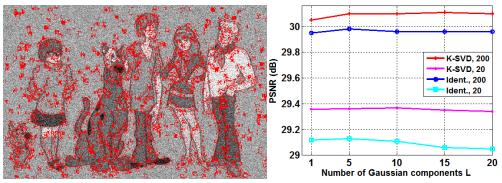
5.1 Denoising performance

When using the identity matrix as a dictionary, image patches are denoised without transformation. Note that the GPID method coincides with the method EPLL proposed by Zoran and Weiss in [5] when $L = 1$. We first observe that most image patches correspond to a single dominant component from Fig 6.4. As expected, more complex images such as textures (Brodatz and Dtd datasets) require more components and have more N_2 patches.

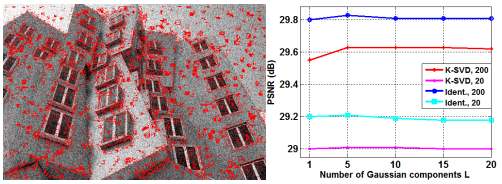
We compute the PSNR for N_2 patches for $L \in \{1,5,10,15,20\}$ and study the distribution of the maximum improvement $(\max_L PSNR) - PSNR_{L=1}$. As shown on Figure 6.4, for five datasets (Cartoon, Urban, Nature, CT Thorax and MRI), the maximum improvement is negligible, less than 0.1dB for all test images except one from Nature. For complex images such as textures (Brodatz and DTD datasets) and the CT Lung images, some images can be modestly improved, up to 0.2dB for $M = 20$. PSNR gains larger than 0.2dB are only observed for complex images (Dtd and CT Lung), with

$M = 200$, i.e. with enough components in the GMM to model distribution details and only for a small fraction of images (around 20%). These gains are require around 200 seconds for a 256×256 image, whereas only 10 seconds are needed for one Gaussian component.

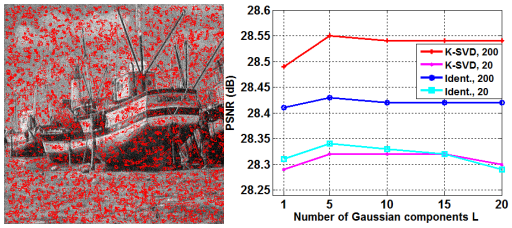
We also analyze the reconstruction error on the central pixel of N_2 patches $||\hat{\mathbf{X}}_{L=1} - \hat{\mathbf{X}}_{L=5}||_{\ell_1}$ on Fig. 6.4. For all images, this difference is less than 2 gray levels per pixel, and cannot be seen by eye.



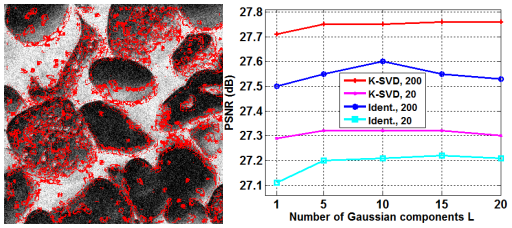
(a) Sas34 (Cartoon), $p = 83.9$, $e = 1.2$



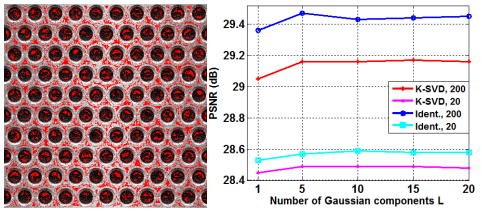
(b) Img75 (Urban), $p = 86.6$, $e = 1.2$



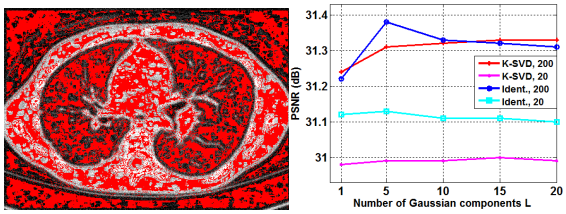
(c) Boat (Nature), $p = 77.1$, $e = 1.1$



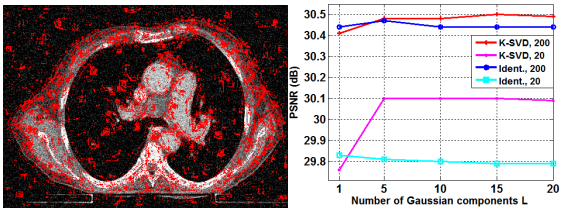
(d) D31 (Brodatz), $p = 79.6$, $e = 1.2$.



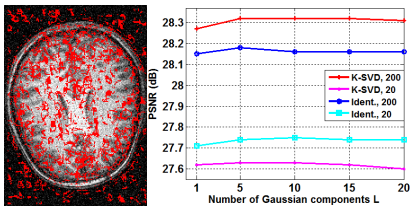
(e) Perforated_0012 (DTD), $p = 79.5$, $e = 1.2$



(f) CT of Lung, $p = 52.7$, $e = 1.2$



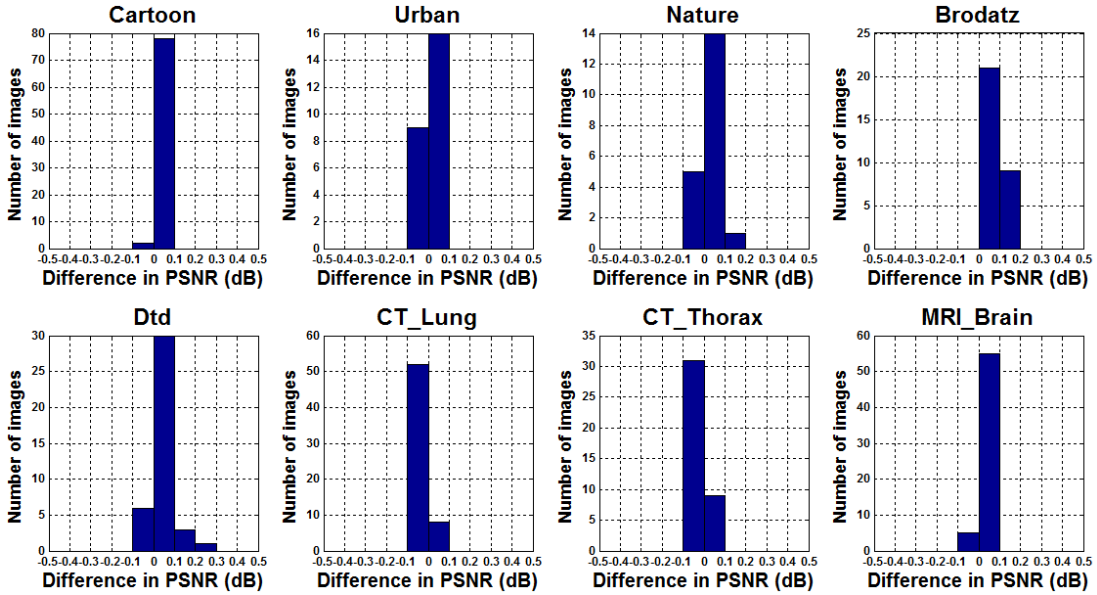
(g) CT of Thorax, $p = 83$, $e = 1.1$



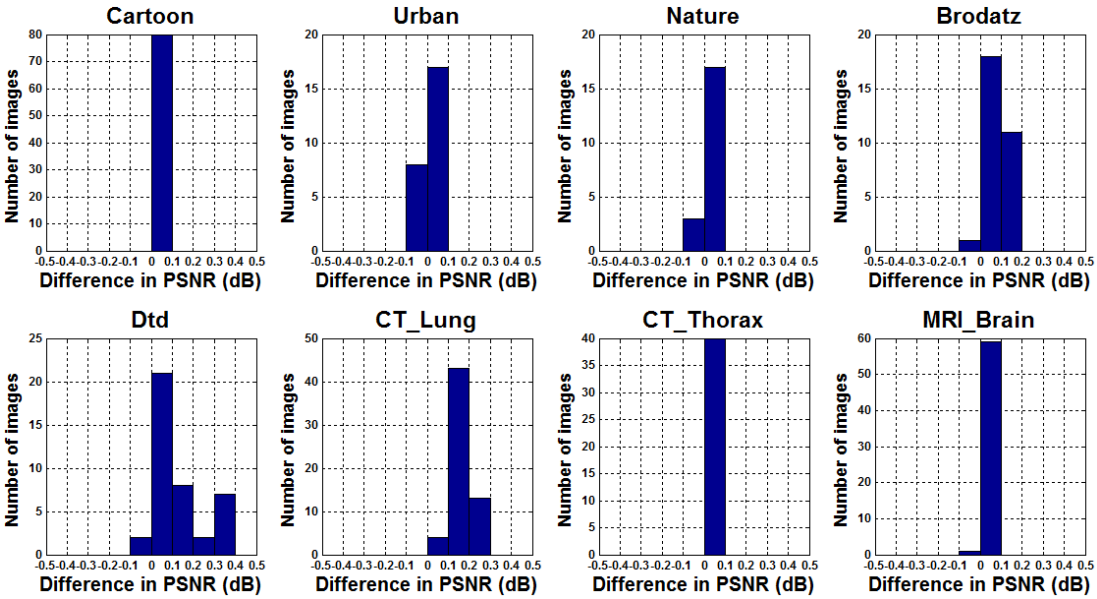
(h) MRI of Brain, $p = 79.4$, $e = 1.2$

Figure 6.3: *Examples from the 8 datasets, test image with N_2 patches in red (left), PSNR as a function of L (right). Captions indicate image name, percentage of N_1 patches and average reconstruction error $||\hat{\mathbf{X}}_{L=1} - \hat{\mathbf{X}}_{L=5}||_{\ell_1}$.*

