UNIVERSITÉ PARIS 13 - Institut Galilée
Laboratoire Analyse, Géométrie et Applications

N° attribué par la bibliothèque
|__|__|__|__|__|__|__|__|__|__|

T H È S E

pour obtenir le titre de

DOCTEUR DE L'UNIVERSITÉ PARIS 13

Spécialité : SIGNAUX ET IMAGES

présentée et soutenue publiquement par

Dinh Hoan TRINH
le 21/06/2013

Denoising and Super-Resolution for Medical Images by Example-based Learning Approach

---

Directeurs de thèse : **Françoise DIBOS, Canh Duong PHAM et Marie LUONG**

Jury :

Rapporteurs :	**Mme. Agnès DESOLNEUX**	- Prof., Directrice de recherche CNRS, UMR 8536, CMLA, ENS Cachan, France
	Mme. Béatrice PETQUET-POPESCU	- Prof., Télécom ParisTech, France
	M. Truong NGUYEN	- Prof., University of California, San Diego, USA
Examinateurs :	**M. Georges KOEPFLER**	- MCF, Paris Descartes (UP5), France
	M. Jean-Marie ROCCHISANI	- Docteur en médecine nucléaire, Hôpital Avicenne, CHU Bobigny, UP13, France
	M. Emmanuel VIENNET	- Prof., Université Paris 13, France
Directeurs et Co-encadrant :	**Mme. Marie LUONG** (Co-encadrant)	- MCF, Université Paris 13, France
	M. Canh Duong PHAM (Co-directeur de thèse)	- Docteur, Vietnam Academy of Science and Technology, Vietnam
	Mme. Françoise DIBOS (Directrice de thèse)	- Prof., Université Paris 13, France

UNIVERSITÉ PARIS 13 - Institut Galilée
Laboratoire Analyse, Géométrie et Applications

N° attribué par la bibliothèque
|__|__|__|__|__|__|__|__|__|__|

T H È S E

pour obtenir le titre de

DOCTEUR DE L'UNIVERSITÉ PARIS 13

Spécialité : SIGNAUX ET IMAGES

présentée et soutenue publiquement par

Dinh Hoan TRINH
le 21/06/2013

Denoising and Super-Resolution for Medical Images by Example-based Learning Approach

---

Directeurs de thèse : **Françoise DIBOS, Canh Duong PHAM et Marie LUONG**

Jury :

Rapporteurs :	**Mme. Agnès DESOLNEUX**	- Prof., Directrice de recherche CNRS, UMR 8536, CMLA, ENS Cachan, France
	Mme. Béatrice PETQUET-POPESCU	- Prof., Télécom ParisTech, France
	M. Truong NGUYEN	- Prof., University of California, San Diego, USA
Examinateurs :	**M. Georges KOEPFLER**	- MCF, Paris Descartes (UP5), France
	M. Jean-Marie ROCCHISANI	- Docteur en médecine nucléaire, Hôpital Avicenne, CHU Bobigny, UP13, France
	M. Emmanuel VIENNET	- Prof., Université Paris 13, France
Directeurs et Co-encadrant :	**Mme. Marie LUONG** (Co-encadrant)	- MCF, Université Paris 13, France
	M. Canh Duong PHAM (Co-directeur de thèse)	- Docteur, Vietnam Academy of Science and Technology, Vietnam
	Mme. Françoise DIBOS (Directrice de thèse)	- Prof., Université Paris 13, France

Acknowledgments

First of all, I would like to express my deepest gratitude to my advisors: Professor DIBOS Françoise, Dr. PHẠM Cảnh Dương and Assoc. Prof. LƯƠNG Marie. Without their incredible guidance and support, I would never have done this work.

I want to thank Dr. ROCCHISANI Jean-Marie for providing me the medical image data and for his continuous discussion on many different aspects and the experimental results associated with these data.

I would like to thank my thesis committee members Professor DESOLNEUX Agnès, Professor PETQUET-POPESCU Béatrice, Professor NGUYỄN Trường, Assoc. Prof. KOEPFLER Georges, Dr. ROCCHISANI Jean-Marie, and Professor VIENNET Emmanuel for reviewing this thesis and their valuable feedback.

I am deeply grateful to Professor PHẠM Huy Điển for enlightening me the first glance of research. I would also like to thank Dr. PHẠM Xuân Hinh and Dr. LÊ Thị Hương for all their help.

I would like to express my gratitude to Dr. ĐỖ Quốc Bảo, Dr. BÙI Thị Thu Cúc, Dr. NGUYỄN Đặng Hồ Hải. They are the first persons I met in Paris. I am lucky to be their friend, and will never forget their support.

I would also like to thank my friends CUNG Thị Ngọc Phượng, ĐẶNG Thành Trung, TRẦN Cường, TẠ Thị Thanh Mai, NGUYỄN Trọng Hiếu and all of my other friends.

Finally, I want to express gratitude to my family. I cannot find any other words for their sacrifice and consideration. I would like to thank my wife for her continuous support and patience during these three and a half years. This work is dedicated to my family, my wife and daughter.

Paris, June 21st, 2013

TRỊNH Đình Hoàn

Contents

Chapter 0

Introduction

Contents

Medical imaging is the technique and process that are used to view the human body. It has been playing a key role in medicine for diagnosis, therapy planning and treatment follow-ups. The influence and impact of digital images on modern society is important, and image processing is now a critical component in science and technology. The rapid progress in computerized medical image reconstruction, and the associated developments in analysis methods and computer-aided diagnosis, has propelled medical imaging into one of the most important sub-fields in scientific imaging [Dou09]. With the widespread use of imaging in medicine, the quality of medical images becomes an important issue.

The arrival of digital medical imaging technologies such as Computerized Tomography (CT), Single Photon Emission Computed Tomography (SPECT) , Positron Emission Tomography (PET), Magnetic Resonance Imaging (MRI), Ultrasound (US) imaging, has revolutionized modern medicine. Despite advances in technology of acquisition and performance of reconstruction algorithms, the quality of medical images is often affected by the limitations of resolution, contrast, noise and artifacts during the acquisition, transmission, and processing of image data. This

makes it more difficult to analyze or extract useful information and hence, the quality of pathological diagnosis and treatment can be seriously affected. Therefore, to achieve the best possible diagnoses, lessen the chance of misdiagnoses and inappropriate treatment, it is necessary to adopt effective image processing methods to obtain images of higher quality and reliability. That is also the motivation for this thesis.

The quality of medical images depends on five principle characteristics as follows [Spr95]:

1. Contrast sensitivity
2. Blurring and visibility of detail
3. Artifacts
4. Visual noise
5. Spatial resolution.

Many techniques have been addressed these characteristics in order to improve the image quality of medical imaging. These techniques can be broadly divided to two: during acquisition and after acquisition. The former includes special processing modules embedded in the image acquisition systems to enhance the quality and also the information content in the generated images. The latter refers to digital image processing techniques used for the acquired images. In the scope of this thesis, we focus on the problems of *denoising* and *spatial resolution enhancement* of digital medical images after acquisition.

Before presenting the objectives and contributions of this work, let us briefly describe about the medical imaging modalities.

0.1 Medical imaging modalities

An imaging modality is an imaging system which uses a particular technique to detect a certain physical signal arising from a patient and produce images. Many medical imaging modalities have been proposed, some of them use *ionizing* radiation, and others use *non-ionizing*. Ionizing radiation is a form of radiation that has sufficient energy to ionize atoms and molecules within the body. In medical imaging, ionizing radiation consists of X-rays and γ-rays, both of which need to be used prudently to avoid causing harmful side-effects on the human body. Non-ionizing radiation, such as high-frequency sound (ultrasound) and radio frequency waves, on the other hand, does not have the potential to damage directly on the body and the risks associated with its use are considered to be very low [Dou09].

