COMPUTED TOMOGRAPHY NOISE REDUCTION BASED ON TOTAL VARIATION MINIMIZATION AND MORPHOLOGICAL COMPONENT ANALYSIS

by

Aryan Khodabandeh

BASc, University of Toronto, Toronto, Canada, 2012

A thesis

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Applied Science

in the program of

Electrical and Computer Engineering

Toronto, Ontario, Canada, 2015

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

COMPUTED TOMOGRAPHY NOISE REDUCTION BASED ON TOTAL VARIATION MINIMIZATION AND MORPHOLOGICAL COMPONENT ANALYSIS

Master of Applied Science

2015

Aryan Khodabandeh
Electrical and Computer Engineering
Ryerson University

Abstract

X-ray Computed Tomography (CT) scans, while useful, emit harmful radiation which is why low-dose image acquisition is desired. However, noise corruption in these cases is a difficult obstacle. CT image denoising is a challenging topic because of the difficulty in modeling noise. In this study, we propose taking an image decomposition approach to removing noise from low-dose CT images. We model the image as the superposition of a structure layer and a noise layer. Total Variation (TV) minimization is used to learn two dictionaries to represent each layer independently, and sparse coding is used to separate them. Finally, an iterative post-processing stage is introduced that uses image-adapted curvelet dictionaries to recover blurred edges. Our results demonstrate that image separation is a viable alternative to the classic K-SVD denoising method.

Acknowledgements

I would like to thank my supervisor Dr. Javad Alirezaie for his support and guidance during my MASc studies at Ryerson University. His trust in me let me arrive at the research topic that most suited me and I was able to finish it by taking advantage of his counsel.

I would like to thank Dr. Paul Babyn for his excellent remarks on my writing and granting us with much needed images and resources.

I am very grateful to my parents for their encouragement and support during all my studies.

A Special thanks to all my colleagues and friends for all their help.

Table of Contents

List of Tables

List of Figures

Chapter 1. Introduction

X-Ray Computed Tomography (CT) is a valuable resource in medical imaging. It creates a comprehensive representation of the body of patients and allows us to see all the different tissues and bones in 2D image slices or in a 3D virtual reconstruction. However, it comes with some risks for the health of the person being scanned because of the ionizing nature of x-ray emissions. As a result, lowering the x-ray dosage during image acquisition is desired. It is well known that as the dose is decreased, noise and streak artifacts become more prominent, degrade the image and lower the Signal to Noise Ratio (SNR) [1]. Therefore, methods of lowering noise in low-dose images are required.

The main source of noise in the raw data (sinogram) is quantum noise which is due to an insufficient number of x-ray photons reaching the detector. Lowering the radiation dose has a direct restrictive effect on how many photons penetrate the patient, thus causing an increase in noise. This noise can be modeled as a Poisson process. After image reconstruction from the sinogram occurs, the noise no longer has a known distribution and becomes non-stationary. As a result, usual denoising methods are not appropriate for dealing with CT images. We propose taking an image decomposition approach to separate the main image structures from unwanted artifacts that arise in low-dose CT images.

Many CT denoising techniques have previously been proposed. They mainly fit into three categories. The first two are similar in their goal to take the statistical nature of CT data and noise into account and then construct the image from the projection data. To achieve this we can either optimize a function whose parameters are image pixels [2] or we can denoise the projection data first and then perform the image reconstruction [3]. The former in particular has recently garnered great attention with the class of algorithms called Statistical Iterative Reconstruction (SIR). With the increase in computing power, newer scanners incorporate this type of image reconstruction rather than the traditional Filtered Back Projection (FBP) method [4]. However, this is still a relatively new movement, and the most common noise removal method is to take the reconstructed image and perform signal denoising on it. This report will focus on this approach.

Edge preservation is a critical aspect of medical image denoising because edges often contain important diagnostic information. Many signal and image denoising algorithms exist. Wavelet thresholding [5] [6], diffusion [7], and total variation (TV) denoising [8] methods are a

few such algorithms. More recently, sparse representation and dictionary learning methods [9] have shown great potential in adaptively analyzing and denoising images.

Many sparse transforms have been proposed over the years. Wavelets [10], curvelets [11], contourlets [12], and shearlets [13] are just a few of the many transforms that can sparsely represent images over a fixed dictionary. More recently, Aharon *et al.* proposed the K-SVD algorithm for adaptively learning the dictionary from image patches [14]. This method is able to find better and sparser representations of images with complex patterns which would not be possible with fixed dictionaries.

The task of image decomposition can be considered a generalization of denoising. Morphological diversity is the idea that an image can be decomposed into two or more layers, each having a different morphology. In 2004, Starck *et al.* introduced a sparse representation approach called Morphological Component Analysis (MCA) [15]. It aims to use mutually incoherent analytical dictionaries, each one good in representing one layer but not the others. The success of this method depends on the choice of each dictionary and how efficient it is in sparsifying the intended layer while being highly inefficient in doing so for the other layers.

Peyré *et al*. extended the MCA algorithm by combining fixed and learned dictionaries [16]. The reason is that some complicated textures may not be effectively represented with any fixed dictionary. In that case learning the dictionary will result in better image separation. Shoham *et al.* showed in [17] that if pre-learned dictionaries exist for each layer, the separation can be done by a directly degenerated block-coordinate-descent algorithm. If pre-learned dictionaries do not exist, they can be alternatively learned and their corresponding layers separated iteratively as Li *et al.* showed in [18].

In this thesis we propose treating noise and unwanted streaks in low-dose CT images as texture that needs to be separated from the main structures. The contributions are outlined here. First the image is smoothed using TV denoising [19] and a dictionary representing the noiseless image is learned from it. A second dictionary representing the noise is learned from the residual between the original image and its smoothed version. Each of these dictionaries is better in coding its own intended morphological content, and sparse representation will result in a piecewise smooth (main structures) layer and a noise layer. Finally, the curvelet transform and dictionary learning are combined to recover edges that falsely end up in the noise layer.

This report is organized as follows. In chapter 2, an overview of CT imaging and its risks is provided; sources of noise and their effect on image quality are examined, and a few methods of dealing with noise are surveyed. In chapter 3, the development of mathematical transforms from orthogonal to overcomplete dictionaries, pursuit methods for seeking sparse representations of signals, and dictionary learning methods are studied. Chapter 4 contains the details of the MCA algorithm leading to the specifics of our proposed method. In chapter 5, the results of testing our algorithm on various CT and natural images are compared to the K-SVD denoising method using the peak signal to noise ratio (PSNR) and structural similarity (SSIM) metrics. Chapter 6 offers some concluding comments about this work and any potential future extensions.

Chapter 2. X-Ray Computed Tomography Imaging

X-ray Computed Tomography (CT) is a diagnostic tool that is used to scan the interior of objects and present them as images based on the various degrees that different materials absorb x-ray photons. It has both industrial and medical uses. The latter in particular is a sensitive topic because of the risks that accompany this type of scanning. X-rays are energetic photons that are able to create ions by liberating electrons from their molecular orbits. Such ionizing radiation is harmful to biological tissue especially DNA which can increase the risk of cancer [20]. By examining CT usage statistics from 1991 to 1996, estimations show that 0.4% of

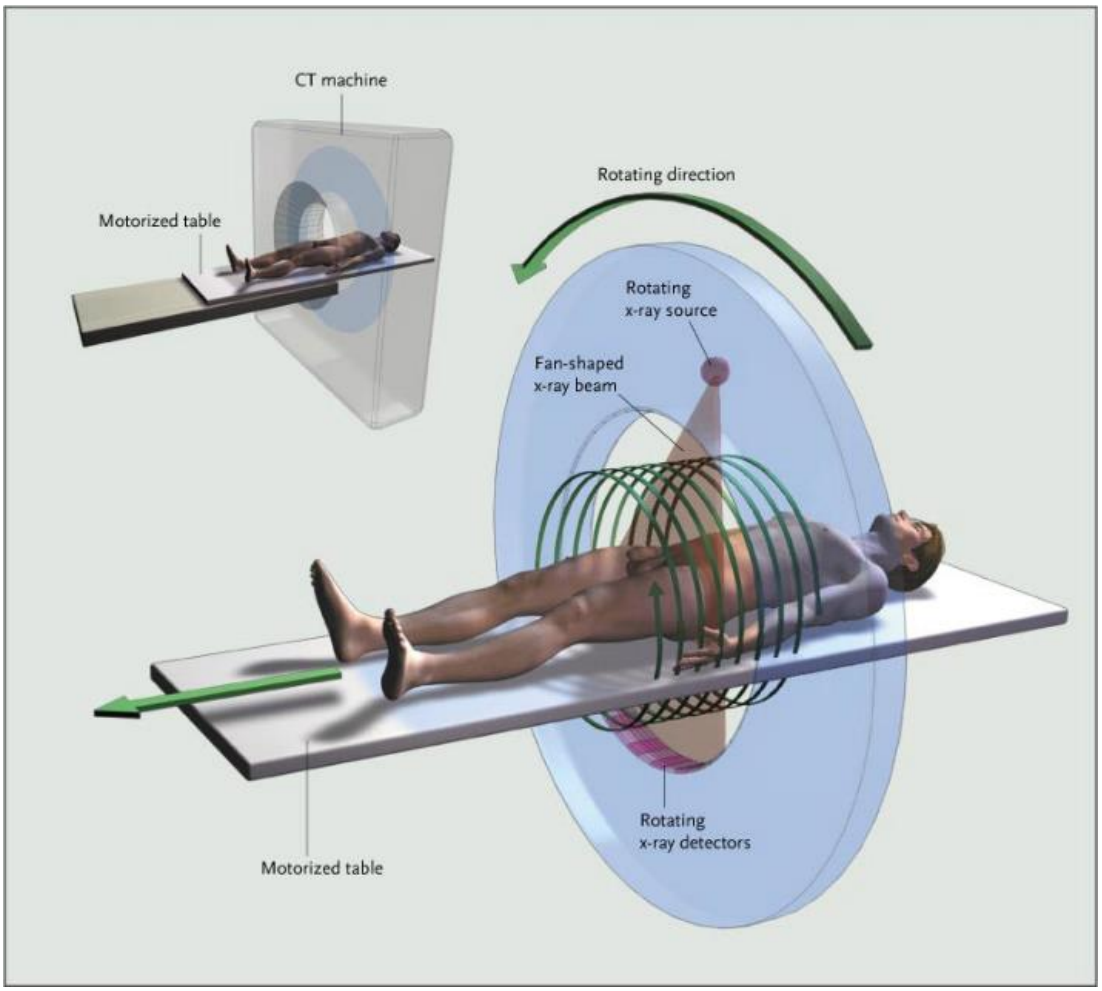


Figure 2.1 Basics of CT imaging [20]

all current cancers in the United States may be linked to CT use in the past [21] [22]. In another study, 1 in every 1800 CT scans was followed by an excess cancer [23].

CT scanning has several advantages over 2D radiography which is another type of x-ray imaging. The latter simply forms an image which is the superposition of all the structures in the scanned object, whereas CT creates a series of image slices that can be viewed individually. This improves contrast and gives the ability to view objects nested inside one another. Estimates show that 67 million CT scans were conducted in 2006 in the U.S. alone [24]. Although the risk to one person undergoing one CT scan may not be large, given how prevalent CT scanning has

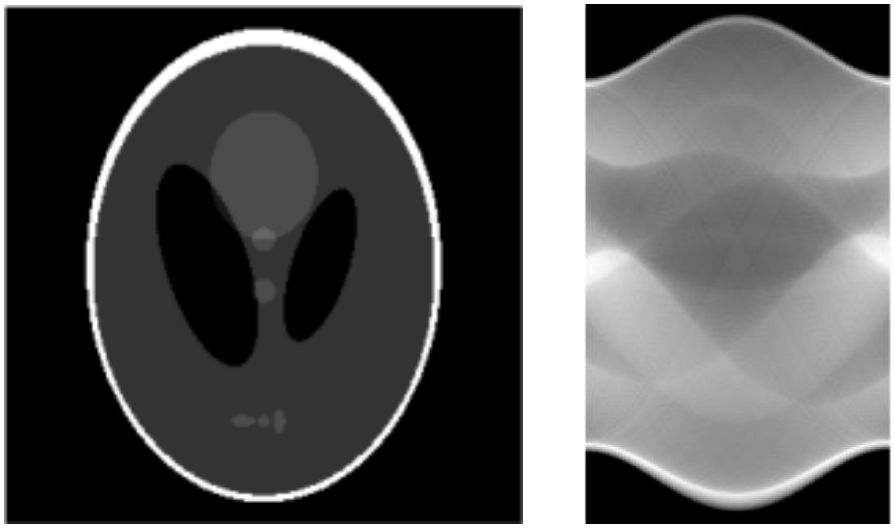


Figure 2.2 Left: Shepp–Logan phantom [25], Right: Its sinogram

become it is considered a public health issue and measures need to be taken to reduce its harm. One way of reducing the radiation absorbed by a person during a scan is to decrease the dose which will consequently result in more noise and a lower Signal to Noise Ratio (SNR) [1]. The focus of this report is to examine methods of decreasing noise in low-dose CT images.

2.1. Procedure Overview

CT scanning involves a motorized bed for the patient with a rotating apparatus around it. An x-ray source sends photons in a fan shaped beam towards the patient and several rows of detectors at the opposite side record the photons that get through. As can be seen in Figure 2.1 this process continues as the source and detectors rotate and the bed is moved through the middle. Consequently, a series of image slices are produced that correspond to slices of the patient's body. Each slice can be viewed individually as a 2D image or a 3D representation of the body can be constructed using special software.

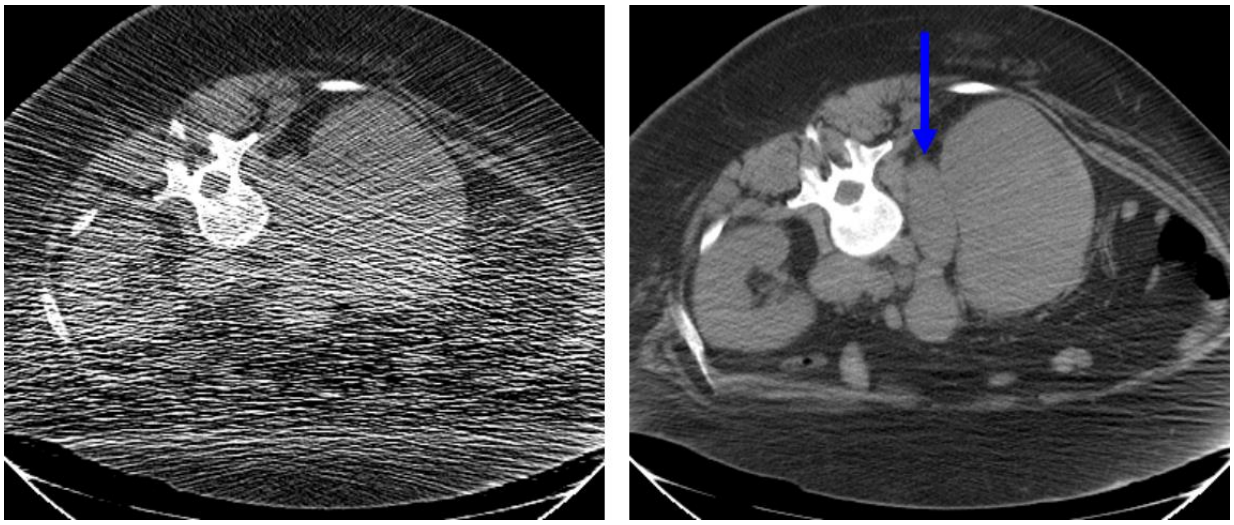
The raw data captured by the detectors, called a sinogram, consists of several projections from different angles of the same body slice. These projections are essentially the radon transform of the scanned object. Figure 2.2 shows the transform of the Shepp—Logan phantom [25]. To get the image back the inverse radon transform of the sinogram needs to be solved.

2.2. Radiation Dose and Image Quality

CT imaging is susceptible to a number of artifacts that are detrimental to image quality. Many factors contribute to these imperfections such as miscalibration of the scanner elements, metallic implants in the patient, and patient motion during the scan. The most prevalent artifacts are caused by beam hardening, Compton scattering, and photon starvation.

X-ray CT imaging takes advantage of the fact that different tissues absorb photons of a given energy to varying degrees. Most CT scanners use polychromatic x-ray beams which mean the photons have a range of energies. This causes the artifact known as beam hardening which occurs when the lower energy photons in the x-ray beam get absorbed more than those with higher energy [26]. It shows up as non-uniformities in the image of a uniform material. These artifacts are usually small and not very noticeable, but they can become significant alongside large bones or metallic objects [27]. Iterative and reconstructive methods of correcting this anomaly have been proposed [28] [29].

Compton scattering involves an x-ray photon interacting with a free electron or one that's loosely bound to an atom. As a result the photon is redirected in a different direction and ends up in a different detector than the one positioned to receive it. This causes the same kind of



60 mA, 120 kVp, slice thickness 5 mm

440 mA, 120 kVp, slice thickness 5 mm

Figure 2.3 Effect of mA on Poisson noise. LEFT: Low dose CT image obtained during a CT-guided biopsy shows extensive Poisson noise. These streaks are the same whether or not the abdomen or arms are partially outside the field of view. RIGHT: Post-biopsy image obtained at $\mathbf{7.3}$ times higher dose has $\sqrt{\mathbf{7.3}} = \mathbf{2.7}$ times less noise. The images show an enlarged retroperitoneal lymph node (arrow) and infiltration of the right kidney in a patient with Hodgkin's lymphoma. [31]

dark streaks as beam hardening because of a higher than expected number of photons in a particular detector [30].

The number of photons that reach the detectors is perhaps the most important factor in determining image quality and consequently the radiation dosage applied to the patient. Under ideal conditions without any of the problems discussed before, the number of photons reaching each detector follows a Poisson distribution with the mean equal to the variance. Therefore, low photon numbers result in poor SNR. Since the measurements of various x-ray projections contribute to a pixel's value in the final image, the noise becomes more complicated and appears as small streaks mostly oriented in the direction of greatest attenuation. In the extreme photon starvation case, the streaks appear as long straight lines. See Figure 2.3 for illustration of this kind of noise.

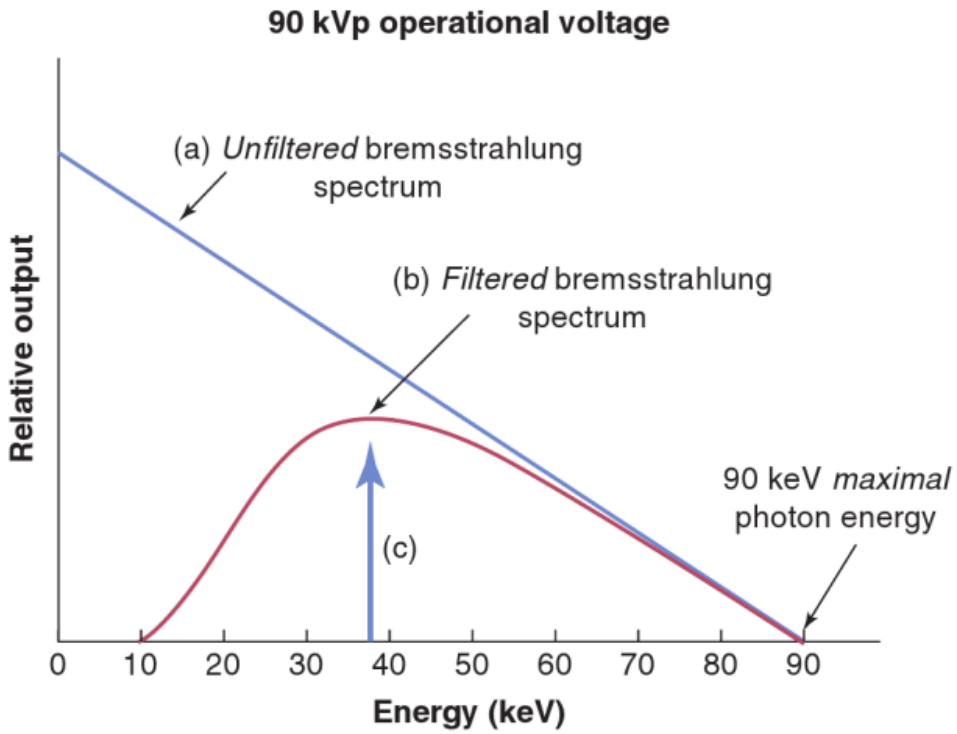


Figure 2.4 The bremsstrahlung energy distribution for a 90-kV acceleration potential difference. The unfiltered bremsstrahlung spectrum (a) shows a greater probability of low-energy x-ray photon production that is inversely linear with energy up to the maximum energy of 90 keV. The filtered spectrum (b) shows the preferential attenuation of the lowest-energy x-ray photons. The vertical arrow (c) indicates the average energy of the spectrum, which is typically 1/3 to 1/2 the maximal energy. [32]

There are a few parameters that have a large impact on the photon numbers detected. They can be changed at the start of the scanning process depending on the body part under examination and other factors decided by the radiologist. The maximum voltage applied across the x-ray tube, called the peak kilovoltage (kVp), relates to the peak energy of the emitted x-ray spectrum (Figure 2.4). A higher value means an increase in the probability of each photon penetrating the tissues. The x-ray tube current, measured as milliamperes (mA) determines the intensity of the beam or the number of x-ray photons emitted. Increasing the current would mean the peak of the x-ray spectrum is elevated upwards. Scan time is the duration of each measurement which means how long the scanner stays in one position to gather photons. Since

this value and the current are related, they are often combined as milliampere-seconds (mAs). Slice thickness is the width of the beam entering each detector which affects the number of photons detected. This parameter also affects spatial resolution which leads to a trade-off between sharpness and noise.