For each dataset, we learn an over-complete dictionary $\mathbf{D}$ with $K = 256$ atoms as in [2]. Figure 6.5 shows a similar situation as Figure 6.4. Most patches in the test

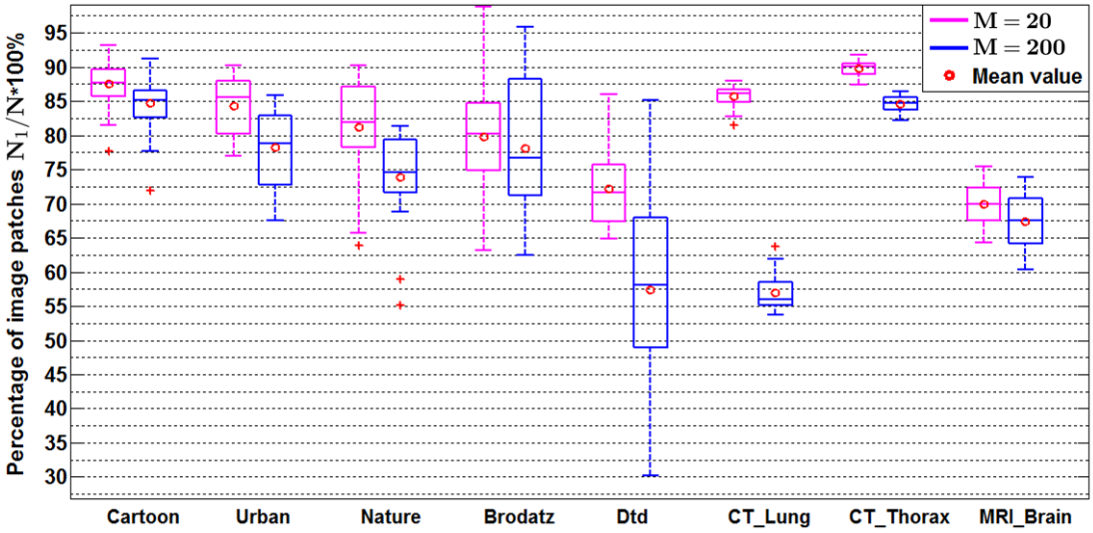


(a) Maximum PSNR improvement for small GMM model with $M = 20$

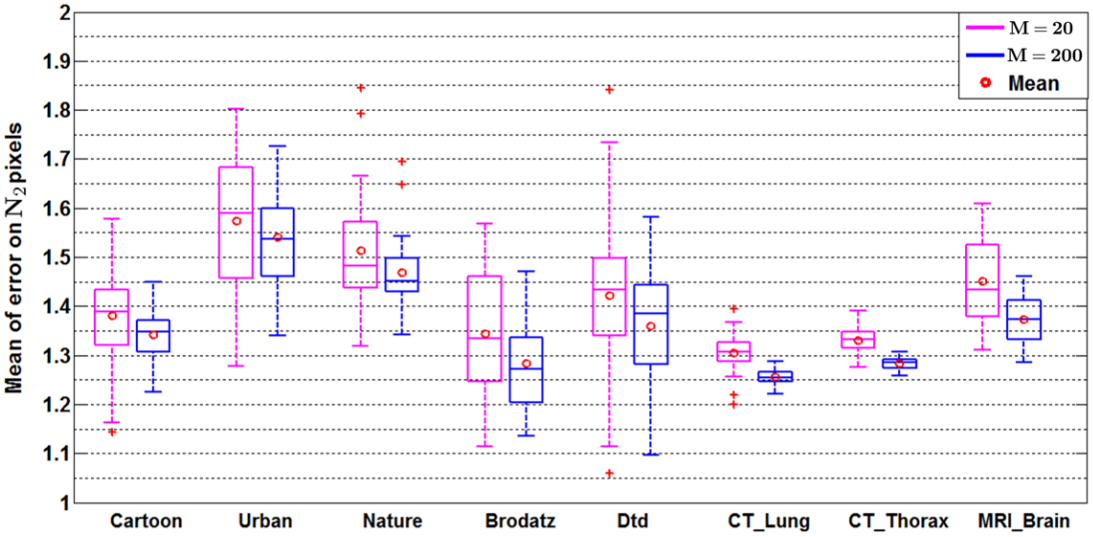


(b) Maximum PSNR improvement for large GMM model with $M = 200$

Figure 6.4: *Denoising performance for the identity dictionary*



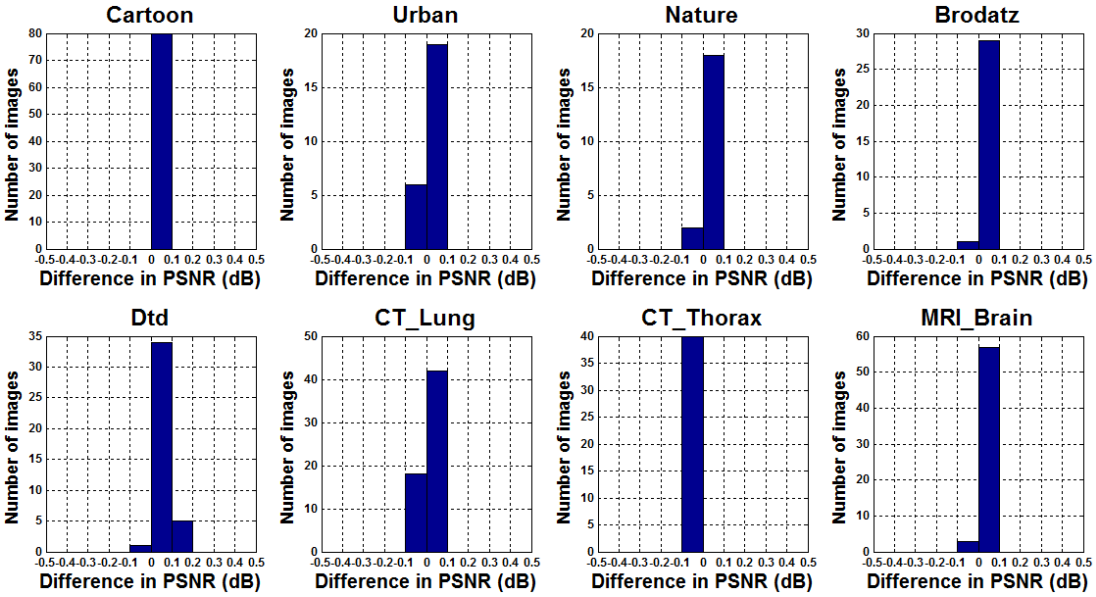
(c) Percentage of N_1 pixels in images of 8 datasets