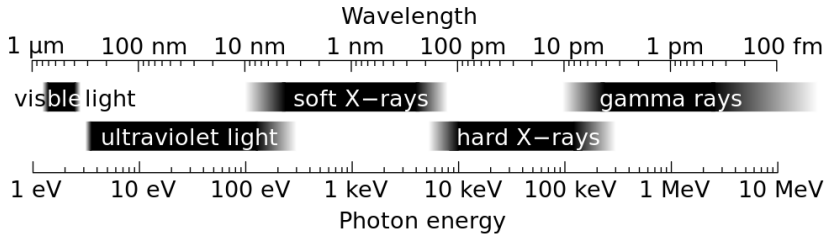


Figure 1: The wavelength and photon energy ranges of the electromagnetic waves including the visible light, ultraviolet light, soft X-rays, hard X-rays, and γ-rays. (source : http://en.wikipedia.org/wiki/X-ray)

In this section we concisely describe about the imaging modalities from X-ray, γ-rays and non-ionizing radiation. For more detailed study, let us mention [Spr95, Jan06, Dou09].

0.1.1 Images from X-rays

X-rays is a form of electromagnetic radiation with a wavelength shorter than ultraviolet rays, but longer than γ-rays, in the range of 0.01 to 10 nanometers, corresponding to photon energies from 100 eV to 100 keV (1eV $= 1.6 \times 10^{-19}$ J) (see Figure 1).

Since it was discovered by Wilhelm Conrad Rontgen in the last decade of the 19th century, X-rays imaging has been widely used in clinical diagnosis. Medical images from X-rays include projection radiography, fluoroscopy, mammography, and CT. These modalities all work on the same basic principle: an X-ray beam is passed through the body where a portion of the X-rays are either absorbed or scattered by the internal structures, and the remaining X-ray pattern is transmitted to a detector for recording or further processing by a computer. In general, objects with higher density (such as bones and calcium deposits) absorb more X-rays than objects with lesser density. Thus, these objects leave a different image on the detector. Specially trained or experienced physicians can read these images to diagnose medical conditions or injuries.

0.1.1.1 X-ray Projection Radiography (X-PR)

The X-ray projection radiography is the historically oldest medical imaging modality. This is also the simplest modality compared with the others modalities. Its principle is transparent with ability of providing directly the images without any reconstruction, as illustrated in Figure 2. The image is formed of intensity values of the X-rays modified after passing through the imaged object, e.g., a part of the

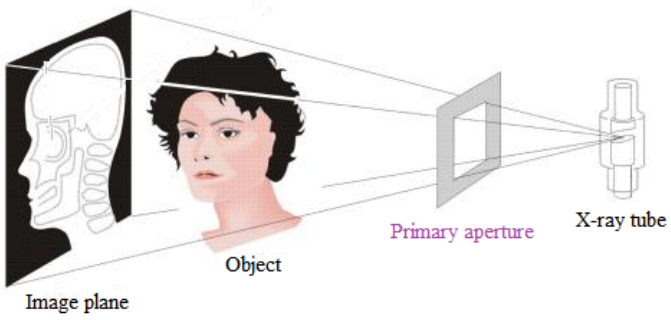


Figure 2: Principle of X-ray projection imaging.

Figure 3: Typical X-ray images (source: http://radiopaedia.org).

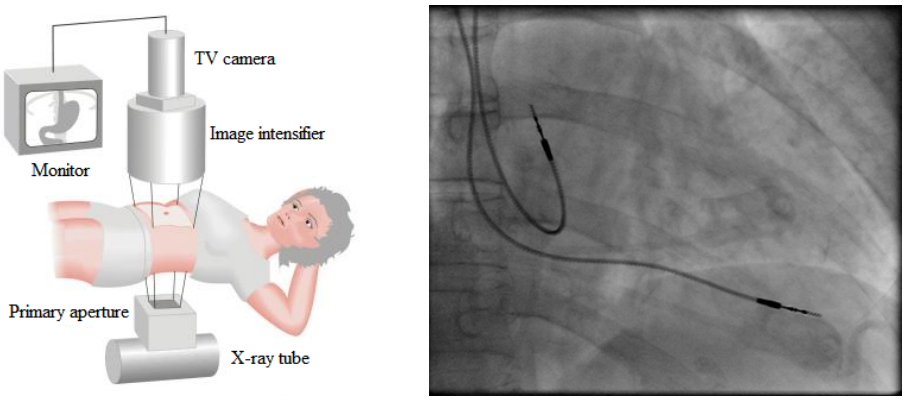
patient's body. Typical examples of X-ray images are shown in Figure 3.

X-ray projection radiography is used in many types of examinations and procedures. Some examples include the possibilities to find orthopedic damage, tumors, pneumonias, foreign objects, etc.

0.1.1.2 Fluoroscopy (FC)

Fluoroscopy is a X-rays-based imaging technique, where images of moving body parts and internal structures of a patient can be seen in real-time. It was invented by Thomas Edison in 1896. Fluoroscopy is used mostly for studies that require observation of dynamic body functions.

During a fluoroscopy procedure, an X-ray beam is passed through the body. The image is transmitted to a monitor so that the body part and its motion can be seen



(a) (b)

Figure 4: (a) Fluoroscopy with image amplifier, (b) a typical fluoroscopy image.

in detail (see Figure 4).

0.1.1.3 Mammography (MG)

Mammography is a type of medical imaging that uses low-energy X-rays to create images of the internal structures of the breasts. These images, called mammograms, are used to find early signs of breast cancer in women such as a dense mass or clusters of calcium (microcalcifications).

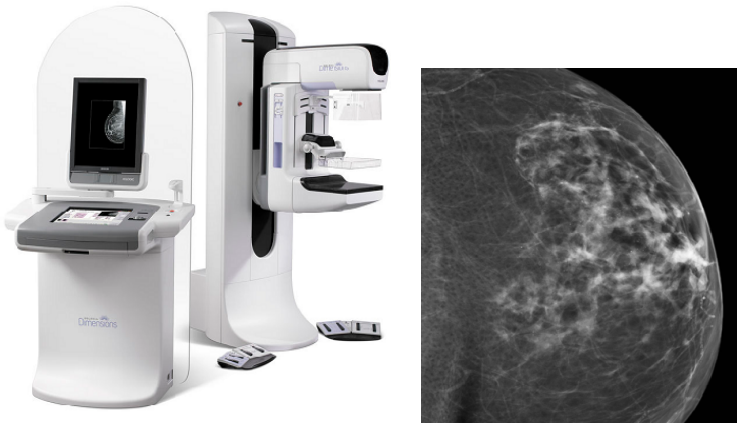
X-ray images of the breast can be captured on film or stored directly onto a computer:

- Screen-film mammography where X-rays are beamed through the breast to a cassette containing a screen and film that must be developed.
- Full field digital mammography where X-rays are beamed through the breast to an image receptor. A scanner converts the information to a digital picture which is sent to a digital monitor and/or a printer.

Figure 5 shows a digital mammography system and a typical mammography image.

0.1.1.4 Computed Tomography (CT)

Computed tomography scanning, also called computerized axial tomography scanning, is a medical imaging technique that uses X-rays to create cross-sectional images or "slices" of the body. These cross-sectional images are used for a variety of diagnostic and therapeutic purposes. It became the historically first tomographic modality entirely based on digital reconstruction of images.



(a) (b)

Figure 5: (a) A digital mammography system (source http://www.medicalexpo.com/), (b) a typical mamography image.

The formation of a CT image includes three phases: scanning phase, reconstruction phase and digital-to analog conversion phase (see Figure 6). The scanning phase produces data, but not an image. The reconstruction phase processes the acquired data and forms a digital image while the visible and displayed analog image is produced by the digital-to analog conversion phase.