The tube current and the scan time are often the only values that are changed in relation to radiation dose. In the interest of the patient's health they can be lowered which will result in more noise. It is the purpose of this report to examine methods of removing noise in the CT images resulting from low photon counts while leaving the actual structures intact.

Filtered Back Projection (FBP) is the most common method of obtaining an image from raw scanner data [33]. First the sinogram is high-pass filtered and then the inverse radon transform is performed on the result. The filtering is necessary to avoid the extensive blurring that would otherwise occur. However, this also emphasizes noise because this algorithm assumes the data is noiseless. Therefore, more complex filters are often used to also dampen noise. This creates a trade-off between lower noise and sharper edges, and different filters are used depending on the situation and achieve one in the expense of the other. The reconstruction process changes the noise distribution to something unknown. Therefore, simple denoising algorithms that assume a known noise model are not optimal for CT images.

2.3. Introduction to Noise Reduction Methods

There are three areas of research that aim to reduce noise and streaks in low-dose CT images: statistical iterative reconstruction, sinogram denoising, and image denoising. In the following sections, they are briefly examined.

2.3.1. Statistical Iterative Reconstruction

Instead of the FBP algorithm, the newest scanners use some kind of iterative algorithm to continually improve the reconstruction of the image by incorporating a statistical model of the noise [2]. This class of algorithms is called Statistical Iterative Reconstruction (SIR). Taking into account the Poisson nature of detected photons after attenuation by the scanned tissue, we can write

$$y_i = Poisson\{I_i e^{-l_i}\}, i \in \{1, \dots, N\} \tag{2.1}$$

where y_i are the recorded measurements for all positions and angles, I_i are the incident x-ray intensities, and $l_i = \sum_j a_{ij}\mu_j$ are the line integrals through the tissue consisting of $j \in \{1, \dots, M\}$ voxels (3D equivalent of pixels). In this formulation, a_{ij} represents the probability of detecting a photon in sensor i that originated in voxel j, and each μ_j is the pixel value in the CT image that we want to find. This relationship is often written in matrix form as $\boldsymbol{l} = \boldsymbol{A\mu}$. The matrix $\boldsymbol{A}$, called the system matrix, models the scanner and its geometry and needs to do this very accurately or the algorithm will not work correctly. The amount of memory required to hold $\boldsymbol{A}$ grows rapidly as the resolution of the image and the number of x-ray projections increases.

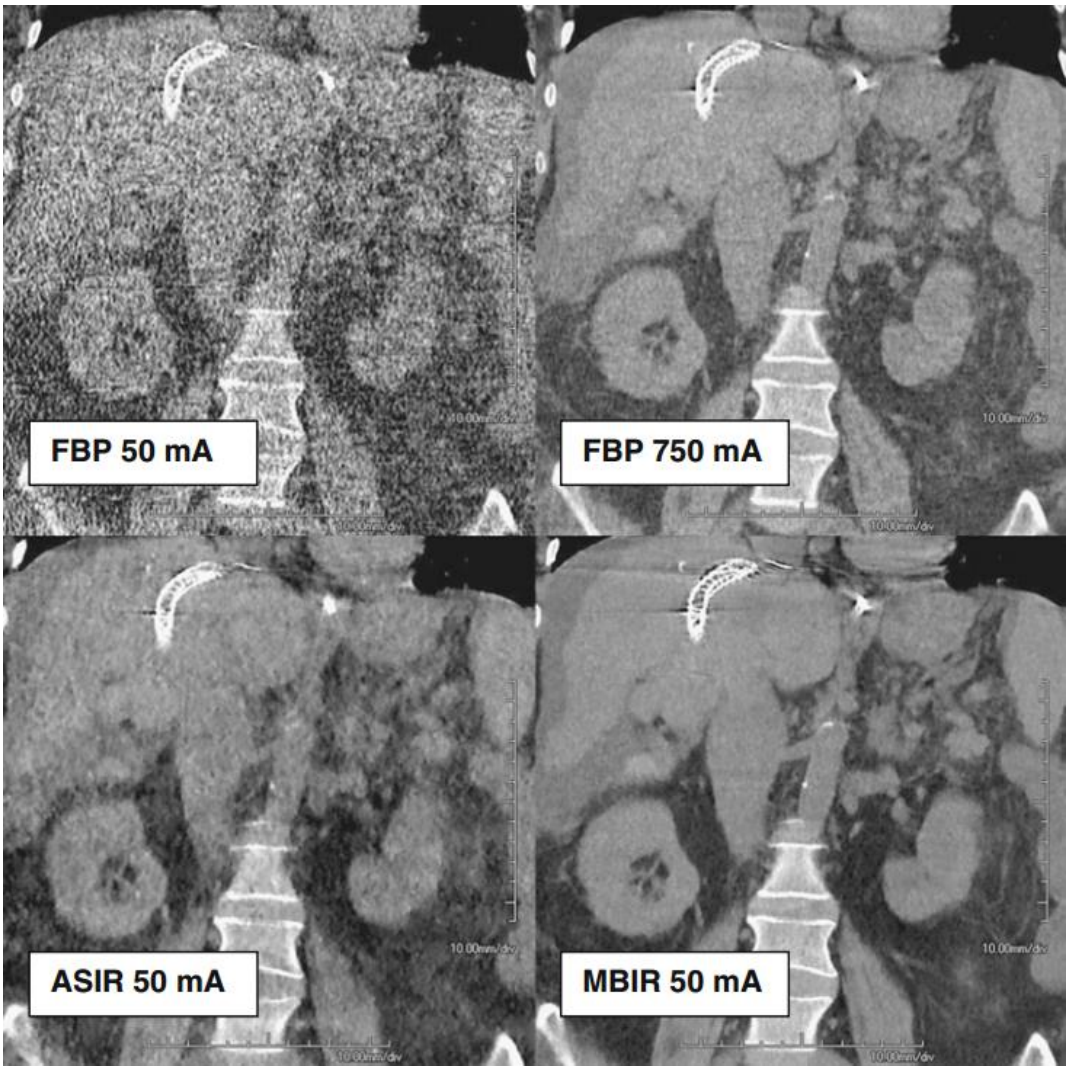


Figure 2.5 Iterative reconstruction using Advanced Statistical Iterative Reconstruction (ASIR) and Model-Based Iterative Reconstruction (MBIR), both from General Electric (Milwaukee, WI, USA). Coronal reformation of a non-contrast CT scan inadvertently obtained with 50 mA and reconstructed with filtered back projection (FBP) shows excessive image noise (TOP LEFT), requiring a repeat scan acquired at 750 mA (TOP RIGHT). Reconstruction of the 50 mA dataset using ASIR shows decreased image noise (BOTTOM LEFT). Dramatic reduction of image noise is achieved by reconstructing the 50 mA dataset with the full iterative reconstruction (MBIR) (BOTTOM RIGHT), which compares favorably with the 750 mA FBP image (TOP RIGHT) which was acquired at 15 times more radiation dose. [4]

One way of solving this problem is in a Bayesian framework as the maximum *a posteriori* (MAP) estimate of $P(\boldsymbol{\mu}|\boldsymbol{y})$ which is equivalent to

$$\widehat{\boldsymbol{\mu}} = \max_{\boldsymbol{\mu}}\{\ln P(\mathbf{y}|\boldsymbol{\mu}) + \ln P(\boldsymbol{\mu})\} \tag{2.2}$$

By using the second-order Taylor series expansion in terms of the unknown image and the natural logarithm of the Poisson distribution for y_i we can write the log likelihood term as

$$\ln P(\mathbf{y}|\boldsymbol{\mu}) = -\frac{1}{2}(\boldsymbol{y} - \boldsymbol{A\mu})^T \boldsymbol{D}(\boldsymbol{y} - \boldsymbol{A\mu}) + f(\boldsymbol{y}) \tag{2.3}$$

where $f(\boldsymbol{y})$ is some function, and $\boldsymbol{D}$ is a diagonal matrix whose coefficients are proportional to the inverse of the variance of the projection measurements: $d_i \propto y_i = I_i e^{-l_i} \cong {}^{1}/_{\sigma_{y_i}^2}$. Finally the image can be estimated as:

$$\widehat{\boldsymbol{\mu}} = \max_{\boldsymbol{\mu}} \left\{ \frac{1}{2}(\boldsymbol{y} - \boldsymbol{A\mu})^T \boldsymbol{D}(\boldsymbol{y} - \boldsymbol{A\mu}) + R(\boldsymbol{u}) \right\} \tag{2.4}$$

where $f(\boldsymbol{y})$ rewritten as $R(\boldsymbol{u})$ is a regularization term.

These types of algorithms start from an initial estimate of the image $\widehat{\boldsymbol{\mu}}_{\mathbf{0}}$ which can be from the FBP reconstruction, then iteratively update it. The regularization term $R(\boldsymbol{u})$ is important in enforcing smoothness in the image and without it the estimates become unstable. It is usually set to the negative of $ln\, P(\boldsymbol{\mu})$ in equation (2.2) where $P(\boldsymbol{\mu})$ represents prior knowledge about the image. Markov random fields are commonly used for regularization.

For all its strengths iterative reconstruction was not used in the past because of high computation times and large memory requirements to store the system matrix $\boldsymbol{A}$. In the past few years, advances in computing power have allowed these methods to become more attractive compared to the traditional FBP reconstruction. Figure 2.5 illustrates some modern realizations of the SIR approach.

2.3.2. Sinogram Denoising

There are some algorithms that try to smooth the sinogram so that it becomes closer to a noise-less set of data that one would get if the photon count was sufficiently high. The FBP algorithm can then be applied to the smoothed sinogram to get the image. In the SIR algorithm the optimization parameters are image pixels, but in this type of algorithm the optimization is done in the projection space. In [34] an adaptive filtering approach was proposed by taking into account the noise property. A multi-dimensional adaptive filtering approach was developed in [35] to enhance the projection data. Penalized likelihood sinogram smoothing techniques have also been proposed [3].

2.3.3. Image Denoising

In cases where the reconstructed low-dose CT image is noisy, image denoising techniques can be used to denoise the image. Edge preservation is crucial in any CT denoising algorithm because a diagnosis requires a sharp image with no small structures lost along with the noise.

Thresholding in the wavelet domain is a method that has showed promise for removing white Gaussian noise in images [5] [6]. Either hard or soft thresholds can be applied to the wavelet coefficients to dampen the effect of noise. This is based on the fact that coefficients representing noise have lower magnitudes than those for structures. Hard thresholding simply sets to zero the coefficients below a preselected value, while soft thresholding lowers all the coefficients according to a continuous function. Therefore the main issue becomes how to

select the threshold to be used. The best results are attained from adaptive methods such as [36] where soft thresholding based on a Bayesian framework is proposed.

Partial differential equations (PDE) have often been used in image processing. One of the first PDE-based edge preserving denoising methods was proposed by Perona and Malik [7]. They introduced a partial differential equation method called anisotropic diffusion where a diffusion tensor is chosen that varies along with the gradient of the image. As such it directs the denoising in the directions with low gradient and does not cross edges, preventing blurring.

Another effective noise reduction method which also preserves edges was proposed by Rudin, Osher, and Fatemi [8]. It is called Total Variation (TV) denoising and is based on minimizing the total variation of a noisy signal subject to a regularization term. This method is widely used in signal processing applications but for highly contaminated images it is unable to differentiate between noise and actual structures which leads to oversmoothing, loss of fine details, and a blocky appearance. Therefore it is often used in conjunction with other methods. For example in [37] dictionary learning and total variation are used to improve magnetic resonance image (MRI) quality.

Recently, the topic of sparse signal representation and dictionary learning has become the subject of much attention. Transform methods such as Fourier and wavelet can be said to sparsely represent signals using a fixed dictionary. For the Fourier transform the dictionary is composed of complex exponentials, and for the wavelet transform it is the various scaled and translated versions of a mother wavelet. Precisely because these dictionaries are predefined, they are non-adaptive. In 2006 the K-SVD [14] algorithm vas introduced for learning a dictionary that is adapted to the signal. Consequently, it was shown to be very effective at removing noise

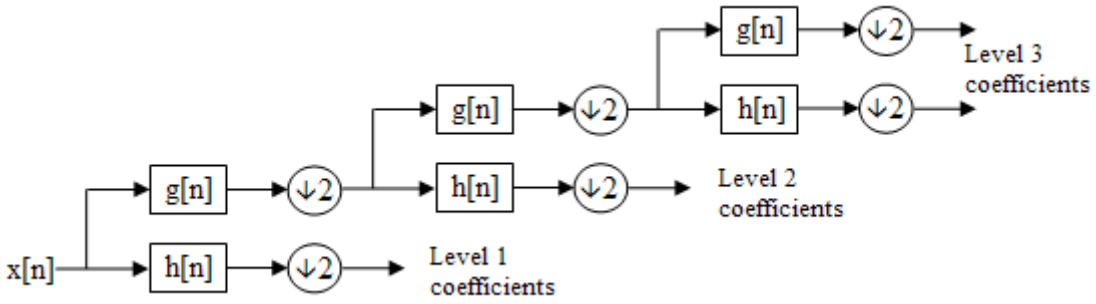


Figure 2.6 The 3 level structure of the forward discrete wavelet transform. h[n] is a high-pass filter and g[n] is the corresponding low-pass filter

from images [9]. Because K-SVD creates an adaptive dictionary, it enables better and sparser representation of signals. Using this scheme for CT image denoising has been explored in works such as [38].

2.3.3.1. Wavelet Thresholding for Image Denoising

The wavelet transform is a multiscale operator that captures both frequency and spatial information. By taking the inner product between a signal and an orthonormal wavelet function at various scales and positions their similarity can be calculated. The wavelet coefficients are given by

$$C(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} f(x)\, \psi^* \left(\frac{x-b}{a}\right) dx \tag{2.5}$$

where $a > 0$ is the scale parameter, b is the position, and ψ^* is the complex conjugate of the wavelet function. The discrete wavelet transform (DWT) is defined for discrete intervals of the wavelet function and is used in digital signal analysis applications.

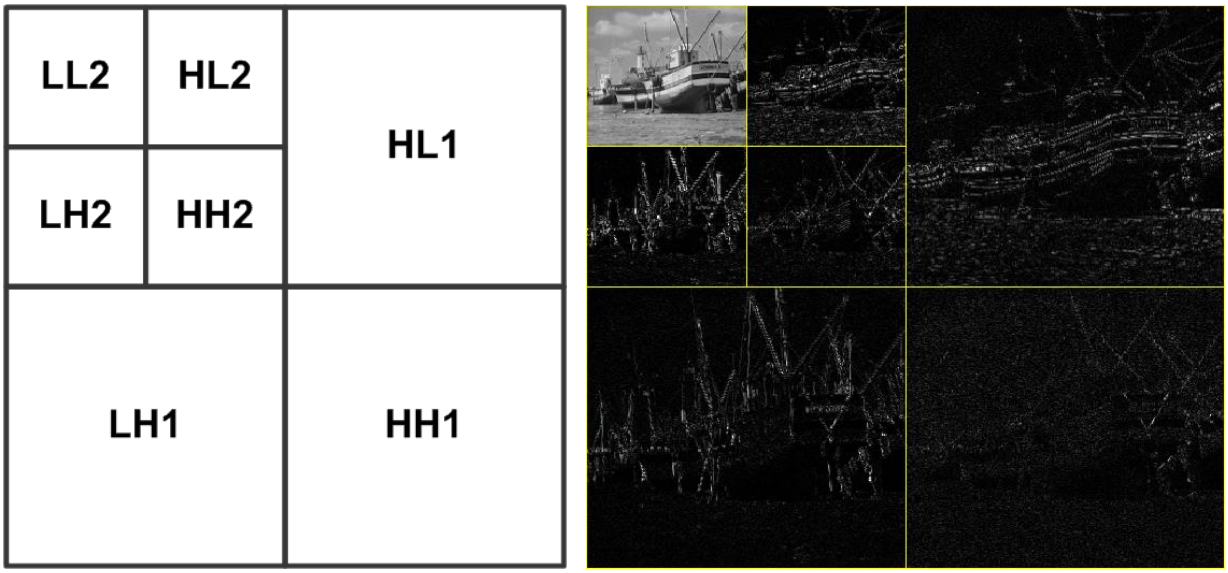


Figure 2.7 LEFT: Sub-bands of the 2D orthogonal wavelet transform, RIGHT: Coefficients of the wavelet decomposition of a sample image

In using the DWT, the signal is both high pass filtered and low pass filtered using two quadrature mirror filters. The results are downsampled by 2 to form the first level coefficients. This procedure can be repeated on the low-passed part of the signal to form the next coarser level and so on. The process is illustrated in Figure 2.6. For 2D signals this process is executed on the rows and columns separately, thus each level contains the high frequency horizontal, vertical, and diagonal details separately as shown in Figure 2.7.

For noisy images contaminated by high frequency noise such as additive white Gaussian noise, the wavelet domain coefficients representing noise are mostly contained in the finer scales. Those with the largest magnitudes have large SNR and mostly represent the image structures. The rest are mostly due to noise. Applying a hard or soft thresholding scheme to the coefficients and inverse-wavelet transforming them will result in an image with decreased noise and well preserved edges.

Selecting the threshold level to be used is difficult. Having a low noise tolerance may result in destroying the actual structures and ending up with a blurry image. A high tolerance may leave a lot of noise behind. Selecting a global threshold is also suboptimal since different image areas might have different amounts of noise.

2.3.3.2. Diffusion

One of the simplest denoising methods is Gaussian filtering which convolves the image $I(x, y)$ with a Gaussian kernel $G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$:

$$(I * G)(x, y) = \sum_u \sum_v I(x - u, y - v)G(u, v) \tag{2.6}$$

This is known as a linear isotropic filtering process. The parameter σ determines the range of the filter and subsequently the amount of smoothing. Based on a diffusion process, filtering the image several times by various values of σ produces a scale space for the image composed of successively more blurred versions of the image. Gaussian filtering satisfies the Laplace equation

$$\frac{\partial I(x, y, \sigma)}{\partial \sigma} = \Delta I(x, y, \sigma) = \frac{\partial^2 I(x, y, \sigma)}{\partial x^2} + \frac{\partial^2 I(x, y, \sigma)}{\partial y^2} \tag{2.7}$$

This equation, also known as isotropic diffusion, spreads out (in all directions) the intensity values of the original image further and further with increasing σ. Although such a procedure is effective for smoothing local noise, it is inadequate when treating images globally because of its blurring effect on edges. As mentioned before, edge preservation is very important especially in

medical imaging because important information is contained in those regions. Therefore, we need some way of removing noise without effecting true edges.

Anisotropic diffusion replaces the linear Laplace equation with a non-linear PDE that avoids the uniform smoothing of its predecessor. Perona and Malik [7] replaced the isotropic formulation of (2.7) by the following:

$$\frac{\partial I(x,y,\sigma)}{\partial \sigma} = div(D(x,y)\nabla I(x,y,\sigma)) \tag{2.8}$$

where $D(x,y)$ is the diffusion tensor which controls the rate of diffusion. It is usually a function of the image gradient which is designed to recognize edges and stop smoothing by following a rule such that

$$D(x,y) \to 0 \quad \text{when} \quad \|\nabla I\| \to \infty$$

For example the following functions can be used:

$$D = e^{-\left(\frac{\|\nabla I\|}{\sigma}\right)^2} \quad \text{and} \quad D = (1 + (\frac{\|\nabla I\|}{\sigma})^2)^{-1} \tag{2.9}$$

where σ can be found experimentally or set to the noise variance.

Even though anisotropic diffusion is able to keep edges intact, it suffers in the actual noise reduction of flat regions where false structures due to noise exist. In such cases, this method can wrongly identify noise as edge information because of the high gradient value.

2.3.3.3. Total Variation Denoising

The basis of total variation (TV) denoising is that noise and false structures that may arise because of it have high total variation defined as

$$\int_{\Omega} \|\nabla y\| dy$$

where y is an image defined on the region Ω. The task of denoising is to reduce oscillations with high TV measures which are due to noise. This method is edge preserving and smoothing only occurs in areas where uncorrelated details exist. The goal is the following minimization:

$$\min_{y} \int_{\Omega} \|\nabla y\| \, dy \qquad subject\ to \qquad \|x - y\|_2 \leq \varepsilon \tag{2.10}$$

where y is the noiseless image we want, x is the noisy image and ε is an error tolerance. The unconstrained form of this formulation in discrete space is

$$\min_{y} \frac{1}{2} \sum_{ij} [y(i,j) - x(i,j)]^2 + \lambda \left[\sum_{ij} \sqrt{|y(i+1,j) - y(i,j)|^2 + |y(i,j+1) - y(i,j)|^2} \right] \tag{2.11}$$

where λ is a regularization parameter. The first term is there to enforce closeness to the input and the second term is the total variation. In this algorithm λ is very important and determines the amount of smoothing that is allowed. Setting it to zero makes the output equal to the input and as it is increased so is the aggressiveness of the algorithm. Some of the algorithms to solve this optimization problem include [19] and [39].

Total variation denoising algorithms are effective in some cases, but the power of this type of minimization is still limited. The TV constraint is global, which makes it unable to directly reflect structures of an object. In addition, it cannot distinguish true structures and

those due to noise. Subsequently, images denoised using TV minimization may lose some fine features and in very noisy cases generate a blocky appearance.

2.3.3.4. Image Denoising Using Sparse Representation And Dictionary Learning

The noise reduction methods discussed so far have been limited in their ability to adapt to the specific signal they consider. With dictionary learning the aim is to create a set of primitive vectors called atoms that are adapted to a specific signal. By selecting only a few atoms and adding those together the original signal can be reconstructed thus the phrase sparse representation. By setting the error tolerance of the reconstruction to the noise variance we can find an approximation of the signal that is free of noise.