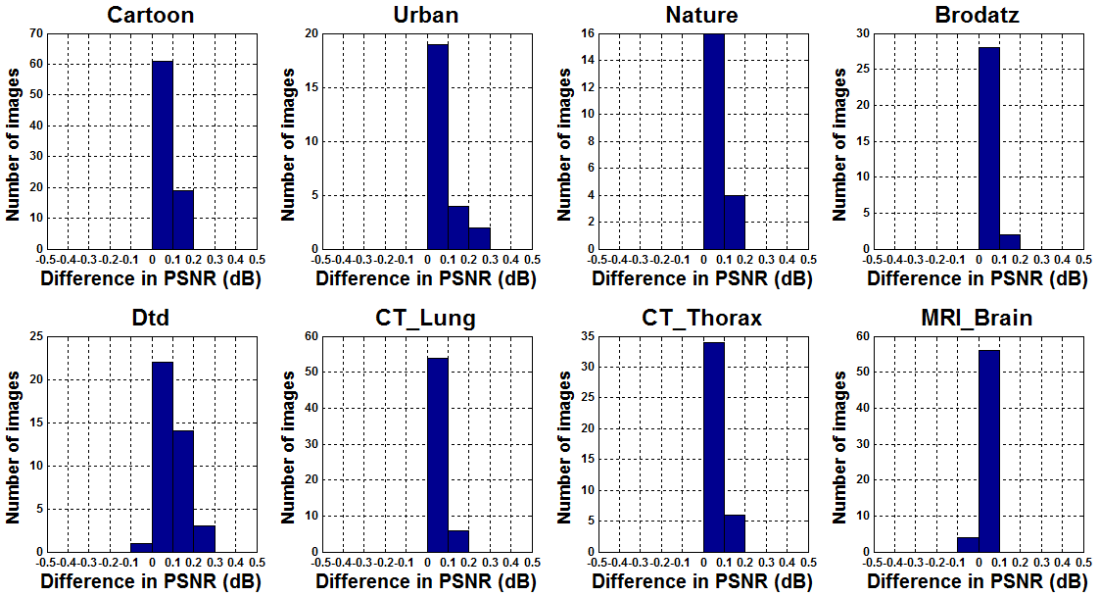
(d) Average reconstruction error $||\hat{\mathbf{X}}_{L=1} - \hat{\mathbf{X}}_{L=5}||_{L_1}$

Figure 6.4: *Denoising performance for the identity dictionary*

images belong to N_1, and most images can be reconstructed with only one component with a penalty less than 0.1dB. PSNR gains larger than 0.2dB can only be observed for a few complex images in the Dtd and Urban datasets. Gray-level differences are lower than for the identity dictionary, around 1.1 gray-levels per pixel.

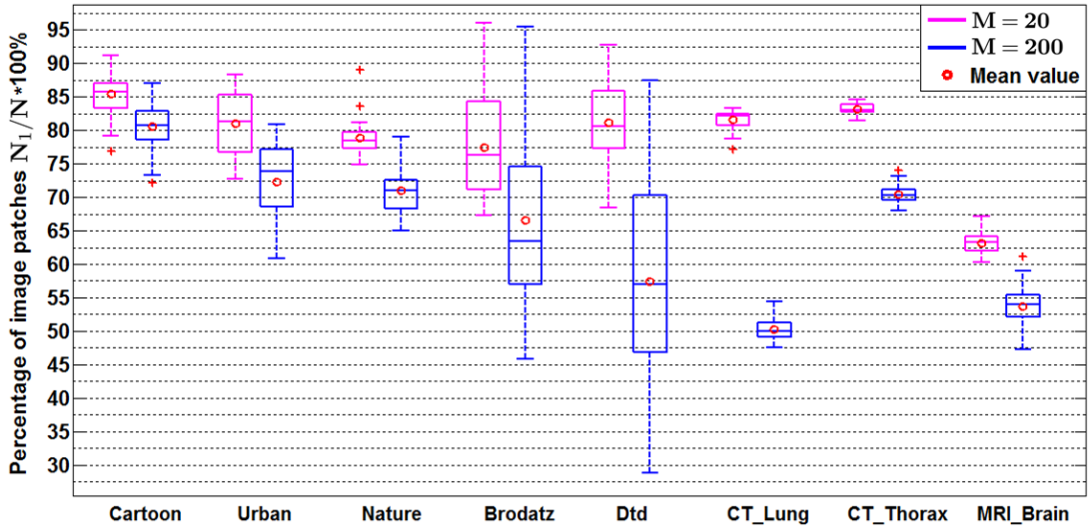


(a) Maximum PSNR improvement for small GMM model with $M = 20$

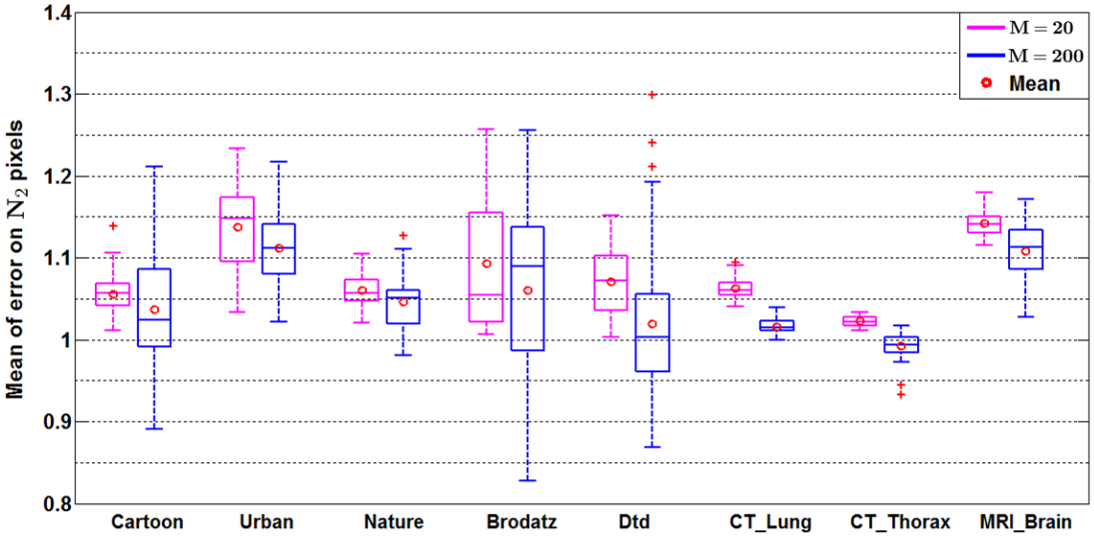


(b) Maximum PSNR improvement for large GMM model with $M = 200$

Figure 6.5: *Denoising performance for the K-SVD dictionary with 256 atoms*



(c) Percentage of N_1 pixels in images of 8 datasets

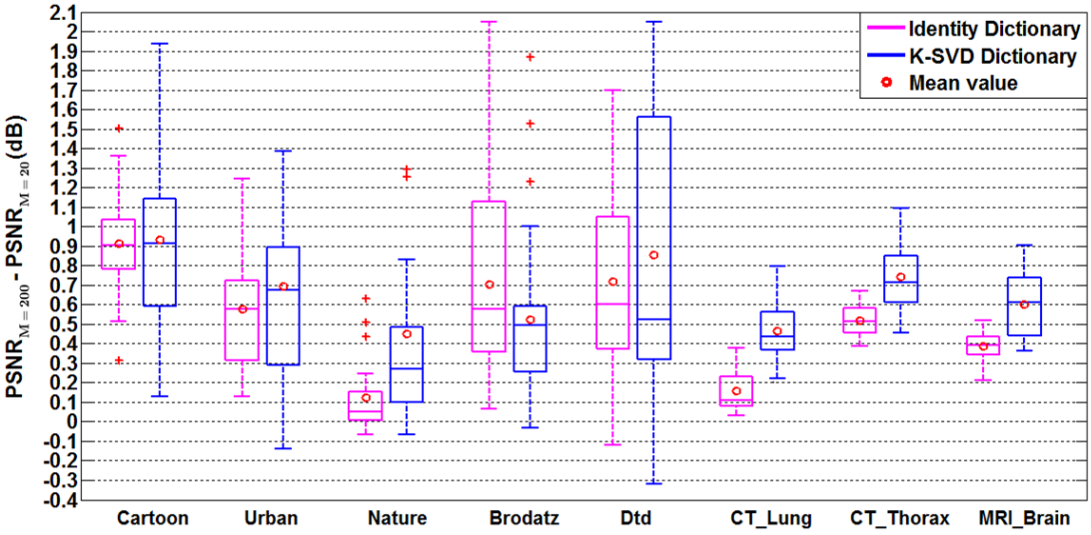


(d) Average reconstruction error $||\hat{\mathbf{X}}_{L=1} - \hat{\mathbf{X}}_{L=5}||_{L_1}$