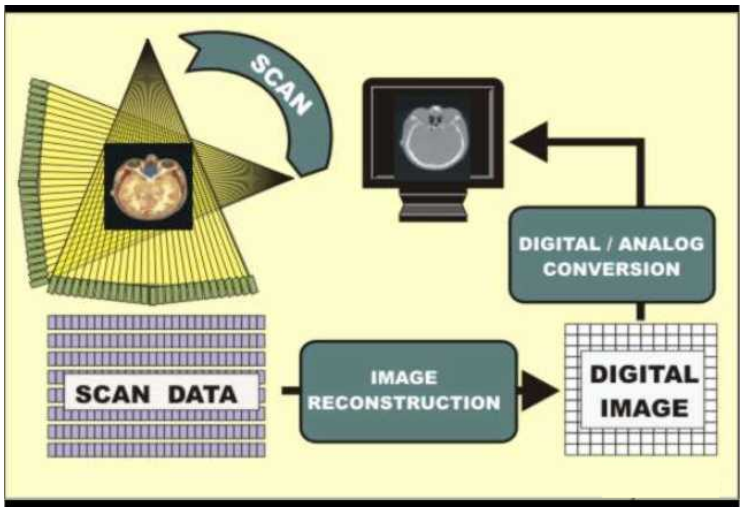


Figure 6: The formation of a CT image (source http://www.sprawls.org/resources/).

In the scanning phase, a fan-shaped X-ray beam is scanned around the body. The amount of X-radiation penetrating the body along each individual ray (pathway) through the body is measured by the detectors that intercept the X-ray beam after it

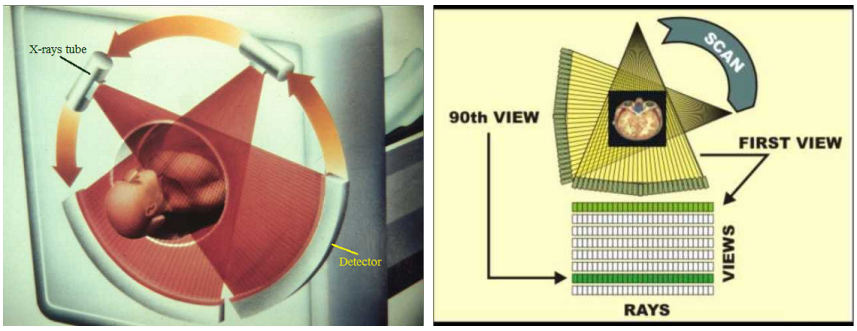


Figure 7: Illustration of the scanning phase (source http://www.sprawls.org/resources/).

passes through the body. The data recorded by the detectors are stored in computer memory for later image reconstruction. Figure 7 illustrates the scanning phase of a slice, where one view represents the projection of the fan-shaped X-ray beam from one specific position of X-ray tube focal spot. A complete scan is performed by rotating the X-ray tube completely around the body and projecting many views. The complete data set produced from a complete scan contains sufficient information for the reconstruction of a slice image.

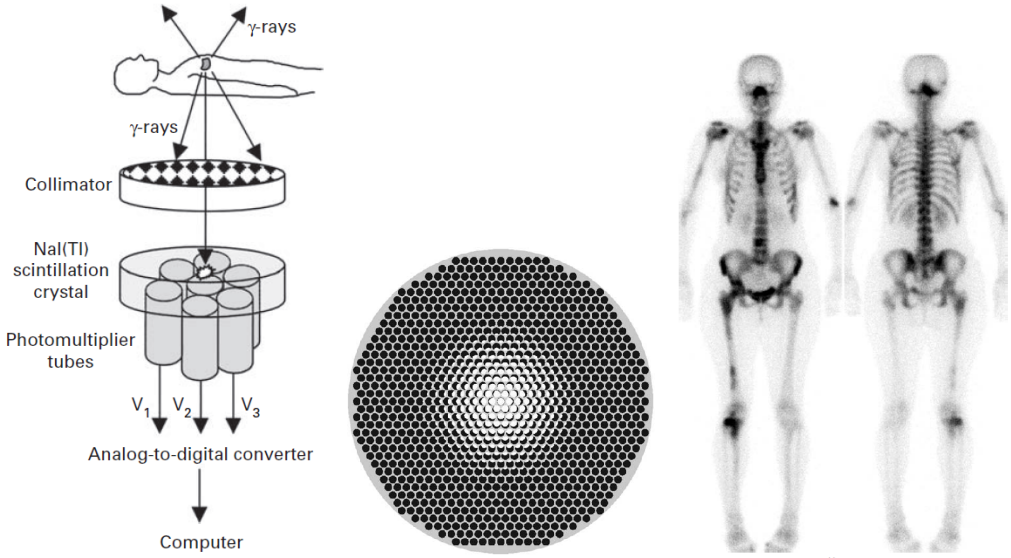
0.1.2 Images from γ-rays

Nuclear imaging (or γ imaging) is a branch of medical imaging that uses the γ-rays emitted from radioactive isotopes attached to pharmaceutical tracers to represent distribution of radionuclides inside the patient's body. These radio-labeled pharmaceutical tracers are ingested or injected into the body where they are circulated and/or metabolized. The γ-rays which they emit during radioactive decay pass out of the body are measured by detectors (*gamma cameras*) placed around the patient, and produce images which show the functional or metabolic activity in the relevant organs. The nuclear imaging is utilized to diagnose and determine the severity of or treat a variety of diseases, including many types of cancers, heart disease, gastrointestinal, endocrine, neurological disorders and other abnormalities within the body.

There are three basic imaging modalities in nuclear medicine: the *planar scintigraphy* (or *planar γ-imaging*), the *single-photon emission computed tomography* (SPECT), and the *positron emission tomography* (PET).

0.1.2.1 Planar scintigraphy

In planar γ-imaging, a single gamma camera, or a dual-head gamma camera to simultaneously take anterior and posterior images, is used to detect the emitted γ-rays. Unlike in radiography, γ-rays are emitted in all directions and pass through the body. Thus, in order to locate the source of the γ-rays, a collimator is placed between the patient and the detector. The collimator is a slab of lead with a geometric array of holes in it (see Figure 8(b)). It only allows γ-photons incident on it almost perpendicularly to pass through; photons from other directions are absorbed by the lead. Thus, only a very small portion of the γ-photons have the direction enabling passage through the collimator. These photons is consequently detected by the gamma detector.



(a) (b) (c)

Figure 8: (a) Schematic diagram for obtaining a planar nuclear medicine image, using a gamma camera. (b) A typical collimator. (c) An example of a planar scintigraphy, (left, anterior; right, posterior) (source: [Dou09]).

The gamma detector consists of a scintillation crystal to convert the γ-rays into low-energy photons (usually in the visible range). The most commonly used scintillation crystal is sodium iodide, doped with thallium, NaI (Tl). These lower-energy photons are subsequently collected by photomultiplier tubes (PMTs). The purpose of the photomultiplier tubes is to convert the light into the electrical signal and amplifies the signal. From the position and outputs of the photomultiplier tubes, the position and the energy of the original γ-photon interaction can be calculated and then used to form an image of the distribution of radioactivity. A schematic

(a) (b)

Figure 9: (a) A dual head system (source: http://www.medgadget.com/). (b) Several SPECT images of brain (source: [MS03]).

diagram for obtaining a planar nuclear medicine image, using a gamma camera, is illustrated in Figure 8(a). Figure 8(c) shows an example of a planar scintigraphy.

0.1.2.2 SPECT imaging

In single-photon emission computed tomography, SPECT, a rotating gamma camera (single or multiple-head) is used to detect the emitted γ-rays from multiple angles, which are used to reconstruct cross-sectional images. The acquired data is then used to reconstruct cross-sectional or three dimensional images.