An image denoising methodology based on K-SVD dictionary learning was developed in [9]. It was shown that to ease the computational difficulty, an image can be divided into small patches. Then the dictionary is learned from the patches in an iterative manner. Finally each patch of the image is reconstructed by a few dictionary atoms and the patches are combined to form a whole image. The details of this process are examined in section 3.3.2.1.

Chapter 3. Sparse Representation and Dictionary Learning

Sparse representation is a way of reducing natural or artificial observations into their elemental constituents. Often these signals have a much more concise representation in a domain other than spatial or temporal where they usually appear. For tasks such as compression or analysis, it is often more efficient and meaningful to transform the signal to another domain or find its sparse representation among a set of basic signals, called atoms, that form a dictionary.

Analytical dictionaries have precise definitions that make them convenient in some situations but they are rigid and inefficient in general. In the pursuit of more adaptability,

sparse coding methods such as [40] and [41] were introduced to allow atoms from the combination of various dictionaries to be selected and added to represent signals. Following this paradigm shift, Olshausen and Field [42] were among the first to propose a way to train a dictionary on examples related to a desired signal. Others such as [43] and [44] soon followed and the K-SVD [14] dictionary learning algorithm is widely used today. The following sections discuss the topics of analytical dictionaries, sparse representation, and dictionary learning in more detail.

3.1. Transforms and Dictionaries

Early in the modern signal processing history, mathematical transforms became prominent tools for tasks such as compression and analysis. Amongst these, the Fourier transform is one of the simplest and most widely known. The introduction of the Fast Fourier Transform (FFT) [45] algorithm made it easy and efficient to implement which increased its popularity. The Fourier transform aims to represent a signal as the summation of orthogonal sinusoidal waveforms and describes it in terms of its global frequency content. The K lowest frequency waveforms are added together to approximate the signal which makes it efficient at dealing with smooth signals. However, in reconstructing discontinuities K needs to be much larger creating a trade-off between efficiency and blurring of edges.

In most practical applications, signals are finite and the Fourier transform implicitly assumes a periodic extension of the signal. This introduces a discontinuity at the boundaries. The Discrete Cosine Transform (DCT) is similar to the Fourier transform but assumes an anti-

symmetric extension of the signal, resulting in continuous boundaries. This makes it a more efficient approximation. As an added benefit, the DCT produces non-complex coefficients. Therefore it is typically preferred in practice.

Over time, more sophisticated transforms were proposed that would allow better descriptions of signals. One issue that was investigated was to allow more localization. This increases efficiency because representative functions more suited to the local characteristics of the signal can be used. The Short Time Fourier Transform (STFT) [46] was created with this in mind. The signal is first multiplied by a window function and then its Fourier transform is calculated. This process can be repeated with a moving window which will result in a time-frequency (or space-frequency) description of the signal (Figure 3.1). This transform allows us to analyze the local properties of a signal.

With the realization that natural signals (especially images) have details at many scales came the introduction of multi-resolution transforms. Among those, the wavelet transform is very well-known. It can represent a signal as a series of dilated and translated versions of a single function called the mother wavelet. Figure 3.1 shows a comparison of the STFT and the

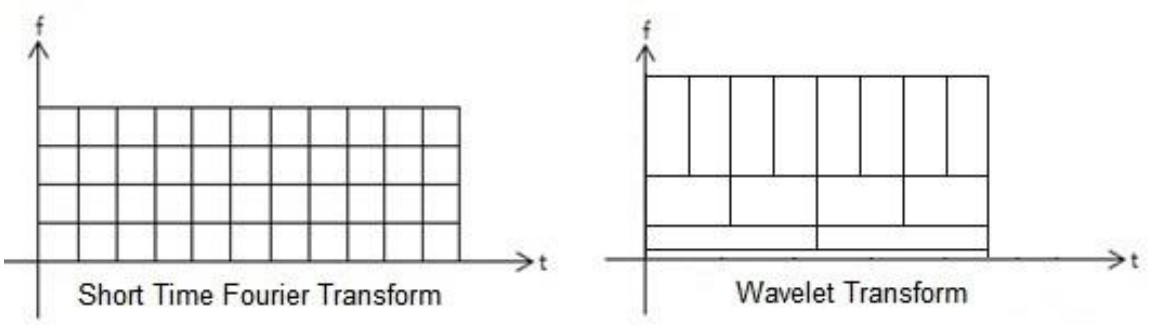


Figure 3.1 Time (or space) resolution at different frequency bands for the short-time Fourier transform and the wavelet transform

wavelet transform in terms of their time-frequency relationships. The STFT has a fixed frequency resolution for a given time window. The wavelet transform can analyze high frequency phenomena with a great time resolution while having excellent frequency resolution for low frequency events.

For all its strengths, the wavelet transform is not optimal for representing images or higher dimensional signals. Towards the end of the 20th century, new transforms were being developed such as the wedgelet [47] and the ridgelet [48] transforms. These efforts ultimately led to the powerful curvelet transform [49] [11] which can represent 2D piecewise smooth functions with curve discontinuities at an optimal rate.

All the transforms discussed can be formulated as dictionaries. A dictionary is composed of a collection of waveforms (called atoms), $\emptyset_n$, and a signal x can be expressed as the superposition of those waveforms scaled by α_n:

$$x = \sum_n \alpha_n \emptyset_n \tag{3.1}$$

In the simplest case the waveforms form a basis and all of them can be combined to uniquely represent every signal. For many years, orthogonal and bi-orthogonal dictionaries were used because of their simplicity. However, they are limited in their representation abilities. This can be seen by considering the principal component analysis (PCA) and the independent component analysis (ICA) algorithms. PCA decomposes a signal into a set of orthogonal functions called principal components which are the eigenvectors of the data covariance matrix. The first principal component is in the direction of greatest variance of the data, the second one in the direction of second-greatest variance, and so on. ICA tries to find the underlying independent signals that produce an observation. The orthogonal constraint that

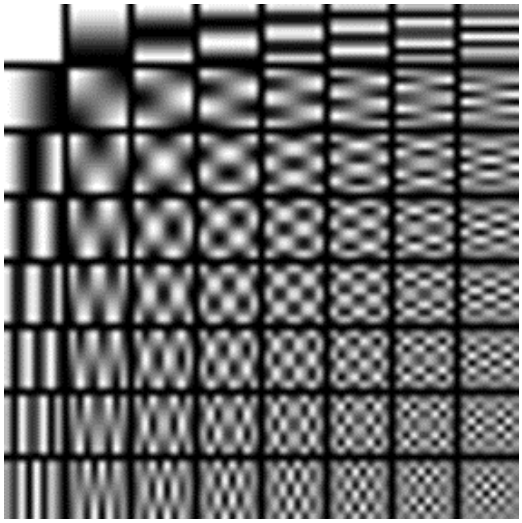


Figure 3.2 Orthogonal DCT dictionary with 64 atoms of size 8x8

exists in these methods limits their expressiveness because they assume the number of causes of an observation is limited to its dimension. This sparked the creation of algorithms that allow the use of overcomplete dictionaries with the number of atoms larger than the dimension of the signal.

The details of the most common analytical dictionaries are illustrated in the following sections.

3.1.1. Discrete Cosine Transform (DCT)

The Discrete Cosine Transform (DCT) represents a signal as the superposition of cosine waveforms with different frequencies. It is similar to the Discrete Fourier Transform (DFT) but deals only with real numbers. More importantly, the DCT is more efficient at representing finite signals. The reason is that the Fourier transform implicitly assumes a periodic extension of a signal which produces discontinuities at the boundaries for most signals. Conversely, the DCT

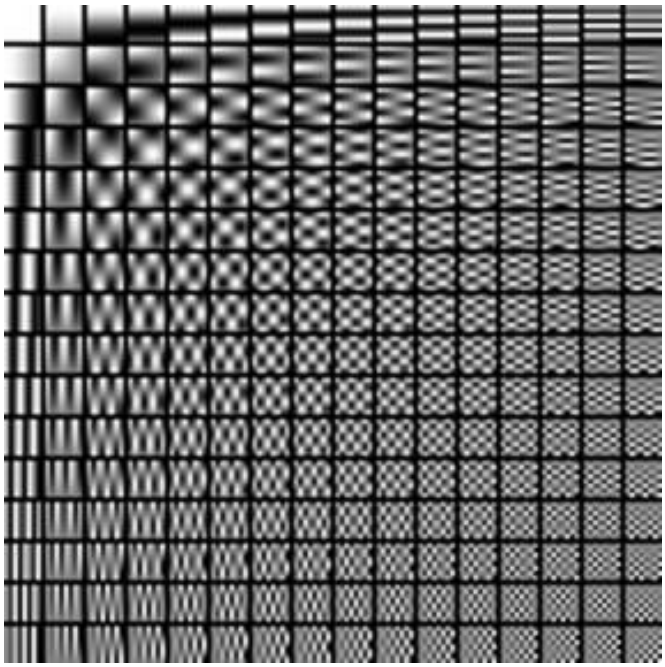


Figure 3.3 Overcomplete DCT dictionary with 256 atoms of size 8x8

assumes an anti-symmetric extension to the signal. This leads to more sinusoids required to represent a signal with DFT than is the case with DCT. The two-dimensional DCT transform of a signal x with dimensions M and N is:

$$A(p,q) = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m,n) \cos\frac{(2m+1)\pi p}{2M} \cos\frac{(2n+1)\pi q}{2N}, \quad \begin{matrix} 0 \le p \le M-1 \\ 0 \le q \le N-1 \end{matrix} \tag{3.2}$$

$$\alpha_p = \begin{cases} 1/\sqrt{M}, & p = 0 \\ \sqrt{2/M}, & 1 \le p \le M-1 \end{cases} \qquad \alpha_q = \begin{cases} 1/\sqrt{N}, & q = 0 \\ \sqrt{2/N}, & 1 \le q \le N-1 \end{cases}$$

The inverse DCT transform is used to reconstruct the signal:

$$x(m,n) = \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} \alpha_p \alpha_q A(p,q) \cos\frac{(2m+1)\pi p}{2M} \cos\frac{(2n+1)\pi q}{2N}, \quad \begin{matrix} 0 \le p \le M-1 \\ 0 \le q \le N-1 \end{matrix} \tag{3.3}$$

This reconstruction is in the form of (3.1) where the basis functions $\alpha_p \alpha_q \cos\frac{(2m+1)\pi p}{2M} \cos\frac{(2n+1)\pi q}{2N}$ form a dictionary and are weighted by the coefficients $A(p,q)$. If

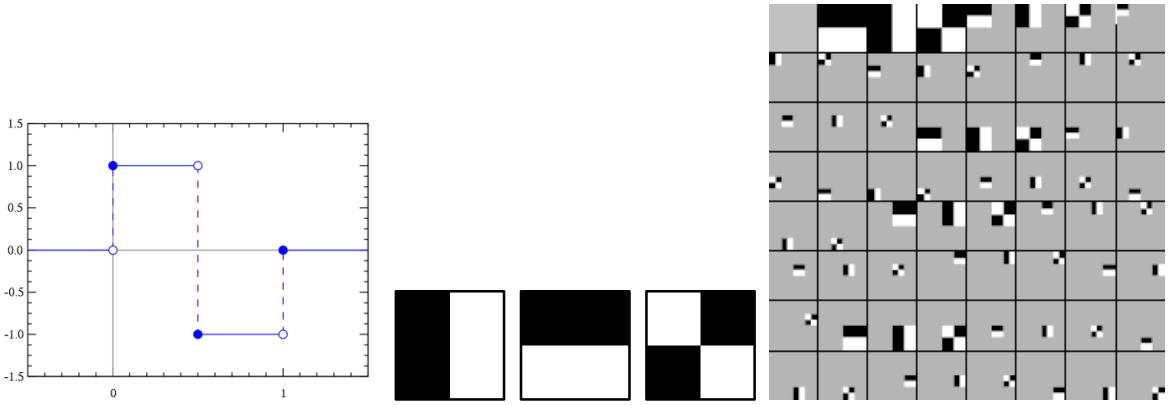


Figure 3.4 LEFT: 1D Haar wavelet, CENTRE: Three configurations of the Haar wavelet in 2D; black is negative and white is positive, RIGHT: Orthogonal Haar dictionary made up of dilations and translations of the Haar wavelets

p and q are integers then the number of atoms equals the size of the signal ($M \times N$) and the dictionary is orthogonal. An example is shown in Figure 3.2.

Because of the simplicity, orthogonal dictionaries were often used in the past, but relaxing this constraint allows sparser representations for signals. This is achieved by allowing non-integer values for p and q thereby increasing the number of atoms beyond the size of the signal. An example of such an overcomplete dictionary is displayed in Figure 3.3.

3.1.2. Wavelet Transform

As previously discussed in section 2.3.3.1, the wavelet transform can be calculated by

$$C(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} f(x)\, \psi^*\left(\frac{x-b}{a}\right) dx \tag{3.4}$$

The basic wavelet ψ is designed to be reversible and computationally efficient. In practice, the translation and scaling parameters are discretized as $a = a_0^m, b = nb_0a_0^m$ where $m, n \in \mathbb{Z}$ and $a_0 > 1, b_0 > 0$. In this case, the Discrete Wavelet Transform (DWT) becomes

$$C(m,n) = \frac{1}{\sqrt{a_0^m}} \int_{-\infty}^{\infty} f(x)\, \psi^* \left(\frac{x - n b_0 a_0^m}{a_0^m} \right) dx \tag{3.5}$$

The signal f can be reconstructed by summing the weighted wavelets:

$$f = \sum_{m,n} C_{m,n}\, \psi_{m,n} \tag{3.6}$$

Commonly the following values are used $a_0 = 2, b_0 = 1$. There are choices of ψ where the collection of wavelets $\psi_{m,n}$ creates an orthonormal basis in which case the wavelets are critically sampled in each scale to exactly span the new detail introduced at that scale. The simplest such wavelet is the Haar wavelet (Figure 3.4). Various other wavelets have been designed by Stromberg [50], Meyer [51], Daubechies [52], and others. In higher dimensions, the DWT is just a separable one-dimensional transform. Therefore for an image, first the columns then the rows are individually transformed by the same methodology as any 1D signal. This makes the DWT translation and rotation sensitive in higher dimensions [53]. To overcome this issue, the Stationary Wavelet Transform (SWT) was introduced by Beylkin [54] which abandons orthogonality in favour of overcompleteness. This is achieved by eliminating the sub-sampling and gathering all translations of the wavelet atoms.

3.1.3. Curvelet Transform

Similar to the wavelet transform, curvelets are waveforms at different scales and locations but with the addition of an orientation parameter. As can be seen in Figure 3.5 curvelets have specific anisotropic support which follows a parabolic scaling law $width \sim length^2$. This is useful for the efficient representation of smooth curves.

Like the wavelet transform, the curvelet transform of a continuous signal $f(x)$ can be calculated by the inner product of the signal and the curvelet function $\varphi(x)$:

$$C(j,l,k) = \int f(x)\, \varphi_{j,l,k}{}^{*}(x) dx \tag{3.7}$$

where *j*, *l*, and *k* are variables of scale, direction and position. Given the basic curvelet $\varphi_{j,0,0}$, the family of curvelet functions is provided by

$$\varphi_{j,l,k}(x) = \varphi_{j,0,0}\left(R_{\theta_{j,l}}\left(x - b_k^{j,l}\right)\right), \qquad j \in \mathbb{N}_0 \tag{3.8}$$

where $R_{\theta_{j,l}}$ indicates the rotation matrix with angle $\theta_{j,l} = \frac{1}{2}\pi l 2^{-\lfloor j/2 \rfloor}, l = \{0,1,\ldots | 0 \le \theta_l < 2\pi\}$ and $b_k^{j,l} = b_{k_1,k_2}^{j,l} = R_{\theta_{j,l}}^{-1}\left(k_1 2^{-j}, k_2 2^{-\frac{j}{2}}\right), k_1, k_2 \in \mathbb{Z}$ indicates the position.

The Fast Discrete Curvelet Transform (FDCT) [11] is able to find the representation of digital signals.

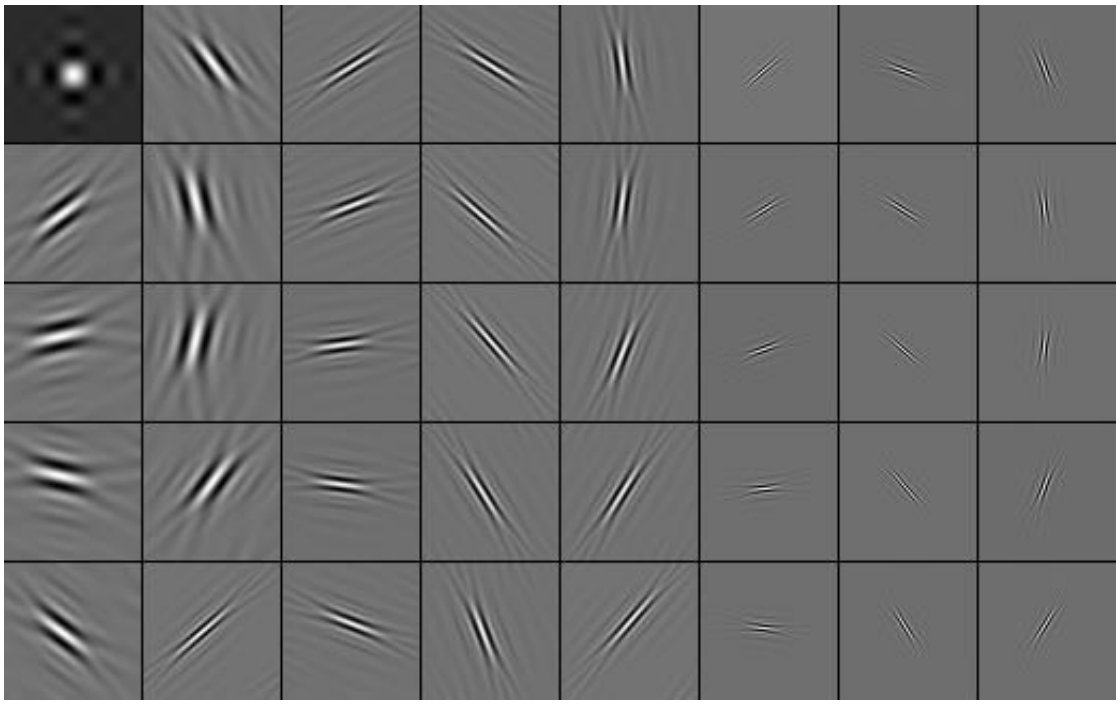


Figure 3.5 A few curvelet atoms at different scales, and orientations

3.2. Sparse Representation

A signal can have more than one optimal representation among the many transforms that exist. The idea to select the best atoms from a set of different analytical dictionaries took shape in such works as [40] and [41] that introduced signal decomposition by matching pursuit and basis pursuit, respectively.

Consider $D \in \mathbb{R}^{N\times K}$ to be an overcomplete ($N \ll K$) dictionary made up of normalized atoms in the form of $\mathbb{R}^{N\times 1}$ column vectors. Then the problem of sparse representation is to find a sparse vector with very few non-zero elements to represent a signal $x \in \mathbb{R}^{N\times 1}$ using a few dictionary atoms. Put formally, vector $\alpha \in \mathbb{R}^{K\times 1}$ is sparse when $\|\alpha\|_0 \ll K$ and the optimization problem is

$$\min_{\alpha} \|\alpha\|_0 \qquad subject\ to \qquad x = D\alpha \tag{3.9}$$

where $\|.\|_0$ is the l_0 norm which is simply a count of the non-zero entries. In general, finding an exact representation of a signal is not feasible which might be due to the presence of noise. Therefore (3.9) is relaxed to allow some error tolerance $\varepsilon \geq 0$:

$$\min_{\alpha} \|\alpha\|_0 \qquad subject\ to \qquad \|x - D\alpha\|_2 \leq \varepsilon \tag{3.10}$$

where the l_2 norm indicates the presence of Gaussian noise. Alternative loss functions can be used for other types of noise. The problem (3.10) seeks the sparsest representation vector given the constraint. An equivalent reformulation is to seek the vector that results in the least error given a sparsity tolerance $L \geq 1$:

$$\min_{\alpha} \|x - D\alpha\|_2 \qquad subject\ to \qquad \|\alpha\|_0 \leq L \tag{3.11}$$

It is also possible to consider the square of the l_2 norm such that $\|x - D\alpha\|_2^2 \leq \varepsilon^2$. Then with an appropriate Lagrange multiplier $\lambda(\varepsilon)$, the above problems can be written as an unconstrained minimization:

$$\min_{\alpha} \frac{1}{2}\|x - D\alpha\|_2^2 + \lambda\|\alpha\|_0 \tag{3.12}$$

Solving this minimization problem is NP-hard and exact solutions cannot be obtained. Nevertheless algorithms exist that find suboptimal solutions in reasonable time. These methods mainly fit into two categories: greedy pursuit and convex relaxation. Greedy algorithms select the locally optimal choice in an iterative process with the hope of reaching a global optimum. By adding or refining a set of selected atoms, they try to minimize the error between the approximated signal and the original, and gradually increase the precision of the estimation. Examples include the matching pursuit (MP) [40] and the orthogonal matching pursuit (OMP) [55].

The l_0 norm is the limit of *p*-norms as *p* approaches zero, but it is not a true norm and is non-convex. Replacing it by the l_1 norm makes the problem convex and allows us to take advantage of the powerful tools that exist for these types of optimization problems. Then (3.12) becomes:

$$\min_{\alpha} \frac{1}{2}\|x - D\alpha\|_2^2 + \lambda\|\alpha\|_1 \tag{3.13}$$

The optimization principles of basis pursuit denoising [41] and Least Absolute Shrinkage and Selection Operator (LASSO) [56] are closely related to this problem. Some algorithms that can be used to solve it include interior point, simplex, homotopy, and gradient descent methods.