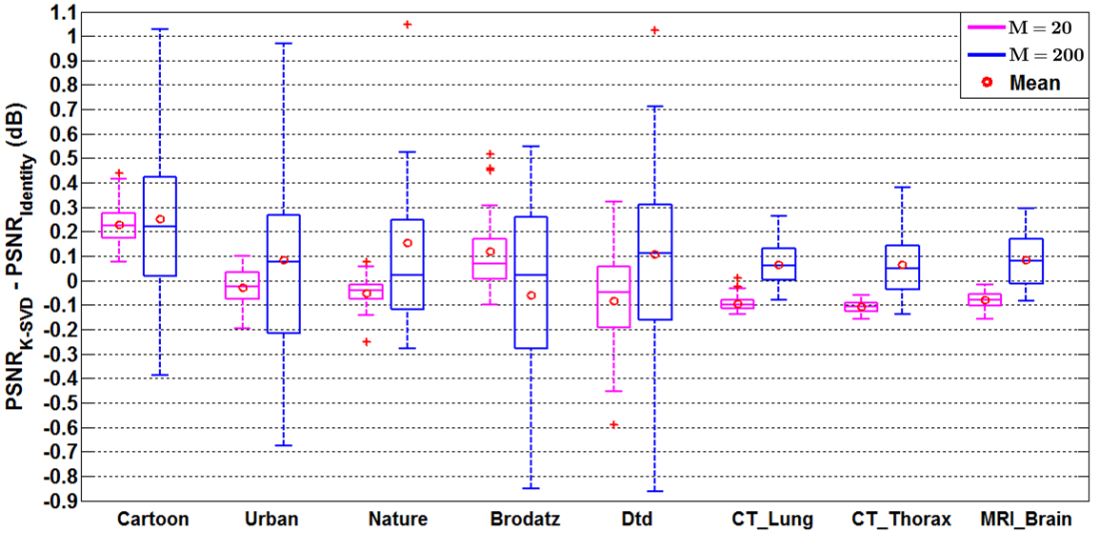
Figure 6.5: *Denoising performance for the K-SVD dictionary with 256 atoms*

5.2 Dictionary choice and model complexity

Using the denoising results from the 8 datasets with $L = 1$, we compare the two dictionaries and GMM sizes in Figure 6.6 (see also the examples in Figure 6.3). We observe that increasing the GMM model complexity is nearly always beneficial, sometimes up to 2dB PSNR gains, and that the K-SVD dictionary tends to benefit more from $M = 200$. The K-SVD dictionary yields slightly better PSNR especially for large GMM models overall, but the results are variable, which implies that dictionary choice is largely image-specific.



(a)



(b)

Figure 6.6: *Effect of model complexity (a) and dictionary choice (b) on the denoising performance for $L = 1$*

6 Conclusion

In a high-dimensional vector space generated by a large given dictionary, we can learn a GMM to approximately represent the distribution of patches in the database and then apply it as a prior model to denoise a degraded image.

However, rather than focus on comparing the performance of the proposed method with other arbitrarily chosen sparse models, our main objective, in this chapter, is to investigate the use of GMM model in solving the optimization problem, due to the lack of justification in the literature. By studying the number of useful components in the GMM for patch-based image denoising on 8 image datasets, we first remark that most of the patches in an input image are well represented by a single prominent component. We have explored denoising with increasing number of components $L \in \{1, 5, 10, 15, 20\}$, and shown that only modest gains can be obtained in terms of PSNR and ℓ_1 reconstruction error (gray-level differences) in all datasets when using more than one component. This verifies the current practice and drastically reduces computational cost. We also demonstrated that much larger improvements can be obtained with a suitable dictionary and GMM model, but reconstruction only requires a single component.

Chapter 7

Perspective and future work

In this thesis, we study patch-based methods for image restoration in the Bayesian framework. Constructing the dictionary used to represent image patches and the prior distribution in dictionary space is a challenging problem in this setting. We have proposed several contributions, in the case of exhaustive dictionaries in Chapters 3 and 4, and in the estimation of the patch distribution in Chapters 5 and 6.

In particular, we have shown that careful selection of the local dictionary improves image denoising and super-resolution in Chapters 3 and 4. The main ingredient was to find matches in the patch database by selecting an appropriate patch distance such as the EMD. In Chapters 5 and 6, we improve image denoising by selecting a better patch prior, first with a computationally efficient procedure for low dimension dictionaries and then with a Gaussian Mixture Model for higher complexity models. All of this points towards the need to adapt the elements of a patch-based Bayesian method to the current restoration problem.

The results presented in this thesis raise a few questions that we would like to address in the future.

Dictionary construction. We have used several possible choices for dictionary construction (filtered exhaustive dictionaries, identity matrix, K-SVD), but we did not explore the properties that a good dictionary should have. This raises the question of the representation space of image patches, of its dimension, of its redundancy, etc.

Prior construction. Our intuition suggests that the prior should be constructed from standard images related to the input image, but it is unclear how specific this needs to be. Current approaches in deep learning and transfer learning suggest that images share common properties and that these are reflected in the prior distribution. In the near future, we would like to compare the dictionaries constructed for different types of images as in Chapter 6 to evaluate if there exists a universal prior or which kind of modifications are required to adapt to different contexts.

Locality. In Chapters 3 and 4, we used filtering to construct locally-adapted sub-dictionaries. The results in Chapter 6 also suggest that the prior distribution is locally simple. Following the insight from Nonlocal Self Similarity methods, would the input image itself provide enough information for estimating a local patch prior? Can this information be combined with a generic prior estimated from an image database?

Bibliography

[1] D. H. Trinh et al. “Novel Example-Based Method for Super-Resolution and Denoising of Medical Images”. In: *IEEE Transactions on Image Processing* 23.4 (Apr. 2014), pp. 1882–1895 *Cited on pages xiv, 3, 4, 7, 21, 22, 25, 26, 28–30, 49, 50, 64, 66, 69, 70, 81.*

[2] M. Aharon, M. Elad, and A. Bruckstein. “K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation”. In: *IEEE Transactions on Signal Processing* 54.11 (Nov. 2006), pp. 4311–4322 *Cited on pages 3, 8, 20, 28, 35, 36, 39, 40, 80, 81, 103, 121.*

[3] J. Yang et al. “Image Super-Resolution Via Sparse Representation”. In: *IEEE Transactions on Image Processing* 19.11 (Nov. 2010), pp. 2861–2873 *Cited on pages 3, 21, 28, 32, 33, 39, 52, 70, 80.*

[4] W. Dong et al. “Nonlocally Centralized Sparse Representation for Image Restoration”. In: *IEEE Transactions on Image Processing* 22.4 (Apr. 2013), pp. 1620–1630 *Cited on pages 3, 22, 25, 39.*

[5] D. Zoran and Y. Weiss. “From learning models of natural image patches to whole image restoration”. In: *2011 International Conference on Computer Vision*. Nov. 2011, pp. 479–486 *Cited on pages 3, 24, 28, 33, 34, 111, 114, 118, 120.*

[6] P Arbeláez et al. “Contour Detection and Hierarchical Image Segmentation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.5 (May 2011), pp. 898–916 *Cited on pages 4, 112, 115.*

[7] Kenneth Clark et al. “The Cancer Imaging Archive (TCIA): Maintaining and Operating a Public Information Repository”. In: *Journal of Digital Imaging* 26.6 (July 2013), pp. 1045–1057 *Cited on pages 4, 115.*

[8] A Di Martino et al. “The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism”. In: *Molecular Psychiatry* 19.6 (June 2013), pp. 659–667 *Cited on pages 4, 115.*

[9] Z. Wang et al. “Image Quality Assessment: From Error Visibility to Structural Similarity”. In: *IEEE Transactions on Image Processing* 13.4 (Apr. 2004), pp. 600–612 *Cited on pages 10, 11.*

[10] Linda G. Shapiro and George C. Stockman. *Computer Vision*. Pearson, 2001 *Cited on page 15.*

[11] R. Keys. “Cubic convolution interpolation for digital image processing”. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 29.6 (Dec. 1981), pp. 1153–1160 *Cited on page 15.*

[12] H. Hou and H. Andrews. “Cubic splines for image interpolation and digital filtering”. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26.6 (Dec. 1978), pp. 508–517 *Cited on page 15.*

[13] C. Tomasi and R. Manduchi. “Bilateral filtering for gray and color images”. In: *Sixth International Conference on Computer Vision*. Jan. 1998, pp. 839–846 *Cited on page 15.*

[14] P. Perona and J. Malik. “Scale-space and edge detection using anisotropic diffusion”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12.7 (July 1990), pp. 629–639 *Cited on page 15.*

[15] Stephen M. Smith and J. Michael Brady. “SUSAN-A New Approach to Low Level Image Processing”. In: *Int. J. Comput. Vision* 23.1 (May 1997), pp. 45–78 *Cited on page 15.*

[16] S. Haykin and B. Widrow. *Least-mean-square adaptive filters*. Hoboken, N.J.: Wiley-Interscience, 2003 *Cited on page 15.*

[17] H. Takeda, S. Farsiu, and P. Milanfar. “Kernel Regression for Image Processing and Reconstruction”. In: *IEEE Transactions on Image Processing* 16.2 (Feb. 2007), pp. 349–366 *Cited on page 15.*

[18] X. Li and M. T. Orchard. “New edge-directed interpolation”. In: *IEEE Transactions on Image Processing* 10.10 (Oct. 2001), pp. 1521–1527 *Cited on page 15.*