The advantage of SPECT over planar scintigraphy is that while planar scintigraphy displays a single view of radiotracer distribution in patients (thyroid, ventilation/perfusion, or whole-body bone images), SPECT can display slice (2-dimensional) or volume (3-dimensional) images of radiotracer distribution. SPECT scanning is mainly used to find out how well an organ is functioning by looking at the supply of blood to its tissues. SPECT scanning is particularly useful for assessing the function of the brain, heart, liver, and lungs. We show in Figure 9 a typical SPECT system and several SPECT images of brain.

0.1.2.3 PET imaging

Positron emission tomography, PET, is a highly specialized imaging technique which enables the visualization of metabolic processes in the body based on detecting γ-rays emitted when certain tracer compounds labeled with positron-emitting radionuclides are injected into the subject of the study. PET imaging is often used to illustrate physiological function (e.g. perfusion, metabolism). It gives information about the chemical activity of tissues or organs. It can also be used to assess blood flow.

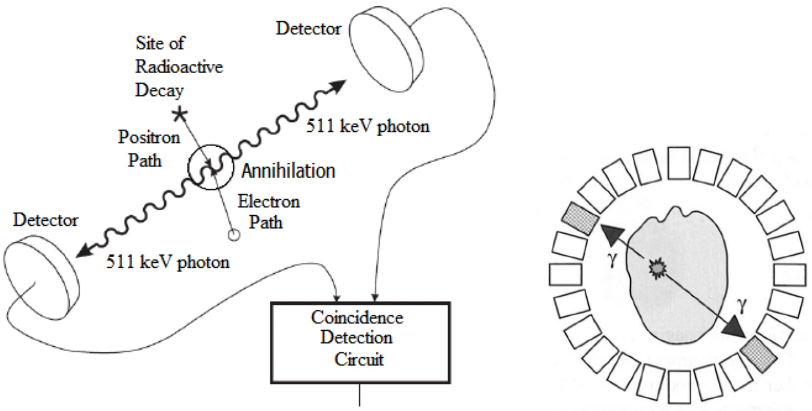
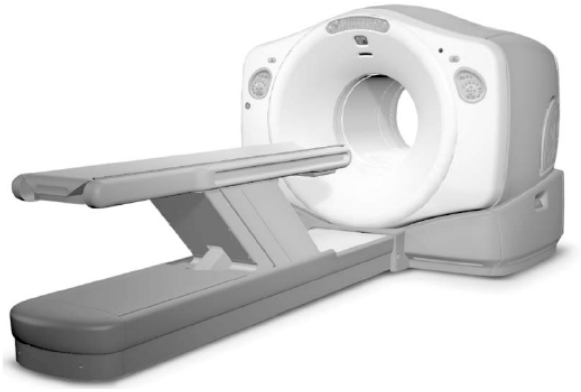


Figure 10: Photon-pair emission in a positron-decay radionuclide and its detection.

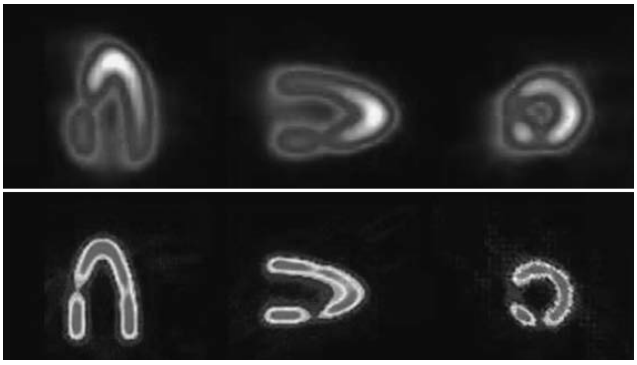
The tracer compound used in PET imaging is created by incorporating a small, positron-emitting radioisotope with a short half-life (such as carbon-11, ^{11}C (20.3 min), nitrogen-13, ^{13}N (9.97 min), oxygen-15, ^{15}O (2.03 min), and fluorine-18, ^{18}F (109.8 min)) into a metabolically active molecule (such as glucose, water or ammonia). Such labeled compounds are known as *radiotracers*. When a positron is emitted within a patient, it travels up to several millimeters while losing its kinetic energy. The positron finally meets with an electron, which leads to annihilation, producing two high-energy (511-keV) photons (γ-rays) traveling from the annihilation site in opposite directions. The patient is surrounded by multiple rings of gamma photon detectors, so that no detector rotation is required (see Figure 10).

In a PET camera, when an incident photon hits on a detector, it generates a timed pulse. These pulses are then combined in coincidence circuitry, and if the pulses fall within a short time-window (around 2-20 nanoseconds), they are deemed to be coincident. Thus, a pair of photons is supposed to be a product of an annihilation event, if their pulses are coincident. Two coincidence γ-photons represent a straight line, and the original positron-emitting radiopharmaceutical must be somewhere along that line. In this way, positional information is gained from the detected radiation without the need for a physical collimator.

A typical PET system is illustrated in Figure 11(a). PET images (Figure 11(b)) have less noise and better spatial resolution than planar scintigraphy and SPECT images [Dou09]. Currently, PET scans are most commonly used to detect cancer, heart problems (such as coronary artery disease and damage to the heart following a heart attack), brain disorders (including brain tumors, memory disorders, seizures) and other central nervous system disorders.



(a)



(b)

Figure 11: (a) An example of PET scans system. (b) A realistic heart phantom imaged along three axes by SPECT with ^{99m}Tc (top row) and PET with 18F-fluorodeoxyglucose (bottom row) (source: [Dou09]).

0.1.3 Medical images obtained with non-ionizing radiation

This section describes two medical imaging modalities obtained with non-ionizing radiation: Ultrasound Imaging and Magnetic Resonance Imaging.

0.1.3.1 Ultrasound Imaging

Ultrasound imaging (sonography) uses high-frequency (around 2-20 MHz) sound waves to view soft tissues such as muscles and internal organs.

In an ultrasound exam, a hand-held transducer is placed in direct contact with the patient's body. The transducer performs several functions as follows. The first function is to sends out the ultrasound pulses when electrical pulses are applied to it. When a beam of ultrasound pulses is passed into a body, most of the ultrasound energy is absorbed and the beam is attenuated. Some of the pulses will be reflected by internal body structures and send echoes back to the surface where they are

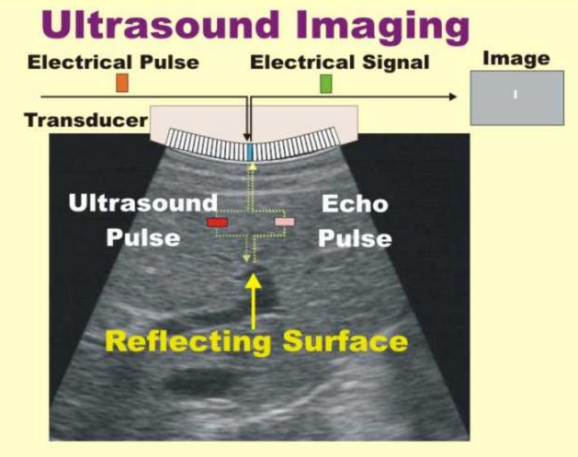


Figure 12: The basic ultrasound imaging process (source: http://www.sprawls.org/ppmi2/USPRO/)

picked up by the transducer and converted back into electrical pulses that are then processed by the system and formed into an image. The image is based on the frequency and strength (amplitude) of the returning sound signal and the time it takes to return from the patient to the transducer. Therefore, the ultrasound image is a display showing the location of reflecting structures or echo sites within the body. The location of a reflecting structure (interface) in the horizontal direction is determined by the position of the beam. In the depth direction, it is determined by the time required for the pulse to travel to the reflecting site and for the echo pulse to return. The basic ultrasound imaging process is illustrated in Figure 12.

There are three modes of ultrasound imaging including: (1) b-mode (the most common mode) - the basic two-dimensional intensity mode, (2) m-mode - to assess moving body parts (e.g. cardiac movements), and (3) Colour mode - pseudo colouring based on the detected cell motion using Doppler analysis.