3.2.1. Greedy Algorithms

Greedy algorithms for sparse representation are pursuit methods. Starting from an estimate of the sparse vector, they iteratively refine it by changing the set of selected atoms and their weights to decrease the signal approximation error. These types of algorithms abandon the brute force approach of exhaustively searching every possible subset of the dictionary. Instead they pick locally optimal solutions. This means selecting the atom, in each iteration, that decreases the reconstruction error the most. Therefore they are able to find one of the many possible solutions in a timely manner.

3.2.1.1. Matching Pursuit

Matching pursuit was introduced in [40] to find time-frequency representations of signals from a dictionary of Gabor functions. Given any redundant dictionary, this algorithm selects atoms iteratively to approximate a signal as closely as needed. First, it selects the atom that produces the maximum inner product with the residual vector (which is just the signal at the start) and calculates the weight attributed to that atom. Then it subtracts the contribution of this atom from the residual vector. The algorithm iterates these steps until the l_2 norm of the residual is equal or less than the required error. Alternatively, the stopping criterion can be the maximum number of atoms to represent the signal. Table 3.1 illustrates the details of the algorithm.

Table 3.1 Matching Pursuit Pseudo-Algorithm

Goal: approximate the solution of $\min_{\alpha}\|\alpha\|_0 \ subject\ to\ \|x - D\alpha\|_2 \leq \varepsilon$

Input: dictionary $D = \{d_k | \ k = 1,2, \dots, K\}$, signal x, error tolerance ε

Initialization:

- iteration number: $i = 0$
- initial vector: $a^0 = 0$
- residual: $r^0 = x - D\alpha^0 = x$
- set of selected atoms: $S^0 = \{\ \}$

Iteration:

- $i = i + 1$
- Calculate error $e(k) = \left\|d_k z_k - r^{i-1}\right\|_2^2, \ \forall k$ using the optimal solution $z_k = \frac{d_k^T r^{i-1}}{\|d_k\|_2^2}$
- Find the minimum k_0, of $e(k)\ \forall k \notin S^{i-1}$, such that $e(k_0) \leq e(k)$
- Add the index of the new atom to the set: $S^i = S^{i-1} \cup \{k_0\}$
- Update atom weight $\alpha^i(k_0) = \alpha^{i-1}(k_0) + z_{k_0}$
- Update residual $r^i = x - D\alpha^i$
- Stop if $\left\|r^i\right\|_2 \leq \varepsilon$, otherwise apply another iteration

Output: $\alpha = \alpha^i$

3.2.1.2. Weak Matching Pursuit

The Weak Matching Pursuit is a computationally more efficient modification of the original algorithm that allows for suboptimal choices of the dictionary atoms. Rather than looking for the atom that maximizes the inner product with the residual, the algorithm settles on the first atom that is a factor t away from the optimal choice. Using the Cauchy-Schwartz inequality we have

$$\frac{(d_k^T r^{i-1})^2}{\|d_k\|_2^2} \leq \max_{1 \leq k \leq m} \frac{(d_k^T r^{i-1})^2}{\|d_k\|_2^2} \leq \left\|r^{i-1}\right\|_2^2 \tag{3.14}$$

Table 3.2 Weak Matching Pursuit Pseudo-Algorithm

Goal: approximate the solution of $\min_{\alpha}\|\alpha\|_0 \ subject\ to\ \|x - D\alpha\|_2 \leq \varepsilon$

Input: dictionary $D = \{d_k | \ k = 1,2, \dots, K\}$, signal x, error tolerance ε, scalar $0 < t < 1$

Initialization:

- iteration number: $i = 0$
- initial vector: $a^0 = 0$
- residual: $r^0 = x - D\alpha^0 = x$
- set of selected atoms: $S^0 = \{\ \}$

Iteration:

- $i = i + 1$
- Calculate error $e(k) = \left\|d_k z_k - r^{i-1}\right\|_2^2$ using the optimal solution $z_k = \frac{d_k^T r^{i-1}}{\|d_k\|_2^2}$ until the following is satisfied: $\frac{(d_{k_0}^T r^{i-1})}{\|d_{k_0}\|_2} \geq t \left\|r^{i-1}\right\|_2$
- Find the minimum k_0, of $e(k)\ \forall k \notin S^{i-1}$, such that $e(k_0) \leq e(k)$
- Add the index of the new atom to the set: $S^i = S^{i-1} \cup \{k_0\}$
- Update atom weight $\alpha^i(k_0) = \alpha^{i-1}(k_0) + z_{k_0}$
- Update residual $r^i = x - D\alpha^i$
- Stop if $\left\|r^i\right\|_2 \leq \varepsilon$, otherwise apply another iteration

Output: $\alpha = \alpha^i$

which sets an upper bound on the maximum possible inner product. Therefore, by calculating $\left\|r^{i-1}\right\|_2^2$ at the beginning of each iteration and searching for the k_0 that gives the smallest error $e(k)$, we can select the first one that gives the following:

$$\frac{(d_{k_0}^T r^{i-1})^2}{\left\|d_{k_0}\right\|_2^2} \geq t^2 \left\|r^{i-1}\right\|_2^2 \geq t^2 \max_{1 \leq k \leq m} \frac{(d_k^T r^{i-1})^2}{\|d_k\|_2^2} \tag{3.15}$$

The details are in Table 3.2.

3.2.1.3. Orthogonal Matching Pursuit (OMP)

The Orthogonal Matching Pursuit (OMP) algorithm [55] is an extension of matching pursuit which ensures the same atom is never selected twice in representing the signal. This means an *N*-dimensional vector converges after a maximum of *N* steps. To achieve this, the weight update of the selected atoms is changed. In each iteration i, a new atom is added to the selected subset S^i and all their weights α_{S^i} are updated by minimizing $\left\|x - D_{S^i}\alpha_{S^i}\right\|_2^2$. The solution can be found by setting the derivative with respect to α_{S^i} to zero:

$$D_{S^i}^T\left(x - D_{S^i}\alpha_{S^i}\right) = -D_{S^i}^T r^i = 0 \tag{3.16}$$

which suggests the atoms that are included in S^i are essentially orthogonal to the residual r^i. The details are shown in Table 3.3.

Table 3.3 Orthogonal Matching Pursuit Pseudo-Algorithm

Goal: approximate the solution of $\min_{\alpha}\|\alpha\|_0 \ subject\ to\ \|x - D\alpha\|_2 \leq \varepsilon$

Input: dictionary $D = \{d_k | \ k = 1,2, \dots, K\}$, signal x, error tolerance ε

Initialization:

- iteration number: $i = 0$
- initial vector: $a^0 = 0$
- residual: $r^0 = x - D\alpha^0 = x$
- set of selected atoms: $S^0 = \{\ \}$

Iteration:

- $i = i + 1$
- Calculate error $e(k) = \left\|d_k z_k - r^{i-1}\right\|_2^2, \ \forall k$ using the optimal solution $z_k = \frac{d_k^T r^{i-1}}{\|d_k\|_2^2}$
- Find the minimum k_0, of $e(k)\ \forall k \notin S^{i-1}$, such that $e(k_0) \leq e(k)$
- Add the index of the new atom to the set: $S^i = S^{i-1} \cup \{k_0\}$
- Update the vector $\alpha^i = \min_{\alpha_{S^i}}\left\|x - D_{S^i}\alpha_{S^i}\right\|_2^2 \ subject\ to\ S^i$
- Update residual $r^i = x - D\alpha^i$
- Stop if $\|r\|_2 \leq \varepsilon$, otherwise apply another iteration

Output: $\alpha = \alpha^i$

3.3. Dictionary Learning

The dictionaries discussed in previous sections were based on precise mathematical functions that are supported by proofs of their optimality and error bounds, as well as fast implicit implementations. However, these analytical dictionaries are too generic and lack enough flexibility in optimal representation of complicated signals. An adaptive dictionary trained from examples close to a signal can allow better and more efficient representations of that signal. Figure 3.6 shows a comparison of learned and analytical dictionaries to illustrate this point. Olshausen and Field [42] were among the first to work on a methodology for signal-adapted dictionaries. The dictionary learning method they developed was based on maximum likelihood estimation.

Given the generative model $x = D\alpha$ for signal $x \in \mathbb{R}^{N\times 1}$, with $D \in \mathbb{R}^{N\times K}$ an overcomplete dictionary and $\alpha \in \mathbb{R}^{K\times 1}$ a sparse vector, the goal of the ML learning method is to maximize the likelihood that x has a sparse representation in D:

$$D = \max_{D} [\log P(x|D)] = \max_{D} \left[\log \int P(x|a, D)P(a)da\right] \tag{3.17}$$

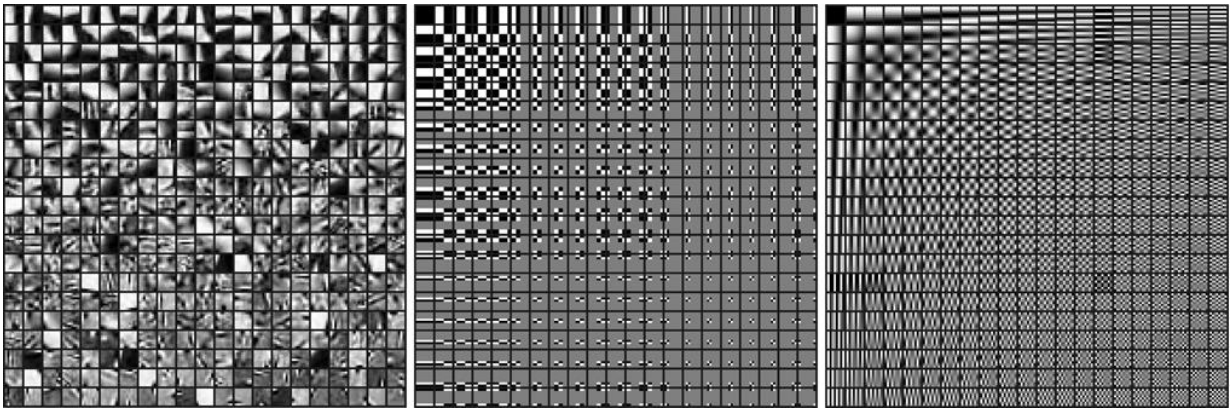


Figure 3.6 Dictionary learned by the K-SVD algorithm and compared to analytical dictionaries [14], LEFT: Learned dictionary, CENTRE: Overcomplete Haar dictionary, RIGHT: Overcomplete DCT dictionary

In general this computation is very difficult. To simplify it, certain constraints have to be put in place such as the assumption that the distribution of $P(a)$ is Laplacian and that the approximation noise error can be modeled as zero-mean Gaussian noise. Other probabilistic dictionary learning methods have been proposed that follow the maximum likelihood [57] or the maximum a posteriori (MAP) [44] frameworks.

The task of dictionary learning is to find a dictionary as well as its corresponding sparse vector to represent a given signal. Put formally, solve

$$\{D, \alpha_i\} = \min_{D,\alpha_i} \sum_{i \in I} \lambda_i \|\alpha_i\|_0 + \|D\alpha_i - x_i\|_2^2 \tag{3.18}$$

where the signal has been split into I smaller sections. For images this is accomplished by breaking it down into overlapping patches of size $n \times n$ with $n = 8$ being a common choice. This is because of the enormous computational cost of operating on large signals. Solving (3.18) is a 2-step iterative process. The first step is to fix D and find sparse representations with expressions synonymous to (3.10) and (3.11) for every image patch $i \in I$:

$$\alpha_i = \min_{\alpha_i} \|\alpha_i\|_0 \quad subject\ to \quad \|D\alpha_i - x_i\|_2^2 \leq \varepsilon \quad \forall i \tag{3.19}$$

$$\alpha_i = \min_{\alpha_i} \|D\alpha_i - x_i\|_2^2 \quad subject\ to \quad \|\alpha_i\|_0 \leq L \quad \forall i \tag{3.20}$$

All the sparse coding methods previously discussed are applicable here.

The second step is to fix α_i for all $i \in I$ and update the dictionary D. Several methods exist to accomplish this. Olshausen and Field used gradient descent. Two other ways of learning the dictionary are the method of optimal directions (MOD) [43] and K-SVD [14] which will be discussed in the following sections.

3.3.1. Method Of Optimal Directions (MOD)

The MOD algorithm [43] follows the 2-step dictionary learning process to solve (3.18). Engan *et al*. used OMP for the sparse representation step, and proposed a closed form solution for the dictionary update step based on the least squares method. Consider the representation mean square error $e_i = \|D\alpha_i - x_i\|_2^2$ for all $i \in I$ expressed as a matrix with each e_i as a column:

$$\|E\|_2^2 = \sum_i \|e_i\|_2^2 = \|DA - X\|_2^2 \tag{3.21}$$

Assuming A is kept constant, we can find a new D to minimize this error by setting the derivative with respect to D to zero: $(DA - X)X^T = 0$. The solution is obtained using the Moore-Penrose pseudo-inverse:

$$D_{new} = XA^T(AA^T)^{-1} \tag{3.22}$$

The MOD algorithm typically converges after a few iterations; however it is hampered by the relative high complexity of the matrix inversion.

3.3.2. K-SVD Algorithm

The K-SVD algorithm [14] is designed to be a natural generalization of K-means clustering. K-means aims to assign Q observations into $K \ll Q$ sets and is a 2-step algorithm similar to dictionary learning. First the observations are grouped into K sets such that their l_2 distance to a given centroid is minimal. This can be considered the extreme case of sparse coding where each observation (signal) vector is only allowed to be represented by a single atom with a

weight of 1. In the second step, the centroids are updated such that the overall distance in each set of observations is minimized. This is synonymous to updating the dictionary.

Similar to dictionary learning algorithms discussed before, K-SVD tries to solve (3.18) by iterating between sparse representation and dictionary updating. The former can be done by any pursuit algorithm and OMP is a common choice because of its simplicity and fast execution. The latter is where the main contribution of K-SVD lies. The dictionary atoms along with their weights are updated one at a time making use of SVD decomposition to minimize approximation error. This greatly reduces computation time and complexity. For a given column (atom) d_k and its corresponding row τ_k in the sparse matrix A, the approximation error (3.21) can be written as:

$$\|X - DA\|_2^2 = \left\| X - \sum_{m=1}^{K} d_m \tau_m \right\|_2^2 = \left\| \left(X - \sum_{m \neq k} d_m \tau_m \right) - d_k \tau_k \right\|_2^2 = \|E_k - d_k \tau_k\|_2^2 \qquad (3.23)$$

The next step is to use SVD to decompose E_k and find alternative d_k and τ_k. However, this will result in filling τ_k with non-zero entries. Therefore to avoid losing sparsity, we need to select the subspace of τ_k that contains non-zero elements. The equivalent term to be minimized is denoted as:

$$\left\| E_k^{\mathcal{R}} - d_k \tau_k^{\mathcal{R}} \right\|_2^2 \qquad (3.24)$$

Performing the SVD decomposition on $E_k^{\mathcal{R}}$ gives us $E_k^{\mathcal{R}} = U \Delta V^T$. The solution to $\hat{d}_k$ becomes the first column of U, and the solution to $\hat{\tau}_k^{\mathcal{R}}$ becomes the first column of V multiplied by $\Delta(1,1)$. Now the entries in $\hat{\tau}_k^{\mathcal{R}}$ can replace their corresponding values in τ_k. More details of the K-SVD algorithm are provided in Table 3.4.

Table 3.4 K-SVD Pseudo-Algorithm

Goal: find a dictionary D to sparsely represent signal $X = \{x_i \in \mathbb{R}^{N\times 1} | i = 1,2,\dots,I\}$ by solving

$$\min_{D,A} \|DA - X\|_2^2 \quad subject\ to \quad \|\alpha_i\|_0 \le L \quad \forall i$$

Input: signal matrix $X \in \mathbb{R}^{N\times I}$, error tolerance ε, sparsity measure L, number of atoms K

Initialization:

- iteration number: $j = 0$
- initial dictionary: set $D^0 \in \mathbb{R}^{N\times K}$ to an overcomplete analytical dictionary or K randomly chosen columns from X and l_2 normalize each column

Iteration:

- $j = j + 1$
- Sparse Coding: find sparse vectors α_i for each signal x_i by solving the following using any pursuit algorithm:

$$\alpha_i^{j-1} = \min_{\alpha_i} \left\| D^{j-1}\alpha_i - x_i \right\|_2^2 \quad subject\ to \quad \|\alpha_i\|_0 \le L \quad \forall i$$

- Dictionary update: for each dictionary atom $\{d_k \in \mathbb{R}^{N\times 1} | k = 1,2,\dots,K\}$ in D^{j-1} follow these steps
  - Define the group of column signals that use atom d_k:

$$\Omega_k = \left\{ i \middle| 1 \le i \le I, \alpha_i^{j-1}(k) \ne 0 \right\}$$

  - Given the row τ_k in A corresponding to d_k, calculate the residual matrix E_k:

$$E_k = X - \sum_{m\ne k} d_m \tau_m$$

  - Select only the columns in E_k corresponding to Ω_k to get $E_k^{\mathcal{R}}$
  - Apply SVD decomposition $E_k^{\mathcal{R}} = U\Delta V^T$. The updated atom $\hat{d}_k$ is the first column of U. The updated coefficient vector $\hat{\tau}_k^{\mathcal{R}}$ becomes the first column of V multiplied by $\Delta(1,1)$. Now the entries in $\hat{\tau}_k^{\mathcal{R}}$ can replace their corresponding values in τ_k
- Stop if $\left\| D^j A^j - X \right\|_2^2 \le \varepsilon$, otherwise apply another iteration

Output: D

3.3.2.1. Image Denoising by K-SVD

In [9] Elad and Aharon formulated how an image can be denoised by sparse representation using a dictionary learned by K-SVD. Consider an image x corrupted with an additive white Gaussian noise n; then the noisy image is:

$$y = x + n \tag{3.25}$$

The problem of minimizing noise while sparse representation and dictionary learning occur can be written as:

$$x = \underset{x,D,\alpha_i}{min}\ \lambda\|y - x\|_2^2 + \sum_i \mu_i\|\alpha_i\|_0 + \|D\alpha_i - E_i x\|_2^2 \tag{3.26}$$

where E_i extracts patches from x. Patches are chosen to be small to ease the computational load. They are also overlapping so that blocking artifacts do not plague the reconstructed image.

Two approaches can be followed to obtain the dictionary. It can be learned from training images that are noise-free, or it can be learned from the patches of the noisy image. It was shown in [9] that the second option leads to comparable results to the first. The reason is that after each iteration, dictionary learning gradually makes the atoms approach the underlying structure of the image. As a result, noise is not learned and the underlying structure of the image can be reconstructed without noise. Solving (3.26) amounts to first fixing x and following the familiar 2-step iterative algorithm composed of:

- Sparse representation: solve (3.19) or (3.20) using any pursuit algorithm
- Dictionary updating: use K-SVD as described in Table 3.4

After the final dictionary and its sparse representation are obtained, the denoised image can be reconstructed by the following closed-form expression:

$$x = \left(\lambda I + \sum_i E_i^T E_i\right)^{-1} \left(\lambda Y + \sum_i E_i^T D\alpha_i\right) \tag{3.27}$$

This is essentially a weighted average of all the patches with some moderation introduced by averaging with the noisy image.

Chapter 4. Methodology

Conventional image denoising methods usually assume a simple noise model such as additive white Gaussian. However, noise in CT images does not have a known distribution. The projection measurements of CT scans are non-stationary [58] [59], and the reconstruction process itself affects the noise in the final image. Looking at CT images in general, the noise and the actual structures can be thought of as a texture layer and a piecewise smooth layer, respectively, superimposed on one another. Recently, the Morphological Component Analysis (MCA) [15] algorithm was introduced that aims to separate texture from the piecewise smooth (cartoon) parts of an image by sparse representation. With the advent of dictionary learning

algorithms in recent years we were motivated to combine these two techniques and decompose low-dose CT images into two morphologically distinct layers thus extracting and discarding unwanted streaks and noise from the main structures.

4.1. Morphological Component Analysis

The image decomposition problem is of the form

$$y = x_1 + x_2 \tag{4.1}$$

where x_1 and x_2 are two morphologically distinct layers that are to be extracted from $y \in \mathbb{R}^N$. The MCA model assumes that there exist two mutually incoherent overcomplete dictionaries. Each one can very sparsely represent one of those layers but results in a non-sparse solution when applied to the other layer. Formally the following assumptions hold true:

- There exists an overcomplete dictionary $D_1 \in \mathbb{R}^{N \times K_1}$ ($K_1 \gg N$) for signal x_1 such that solving $\alpha_1 = \underset{\alpha}{min} \|\alpha\|_0 \ subject\ to: x_1 = D_1\alpha$, leads to a sparse solution.
- There exists an overcomplete dictionary $D_2 \in \mathbb{R}^{N \times K_2}$ ($K_2 \gg N$) for signal x_2 such that solving $\alpha_2 = \underset{\alpha}{min} \|\alpha\|_0 \ subject\ to: x_2 = D_2\alpha$, leads to a sparse solution.
- For signal x_1, solving $\alpha_1 = \underset{\alpha}{min} \|\alpha\|_0 \ subject\ to: x_1 = D_2\alpha$, and for signal x_2 solving $\alpha_2 = \underset{\alpha}{min} \|\alpha\|_0 \ subject\ to: x_2 = D_1\alpha$, both lead to non-sparse solutions.

where $\|.\|_0$ is a count of non-zero entries. Therefore, the two dictionaries D_1 and D_2 are able to discriminate between the two morphologically distinct layers x_1 and x_2.