[19] Lei Zhang and Xiaolin Wu. “An edge-guided image interpolation algorithm via directional filtering and data fusion”. In: *IEEE Transactions on Image Processing* 15.8 (Aug. 2006), pp. 2226–2238 *Cited on page 15.*

[20] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. “Nonlinear Total Variation Based Noise Removal Algorithms”. In: *Phys. D* 60 (Nov. 1992), pp. 259–268 *Cited on pages 16, 17.*

[21] David L. Donoho and Iain M. Johnstone. “Ideal spatial adaptation by wavelet shrinkage”. In: *Biometrika* 81 (1994), pp. 425–455 *Cited on pages 16, 18, 19.*

[22] S. G. Chang, Bin Yu, and M. Vetterli. “Adaptive wavelet thresholding for image denoising and compression”. In: *IEEE Transactions on Image Processing* 9.9 (Sept. 2000), pp. 1532–1546 *Cited on pages 16, 19.*

[23] J. Portilla et al. “Image denoising using scale mixtures of Gaussians in the wavelet domain”. In: *IEEE Transactions on Image Processing* 12.11 (Nov. 2003), pp. 1338–1351 *Cited on pages 16, 18, 19, 24.*

[24] L. Sendur and I. W. Selesnick. “Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency”. In: *IEEE Transactions on Signal Processing* 50.11 (Nov. 2002), pp. 2744–2756 *Cited on pages 16, 18, 19.*

[25] Antonin Chambolle. “An Algorithm for Total Variation Minimization and Applications”. In: *J. Math. Imaging Vis.* 20 (Jan. 2004), pp. 89–97 *Cited on page 17.*

[26] Michael K. Ng et al. “On Semismooth Newton’s Methods for Total Variation Minimization”. In: *Journal of Mathematical Imaging and Vision* 27.3 (Mar. 2007), pp. 265–276 *Cited on page 17.*

[27] Michael K. Ng, Pierre Weiss, and Xiaoming Yuan. “Solving Constrained Total-variation Image Restoration and Reconstruction Problems via Alternating Direction Methods”. In: *SIAM Journal on Scientific Computing* 32.5 (Jan. 2010), pp. 2710–2736 *Cited on page 17.*

[28] Pascal Getreuer. “Rudin-Osher-Fatemi Total Variation Denoising using Split Bregman”. In: *Image Processing On Line* 2 (May 2012), pp. 74–95 *Cited on page 17.*

[29] O. Scherzer. “Denoising with higher order derivatives of bounded variation and an application to parameter estimation”. In: *Computing* 60.1 (Mar. 1998), pp. 1–27 *Cited on page 17.*

[30] Tony Chan, Antonio Marquina, and Pep Mulet. “High-Order Total Variation-Based Image Restoration”. In: *SIAM Journal on Scientific Computing* 22.2 (2000), pp. 503–516 *Cited on pages 17, 18.*

[31] K. Bredies, K. Kunisch, and T. Pock. “Total Generalized Variation”. In: *SIAM Journal on Imaging Sciences* 3.3 (2010), pp. 492–526 *Cited on page 17.*

[32] F. Knoll et al. “Second Order Total Generalized Variation (TGV) for MRI”. In: *Magnetic Resonance in Medicine* 65.2 (2011), pp. 480–491 *Cited on page 17.*

[33] İlker Bayram and M. E. Kamasak. "Directional Total Variation". In: *IEEE Signal Processing Letters* 19.12 (Dec. 2012), pp. 781–784 *Cited on page 17.*

[34] Qiang Wang et al. "Single-image super-resolution using directional total variation regularization and alternating direction method of multiplier solver". In: *Journal of Electronic Imaging* 24.2 (Apr. 2015), p. 023026 *Cited on page 17.*

[35] Lin Li et al. "Single image super-resolution using combined total variation regularization by split Bregman Iteration". In: *Neurocomputing* 142 (Oct. 2014), pp. 551–560 *Cited on page 18.*

[36] Chao Ren, Xiaohai He, and Truong Nguyen. "Single Image Super-Resolution via Adaptive High-Dimensional Non-Local Total Variation and Adaptive Geometric Feature". In: *IEEE Transactions on Image Processing* (2016), pp. 90–106 *Cited on page 18.*

[37] D. L. Donoho. "De-noising by Soft-thresholding". In: *IEEE Trans. Inf. Theor.* 41.3 (May 1995), pp. 613–627 *Cited on page 19.*

[38] Raymond H. Chan et al. "Wavelet Algorithms for High-Resolution Image Reconstruction". In: *SIAM Journal on Scientific Computing* 24.4 (Jan. 2003), pp. 1408–1432 *Cited on page 19.*

[39] Hui Ji and C. Fermuller. "Robust Wavelet-Based Super-Resolution Reconstruction: Theory and Algorithm". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.4 (Apr. 2009), pp. 649–660 *Cited on page 19.*

[40] Do Kyung Shin and Young Shik Moon. "Super-Resolution Image Reconstruction Using Wavelet Based Patch and Discrete Wavelet Transform". In: *Journal of Signal Processing Systems* 81.1 (Aug. 2014), pp. 71–81 *Cited on page 19.*

[41] Selen Ayas and Murat Ekinci. "Single image super resolution based on sparse representation using discrete wavelet transform". In: *Multimedia Tools and Applications* (Oct. 2017) *Cited on page 19.*

[42] Zongqing Lu et al. "Wavelet-based single image super-resolution with an overall enhancement procedure". In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Mar. 2017 *Cited on page 19.*

[43] S. G. Mallat and Zhifeng Zhang. "Matching pursuits with time-frequency dictionaries". In: *IEEE Transactions on Signal Processing* 41.12 (Dec. 1993), pp. 3397–3415 *Cited on pages 20, 80, 81.*

[44] M. Elad and M. Aharon. "Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries". In: *IEEE Transactions on Image Processing* 15.12 (Dec. 2006), pp. 3736–3745 *Cited on pages 20, 38, 80, 81.*

[45] Roman Zeyde, Michael Elad, and Matan Protter. "On Single Image Scale-Up Using Sparse-Representations". In: *Curves and Surfaces*. 2012, pp. 711–730 *Cited on pages 20, 39, 80.*

[46] Junfeng Yang and Yin Zhang. "Alternating Direction Algorithms for ℓ_1-Problems in Compressive Sensing". In: *SIAM Journal on Scientific Computing* 33.1 (2011), pp. 250–278 *Cited on page 21.*

[47] Shenlong Wang et al. "Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. June 2012, pp. 2216–2223 *Cited on pages 21, 39.*

[48] A. Y. Yang et al. "Fast ℓ_1-Minimization Algorithms for Robust Face Recognition". In: *IEEE Transactions on Image Processing* 22.8 (Aug. 2013), pp. 3234–3246 *Cited on page 21.*

[49] De-An Huang and Yu-Chiang Frank Wang. "Coupled Dictionary and Feature Space Learning with Applications to Cross-Domain Image Synthesis and Recognition". In: *2013 IEEE International Conference on Computer Vision*. IEEE, Dec. 2013, pp. 2496–2503 *Cited on page 21.*

[50] D. H. Trinh et al. "An effective example-based learning method for denoising of medical images corrupted by heavy Gaussian noise and poisson noise". In: *2014 IEEE International Conference on Image Processing*. Oct. 2014, pp. 823–827 *Cited on pages 21, 26, 51.*

[51] Z. Xu et al. "$L_{1/2}$ Regularization: A Thresholding Representation Theory and a Fast Solver". In: *IEEE Transactions on Neural Networks and Learning Systems* 23.7 (July 2012), pp. 1013–1027 *Cited on pages 21, 81.*

[52] A. Rakotomamonjy et al. "$\ell_p - \ell_q$ Penalty for Sparse Linear and Sparse Multiple Kernel Multitask Learning". In: *IEEE Transactions on Neural Networks* 22.8 (Aug. 2011), pp. 1307–1320 *Cited on page 21.*

[53] Qin Lyu et al. "A Comparison of Typical ℓ_p Minimization Algorithms". In: *Neurocomput.* 119 (Nov. 2013), pp. 413–424 *Cited on page 21.*

[54] S. Guo, Z. Wang, and Q. Ruan. "Enhancing sparsity via ℓ_p $(0 < p < 1)$ minimization for robust face recognition". In: *Neurocomputing* 99 (2013), pp. 592–602 *Cited on page 21.*

[55] F. Jia et al. "Image denoising using hyper-Laplacian priors and gradient histogram preservation model". In: *2014 12th International Conference on Signal Processing*. Oct. 2014, pp. 811–815 *Cited on pages 21, 81.*