Main advantages of Ultrasound imaging are (a) non-radiation, (b) inexpensive, (c) excellent for cyst, foreign bodies and obstetric imaging, and (d) real time imaging [WBO08]. However, Ultrasound image contains more noise content than any other imaging modality, especially speckle noise.

0.1.3.2 Magnetic Resonance Imaging

Magnetic resonance imaging (MRI) is a non-ionizing medical imaging technique that uses strong magnetic fields (around 1-2 tesla (T)) and radiofrequency (200MHz-2 GHz) electromagnetic radiation to produce cross-sectional images of organs and internal structures in the body.

In most MRI devices, the strong magnetic fields around a patient's body are created by superconducting magnets, in which electric current is passed through coils of superconducting wire whose electrical resistance is virtually zero. Human body is made up of atoms. Thus, when the atoms in the body are exposed to a strong magnetic field, they line up parallel to each other. Radio waves are sent from a transmitter then briefly knock the atoms out of alignment. As they realign, the atoms emit tiny signals, which are received by receiver in the machine, and these signals are used to produce digital images of the area of interest.

MRI provides good contrast between the different soft tissues of the body, which makes it especially useful in imaging the brain, muscles, the heart, and cancers compared with other medical imaging techniques such as X-ray radiography, Ultrasound, and CT.

0.2 Objectives

The motivation of this thesis is to deal with the problem of image quality for medical imaging. As mentioned above, there are many different factors which affect the quality of medical images in which especially noise and spatial resolution are important factors. On the one hand, low-resolution reduces the level of detail discernible in image, e.g. structures of smaller size than the image resolution would not be detected, or smaller objects that are adjacent to one another would not be differentiated. On the other hand, noise which is the major limiting factor of medical image quality, may affect adversely the contrast and the visibility of details which could be of vital information, thus compromising the accuracy and the reliability of pathological diagnosis or surgery purposes.

In the scope of this thesis, we focus on denoising and enhancing spatial resolution which can help improving the image quality and accuracy of useful information and hence the diagnosis.

Generally, denoising allows reducing or removing noise in an image, restoring actual information, and helping the medical experts to reliably distinguish useful information. However, due to spatial resolution of the acquisition system, certain details or features which could be of crucial information and interest the experts, would not be discriminated by denoising. Enhancing spatial resolution is then an alternative solution to improve the resolution, i.e. to detect and discriminate the smallest possible details that can be seen. Super-resolution is general technique to alleviate this problem. Until now, existing super-resolution methods offer promising performances, but these methods can not effectively deal with noisy images. The challenge is also to incorporate denoising and super-resolution in the same framework

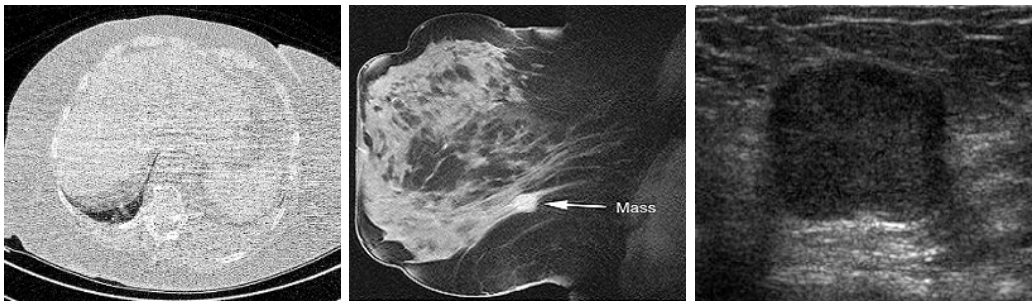
offering the possibility to improve spatial resolution with the best quality as possible.

The objective of this work is to investigate effective methods for denoising and super-resolution in order to improve the quality and the spatial resolution of medical images. Furthemore, we are motivated by the challenge to integrate denoising and super-resolution in the same framework.

The problem of image denoising and the problem of image super-resolution can be described in the following subsections.

0.2.1 Image Denoising Problem

Noise is the unwanted, random fluctuations in the pixel values of an image, result in the degradation of the image quality [Dou09] (as can be seen in Figure 13). Denoising is thus an important application of image processing to improve the quality of image. To propose an effective denoising method which would be consistent with medical image data, understanding the characteristics of noise is important. In the following part, we briefly present the characteristics of noise on the medical imaging modalities.



(a) CT image of abdomen (b) MRI image of breast (c) US image of high-grade breast cancer

Figure 13: Noisy images, source (http://rad.usuhs.edu)

0.2.1.1 Noise properties in medical imaging

In a digital imaging system, there are two principal sources of noise that are *photon noise* (also known as *quantum noise*), which arises from the discrete nature of electromagnetic radiation and its interactions with matter, and *electronic noise* in detectors or amplifiers. In addition, the process of digitization is also a cause for adding noise (quantization noise) to an image. Quantum noise usually obeys the Poisson distribution law, and electronic noise is almost Gaussian. In fact, the noise produced within an imaging system is a combination of several noise sources, and it may not be possible to identify them clearly.

In medical X-ray and γ-ray imaging systems, the image is constructed based on the number of photons detected per unit time. Thus, the image has a spatial and temporal randomness. The quantum noise is a fundamental and unavoidable noise source in medical imaging. In a good medical imaging system, quantum noise is the dominant source of random fluctuation [Dou09]. Due to the fact that quantum noise is characterized by Poisson statistics, Poisson noise can be found in all types of images formed with X-rays or γ-rays. However, in the case of low-dose CT images, the noise distribution has an approximated Gaussian functional, instead of usually assumed Poisson distribution [LS92, LLHL02].

In magnetic resonance imaging, the signal is measured through a quadrature detector that gives the real and the imaginary signals. The noise in each signal is assumed to have a Gaussian distribution with standard deviation σ. The real (R) and the imaginary (Im) images are reconstructed from the acquired data by the complex Fourier transform. Because the Fourier transform is a linear and orthogonal transform, it will preserve the Gaussian characteristics of the noise. However, the final magnitude image is formed by calculating the magnitude ($\sqrt{R^2 + Im^2}$), pixel by pixel, from the real and the imaginary images. This is a nonlinear mapping and therefore the noise distribution is no longer Gaussian, it is known as the Rician noise [GP95].

In medical ultrasound imaging, the image quality is often degraded by an inherent imaging artifact called speckle noise. This degradation is unavoidable interference effects, which will be caused by scattering of the ultrasound beam from tissue inhomogeneities [AT79]. An example of ultrasound image corrupted by speckle noise is shown in Figure 13(c).

0.2.1.2 Image denoising

Denoising is an essential step to improve the image quality. It is also a required pre-processing step in many applications in image processing and pattern recognition, from simple image segmentation tasks to higher-level computer vision ones as tracking and object detection. Therefore, estimating an image that is degraded by noise has been of high interest to a wide community of researchers.

Basically, the goal of image denoising is to effectively remove noise, while retaining important features such as edges and fine details as much as possible. In the domain of image denoising, a lot of effective methods have been proposed such as the total variation-based methods, sparcifying transform-based methods or non-local means-based methods. However, most of the state-of-the-art methods are designed for removing Gaussian white noise, while as mentioned above, noise in medical image is often very complex. Moreover, unlike conventional denoising methods, medical

image denoising needs to satisfy some following conditions:

1. It has to be consistent with the type of noise in the image.

2. It should effectively denoise while preserving edges, as well as small structures as much as possible, because subtle details may reveal critical pathological information.

Although many denoising methods have been proposed, satisfying the two conditions above is still a hard challenge. Therefore, studying effective denoising methods adapted to each specific type of medical imaging remains one of the challenges in the study of medical imaging.