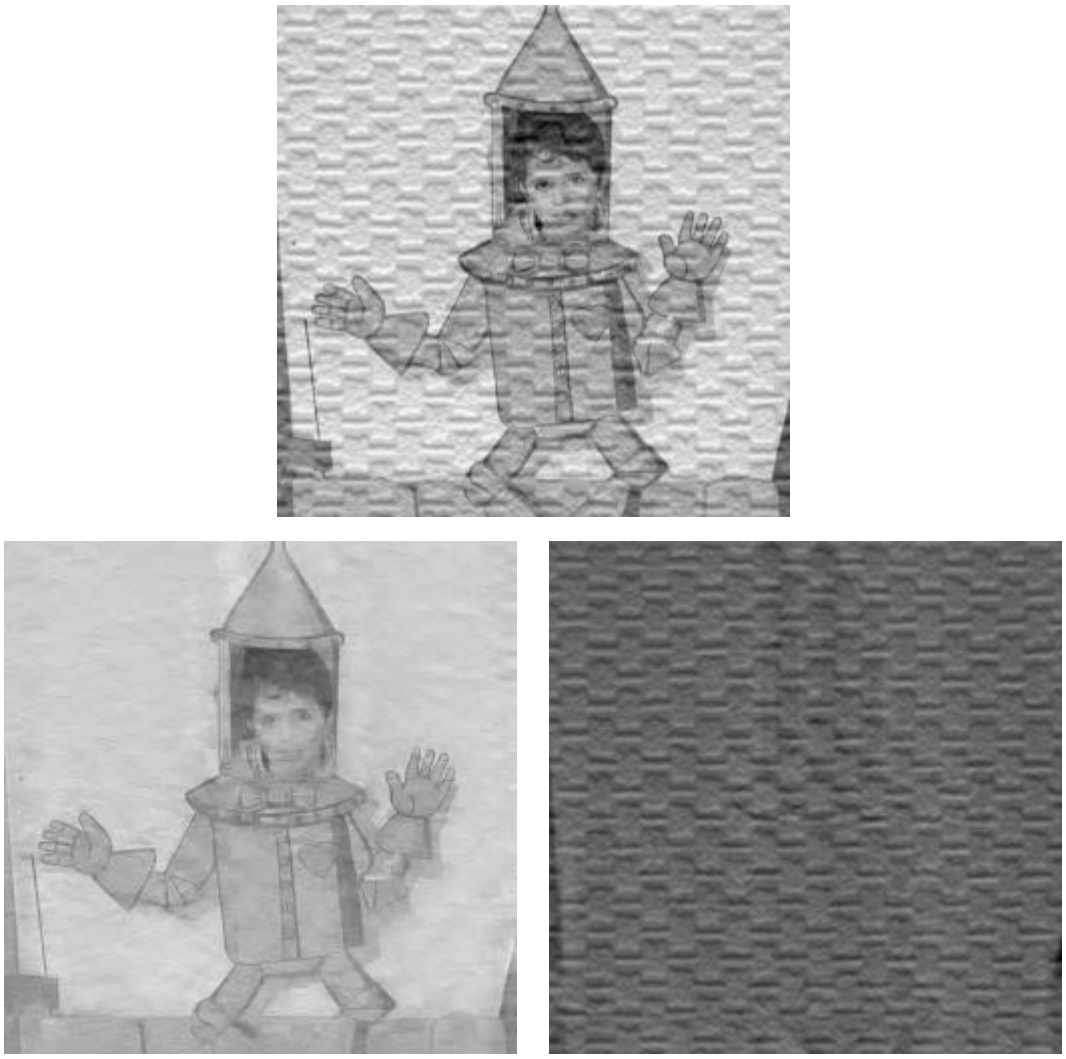


Figure 4.1 TOP: Original simulated mixture, BOTTOM LEFT: Recovered cartoon image using the curvelet transform, BOTTOM RIGHT: Separated texture part using DCT. [15]

The Original MCA algorithm called for selecting an appropriate fixed dictionary for each of x_1 and x_2. For example, we can select the overcomplete discrete cosine transform (DCT) to represent oscillatory texture content and the curvelet transform can be used to represent the piecewise smooth parts of the image as shown in Figure 4.1. For an image *y* with the dictionaries selected, the optimization task becomes

$$\{\alpha_1, \alpha_2\} = \min_{\{\alpha_1,\alpha_2\}} \|\alpha_1\|_0 + \|\alpha_2\|_0 \quad subject\ to: \quad y = D_1\alpha_1 + D_2\alpha_2 \tag{4.2}$$

The problem as it is posed in (4.2) is non-convex and hard to solve. Additionally, in general it is not possible to cleanly separate the two layers as there might be some content that does not belong to either dictionary. Fortunately the matching and pursuit algorithms are able to find an approximate solution [40] [41]. So the l_0-norm can be relaxed to an l_1-norm and the constraint is changed to a penalty:

$$\{\alpha_1, \alpha_2\} = \min_{\{\alpha_1,\alpha_2\}} \|\alpha_1\|_1 + \|\alpha_2\|_1 + \lambda\|y - D_1\alpha_1 - D_2\alpha_2\|_2^2 \tag{4.3}$$

In [15], an iterative shrinkage algorithm based on the block coordinate relaxation method was used to solve this problem.

In general, images might contain complicated textures which may not be effectively represented with any fixed dictionary. Therefore, using dictionary learning to create a set of atoms adapted to the image will result in better image separation.

4.2. Learning the Morphological Content

Peyré *et al.* [16] extended the MCA algorithm by combining fixed and learned dictionaries. They suggested using a fixed dictionary such as wavelets or curvelets to represent the piecewise smooth content and learning a dictionary based on patches of the image (manually selected) that contain mostly texture. This would allow better image separation as the learned dictionary is adapted to the texture content.

With the success of dictionary learning and its ability to adapt to any image, it makes sense to represent both the smooth and texture parts of an image by learning their content separately. Indeed this is possible as shown in [17]. The same paper also argues that given two

previously learned dictionaries, it is faster (with no noticeable degradation in the results) to use a direct degenerated block-coordinate-descent algorithm rather than an iterative one as was done previously in [15] and [16].

The task of sparse representation is usually done by the Orthogonal Matching Pursuit (OMP) algorithm which needs to operate on small patches to be computationally feasible. Consider an image broken down into I patches of size $\sqrt{N} \times \sqrt{N}$ and each patch $i \in I$ formed into a column vector of size $N \times 1$. The image decomposition problem can be formulated as the generalization of the denoising problem in [9]:

$$\min_{x_1,x_2,\alpha_i,\beta_i} \lambda\|y - x_1 - x_2\|_2^2 + \sum_i \mu_i\|\alpha_i\|_0 + \|D_1\alpha_i - E_i x_1\|_2^2 + \rho_i\|\beta_i\|_0 + \|D_2\beta_i - E_i x_2\|_2^2 \quad (4.4)$$

where E_i extracts the i^{th} patch from the image, μ and ρ are Lagrange multipliers, and λ is a regularization parameter. The first term exists to enforce input-output proximity in the presence of noise. The rest express our belief that two distinct dictionaries D_1 and D_2 can sparsely represent x_1 and x_2 respectively.

We do not have initial values for x_1 and x_2, so to solve (4.4) the following approximation is used:

$$\sum_i \|D_1\alpha_i - E_i x_1\|_2^2 + \sum_i \|D_2\beta_i - E_i x_2\|_2^2 \approx \sum_i \left\| [D_1, D_2] \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} - E_i(x_1 + x_2) \right\|_2^2 \quad (4.5)$$

which is true with uncorrelated separation errors:

$$(E_i x_1 - D_1\alpha_i)^T (E_i x_2 - D_2\beta_i) = 0 \quad (4.6)$$

Therefore we can write (4.4) as

$$\min_{x_1,x_2,\alpha_i,\beta_i} \lambda\|y - (x_1 + x_2)\|_2^2 + \sum_i \mu_i\|\alpha_i\|_0 + \rho_i\|\beta_i\|_0 + \left\| [D_1, D_2] \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} - E_i(x_1 + x_2) \right\|_2^2 \quad (4.7)$$

Figure 4.2 Main stages of the proposed algorithm

Solving this problem involves three stages. First we need to learn dictionaries D_1 and D_2 from training patches that represent piecewise smooth structures and noise respectively in a CT image. Then we can set $y = x_1 + x_2$ in (4.7) and find the sparse vectors using OMP. Finally having found the parameters to represent x_1 and x_2, we can proceed to construct them. After this process is completed, there is a need to recover some edges that may end up in the wrong layer. Figure 4.2 shows the overall profile of the proposed algorithm and Figure 4.5 presents a more detailed flowchart. The details are provided in the following sections.

4.2.1. Pre-learning of Dictionaries

To ensure successful image decomposition the two dictionaries that are to be used need to be mutually incoherent to avoid ambiguity in the sparse representation stage. The first dictionary needs to represent the noiseless structures of the CT image. We propose using a simple and fast denoising method on the given image to prepare training samples for dictionary learning. This denoising method should be edge preserving and smooth away all the noise. Even if some small details get lost at this stage, the impact on the final outcome is negligible. The purpose of this stage is to provide us with a noiseless starting point from which a dictionary

representing only the main structures can be produced. Total Variation (TV) denoising [8] is a good candidate that meets our requirements.

Using Chambolle's method [19] we solve the TV minimization for the low-dose CT image y to smooth away the noise and unwanted streaks and find $\hat{y}_s$:

$$\min_{\hat{y}_s} \frac{1}{2} \sum_{ij} [\hat{y}_s(i,j) - y(i,j)]^2 + \lambda \left[\sum_{ij} \sqrt{|\hat{y}_s(i+1,j) - \hat{y}_s(i,j)|^2 + |\hat{y}_s(i,j+1) - \hat{y}_s(i,j)|^2} \right] \quad (4.8)$$

In the algorithm, the regularization parameter λ controls the strength of the smoothing. We need to find the optimum value of λ that does not blend or destroy small structures yet is able to produce a reasonably smooth output. This is achieved experimentally by running the TV algorithm a few times with a series of linearly increasing λ values as inputs and finding the corner of the L-curve [60].

The L-curve is a plot of the l_2-norm of the solution $\|\hat{y}_s\|_2$ against the l_2-norm of the residual $\|\hat{y}_s - y\|_2$ as shown in Figure 4.3. The corner of the curve is identified as the point where the change in slope is most rapid. This can be calculated to be where the magnitude of the second derivative of $\|\hat{y}_s\|_2$ with respect to $\|\hat{y}_s - y\|_2$ is at its maximum. In the example in Figure 4.3, this point occurs at $\|\hat{y}_s - y\|_2 \cong 9 \times 10^2$ where the second derivative is furthest from zero. Therefore the corner corresponds to the value of λ that resulted in that residual value. The regularization parameter is chosen in this manner and a dictionary representing image structures is learned from the corresponding TV smoothed output.

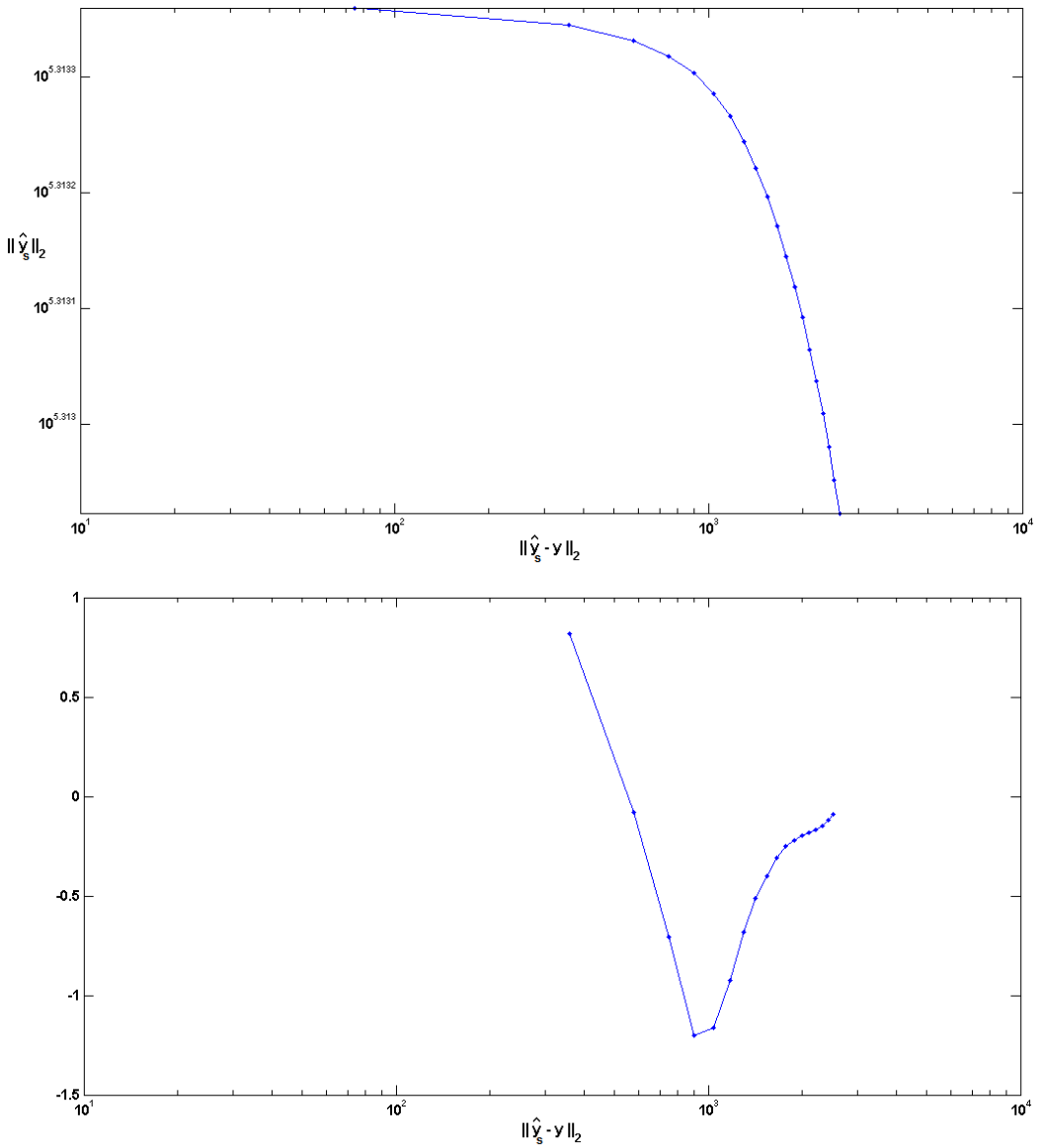


Figure 4.3 TOP: Log-log plot of the solution norm $\|\hat{y}_s\|_2$ against residual norm $\|\hat{y}_s - y\|_2$, BOTTOM: Second derivative of $\|\hat{y}_s\|_2$ with respect to $\|\hat{y}_s - y\|_2$

To learn the dictionary representing noise we subtract the TV denoised image from the original noisy one and discard patches that inevitably have leftover edges in them. What is left is a fairly accurate representation of the noise without any strong edge information. We then learn the dictionary from these patches. The next step is to find the sparse vectors.

4.2.2. Sparse Coding and Image Separation

Now that we have the dictionaries, we need to find their corresponding sparse vectors. Fixing D_1 and D_2 to the obtained dictionaries and setting $y = x_1 + x_2$ in (4.7) leads to the sparse representation expression

$$\min_{\alpha_i,\beta_i} \sum_i \left\| \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} \right\|_0 \quad s.t. \quad \left\| [D_1, D_2] \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} - E_i y \right\|_2^2 < \varepsilon \quad \forall i \tag{4.9}$$

Note that for the right choice of μ and ρ, the following equality $\left\| \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} \right\|_0 = \|\alpha_i\|_0 + \|\beta_i\|_0$ holds true [9].

This problem can be solved by the OMP algorithm the same way it is done for a single dictionary because we are essentially concatenating D_1 and D_2 together and seeking a sparse vector to represent the input image using all the atoms contained in both dictionaries. After finding the sparse vector, it is easy to split it into α and β according to which dictionary they belong to. Therefore each patch of the noisy image is approximated as closely as we want (by setting ε). The atoms from one dictionary are better at capturing structural information in each patch while atoms from the other dictionary are better at doing so for the unwanted artifacts and noise.

Since we are trying to capture all the information in a given image and send them to their respective layers (x_1 or x_2), ε should ideally be zero. However for time constraints and because there is no reason to capture all the noise, ε can be set to an estimated Gaussian noise variance.

To get x_1 and x_2, fix the dictionaries and the sparse vectors in (4.4) and we have

$$\min_{x_1,x_2} \lambda\|y - x_1 - x_2\|_2^2 + \sum_i \|D_1\alpha_i - E_i x_1\|_2^2 + \sum_i \|D_2\beta_i - E_i x_2\|_2^2 \tag{4.10}$$

the solution of which amounts to the following:

$$x_1 = \left(\lambda I + \sum_i E_i^T E_i\right)^{-1} \left(\lambda\left(y - \sum_i E_i^T D_2\beta_i\right) + \sum_i E_i^T D_1\alpha_i\right) \tag{4.11}$$

$$x_2 = \left(\lambda I + \sum_i E_i^T E_i\right)^{-1} \left(\lambda\left(y - \sum_i E_i^T D_1\alpha_i\right) + \sum_i E_i^T D_2\beta_i\right) \tag{4.12}$$

Despite trying to separate only noise from the actual image, some edges will end up in the wrong layer blurring the desired layer. The reason is that the high frequency nature of noise makes it morphologically similar to the edges of actual image structures. Therefore atoms from the noise dictionary tend to represent some edge information as well as unwanted streaks. In the next section we develop an iterative procedure based on the curvelet transform to reconstruct the edges in the denoised image.

4.2.3. Iterative Edge Reconstruction Using Curvelets

In the decomposition process, it is inevitable that some edges are wrongly attributed to the noise dictionary and are sent to the noise layer. Therefore some processing is required to identify the missing edges and restore them to the structure layer. We use the curvelet transform [11] to produce a dictionary of adapted curvelet atoms and extract the edges from the noise layer. This transform is similar to the wavelet transform but better in representing curves. This property makes it useful in recovering CT image edges.

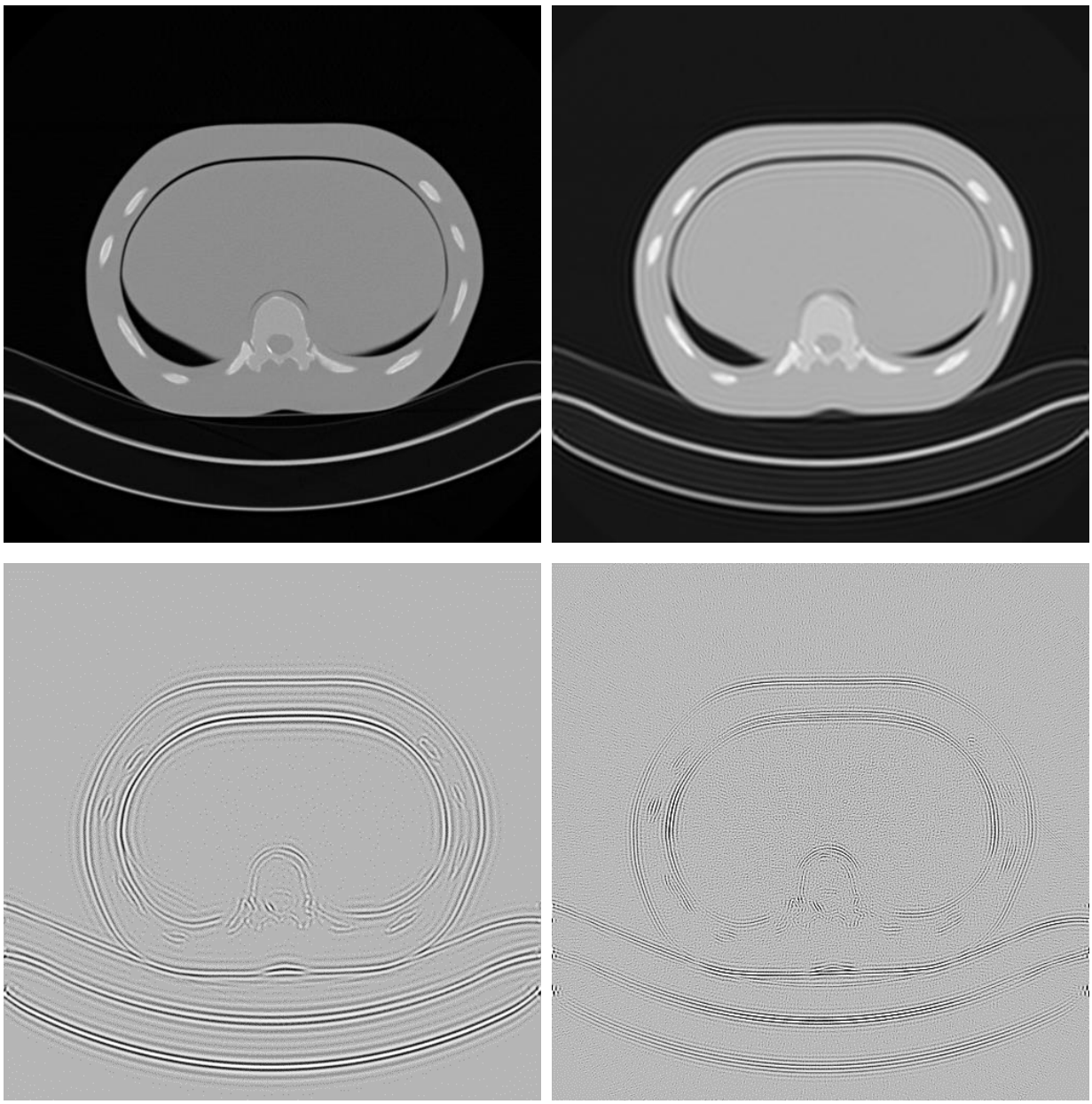


Figure 4.4 TOP LEFT: Phantom CT image [61], TOP RIGHT: Low frequency components of image represented by coarsest curvelets (scale=1) in a 3-scale transform, BOTTOM LEFT: Mid-range frequency components of image represented by curvelets at scale=2, BOTTOM RIGHT: Highest frequency components of image represented by curvelets at scale=3