[56] F. Cao et al. “Image Super-Resolution via Adaptive ℓ_p $(0 < p < 1)$ Regularization and Sparse Representation”. In: *IEEE Transactions on Neural Networks and Learning Systems* 27.7 (July 2016), pp. 1550–1561 *Cited on page 21.*

[57] J. Kong et al. “Generalized ℓ_p-regularized representation for visual tracking”. In: *Neurocomputing* 213 (2016), pp. 155–161 *Cited on page 21.*

[58] Lei Zhang, Meng Yang, and Xiangchu Feng. “Sparse representation or collaborative representation: Which helps face recognition?” In: *2011 International Conference on Computer Vision*. IEEE, Nov. 2011, pp. 471–478 *Cited on page 21.*

[59] R. Timofte, V. De, and L. V. Gool. “Anchored Neighborhood Regression for Fast Example-Based Super-Resolution”. In: *IEEE International Conference on Computer Vision*. Dec. 2013, pp. 1920–1927 *Cited on page 21.*

[60] Radu Timofte, Vincent De Smet, and Luc Van Gool. “A+: Adjusted Anchored Neighborhood Regression for Fast Super-Resolution”. In: *Asian Conference on Computer Vision (ACCV)*. Nov. 2014, pp. 111–126 *Cited on page 21.*

[61] Yongbing Zhang et al. “CCR: Clustering and Collaborative Representation for Fast Single Image Super-Resolution”. In: *IEEE Transactions on Multimedia* 18.3 (Mar. 2016), pp. 405–417 *Cited on pages 21, 22, 25.*

[62] Wenming Yang et al. “Consistent Coding Scheme for Single-Image Super-Resolution Via Independent Dictionaries”. In: *IEEE Transactions on Multimedia* 18.3 (Mar. 2016), pp. 313–325 *Cited on pages 21, 22.*

[63] H. Zou and T. Hastie. “Regularization and variable selection via the Elastic Net”. In: *Journal of the Royal Statistical Society, Series B* 67 (2005), pp. 301–320 *Cited on pages 21, 93.*

[64] H. Zou, T. Hastie, and R. Tibshirani. “Sparse Principal Component Analysis”. In: *Journal of Computational and Graphical Statistics* 15 (2006), pp. 265–286 *Cited on page 21.*

[65] A. Buades, B. Coll, and J. M. Morel. “A non-local algorithm for image denoising”. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 2. June 2005, pp. 60–65 *Cited on pages 21, 92.*

[66] A. Buades, B. Coll, and J. M. Morel. “A Review of Image Denoising Algorithms, with a New One”. In: *Multiscale Modeling & Simulation* 4.2 (2005), pp. 490–530 *Cited on page 21.*

[67] C. Kervrann and J. Boulanger. “Local Adaptivity to Variable Smoothness for Exemplar-Based Image Regularization and Representation”. In: *International Journal of Computer Vision* 79.1 (Aug. 2008), pp. 45–69 *Cited on page 21.*

[68] C.-A. Deledalle, V. Duval, and J. Salmon. "Non-local Methods with Shape-Adaptive Patches (NLM-SAP)". In: *Journal of Mathematical Imaging and Vision* 43.2 (June 2012), pp. 103–120 *Cited on page 21.*

[69] P. Chatterjee and P. Milanfar. "Patch-Based Near-Optimal Image Denoising". In: *IEEE Transactions on Image Processing* 21.4 (Apr. 2012), pp. 1635–1649 *Cited on page 21.*

[70] X. Zhang, X. Feng, and W. Wang. "Two-Direction Nonlocal Model for Image Denoising". In: *IEEE Transactions on Image Processing* 22.1 (Jan. 2013), pp. 408–412 *Cited on page 21.*

[71] W. Dong et al. "Sparsity-based image denoising via dictionary learning and structural clustering". In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. June 2011, pp. 457–464 *Cited on pages 22, 25, 39.*

[72] W. Dong et al. "Image Deblurring and Super-Resolution by Adaptive Sparse Domain Selection and Adaptive Regularization". In: *IEEE Transactions on Image Processing* 20.7 (July 2011), pp. 1838–1857 *Cited on pages 22, 25, 39.*

[73] Kaibing Zhang et al. "Learning Multiple Linear Mappings for Efficient Single Image Super-Resolution". In: *IEEE Transactions on Image Processing* 24.3 (Mar. 2015), pp. 846–861 *Cited on pages 22, 25.*

[74] H. Chang, D.-Y. Yeung, and Y. Xiong. "Super-resolution through neighbor embedding". In: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2004)*. Vol. 1. June 2004, pp. 1275–1282 *Cited on pages 22, 28, 30, 50, 52, 64, 66, 70.*

[75] H. Ji et al. "Robust video denoising using low rank matrix completion". In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. June 2010, pp. 1791–1798 *Cited on page 23.*

[76] H. Ji et al. "Robust Video Restoration by Joint Sparse and Low Rank Matrix Approximation". In: *SIAM Journal on Imaging Sciences* 4.4 (2011), pp. 1122–1142 *Cited on page 23.*

[77] S. Gu et al. "Weighted Nuclear Norm Minimization with Application to Image Denoising". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. June 2014, pp. 2862–2869 *Cited on pages 23, 24.*

[78] Q. Guo et al. "An Efficient SVD-Based Method for Image Denoising". In: *IEEE Transactions on Circuits and Systems for Video Technology* 26.5 (May 2016), pp. 868–880 *Cited on page 23.*

[79] X. Jia, X. Feng, and W. Wang. “Rank Constrained Nuclear Norm Minimization with Application to Image Denoising”. In: *Signal Process.* 129 (Dec. 2016), pp. 1–11 *Cited on page 23.*

[80] Y. Xie et al. “Weighted Schatten p-Norm Minimization for Image Denoising and Background Subtraction”. In: *IEEE Transactions on Image Processing* 25.10 (Oct. 2016), pp. 4842–4857 *Cited on pages 23, 24.*

[81] S. Gu et al. “Weighted Nuclear Norm Minimization and Its Applications to Low Level Vision”. In: *Int. J. Comput. Vision* 121.2 (Jan. 2017), pp. 183–208 *Cited on pages 23, 24.*

[82] Z. Zha et al. “Analyzing the group sparsity based on the rank minimization methods”. In: *2017 IEEE International Conference on Multimedia and Expo (ICME)*. July 2017, pp. 883–888 *Cited on pages 23, 24.*

[83] J. Wright et al. “Robust Principal Component Analysis: Exact Recovery of Corrupted Low-Rank Matrices via Convex Optimization”. In: *Advances in Neural Information Processing Systems 22*. 2009, pp. 2080–2088 *Cited on page 23.*

[84] J.-F. Cai, E. J. Candès, and Z. Shen. “A Singular Value Thresholding Algorithm for Matrix Completion”. In: *SIAM Journal on Optimization* 20.4 (2010), pp. 1956–1982 *Cited on page 23.*

[85] Gilles Celeux and Gérard Govaert. “Gaussian parsimonious clustering models”. In: *Pattern Recognition* 28.5 (1995), pp. 781–793 *Cited on page 24.*

[86] H. Permuter, J. Francos, and I. H. Jermyn. “Gaussian mixture models of texture and colour for image database retrieval”. In: *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP ’03)*. Vol. 3. Apr. 2003, pp. 569–572 *Cited on page 24.*

[87] Haim Permuter, Joseph Francos, and Ian Jermyn. “A Study of Gaussian Mixture Models of Color and Texture Features for Image Classification and Segmentation”. In: *Pattern Recogn.* 39.4 (Apr. 2006), pp. 695–706 *Cited on page 24.*

[88] Zhaoxia Fu and Liming Wang. “Color Image Segmentation Using Gaussian Mixture Model and EM Algorithm”. In: *Multimedia and Signal Processing*. 2012, pp. 61–66 *Cited on page 24.*

[89] D. Zoran and Y. Weiss. “Natural Images, Gaussian Mixtures and Dead Leaves”. In: *Advances in Neural Information Processing Systems 25*. 2012, pp. 1736–1744 *Cited on pages 24, 111, 114, 118.*

[90] G. Yu, G. Sapiro, and S. Mallat. "Solving Inverse Problems With Piecewise Linear Estimators: From Gaussian Mixture Models to Structured Sparsity". In: *IEEE Transactions on Image Processing* 21.5 (May 2012), pp. 2481–2499 *Cited on pages 24, 111, 114, 118.*

[91] Yi-Qing Wang and Jean-Michel Morel. "SURE Guided Gaussian Mixture Image Denoising". In: *SIAM Journal on Imaging Sciences* 6.2 (2013), pp. 999–1034 *Cited on pages 24, 114.*