0.2.2 Super-Resolution Problem

Essentially, image resolution is the ability to differentiate small objects. Precisely, it refers to the size of the smallest detail that can be distinguished in an image. In medical imaging, image resolution is limited by several factors such as hardware, time limitations and so forth, leading to a degradation of image quality. In many cases, the acquired images have to be upsampled to match to a specific resolution. In such cases, image interpolation techniques such as B-spline interpolation, cubic interpolation have been traditionally applied. However, the resulting images often suffer from blurring especially in edges. Moreover, the classical interpolation methods are often not effective in the case of image being affected by noise.

Super-Resolution (SR) methods are known as more effective methods to enhance image resolution. The term *super-resolution* is used for naming any technique that exploits the knowledge contained in one or several low-resolution (LR) images to produce an image that has a higher spatial resolution than that of the original images. Up to now, a large number of SR approaches have been developed. Broadly, these methods may be divided into two main groups: multi-image super-resolution and single-image super-resolution.

1. Multi-image SR methods usually use information extracted from a few LR images of the same scene. The first and most important step of these methods is motion estimation, or registration between LR images. Precision of the estimation is crucial for the success of the whole method. Most of the multi-image SR methods assume that shifts are exactly known. Such assumption is only feasible when considering a static scene. Indeed, in medical imaging, it is not easy to estimate accurately the motion between LR images, because it may involve not only horizontal and vertical shifts, but also rotation angles and possibly other complex movements.

2. In the second group, also known as example-based image SR techniques or single-image SR, a high resolution (HR) image is estimated from only one LR image, with the help of one or more example images. This approach is very popular in recent years. The advantage of the single-image SR methods is that they do not require LR images taken from the same scene as well as registration. Nevertheless, the limitation of this approach is that it can be effective only if the example images are similar to the image to be enhanced.

We are interested in the example-based SR techniques of the second group. This is mainly due to two reasons:

- First, this approach does not require motion estimation.
- Second, the example-based approach is a learning approach which requires having examples from data. This is possible in medical imaging, because many images of different modalities are available. Good quality medical images can be used as examples and for training.

We aim at proposing novel and effective example-based SR solutions which should be consistent with medical imaging.

0.3 Our Contributions

This thesis introduces some effective methods for denoising and super-resolution of medical images. Our proposed methods in this work are based on the interesting observation that lots of images are taken from the same type of subject at similar locations, and many of them can be considered as *standard images* (high quality or acceptably and proven as noiseless images or passed a preprocessing step) by experts. These images can be referred to as examples. Note that these images can be taken on different patients. Therefore, it will be very helpful if we could use a collection of such standard images for improving spatial resolution as well as removing noise on other images. In the proposed methods, we assume that with a given input image, there exists a set of standard images acquired at nearly the same location. These images will be used as prior information in the proposed models, making them consistent with medical images. By this way, we can also take the benefits offered by the redundancy of information, i.e., the repetition of local structures of images that tends to recur many times inside the images taken at nearby location in the human body. For our models, we adopt learning approach to take account for the available standard images also referred to as examples, in a convenient way and take advantages of this approach. Our proposed models are all dictated by

Table 1: The applications of the proposed denoising methods

Method	Type of medical imaging						
	CT	X-PR	FC	MG	PET/SPECT	MRI	US
KRRD	*Yes*					*Yes*	
kNND	*Yes*	*Yes*	*Yes*	*Yes*	*Yes*		
SWMGD	*Yes**						

example-based and learning approach. So, in this thesis, we propose three example-based methods for denoising and two example-based methods for super-resolution with the ability of dealing efficiently with noisy images.

Our contributions related to denoising and super-resolution are summarized in the next section.

0.3.1 Denoising Methods

To cope with noise, we propose three novel learning-based methods:

1. Kernel Ridge Regression-based Denoising method (KRRD)
2. k-Nearest Neighbor-based Denoising method (kNND)
3. A Sparse Weight Model for Gaussian Denoising (SWMGD).

The details of these methods are described in Chapter 2. Based on statistics of noise distributions on medical imaging modalities, the applications of the proposed denoising methods can be summarized as in Table 1. Note that in Table 1 we use notation Yes^* for SWMGD to indicate that it is a good method for CT image denoising compared with the other two methods.

In the following, we present the main idea of the proposed methods.

0.3.1.1 The first method (KRRD)

We have proposed a new approach for reducing Gaussian noise and Rician noise in medical images, based on the kernel ridge regression. In the proposed method, denoising is performed pixel-wise, and consists of two independent phases: *training phase* and *denoising phase*.

Training phase: The training phase is performed in the three steps as follows:

1. A training set is established from the database of standard images for a given noise (type of noise and noise level).

2. A robust classification of the training set is performed to obtain different groups according to some image features, namely region homogeneity, texture/edge zones and luminance.
3. A nonlinear regression function is determined for each training group.

The training phase can be realized off-line. The result of this phase is a set of the trained regression functions. These functions learn the relationships from the examples of the training set.

Denoising phase: In this phase, the denoising of an input noisy image is performed based on the regression functions trained in the training phase. This method can be consistent with the Gaussian noise or Rician noise.

0.3.1.2 The second method (kNND)

We introduce another learning-based solution for the problem of denoising in medical images. This method is proposed to overcome some limitations of the first method. In particular, this method avoids using simulated noise in the training phase as in the first method. Hence, the training set of this method is constructed by another way compared with the first method. Unlike the first method where learning of the regression functions is performed off-line through the training phase, in this method the learning the regression function and denoising for a given input are performed in parallel based on the k nearest neighbors (with respect to the input) among the training examples. A new model for determining the regression function is also proposed. Denoising by this method is efficient for Gaussian noise or Poisson noise.

0.3.1.3 The third method (SWMGD)

This is a denoising solution via sparse representation. This method is designed for removing Gaussian noise, e.g. for CT images. As in two previous methods, we use in this method a database of good image patches selected from a given set of standard images. Here, denoising is performed patch-wise with the help of the constructed database. Our main idea is that the desired noise-free patch can be represented as a sparse positive linear combination of the example patches in the database. To realize this idea, we establish an optimization problem such that its solution is a sparse solution in which most of the non-zero coefficients of the solution vector corresponds to the patches in the database which can be considered as the good candidates for the denoised output. In this method we use the assumption that on each image patch, noise can be locally approximated by Gaussian distribution. Note that, the noise levels of different patches may be different. Thus, one of the

advantages of this method is that it can effectively deal with the case of images corrupted by an additive noise where noise level can be unstable.

0.3.2 Single-Image Super-Resolution Methods

The single-image super-resolution approach, also referred to as example-based super-resolution, is very popular in recent years by its effectiveness. Example-based methods perform super-resolution on each image patch with the help of database of high-resolution and low-resolution patch pairs. Many methods have been proposed in the literature, the difference between them often comes from the example database construction and the super-resolution model on patch.

In this thesis, we focus on studying this approach for improving medical image resolution. We propose two example-based methods for single-image super-resolution. In our methods, the database is constructed such that it can be effectively utilized even if the low-resolution image is a noisy one. The proposed methods can be applied to noiseless images or noisy images. We assume that noise can be approximated as having a Gaussian distribution such as CT images or high intensity MRI images.

0.3.2.1 The first SR method (SPOCH)

This is a geometric method, where the projection of a point onto the convex hull of a finite set of points in high dimension space is first applied to the problem of single image super-resolution. The method can be considered as a k-nearest neighbor-based approach. As in the other k-nearest neighbor-based methods, to perform super-resolution for a given input low-resolution patch, a subset of k best candidates for the desired output patch is searched from the database of high-resolution patches. In particular, we proposed a statistical criterion for the similarity in the k-nearest neighbor search. Then, the output HR patch is estimated from these k best candidates. However, unlike the other methods, in our proposed method, the image patches are regarded as points in a high dimensional vector space. To estimate the output, we assume that the high-resolution point lies on the convex hull of the k candidate points and closest to the input point. Consequently, the projection of a "coarse" HR the input point onto the convex hull of the HR candidate points is defined or estimated as the true HR output.