The curvelet transform is a multiscale operator. Consider the conversion of an image into an *M*-scale curvelet representation. The coefficients at the coarsest scale essentially contain the lowest frequency components of the image which is not of interest here. In fact for our purpose

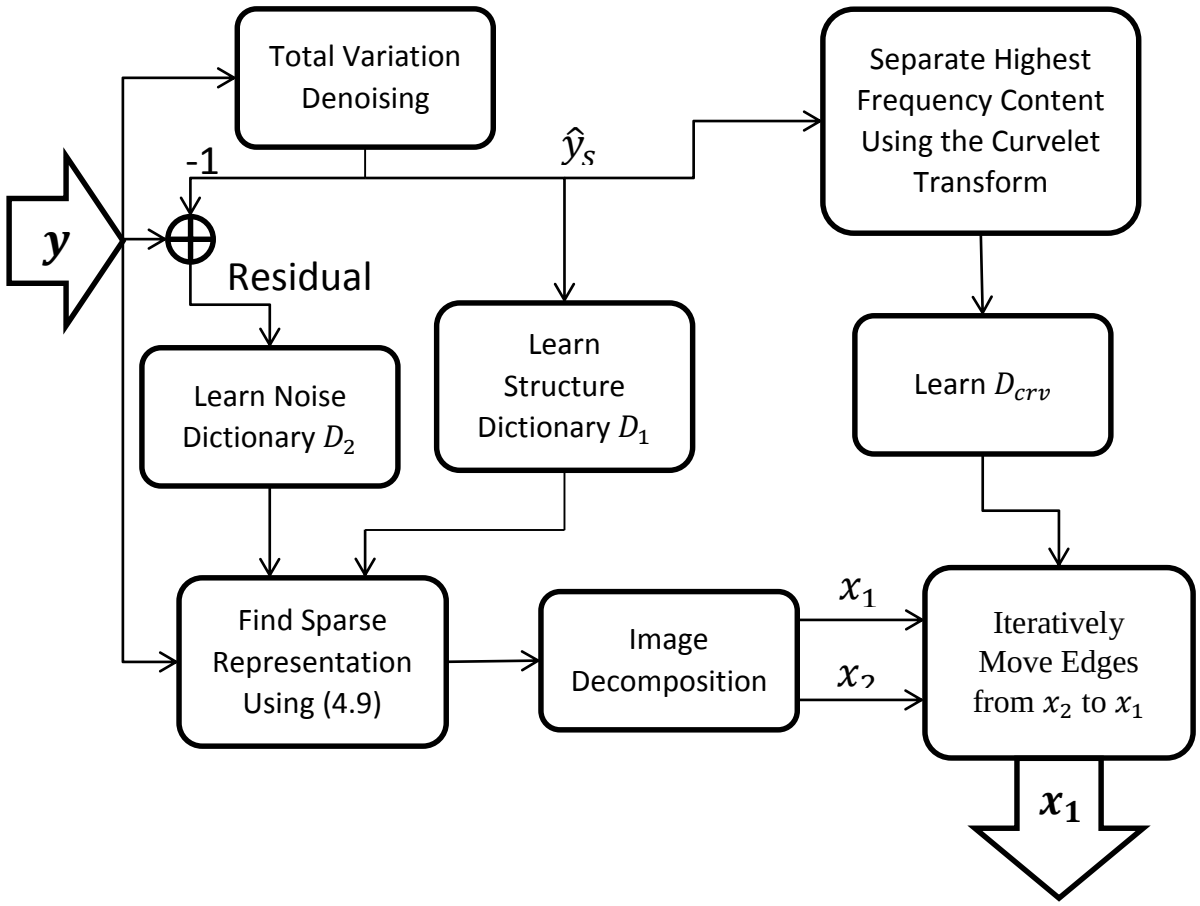


Figure 4.5 Flowchart of the proposed algorithm

of edge recovery, the two finest scales are all that is needed. By setting the coefficients of all but the finest scale to zero, and performing the inverse transform, we end up with the image edges at the highest curvelet resolution (Figure 4.4). We can use this result to learn a dictionary of high resolution curvelet atoms adapted to the image. Following the same procedure with the second-finest scale gives us a set of adapted curvelet atoms at the two highest resolutions.

We learn the adapted curvelet dictionary from the TV denoised image to avoid the noise of the original image. Then we iterate the following steps:

Table 4.1 Pseudo-Algorithm of the Proposed Method

Goal: decompose input image y into a noiseless layer x_1 and a noisy layer x_2

Input: noisy image y, number of atoms *K*, noise variance σ^2

1. **Pre-learn D_1 and D_2**
   - Perform TV denoising on y to find $\hat{y}_s$ using the optimum λ as explained in section 4.2.1
   - Learn D_1 from $\hat{y}_s$
   - Remove the strongest edges from the residual $y - \hat{y}_s$ and learn D_2 from it
2. **Sparse Coding and Image Separation**
   - Solve (4.9) for y using OMP and the concatenated dictionary $[D_1, D_2]$
   - Separate the sparse vector into α and β corresponding to D_1 and D_2 respectively
   - Using equations (4.11) and (4.12) decompose y into x_1 and x_2 corresponding to D_1 and D_2 respectively
3. **Edge Reconstruction**
   - Curvelet transform $\hat{y}_s$ into 3 scales
   - Repeat for *m=2* and *m=3*
     - Set all but the coefficients at scale *m* to zero
     - Inverse curvelet transform
     - Learn a dictionary from the result
   - Combine the adapted curvelet dictionaries to form D_{crv}
   - Iterate until some error criterion is reached (e.g. highest PSNR and SSIM)
     - k represents the current iteration
     - Take the l_2 norm of $x_2^{(k-1)}$ and set the lower 90% to zero so only the strongest edges $\mathcal{E}$ remain
     - Set *L=2* in equation (3.20) and use OMP and D_{crv} to represent $\mathcal{E}$ thus forming noiseless edges $\mathcal{E}_s$
     - Update $x_1^{(k)} = x_1^{(k-1)} + \mathcal{E}_s$ and $x_2^{(k)} = x_2^{(k-1)} - \mathcal{E}_s$

Output: denoised image x_1

- Hard threshold the noise layer x_2 so only the strongest edges remain
- Sparsely represent the residual using OMP with no more than 2 atoms from the curvelet dictionary
- Add the result to the structure layer x_1 and subtract it from the noise layer x_2

This ensures only the strongest edges are added to x_1 in each iteration. By continuously selecting edges and adding them to the structure layer a point is reached when only noise

remains and we can stop. In our experiments we measure the Peak Signal to Noise Ratio (PSNR) and the Structural Similarity (SSIM) index to stop the iterations when they reach their peak.

The flow of our proposed method is depicted in Figure 4.5 and the details are illustrated in Table 4.1.

Chapter 5. Results and Discussion

5.1. Setup of the Algorithm

The following sections outline the parameters of the proposed algorithm that were used for testing and illustrate some of the dictionaries that were created. In section 5.2 the results of our experiments are presented and compared to the K-SVD denoising method of [9].

5.1.1. Preprocessing and Dictionary Learning

In order to learn two dictionaries representing structure and noise artifacts respectively, we apply the total variation (TV) denoising. As explained in the previous chapter, λ is a

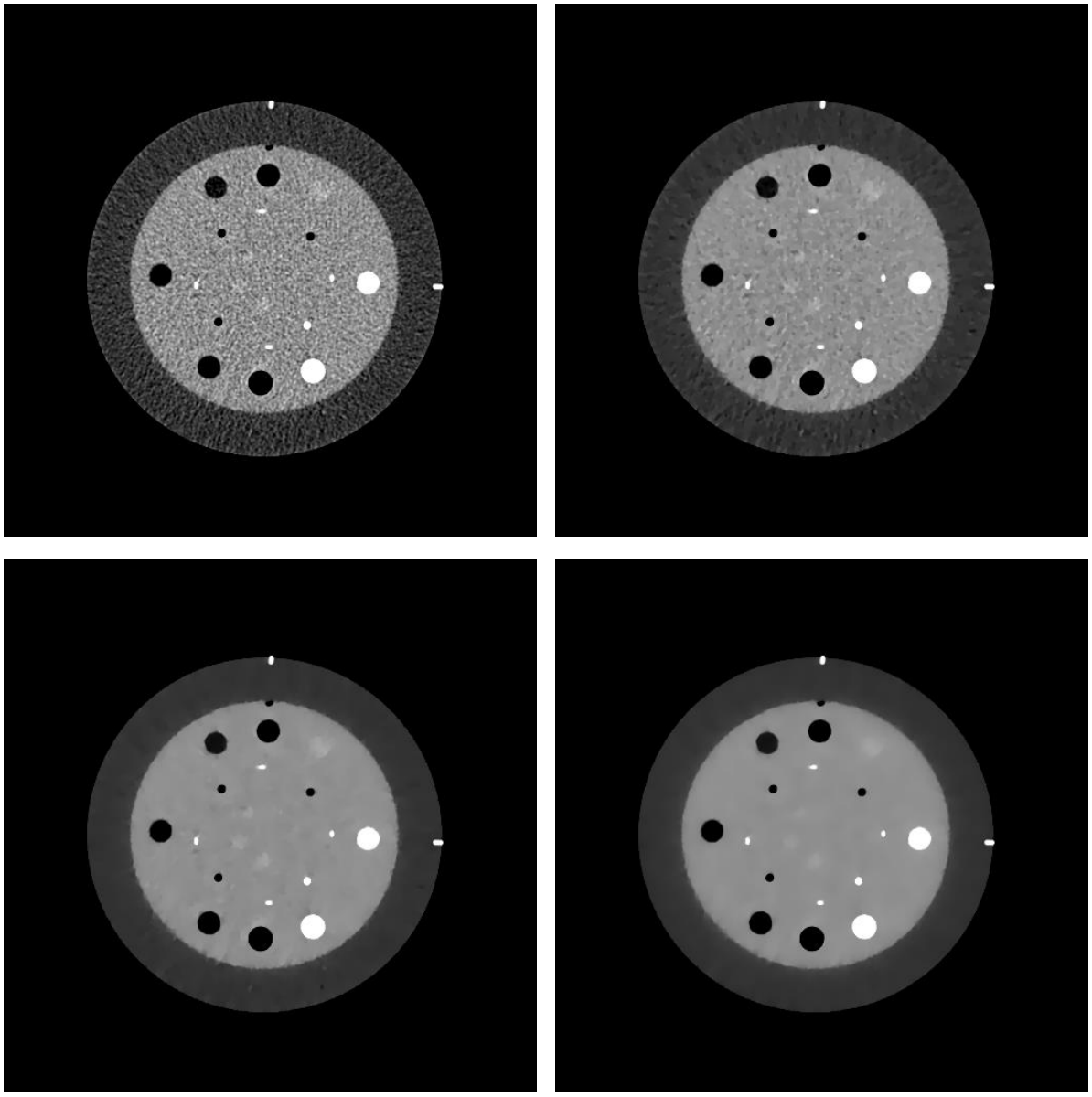


Figure 5.1 Phantom images. TOP LEFT: low-dose, TOP RIGHT: not enough smoothing because of small $\boldsymbol{\lambda}$, BOTTOM LEFT: optimal smoothing, BOTTOM RIGHT: too much smoothing with large $\boldsymbol{\lambda}$

regularization term that determines the amount of smoothing that occurs. Figure 5.1 shows various degrees of smoothing.

We need to produce a very smooth image to ensure the structure dictionary does not learn the morphology of the unwanted artifacts. The procedure outlined in 4.2.1 finds the optimal λ for a good balance between denoising and distance to the original. To this end, we

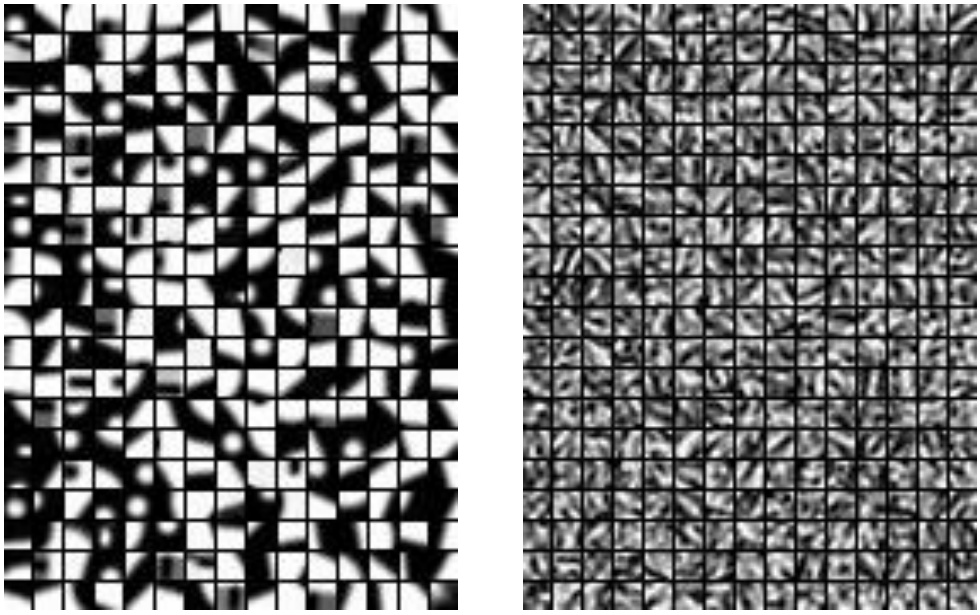


Figure 5.2 Dictionaries learned using the optimally smoothed image of Figure 5.1. LEFT: Structure dictionary, RIGHT: Noise dictionary

run the TV algorithm several times using a fixed range of λ values, and find the λ suggested by the corner of the L-curve. To ensure enough smoothing, we select the smoothed result corresponding to the next highest λ.

Next, a dictionary of 300 atoms is learned from 75% overlapping patches of size 8×8 from the smoothed image. The noisy residual is formed, patches containing edges removed using the canny edge detector and another dictionary (same dimensions) representing noise is learned from it. In both cases the K-SVD algorithm is used and the number of iterations is set to 10. The dictionaries corresponding to Figure 5.1 are shown in Figure 5.2.

5.1.2. Sparse Representation

Once the dictionaries are obtained the separation of their corresponding morphological content can begin. OMP is used for the sparse representation stage because it is efficient and

quick. Our algorithm relies on morphology rather than noise variance; therefore, the stopping criterion for OMP is the maximum number of atoms taken from both dictionaries to represent the given image. In our experiments, the ratio of atoms taken from the noise dictionary compared to the structure dictionary was very high; the latter only contributed around 2 to 3 atoms for each patch in all tests. Therefore, the stopping criterion for OMP is set to 20 atoms to allow enough structure atoms to be selected.

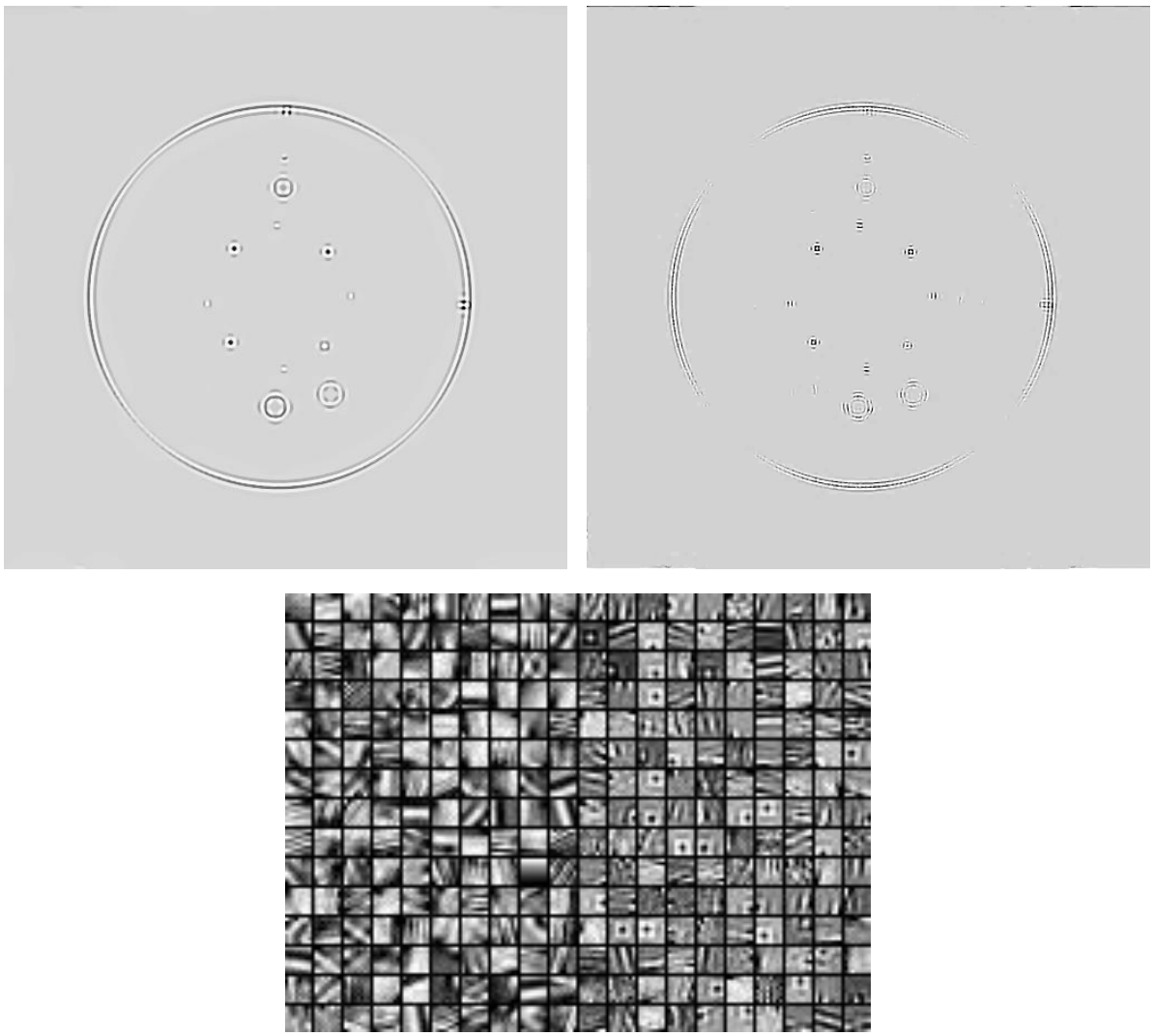


Figure 5.3 TOP: High frequency components of smoothed image in Figure 5.1 used to train dictionary, BOTTOM: Learned dictionary. Left half and right half correspond to the top images respectively

5.1.3. Adapted Curvelet Dictionary

Following the procedure of section 4.2.3, the fast curvelet transform [11] is used to isolate high frequency contents of the TV smoothed image and produce a dictionary to use for edge recovery. Figure 5.3 shows the training images corresponding to Figure 5.1 and the dictionary learned from them. Each half of the dictionary is learned from one of the training images.

5.2. Results

Experiments were performed on 3 sets containing 8, 6, and 10 images respectively, selected from low-dose CT scans of 3 different phantoms. Furthermore, 7 standard images (cameraman, Lena, peppers, etc.) were corrupted by Poisson noise to various degrees to produce 6 noisy images each, and these were tested. Altogether, 24 CT and 42 standard images were processed by the proposed algorithm and K-SVD denoising [9] to compare the two methods. All images are of size 512 × 512 pixels. More details are provided in the following pages. To evaluate the effectiveness of the algorithms the peak signal to noise ratio (PSNR) and the structural similarity (SSIM) index were calculated in each case. The same parameters used to learn dictionaries for the proposed method were input to the K-SVD denoising algorithm: 300 column atoms of size 64, and 10 iterations.

The CT images are from scans of 3 different phantoms and contain several registered high-dose (used as ground truth) and low-dose images that allow quantitative analyses to be performed. For the first phantom, the following parameters were used to scan: voltage of

120kVp, slice thickness of 0.75mm, effective dose of 210mAs for high-dose images, and effective dose of 60mAs for the low-dose images. Eight images were randomly chosen from this set, were denoised, and the results were recorded in Table 5.1. Figure 5.4 shows two of the results and the zoomed views for better examination of details.

In the second group, another phantom was scanned once with slice thickness of 5mm, and again with slice thickness of 1.25mm. For all the images, the effective dose of the high-dose and low-dose images was 21mAs and 5mAs respectively, and the voltage was 120kVp. Table 5.2 shows the denoising results. The first two rows correspond to the 5mm slices, and the rest to the 1.25mm slices. Figure 5.5 shows one of the results.

The third group are images of an anthropomorphic phantom [61] which was scanned using the following parameters: voltage of 120kVp, slice thickness of 3mm, effective dose of 200mAs for high-dose images, and effective dose of 25mAs for the low-dose images. The results of denoising 10 random images from this group are shown in Table 5.3 and Figure 5.6.

The PSNR results of the proposed method are always within close to 1dB of PSNR for K-SVD denoising. However, the SSIM metric is a better indicator of visual quality which is very important in medical imaging. The proposed algorithm is superior in that respect. K-SVD denoising assumes that each update of the dictionary brings it closer to representing the true image. However, the results show that this assumption is not always correct. Looking at Figure 5.4 and Figure 5.6 it can be seen that some parts of the images have been completely smoothed while in other areas, streaks still remain. This is because K-SVD only learns the true underlying structures if the noise is close to Gaussian. However, CT noise has a more complex distribution.

An important advantage for the proposed method over many denoising algorithms is that it learns the morphological signature of the noise. It is a lot less dependent on knowing the noise variance which is often difficult to find especially for CT images. For this reason, one needs to run K-SVD denoising a number of times, inputting various estimates of the noise variance to find the best result. Even then it tends to smooth out certain regions too much while leaving others under-processed.