[92] M. Niknejad, H. Rabbani, and M. Babaie-Zadeh. "Image Restoration Using Gaussian Mixture Models With Spatially Constrained Patch Clustering". In: *IEEE Transactions on Image Processing* 24.11 (Nov. 2015), pp. 3624–3636 *Cited on pages 24, 111, 114, 118.*

[93] J. Xu et al. "Patch Group Based Nonlocal Self-Similarity Prior Learning for Image Denoising". In: *2015 IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015, pp. 244–252 *Cited on pages 24, 111, 114, 118.*

[94] X. Lu et al. "Image-Specific Prior Adaptation for Denoising". In: *IEEE Transactions on Image Processing* 24.12 (Dec. 2015), pp. 5469–5478 *Cited on pages 24, 25, 114.*

[95] E. Luo, S. H. Chan, and T. Q. Nguyen. "Adaptive Image Denoising by Mixture Adaptation". In: *IEEE Transactions on Image Processing* 25.10 (Oct. 2016), pp. 4489–4503 *Cited on pages 24, 25.*

[96] A. Teodoro, M. Almeida, and M. Figueiredo. "Single-frame Image Denoising and Inpainting Using Gaussian Mixtures". In: *Proceedings of the International Conference on Pattern Recognition Applications and Methods - Volume 2*. ICPRAM 2015. 2015, pp. 283–288 *Cited on page 24.*

[97] A. M. Teodoro, J. M. Bioucas-Dias, and M. A. T. Figueiredo. "Image restoration with locally selected class-adapted models". In: *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*. Sept. 2016, pp. 1–6 *Cited on pages 24, 25, 111.*

[98] Shibin Parameswaran et al. "Accelerating GMM-based patch priors for image restoration: Three ingredients for a 100× speed-up". In: *CoRR* (2017). arXiv: 1710.08124 *Cited on page 25.*

[99] K. Dabov et al. "Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering". In: *IEEE Transactions on Image Processing* 16.8 (Aug. 2007), pp. 2080–2095 *Cited on pages 25, 92.*

[100] J. Mairal et al. “Non-local sparse models for image restoration”. In: *2009 IEEE 12th International Conference on Computer Vision*. Sept. 2009, pp. 2272–2279 *Cited on pages 25, 26.*

[101] S. Bengio et al. “Group Sparse Coding”. In: *Advances in Neural Information Processing Systems 22*. 2009, pp. 82–89 *Cited on pages 25, 26.*

[102] P. Chatterjee and P. Milanfar. “Clustering-Based Denoising With Locally Learned Dictionaries”. In: *IEEE Transactions on Image Processing* 18.7 (July 2009), pp. 1438–1451 *Cited on page 25.*

[103] J. Zhang, D. Zhao, and W. Gao. “Group-Based Sparse Representation for Image Restoration”. In: *IEEE Transactions on Image Processing* 23.8 (Aug. 2014), pp. 3336–3351 *Cited on page 25.*

[104] Viren Jain and Sebastian Seung. “Natural Image Denoising with Convolutional Networks”. In: *Advances in Neural Information Processing Systems 21*. 2009, pp. 769–776 *Cited on page 27.*

[105] C. Dong et al. “Image Super-Resolution Using Deep Convolutional Networks”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.2 (Feb. 2016), pp. 295–307 *Cited on page 27.*

[106] J. Kim, J. K. Lee, and K. M. Lee. “Accurate Image Super-Resolution Using Very Deep Convolutional Networks”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2016, pp. 1646–1654 *Cited on page 27.*

[107] J. Kim, J. K. Lee, and K. M. Lee. “Deeply-Recursive Convolutional Network for Image Super-Resolution”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2016, pp. 1637–1645 *Cited on page 27.*

[108] W. Shi et al. “Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2016, pp. 1874–1883 *Cited on page 27.*

[109] Kai Zhang et al. “Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising”. In: *IEEE Transactions on Image Processing* 26.7 (July 2017), pp. 3142–3155 *Cited on page 27.*

[110] K. Zhang et al. “Learning Deep CNN Denoiser Prior for Image Restoration”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. July 2017, pp. 2808–2817 *Cited on page 27.*

[111] S. Lefkimmiatis. “Non-local Color Image Denoising with Convolutional Neural Networks”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. July 2017, pp. 5882–5891 *Cited on page 27.*

[112] N. Divakar and R. V. Babu. “Image Denoising via CNNs: An Adversarial Approach”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. July 2017, pp. 1076–1083 *Cited on page 27.*

[113] H. C. Burger, C. J. Schuler, and S. Harmeling. “Image denoising: Can plain neural networks compete with BM3D?” In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. June 2012, pp. 2392–2399 *Cited on page 27.*

[114] Junyuan Xie, Linli Xu, and Enhong Chen. “Image Denoising and Inpainting with Deep Neural Networks”. In: *Advances in Neural Information Processing Systems 25*. 2012, pp. 341–349 *Cited on pages 27, 92.*

[115] Forest Agostinelli, Michael R Anderson, and Honglak Lee. “Adaptive Multi-Column Deep Neural Networks with Application to Robust Image Denoising”. In: *Advances in Neural Information Processing Systems 26*. 2013, pp. 1493–1501 *Cited on page 27.*

[116] Chao Dong et al. “Learning a Deep Convolutional Network for Image Super-Resolution”. In: *European Conference on Computer Vision (ECCV 2014)*. 2014, pp. 184–199 *Cited on page 27.*

[117] Y. Tai, J. Yang, and X. Liu. “Image Super-Resolution via Deep Recursive Residual Network”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. July 2017, pp. 2790–2798 *Cited on page 27.*

[118] Y. Chen and T. Pock. “Trainable Nonlinear Reaction Diffusion: A Flexible Framework for Fast and Effective Image Restoration”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (June 2017), pp. 1256–1272 *Cited on page 27.*

[119] Honglak Lee et al. “Efficient Sparse Coding Algorithms”. In: *Proceedings of the 19th International Conference on Neural Information Processing Systems*. 2006, pp. 801–808 *Cited on page 33.*

[120] Joel A. Tropp and Anna C. Gilbert. “Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit”. In: *IEEE Transactions on Information Theory* 53.12 (Dec. 2007), pp. 4655–4666 *Cited on pages 36, 93.*

[121] Stphane Mallat. *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. 3rd. Academic Press, 2008 *Cited on page 38.*

[122] Guoshen Yu and Guillermo Sapiro. “DCT Image Denoising: a Simple and Effective Image Denoising Algorithm”. In: *Image Processing On Line* 1 (2011), pp. 292–296 *Cited on page 38.*

[123] I.T. Jolliffe. *Principal component analysis*. New York: Springer Verlag, 2002 *Cited on page 39.*

[124] R. Vidal, Yi Ma, and S. Sastry. “Generalized principal component analysis (GPCA)”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.12 (Dec. 2005), pp. 1945–1959 *Cited on page 39.*

[125] S. Bacchelli and S. Papi. “Image denoising using principal component analysis in the wavelet domain”. In: *Journal of Computational and Applied Mathematics* 189.1 (2006), pp. 606–621 *Cited on page 39.*

[126] Q. Liu et al. “Adaptive sparse coding on PCA dictionary for image denoising”. In: *The Visual Computer* 32.4 (Apr. 2016), pp. 535–549 *Cited on page 39.*

[127] Julien Mairal et al. “Online Dictionary Learning for Sparse Coding”. In: *Proceedings of the 26th Annual International Conference on Machine Learning*. 2009, pp. 689–696 *Cited on page 39.*

[128] R. Rubinstein, M. Zibulevsky, and M. Elad. “Double Sparsity: Learning Sparse Dictionaries for Sparse Signal Approximation”. In: *IEEE Transactions on Signal Processing* 58.3 (Mar. 2010), pp. 1553–1564 *Cited on page 39.*

[129] J. Mairal, F. Bach, and J. Ponce. “Task-Driven Dictionary Learning”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.4 (Apr. 2012), pp. 791–804 *Cited on page 39.*

[130] Yossi Rubner. In: *International Journal of Computer Vision* 40.2 (2000), pp. 99–121 *Cited on pages 44, 45.*

[131] Qin Lv, Moses Charikar, and Kai Li. “Image similarity search with compact data structures”. In: *Proceedings of the Thirteenth ACM conference on Information and knowledge management - CIKM ’04*. 2004, pp. 208–217 *Cited on pages 44, 45.*

[132] Ofir Pele and Michael Werman. “A Linear Time Histogram Metric for Improved SIFT Matching”. In: *Lecture Notes in Computer Science*. 2008, pp. 495–508 *Cited on pages 44, 45.*

[133] Ofir Pele and Michael Werman. “Fast and robust Earth Mover’s Distances”. In: *2009 IEEE 12th International Conference on Computer Vision*. IEEE, Sept. 2009, pp. 460–467 *Cited on pages 44, 45, 64, 67.*