Although the experimental results have demonstrated the performance of the proposed method over some other state-of-the-art methods, the proposed method is highly dependent on the similarity of the input LR image with the example images. Another drawback is that it often requires a large database of HR and LR patch

pairs. Moreover, this method is often limited by the choice of k and the quality of the nearest neighbors. Due to blurring, down-sampling, and noisy data, it is very difficult to propose an ideal metric for measuring the similarity between image patches.

0.3.2.2 The second SR method (SROW)

The second method has been proposed to overcome some of restrictions in the first method. We proposed a positive sparse representation model for perform super resolution on image patch. Especially, by adapting the SWMGD denoising model (the third denoising model in subsection 0.3.1.3) in the SR framework, this model provides very efficiently super-resolution solution for noisy images. Thus, this method is very useful in the case of noise present in the low-resolution image. The results of the proposed method for medical images are very promising, demonstrating the ability of the method for the potential benefit of diagnosis.

0.4 Thesis organization

The thesis is organized as follows:

Chapter 1: This chapter presents some recent state-of-the-art denoising methods on spatial domain and their applications in medical image denoising.

Chapter 2: This chapter introduces the proposed methods for medical image denoising. First, we describe the denoising method using the kernel regression techniques, where we present, in details, how to construct a denoising machine through learning the regression from a training set. Then, we present the second method where denoising is performed through learning on k-nearest neighbors. Finally, the denoising method by sparse weight model is presented. The proposed methods are evaluated using several objective and subjective comparative tests with respect to some state-of-the-art methods. We also give in this chapter a comparative study of the three proposed denoising methods.

Chapter 3: This chapter first describes an overview of the single-image image super resolution approaches. The proposed methods are then presented and compared with some other methods with both objective and subjective evaluations.

Chapter 4: This chapter provides a summary of the work presented in the previous chapters, the contributions to knowledge already achieved in this research, and directions for future work.

0.5 Publications

Journal Paper:

1. **D. H. Trinh**, M. Luong, F. Dibos, J. M. Rocchisani, C. D. Pham, H. D. Pham, *"An Optimal Weight Method for CT Image Denoising,"* Journal of Electronic Science and Technology, Vol. 10, Issue: 2, pp.124–129, 2012.

International Conference Papers:

1. **D. H. Trinh**, M. Luong, F. Dibos, C. D. Pham, J. M. Rocchisani, H. D. Pham, *"An Optimal Weight Model for Single Image Super-Resolution,"* International Conference on Digital Image Computing: Techniques and Applications (DICTA-2012), pp. 1–8, Fremantle, Australia, 2012.

2. **D. H. Trinh**, M. Luong, C. D. Pham, F. Dibos, H. D. Pham, J. M. Rocchisani, *"Image Resolution Enhancement by Projection Onto Convex Hull,"* the 12th IEEE International Symposium on Signal Processing and Information Technology (ISSPIT-2012), Ho Chi Minh City, Vietnam, December 2012.

3. **D. H. Trinh**, M. Luong, J. M. Rocchisani, C. D. Pham, F. Dibos, *"Adaptive Medical Image Denoising using Support Vector Regression,"* the 14th International Conference on Computer Analysis of Images and Pattern(CAIP-2011), Part I, LNCS 2011, Springer-Verlag Berlin, Volume 6854/2011, pp. 494–502, 2011.

4. **D. H. Trinh**, M. Luong, J. M. Rocchisani, C. D. Pham, F. Dibos, *"Medical Image Denoising using Kernel Ridge Regression,"* the 18th IEEE International Conference on Image Processing (ICIP-2011), pp. 1597–1600, Brussels, Belguim, 2011.

5. **D. H. Trinh**, M. Luong, J. M. Rocchisani, C. D. Pham, F. Dibos, L. T. Nguyen, *"MR Image Denoising using Non-linear Regression and Fuzzy C-means Clustering,"* IEEE International Conference on Advanced Technologies for Communications (ATC-2011), pp. 256–259, Vietnam, 2011.

Chapter 1

Survey of image denoising approaches

Contents

1.1 Introduction

Today, with the widespread use of imaging in medicine, in particular for pathologic diagnosis and survey, the quality of medical images becomes an important issue. To achieve the best possible diagnosis it is important that medical images be sharp, and free of noise and artifacts. While the technologies for acquiring medical images

continue to improve, resulting in images of higher and higher resolution and quality, noise remains an issue for many medical images. Generally, medical images often get complex types of noise introduced by various acquisitions, transmission, storage, display devices. Denoising is a crucial task as noise may lead to improper diagnosis. Therefore, removing noise remains one of the major challenges in medical imaging.

Given a noisy version $\mathbf{f}$ of the free-noise image $\mathbf{u}$ corrupted by noise $\mathbf{n}$, the image denoising problem is to recover $\mathbf{u}$ from its noisy version $\mathbf{f}$. To this end, many solutions have been proposed such as [LFB94, PM90, ROF92, BCM06, EA06, TFM07, DFKE07, DD12]. The denoising solutions derive from various disciplines such as linear and nonlinear filtering, spectral and multiresolution analysis, probability theory, statistics, partial differential equations. These methods rely on some explicit or implicit assumptions about the true image in order to separate it from noise component. In medical imaging, edges, textures and subtle details could very well reveal crucial information about the patients. Therefore, denoising without affecting the image details is always of high interest topic in the medical image processing domain.

The classical filters such as the mean filter, the median filter, the Gaussian filter [LFB94], etc. are only effective for homogeneous regions. Even though these methods are often simple and easy to apply, their effectiveness is limited since these often lead to blur or over-smoothing in high frequency regions such as edges and textures. In order to overcome this drawback, many edge preserving filters have been proposed. The anisotropic diffusion filter [PM90] attempts to preserve edges by convolving the image in the orthogonal direction of the local gradient. Although straight edges can be well preserved, curved edges or features are usually degraded [BCM06]. In the last decade, many state-of-the-art denoising approaches have been proposed including such as the Total Variation-based methods [ROF92, BKP10, KBPS11], the Non-local means-based filters [BCM05, BCM06], the Data-adaptive Kernel Regression-based methods [TFM07], the Sparse Representation-based methods [EA06, AEB06, DlZS11], the Block matching with 3D filtering (BM3D) [DFKE07], and many of them have been proven to be effective in reducing some common types of noise (e.g. Gaussian noise, Poisson noise, Rician noise, speckle noise) in medical images.

The purpose of this chapter is to briefly describe some recent advances in the image denoising domain and the applicability of them for reducing noise of medical images. First of all, we give in Section 1.2 the basic definition of continuous and discrete images that will be used in the sequel of this thesis. Then, several common noise models which are found in medical images are presented in Section 1.3. Section 1.4 presents some common image quality metrics used to evaluate

the performance of image processing methods. Some recent state-of-the-art image denoising approaches are presented in Section 1.5. Finally, the conclusion is given in Section 1.6.

1.2 Image Definition

We start by expressing a mathematical representation of an image in a discrete and continuous setting.

Definition 1.2.1 (Continuous Image [[Mod04], Def. 3.1])
Let $\Omega \subset \mathbb{R}^d$, $d \in \mathbb{N}$, be the image spatial domain. A function $\mathbf{u} : \Omega \to \mathbb{R}$ is called a d-dimensional image if the following conditions are fulfilled,
(1) $\mathbf{u}$ has a compact support, if Ω is not bounded,
(2) $0 \leq \mathbf{u}(x) < \infty$ for all $x \in \Omega$,
(3) $\int_\Omega \mathbf{u}(x)dx < \infty$.

This representation of images is an elegant way to deliver a simple basis for the analysis and construction of mathematical methods. It is actually only an idealization, which cannot be realized on any computer and does not correspond to the reality of applications. Therefore, we are also interested in digital image.