To test the effectiveness of the proposed algorithm in general, 7 natural images were corrupted with Poisson noise to various degrees and were subsequently denoised. Some examples can be seen in Figure 5.7 and Figure 5.8. Table 5.4 shows the PSNR and the SSIM indices. Figure 5.9 plots the average over all 7 images and demonstrates that for lightly degraded images, both methods work similarly. However, as the noise becomes more severe, the proposed method out-performs K-SVD denoising.

Our tests show that for low-dose CT images or other severely corrupted images where it is difficult to distinguish true edges and false structures, the proposed method performs well to reconstruct the real image beneath the noise.

Table 5.1 PSNR and SSIM values for the results of the first set of phantom CT images

	Low dose		K-SVD		Proposed	
#	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
1	35.6831	0.8656	37.7097	0.9129	36.6571	0.9106
2	36.0689	0.8721	41.3300	0.8847	40.4529	0.9463
3	40.9773	0.9392	44.9002	0.9152	44.3386	0.9614
4	42.0507	0.9563	46.7823	0.9178	45.8201	0.9743
5	42.1265	0.9621	46.7936	0.7761	45.5155	0.9840
6	38.7550	0.8584	39.7400	0.8247	39.6555	0.8769
7	44.7696	0.9775	49.1444	0.9495	48.5232	0.9903
8	41.8519	0.9592	46.8806	0.8089	46.2781	0.9845
Average	40.2854	0.9238	44.1601	0.8737	43.4051	0.9535

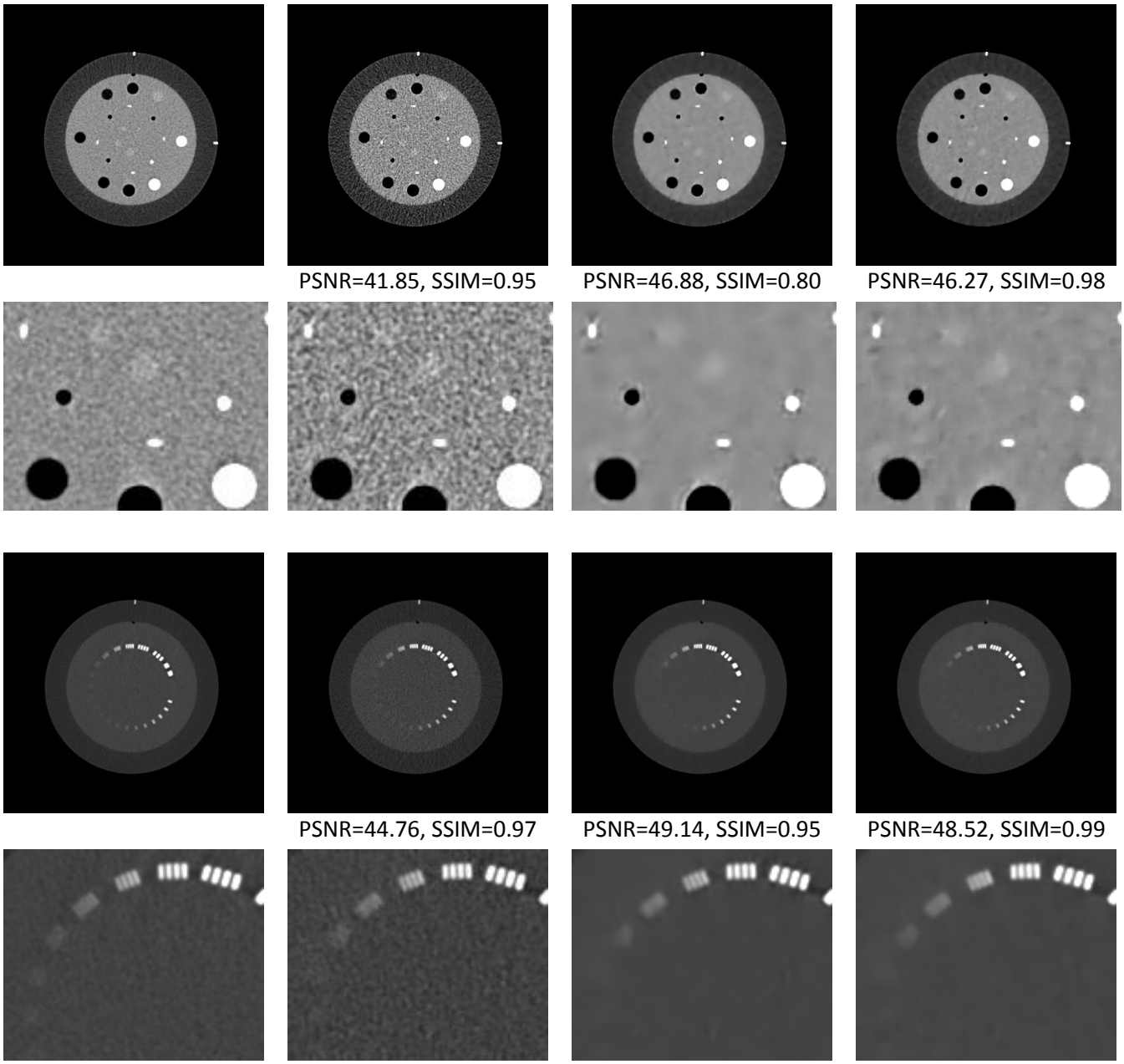


Figure 5.4 Results of denoising the first set of phantom CT images with their zoomed views below them. From the left: FIRST COLUMN: High-dose, SECOND COLUMN: Low-dose, THIRD COLUMN: K-SVD denoising, FOURTH COLUMN: Proposed method

Table 5.2 PSNR and SSIM values for the results of the second set of phantom CT images. First two rows correspond to images with 5mm slice thickness. Last four rows correspond to images with 1.25mm slice thickness

	Low dose		K-SVD		Proposed	
#	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
1	46.9374	0.9911	48.7634	0.9345	48.2077	0.9959
2	47.0486	0.9912	48.8657	0.9053	48.1017	0.9956
3	41.8767	0.9632	43.7096	0.9017	43.4323	0.9836
4	41.8661	0.9629	43.7103	0.8855	42.8274	0.9790
5	41.8188	0.9622	43.6533	0.9006	43.2037	0.9834
6	41.9235	0.9632	43.7923	0.9098	42.8346	0.9737
Average	43.5785	0.9723	45.4158	0.9063	44.7679	0.9852

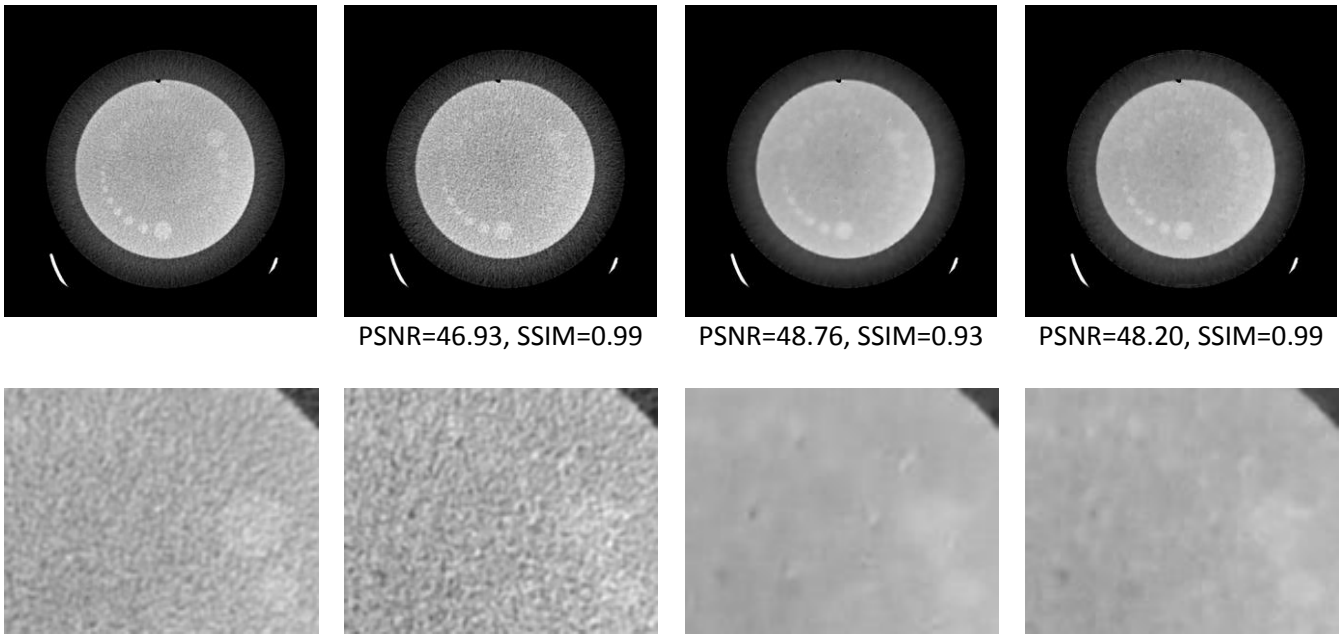


Figure 5.5 Results of denoising the second set of phantom CT images (5mm thickness) with their zoomed views below them. From the left: FIRST COLUMN: High-dose, SECOND COLUMN: Low-dose, THIRD COLUMN: K-SVD denoising, FOURTH COLUMN: Proposed method

Table 5.3 PSNR and SSIM values for the results of the third set of phantom CT images

	Low dose		K-SVD		Proposed	
#	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
1	36.6155	0.9378	40.7097	0.8445	39.8581	0.9706
2	37.7416	0.9450	41.4014	0.8762	40.9290	0.9684
3	37.7706	0.9650	40.9806	0.8972	39.8865	0.9862
4	38.4865	0.9758	40.2136	0.8910	39.6846	0.9875
5	39.3263	0.9796	41.5197	0.8820	40.7554	0.9916
6	39.7248	0.9793	42.0855	0.9054	40.9162	0.9914
7	40.0498	0.9806	42.2478	0.8918	41.4509	0.9922
8	39.5338	0.9799	41.5934	0.8983	40.8602	0.9917
9	39.5098	0.9801	41.4765	0.8794	40.8934	0.9901
10	40.8774	0.9827	43.1494	0.9105	42.3225	0.9918
Average	38.9636	0.9706	41.5378	0.8876	40.7557	0.9862

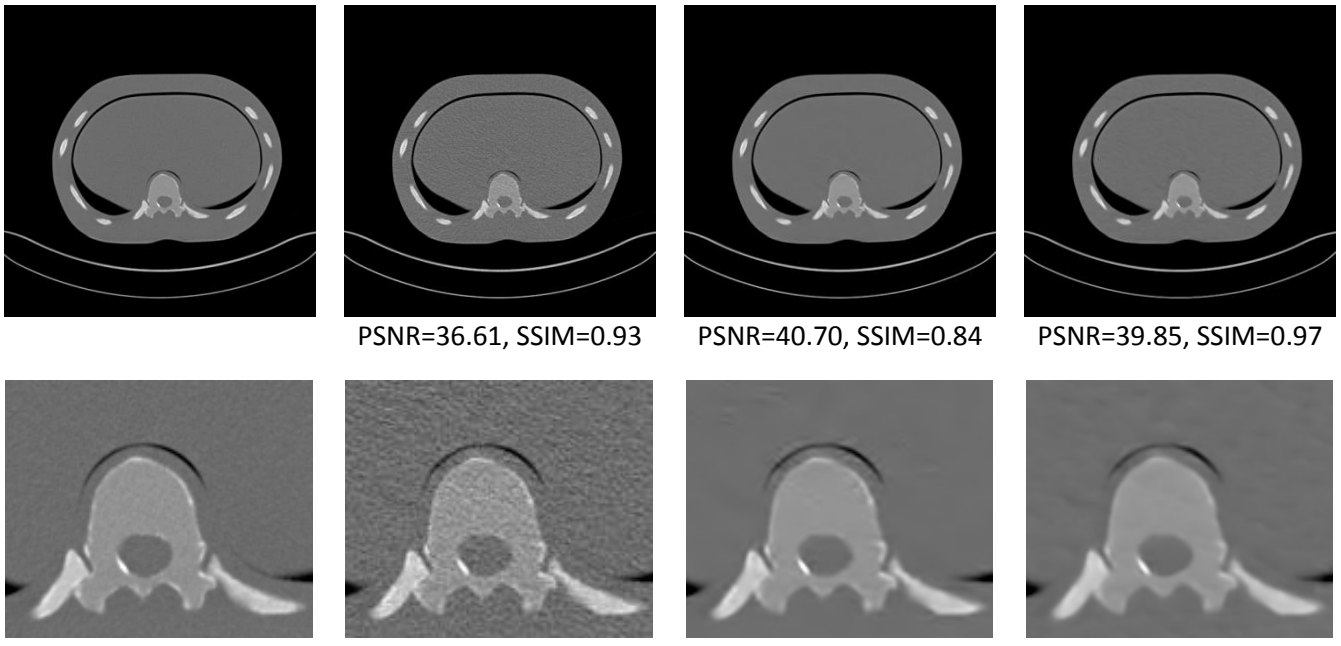


Figure 5.6 Results of denoising the third set of phantom CT images [62] with their zoomed views below them. From the left: FIRST COLUMN: High-dose, SECOND COLUMN: Low-dose, THIRD COLUMN: K-SVD denoising, FOURTH COLUMN: Proposed method

Table 5.4 PSNR (TOP) and SSIM (BOTTOM) values for the results of the natural images corrupted with Poisson noise. Standard deviation values are average estimates across all similarly corrupted images.
LEFT: Low-dose, TOP RIGHT: K-SVD, BOTTOM RIGHT: Proposed method

σ	5.77		9.17		12.01		24.68		34.50		75.96	
cameraman	36.66	41.09	29.65	35.62	26.64	33.24	19.66	27.47	16.64	24.92	9.65	18.65
		39.45		34.12		32.27		29.56		28.05		23.48
lena	36.85	39.13	29.86	35.64	26.83	33.83	19.86	29.31	16.84	27.18	9.86	22.58
		38.62		34.02		31.90		30.07		28.47		23.81
peppers	37.00	36.48	30.04	34.36	27.01	32.82	20.06	28.31	17.02	25.97	10.04	20.24
		37.78		32.77		31.65		29.93		28.50		24.23
baboon	36.14	30.28	29.16	28.41	26.13	27.17	19.13	23.42	16.12	21.70	9.14	18.99
		36.43		29.92		27.77		21.68		20.73		19.19
barbara	37.02	37.98	30.01	34.53	27.01	32.58	20.04	27.50	17.01	25.38	10.04	21.07
		37.96		32.34		30.42		24.69		23.31		21.63
boat	37.02	36.61	30.01	33.99	26.98	32.37	20.00	28.22	17.00	26.39	10.01	22.43
		37.80		32.60		30.74		27.57		26.62		23.07
couple	37.26	38.06	30.28	34.26	27.27	32.37	20.28	27.85	17.30	26.01	10.27	22.12
		38.03		32.87		30.82		26.95		26.18		23.02
average	**36.85**	**37.09**	**29.86**	**33.83**	**26.84**	**32.05**	**19.86**	**27.44**	**16.85**	**25.36**	**9.86**	**20.87**
		38.01		**32.66**		**30.80**		**27.21**		**25.98**		**22.63**

σ	5.77		9.17		12.01		24.68		34.50		75.96	
cameraman	0.97	0.98	0.90	0.95	0.84	0.93	0.63	0.81	0.53	0.74	0.26	0.49
		0.98		0.96		0.94		0.88		0.87		0.65
lena	0.98	0.98	0.90	0.96	0.84	0.94	0.64	0.85	0.53	0.76	0.27	0.53
		0.98		0.95		0.93		0.89		0.80		0.59
peppers	0.98	0.97	0.92	0.96	0.87	0.94	0.65	0.86	0.53	0.78	0.24	0.53
		0.99		0.96		0.95		0.92		0.86		0.68
baboon	0.99	0.94	0.96	0.92	0.92	0.90	0.71	0.76	0.55	0.65	0.22	0.35
		0.99		0.96		0.94		0.76		0.70		0.50
barbara	0.98	0.98	0.93	0.97	0.89	0.94	0.70	0.82	0.58	0.73	0.27	0.40
		0.99		0.95		0.93		0.80		0.74		0.59
boat	0.98	0.98	0.93	0.96	0.88	0.94	0.68	0.85	0.55	0.77	0.25	0.48
		0.99		0.95		0.92		0.86		0.82		0.61
couple	0.99	0.98	0.95	0.96	0.90	0.94	0.71	0.85	0.58	0.76	0.27	0.51
		0.99		0.96		0.93		0.85		0.82		0.62
average	**0.98**	**0.97**	**0.93**	**0.95**	**0.88**	**0.93**	**0.68**	**0.83**	**0.55**	**0.74**	**0.25**	**0.47**
		0.99		**0.96**		**0.93**		**0.85**		**0.80**		**0.61**

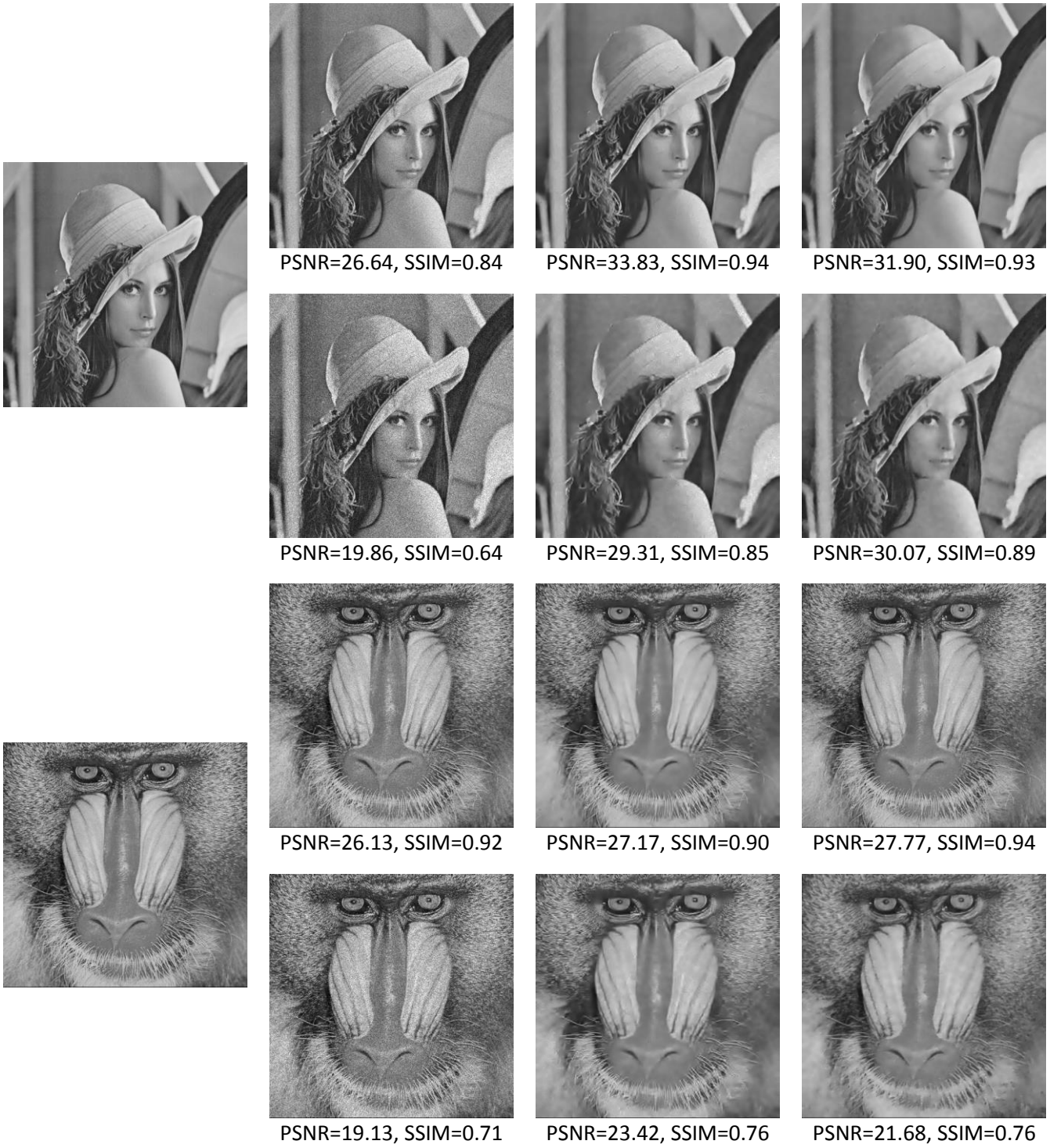


Figure 5.7 Results of denoising natural images corrupted with Poisson noise. From the left: FIRST COLUMN: Original, SECOND COLUMN: Noisy images ($TOP\ \sigma = 12.01,\ BOTTOM\ \sigma = 24.68$), THIRD COLUMN: K-SVD denoising, FOURTH COLUMN: Proposed method