[134] Zhenghua Yu and G. Herman. “On the Earth Mover’S Distance as a Histogram Similarity Metric for Image Retrieval”. In: *2005 IEEE International Conference on Multimedia and Expo*. 2005 *Cited on pages 44, 45.*

[135] Min-Hee Jang et al. “Accurate Approximation of the Earth Mover’s Distance in Linear Time”. In: *Journal of Computer Science and Technology* 29.1 (Jan. 2014), pp. 142–154 *Cited on page 44.*

[136] Jianguo Zhang et al. “Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study”. In: *Conference on Computer Vision and Pattern Recognition Workshop (CVPRW’06)*. IEEE, 2006. DOI: 10.1109/cvprw.2006.121 *Cited on page 44.*

[137] K. Grauman and T. Darrell. “Fast contour matching using approximate Earth mover’s distance”. In: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2004, pp. 220–227 *Cited on pages 44, 45.*

[138] Peihua Li, Qilong Wang, and Lei Zhang. “A Novel Earth Mover’s Distance Methodology for Image Matching with Gaussian Mixture Models”. In: *2013 IEEE International Conference on Computer Vision*. IEEE, Dec. 2013, pp. 1689–1696 *Cited on page 44.*

[139] C.S. Mendoza et al. “Multi-dimensional earth mover’s distance active contours”. In: *IEEE International Conference on Image Processing*. Sept. 2011, pp. 3173–3176 *Cited on page 44.*

[140] Peihua Li and Qilong Wang. “Variational Earth Mover’s Distance for Image Segmentation”. In: *International Conference on Image and Graphics*. July 2013, pp. 236–242 *Cited on page 44.*

[141] S. Peleg, M. Werman, and H. Rom. “A unified approach to the change of resolution: space and gray-level”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11.7 (July 1989), pp. 739–742 *Cited on page 44.*

[142] Haibin Ling and K. Okada. “An Efficient Earth Mover’s Distance Algorithm for Robust Histogram Comparison”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.5 (May 2007), pp. 840–853 *Cited on page 45.*

[143] S. Shirdhonkar and D.W. Jacobs. “Approximate earth mover’s distance in linear time”. In: *IEEE Conference on Computer Vision and Pattern Recognition*. June 2008, pp. 1–8 *Cited on page 45.*

[144] F. J. Anscombe. “The transformation of Poisson, binomial and negative-binomial data”. In: *Biometrika* 35.3-4 (1948), pp. 246–254 *Cited on page 49.*

[145] Yi Tang et al. “Single-image super-resolution via local learning”. In: *International Journal of Machine Learning and Cybernetics* 2.1 (Feb. 2011), pp. 15–23 *Cited on page 50.*

[146] Qiang Wu, Yiming Ying, and Ding-Xuan Zhou. “Multi-kernel regularized classifiers”. In: *Journal of Complexity* 23.1 (Feb. 2007), pp. 108–134 *Cited on page 51.*

[147] Fei Sha, Lawrence K. Saul, and Daniel D. Lee. “Multiplicative Updates for Nonnegative Quadratic Programming in Support Vector Machines”. In: *Advances in Neural Information Processing Systems 15*. Ed. by S. Becker, S. Thrun, and K. Obermayer. 2003, pp. 1065–1072 *Cited on page 68.*

[148] Michal Irani and Shmuel Peleg. “Motion Analysis for Image Enhancement: Resolution, Occlusion, and Transparency”. In: *Journal of Visual Communication and Image Representation* 4.4 (Dec. 1993), pp. 324–335 *Cited on page 69.*

[149] K. Kreutz-Delgado et al. “Dictionary Learning Algorithms for Sparse Representation”. In: *Neural Computation* 15.2 (Feb. 2003), pp. 349–396 *Cited on page 81.*

[150] Bruno A. Olshausen and David J. Field. “Sparse coding with an overcomplete basis set: A strategy employed by V1?” In: *Vision Research* 37.23 (Dec. 1997), pp. 3311–3325 *Cited on page 81.*

[151] H. Lu, A.C. Kot, and Y.Q. Shi. “Distance-Reciprocal Distortion Measure for Binary Document Images”. In: *IEEE Signal Processing Letters* 11.2 (Feb. 2004), pp. 228–231 *Cited on page 93.*

[152] Julian Besag. “On the statistical analysis of dirty pictures”. In: *Journal of the Royal Statistical Society: Series B* 48.3 (1986), pp. 48–259 *Cited on page 98.*

[153] V. Kolmogorov and R. Zabih. “What energy functions can be minimized via graph cuts?” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.2 (Feb. 2004), pp. 147–159 *Cited on page 98.*

[154] Fahad Shahbaz Khan et al. “Color attributes for object detection”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, June 2012, pp. 3306–3313 *Cited on page 115.*

[155] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. “Single image super-resolution from transformed self-exemplars”. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2015, pp. 5197–5206 *Cited on page 115.*

[156] Phil Brodatz. *Textures: A Photographic Album for Artists and Designers*. Dover Pubns, 1966 *Cited on page 115.*

[157] Mircea Cimpoi et al. "Describing Textures in the Wild". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, June 2014, pp. 3606–3613 *Cited on page 115.*

[158] A. P. Dempster, N. M. Laird, and D. B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the Royal Statistical Society, Series B* 39.1 (1977), pp. 1–38 *Cited on page 117.*

Titre : **Approaches bayésiennes par patchs pour l'amélioration de la qualité des images.**

Mots-clefs : Débruitage d'images, super-résolution d'images, maximum à posteriori, modèle de mélange gaussien, earth mover's distance, modèle parcimonieux, distribution des patchs, restauration d'images basée sur l'exemple.

Résumé : Les travaux présentés dans cette thèse concernent les approches bayésiennes par patchs des problèmes d'amélioration de la qualité d'images. Notre contribution réside en le choix du dictionnaire construit grâce à un ensemble d'images de haute qualité et en la définition et l'utilisation d'un modèle à priori pour la distribution des patchs dans l'espace du dictionnaire. Nous avons montré qu'un choix attentif du dictionnaire représentant les informations locales des images permettait une amélioration de la qualité des images dégradées. Plus précisément, d'un dictionnaire construit de façon exhaustive sur les images de haute qualité nous avons sélectionné, pour chaque patch de l'image dégradée, un sous dictionnaire fait de ses voisins les plus proches. La similarité entre les patchs a été mesurée grâce à l'utilisation de la distance du cantonnier (Earth Mover's Distance) entre les distributions des intensités de ces patchs. L'algorithme de super résolution présenté a conduit à de meilleurs résultats que les algorithmes les plus connus. Pour les problèmes de débruitage d'images nous nous sommes intéressés à la distribution à priori des patchs dans l'espace du dictionnaire afin de l'utiliser comme pré requis pour régulariser le problème d'optimisation donné par le Maximum à Postériori. Dans le cas d'un dictionnaire de petite dimension, nous avons proposé une distribution constante par morceaux. Pour les dictionnaires de grande dimension, la distribution à priori a été recherchée comme un mélange de gaussiennes (GMM). Nous avons finalement justifié le nombre de gaussiennes utiles pour une bonne reconstruction apportant ainsi un nouvel éclairage sur l'utilisation des GMM.

Title : **Patch-based Bayesian approaches for image restoration.**

Keywords : Image denoising, image super-resolution, Bayesian Maximum A Posteriori, Gaussian Mixture Model, Earth Mover's Distance, Sparse models, patches distribution, example-based image restoration.

Abstract : In this thesis, we investigate the patch-based image denoising and super-resolution under the Bayesian Maximum A Posteriori framework, with the help of a set of high quality images which are known as standard images. Our contributions are to address the construction of the dictionary, which is used to represent image patches, and the prior distribution in dictionary space. We have demonstrated that the careful selection of dictionary to represent the local information of image can improve the image reconstruction. By establishing an exhaustive dictionary from the standard images, our main attribute is to locally select a sub-dictionary of matched patches to recover each patch in the degraded image. Beside the conventional Euclidean measure, we propose an effective similarity metric based on the Earth Mover's Distance (EMD) for image patch-selection by considering each patch as a distribution of image intensities. Our EMD-based super-resolution algorithm has outperformed comparing to some state-of-the-art super-resolution methods. To enhance the quality of image denoising, we exploit the distribution of patches in the dictionary space as a an image prior to regularize the optimization problem. We develop a computationally efficient procedure, based on piece-wise constant function estimation, for low dimension dictionaries and then proposed a Gaussian Mixture Model (GMM) for higher complexity dictionary spaces. Finally, we justify the practical number of Gaussian components required for recovering patches. Our researches on multiple datasets with combination of different dictionaries and GMM models have complemented the lack of evidence of using GMM in the literature.