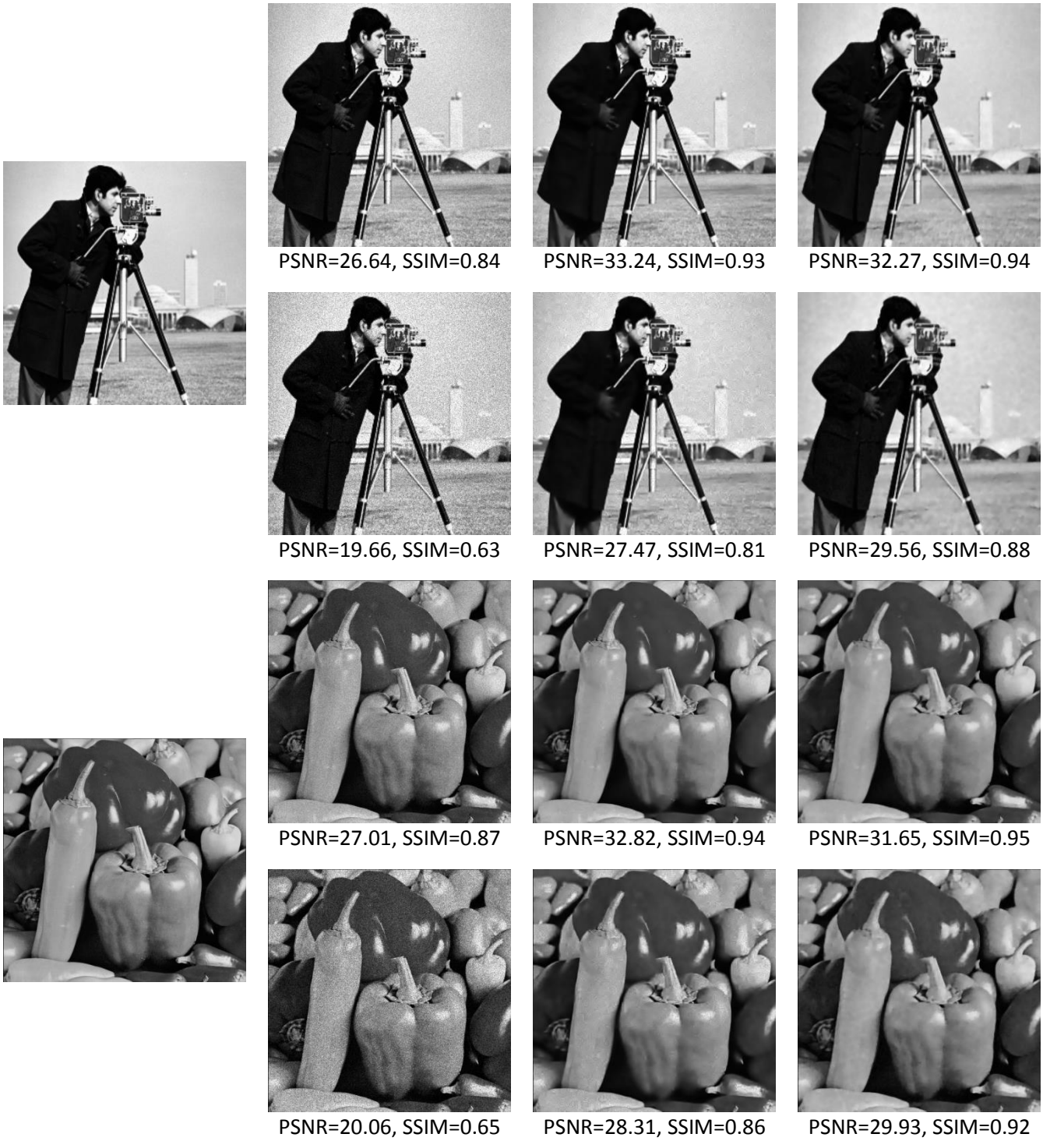


Figure 5.8 More results of denoising natural images corrupted with Poisson noise. From the left: FIRST COLUMN: Original, SECOND COLUMN: Noisy images ($TOP\ \sigma = 12.01,\ BOTTOM\ \sigma = 24.68$), THIRD COLUMN: K-SVD denoising, FOURTH COLUMN: Proposed method

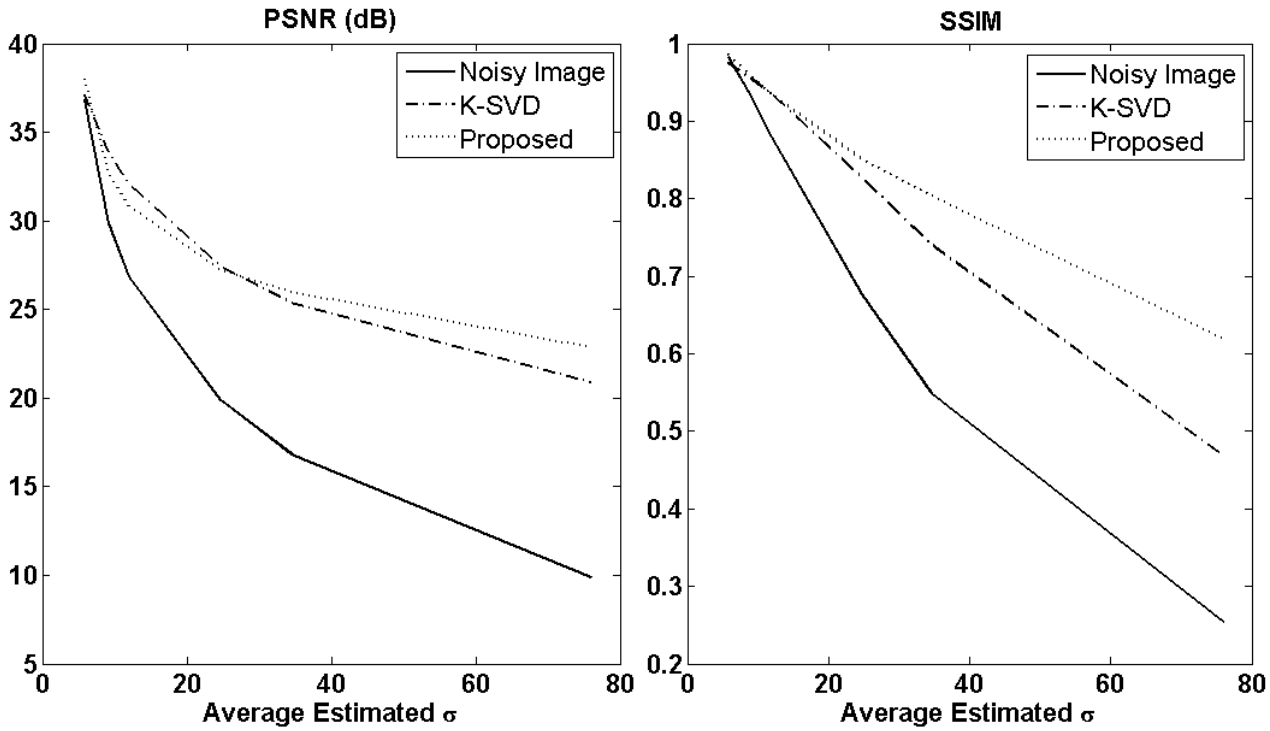


Figure 5.9 Average PSNR and SSIM values for natural images with increasing noise standard deviation

Chapter 6. Conclusion and Future Works

6.1. Total Variation Dictionary Learning

In order to separate noise from the true structures in an image we presented a preprocessing stage where a noisy image is smoothed by minimizing the total variation. The regularization method known as L-curve is used to find an optimally smoothed image that is balanced in terms of retaining structural detail and discarding noise. This smoothed image is used to train a dictionary representing the piecewise smooth parts of the image. Subsequently, the residual of the smoothed image is used to learn another dictionary representing noise and false structures. Care is taken to discard patches of the residual that may contain any strong edge presence. This makes sure that the dictionary atoms do not adapt to edge information.

This methodology creates two mutually incoherent dictionaries. When they are used together to represent the noisy image, it becomes possible to accurately reconstruct it and separate the morphological content represented by each dictionary. The goal is to decompose the image to the superposition of a noisy layer and a structure layer.

6.2. Sparse Representation and Image Separation

Once the dictionaries are obtained, they can be concatenated together and used to sparsely represent the noisy image. We use the OMP algorithm and set the stopping condition as the maximum number of atoms to use to represent each patch. The result is a sparse matrix that can be split according to which dictionary they belong to. This allows us to reconstruct the image layers individually according to which dictionary they belong to.

There is a problem that is encountered in this separation process. Despite our best efforts, certain edges are represented by both dictionaries simultaneously and some are completely mistaken as noise. Because of this we developed an iterative method of identifying edges that appear in the noise layer, and add them back to the structure layer.

6.3. Edge Recovery using Adapted Curvelets

The curvelet transform provides an optimal way of representing smooth curves and image discontinuities. For this reason we chose it as the basis for learning a dictionary adapted to edges. As mentioned before, we want to capture the edge information imbedded in the noise layer. Naively using this dictionary to represent edges in the noise layer will result in capturing

some noise as well. Therefore, we showed how to iteratively select the strongest edges and very sparsely represent them to make sure only the main structures are returned back to the main layer. Continuing this procedure, a point is reached when most edges are recovered and the algorithm should stop. Our results show the superiority of our noise removal method compared to simple K-SVD denoising.

6.4. Future Improvements

The success and accuracy of image decomposition into distinct morphological components using sparse representation is highly dependent on the dictionaries assigned to each layer. In this report we used the smoothed image and its residual to learn the dictionaries. More accurate descriptions of noise could help design a more adaptive dictionary. Furthermore, it could be worthwhile to explore some recently proposed analytical dictionaries which could help distinguish true edges from random streaks in CT images. For example, the bandelet transform [62] [63] fits a specifically optimized dictionary to an image by taking advantage of geometric regularities. Another dictionary is obtained from the directionlet transform [64] which builds oriented and anisotropic wavelets based on local image directionality.

References

[1] C. Martin, D. Sutton and P. Sharp, "Balancing patient dose and image quality," *Applied Radiation and Isotopes,* vol. 50, no. 1, pp. 1-19, 1999.

[2] J.-B. Thibault, K. D. Sauer, C. A. Bouman and J. Hsieh, "A three-dimensional statistical approach to improved image quality for multislice helical CT," *Medical Physics,* vol. 34, no. 11, p. 4526, 2007.

[3] P. J. L. Rivière, "Penalized-likelihood sinogram smoothing for low-dose CT," *medical physics,* vol. 32, no. 6, pp. 1676-83, 2005.

[4] D. Fleischmann and F. E. Boas, "Computed tomography—old ideas and new technology," *European radiology,* vol. 21, no. 3, pp. 510-517, 2011.

[5] D. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika,* vol. 81, no. 3, pp. 425-455, 1994.

[6] D. Donoho, "De-noising by soft-thresholding," *Information Theory, IEEE Transactions on,* vol. 41, p. 613–627, 1995.

[7] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 12, no. 7, pp. 629-639, 1990.

[8] L. I. Rudin, S. Osher and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena,* vol. 60, no. 1, pp. 259-268, 1992.

[9] M. Elad and M. Aharon, "Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries," *Image Processing, IEEE Transactions on,* vol. 15, no. 12, pp. 3736-45, 2006.

[10] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 11, no. 7, pp. 674-693, 1989.

[11] E. Candès, L. Demanet, D. Donoho and L. Ying, "Fast discrete curvelet transforms," *Multiscale Modeling & Simulation,* vol. 5, no. 3, pp. 861-899, 2006.

[12] M. N. Do and M. Vetterli, "Contourlets: a new directional multiresolution image representation," in *Signals, Systems and Computers, 2002. Conference Record of the Thirty-Sixth Asilomar Conference on*, 2002.

[13] D. Labate, W.-Q. Lim, G. Kutyniok and G. Weiss, "Sparse multidimensional representation using shearlets," in *Optics & Photonics 2005. International Society for Optics and Photonics*, 2005.

[14] M. Aharon, M. Elad and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *Signal Processing, IEEE Transactions on,* vol. 54, no. 11, pp. 4311-22, 2006.

[15] J.-L. Starck, M. Elad and D. Donoho, "Redundant multiscale transforms and their application for morphological component separation," *Advances in Imaging and Electron Physics,* vol. 132, no. 82, pp. 287-348, 2004.

[16] G. Peyré, J. Fadili and J.-L. Starck, "Learning adapted dictionaries for geometry and texture separation," in *Proc. SPIE 6701 Wavelets XII 67011T*, San Diego, CA, 2007.

[17] N. Shoham and M. Elad, "Algorithms for signal separation exploiting sparse representations, with application to texture image separation," in *Electrical and Electronics Engineers in Israel, IEEE 25th Convention of*, Eilat, Israel, 2008.

[18] Y. Li and X. Feng, "Image decomposition via learning the morphological diversity," *Pattern Recognition Letters,* vol. 33, no. 2, p. 111–120, 2012.

[19] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical imaging and vision,* vol. 20, no. 1-2, pp. 89-97, 2004.

[20] D. J. Brenner and E. J. Hall, "Computed Tomography — An Increasing Source of Radiation Exposure," *New England Journal of Medicine,* vol. 357, no. 22, pp. 2277-2284, 2007.

[21] United Nations Scientific Committee on the Effects of Atomic Radiation (UNSCEAR), "Sources and effects of ionizing radiation," Report to the General Assembly, New York: United Nations, 2000.

[22] A. B. d. González and S. Darby, "Risk of cancer from diagnostic X-rays: estimates for the UK and 14 other countries," *The Lancet,* vol. 363, no. 9406, pp. 345-351, 2004.

[23] J. D. Mathews, A. V. Forsythe, Z. Brady, et al, "Cancer risk in 680 000 people exposed to computed tomography scans in childhood or adolescence: data linkage study of 11 million Australians," *British Medical Journal,* vol. 346, no. f2360, 2013.

[24] National Council on Radiation Protection and Measurements, "Ionizing radiation exposure of the population of the United States (NCRP Report No 160)," Bethesda, Maryland, 2009.

[25] L. Shepp and B. Logan, "Reconstructing Interior Head Tissue from X-Ray Transmissions," *Nuclear Science, IEEE Transactions on,* vol. 21, no. 1, pp. 228 - 236, 1974.

[26] A. C. Kak and M. Slaney, Principles of Computerized Tomographic Imaging, IEEE Press, 1988.

[27] L. W. Goldman, "Principles of CT: Radiation Dose and Image Quality," *Journal of Nuclear Medicine Technology,* vol. 35, no. 4, pp. 213-225, 2007.

[28] J. Hsieh, R. C. Molthen, C. A. Dawson and R. H. Johnson, "An iterative approach to the beam hardening correction in cone beam CT," *Medical Physics,* vol. 27, no. 1, pp. 23-29, 2000.

[29] C. H. Yan, R. Whalen, G. Beaupre, S. Yen and S. Napel, "Reconstruction algorithm for polychromatic CT imaging: application to beam hardening correction," *Medical Imaging, IEEE Transactions on,* vol. 19, no. 1, pp. 1-11, 2000.

[30] P. Joseph and R. Spital, "The effects of scatter in x-ray computed tomography," *Medical Physics,* vol. 9, no. 4, pp. 464-72, 1982.

[31] F. E. Boas and D. Fleischmann, "Computed tomography artifacts: Causes and reduction techniques," *Imaging in Medicine,* vol. 4, no. 2, pp. 229-240, 2012.

[32] J. T. Bushberg and J. M. Boone, in *The Essential Physics of Medical Imaging*, 3rd ed., Lippincott Williams & Wilkins, 2011, p. 173.

[33] J. Hsieh, Computed tomography: Principles, design, artifacts, and recent advances, Bellingham, WA: SPIE Press, 2003.

[34] J. Hsieh, "Adaptive streak artifact reduction in computed tomography," *Medical Physics,* vol. 25, p. 2139–47, 1998.

[35] M. Kachelriess, O. Watzke and W. A. Kalender, "Generalized multidimensional adaptive filtering for conventional and spiral single-slice, multi-slice, and cone-beam CT," *Medical Physics,* vol. 28, p. 475–490, 2001.

[36] S. Chang, B. Yu and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *Image Processing, IEEE Transactions on,* vol. 9, no. 9, pp. 1532-46, 2000.

[37] Y. Chen, X. Ye and F. Huang, "A novel method and fast algorithm for MR image reconstruction with significantly under-sampled data," *Inverse Problems and Imaging,* vol. 4, no. 2, pp. 223-240, 2010.

[38] R. Rubinstein, M. Zibulevsky and M. Elad, "Double Sparsity: Learning Sparse Dictionaries for Sparse Signal Approximation," *Signal Processing, IEEE Transactions on,* vol. 58, no. 3, pp. 1553-64, 2009.

[39] Y. Wang, J. Yang, W. Yin and Y. Zhang, "A New Alternating Minimization Algorithm for Total Variation Image Reconstruction," *SIAM Journal on Imaging Sciences,* vol. 1, no. 3, p. 248–272, 2008.

[40] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *Signal Processing, IEEE Transactions on,* vol. 41, no. 12, p. 3397–415, 1993.

[41] S. S. Chen, D. L. Donoho and M. A. Saunders, "Atomic Decomposition by Basis Pursuit," *SIAM journal on scientific computing,* vol. 20, no. 1, pp. 33-61, 1998.

[42] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1?," *Vision research,* vol. 37, no. 3, pp. 3311-25, 1997.

[43] K. Engan, S. Aase and J. Hakon Husoy, "Method of optimal directions for frame design," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, 1999.

[44] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, et al, "Dictionary Learning Algorithms for Sparse Representation," *Neural Computation,* vol. 15, no. 2, pp. 349-396, 2003.

[45] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," *Math. Comp.,* vol. 19, pp. 297-301, 1965.

[46] J. B. Allen and L. Rabiner, "A unified approach to short-time Fourier analysis and synthesis," *Proceedings of the IEEE,* vol. 65, no. 11, pp. 1558-64, 1977.

[47] D. L. Donoho, "Wedgelets: Nearly minimax estimation of edges," *The Annals of Statistics,* vol. 27, no. 3, pp. 859-897, 1999.

[48] E. J. Candès and D. L. Donoho, "Ridgelets: A key to higher-dimensional intermittency?," *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences,* vol. 357, no. 1760, pp. 2495-509, 1999.

[49] E. J. Candès and D. L. Donoho, "Curvelets: A surprisingly effective nonadaptive representation for objects with edges," *Curves and Surfaces,* 1999.

[50] J. O. Stromberg, "A modified haar system and higher order spline systems," in *Conference in harmonic analysis in honor of Antoni Zygmund*, 1981.

[51] Y. Meyer, "Principe D'incertitude, bases hilbertiennes et algebres d'operateurs," in *Seminaire Bourbaki*, 1985.

[52] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications on Pure and Applied Mathematics,* vol. 41, p. 909–996, 1988.

[53] E. P. Simoncelli, W. T. Freeman, E. H. Adelson and D. J. Heeger, "Shiftable multiscale transforms," *Information Theory, IEEE Transactions of,* vol. 38, no. 2, pp. 587-607, 1992.

[54] G. Beylkin, "On the representation of operators in bases of compactly supported wavelets," *SIAM Journal on Numerical Analysis,* vol. 29, no. 6, pp. 1716-40, 1992.

[55] Y. C. Pati, R. Rezaiifar and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Asilomar Conference on Signals, Systems and Computers*, 1993.

[56] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society,* vol. 58, no. 1, pp. 267-288, 1996.

[57] M. S. Lewicki and T. J. Sejnowski, "Learning Overcomplete Representations," *Neural Computation,* vol. 12, no. 2, pp. 337-365, 2000.

[58] J. Hsieh, "Nonstationary noise characteristics of the helical scan and its impact on image quality and artifacts," *Medical Physics ,* vol. 24, no. 9, pp. 1375-84, 1997.

[59] H. Lu, I.-T. Hsiao, X. Li and Z. Liang, "Noise properties of low-dose CT projections and noise treatment by scale transformations," *Nuclear Science Symposium Conference Record, 2001 IEEE,* vol. 3, pp. 1662-66, 2001.

[60] P. C. Hansen, "Analysis of Discrete Ill-Posed Problems by Means of the L-Curve," *SIAM Review,* vol. 34, no. 4, pp. 561-580, 1992.

[61] M. A. Gavrielides, L. M. Kinnard, K. J. Myers, et al, "A resource for the assessment of lung nodule size estimation methods: database of thoracic CT scans of an anthropomorphic phantom," *Optics Express,* vol. 18, no. 14, pp. 15244-55, 2010.

[62] E. Le Pennec and S. Mallat, "Sparse geometric image representations with bandelets," *Image Processing, IEEE Transactions on,* vol. 14, no. 4, pp. 423-438, 2005.

[63] G. Peyré and S. Mallat, "Surface compression with geometric bandelets," *ACM Transactions on Graphics (TOG),* vol. 24, no. 3, pp. 601-608, 2005.

[64] V. Velisavljevic, B. Beferull-Lozano, M. Vetterli and P. L. Dragotti, "Directionlets: anisotropic multidirectional representation with separable filtering," *Image Processing, IEEE Transactions on,* vol. 15, no. 7, pp. 1916-33, 2006.

List of Submitted Publications

Aryan Khodabandeh, Javad Alirezaie, and Paul Babyn, "Computed Tomography Image Denoising by Learning to Separate Morphological Diversity," submitted to the 38th International Conference on Telecommunications and Signal Processing (TSP) 2015, Prague, Czech Republic

Aryan Khodabandeh, Javad Alirezaie, and Paul Babyn, "Reducing Image Noise In Low Dose Computed Tomography (CT) By Learning The Morphologies Of Image Structure And Noise," submitted to the 29th International Congress and Exhibition of Computer Assisted Radiology and Surgery, 2015, Barcelona, Spain

Glossary

CT Computed Tomography

DCT Discrete Cosine Transform

DFT Discrete Fourier Transform

DWT Discrete Wavelet Transform

ICA Independent Component Analysis

K-SVD Generalized K-Means Singular Value Decomposition

LASSO Least Absolute Shrinkage and Selection Operator

MAP Maximum A Posteriori

MCA Morphological Component Analysis

MOD Method of Optimal Directions

MP Matching Pursuit

OMP Orthogonal Matching Pursuit

PCA Principle Component Analysis

PDE Partial Differential Equation

PSNR Peak Signal to Noise Ratio

SSIM Structural Similarity Index

STFT Short Time Fourier Transform

SVD Singular Value Decomposition

TV Total Variation