Definition 1.2.2 (Discrete Image)
Let $h > 0$ and $n_1, n_2, \ldots, n_d \in \mathbb{N}$. A grid matrix $\mathbf{u} \in \mathbb{R}^{n_1 \times n_2 \times \ldots \times n_d}$ is called a d-dimensional image if the following conditions are satisfied,
(1) $0 \leq \mathbf{u}(i_1, i_2, \ldots, i_d) < \infty$, $(i_1, i_2, \ldots, i_d) \in \Omega = \{1, \ldots, n_1\} \times \ldots \times \{1, \ldots, n_d\}$,
(2) $\sum_{i_1=1}^{n_1} \ldots \sum_{i_d=1}^{n_d} \mathbf{u}(i_1, i_2, \ldots, i_d) < \infty$.

Here, $\mathbf{u}(i_1, i_2, \ldots, i_d) \in \mathbb{R}$ describes the intensity values of a digital image at the nodal points $x_{i_1,i_2,\ldots,i_d} = (i_1 h, i_2 h, \ldots, i_d h)$ of the pixel grid $\mathbf{x} = (x_{i_1,i_2,\ldots,i_d})$, the parameter h controls the resolution of the image. In Figure1.1, h controls the horizontal and vertical distance of the pixel $x_{i_1,i_2,\ldots,i_d}$.

In the following, we list some common models of random noise found in medical images, including the additive Gaussian noise, the Poisson noise, the Rician noise and the Speckle noise.

1.3 Image noise models

Let us denote by $\mathbf{f} : \Omega \to \mathbb{R}$ the observed image (noisy image) and $\mathbf{u}$ the ground truth image (noise-free image). As $\mathbf{f}$ is corrupted by random noise, each pixel value

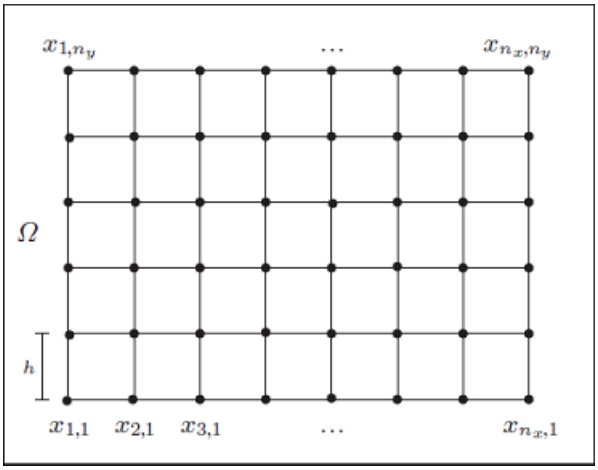


Figure 1.1: Pixel grid with nodes in the case of 2D-image

$\mathbf{f}(x)$ of $\mathbf{f}$ is the realization of a random variable F_x. We will denote by $F = (F_x)$ the random vector consisting of independent random variables F_x and $p_F(\cdot)$ the probability density function (PDF) of F. Usually, it is assumed that the random variables F_x are statistically pairwise independent and identically distributed. Then, the conditional probability density function, $p_F(\mathbf{f}|\mathbf{u})$, can be written as

$$p_F(\mathbf{f}|\mathbf{u}) = \prod_{x\in\Omega} p_{F_x}(\mathbf{f}(x)|\mathbf{u}(x)). \tag{1.1}$$

where p_{F_x} is the PDF of F_x.

1.3.1 Additive Gaussian noise

A random variable F is said to have a Gaussian distribution if its PDF is

$$p_F(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, \tag{1.2}$$

where the parameters μ and σ^2 are the mean (or expectation) and the variance, respectively. The normal or Gaussian distribution of F is usually represented by,

$$F \sim \mathcal{N}(\mu, \sigma^2).$$

An image corrupted by Gaussian noise is often modeled as

$$\mathbf{f} = \mathbf{u} + \mathbf{n}, \tag{1.3}$$

with $\mathbf{n}(x) \sim \mathcal{N}(0, \sigma^2)$ is noise component. Then, the conditional PDF (1.1) for Gaussian noise is

$$p_F(\mathbf{f}|\mathbf{u}) = \prod_{x\in\Omega} p_{F_x}(\mathbf{f}(x)|\mathbf{u}(x)) = e^{-\int_\Omega \left(\ln(\sigma\sqrt{2\pi}) + \frac{1}{2\sigma^2}(\mathbf{f}(x)-\mathbf{u}(x))^2\right)dx}. \tag{1.4}$$

In medical imaging, noise may come from different sources such as image acquisition, transmission, storage, display devices, etc. According to the central limit theorem which states that the sum of many random variables with various probability density functions results in a variable with a Gaussian PDF. Therefore, Gaussian noise can be found in most of medical images, especially in CT images [LS92, LLHL02] and sometimes in high intensity MRI [Now99].

1.3.2 Poisson noise

Noise in the image $\mathbf{f}$ is assumed to have a Poisson distribution if for each pixel $x \in \Omega$ we have

$$p_{F_x}(\mathbf{f}(x)|\mathbf{u}(x)) = \frac{e^{-\mathbf{u}(x)}\mathbf{u}(x)^{\mathbf{f}(x)}}{\mathbf{f}(x)!}. \tag{1.5}$$

Thus,

$$p_F(\mathbf{f}|\mathbf{u}) = \prod_{x\in\Omega} \frac{e^{-\mathbf{u}(x)}\mathbf{u}(x)^{\mathbf{f}(x)}}{\mathbf{f}(x)!}. \tag{1.6}$$

Furthermore, the expectation value of the Poisson variable $\mathbf{f}(x)$ is $\mathbf{u}(x)$ and equal to its variance

$$\mathrm{E}\{\mathbf{f}(x)|\mathbf{u}(x)\} = \mathrm{Var}\{\mathbf{f}(x)|\mathbf{u}(x)\} = \mathbf{u}(x). \tag{1.7}$$

In medical imaging, the Poisson noise arises in systems involving counting procedures such as in standard X-ray projection radiography [GBG04], Fluoroscopy, Mammography, PET/SPECT [VSK85, OF97].

Poisson noise component can be formally defined as

$$\eta(x) = \mathbf{f}(x) - \mathrm{E}\{\mathbf{f}(x)|\mathbf{u}(x)\}. \tag{1.8}$$

It is easy to see that $\mathrm{E}\{\eta(x)|\mathbf{u}(x)\} = 0$ and $\mathrm{Var}\{\eta(x)|\mathbf{u}(x)\} = \mathrm{Var}\{\mathbf{f}(x)|\mathbf{u}(x)\} = \mathbf{u}(x)$. Since the noise variance depends on the true intensity value, Poisson noise is signal dependent. We will show that it is possible to transform the Poisson process in a standard normal process.

1.3.2.1 Variance stabilizing transformation

Due to the data-dependence of the noise variance, the Poisson denoising thus becomes more difficult. One of the most popular solutions is to use a variance stabilizing transformation. We introduce here the Anscombe transformation [Ans48]:

$$\mathcal{T}(z) = 2\sqrt{z + \frac{3}{8}}. \tag{1.9}$$

Applying the Anscombe transformation to the noisy image $\mathbf{f}$ gives a new image whose noise is asymptotically additive standard normal. Thus, one can use a Gaussian denoising method to the transformed image $\mathcal{T}(\mathbf{f})$; the final estimate is then obtained by applying an inverse Anscombe transformation to the denoised data. In the following subsection we will present some inverse transformation of the Anscombe transformation.

1.3.2.2 Unbiased inverse transformations

Assume that D is the denoised version of $\mathcal{T}(\mathbf{f})$. Then, D can be considered as an estimate of $\mathrm{E}\{\mathcal{T}(\mathbf{f})|\mathbf{u}\}$. In order to obtain the desired estimate of $\mathbf{u}$, we need to apply an inverse transformation of $\mathcal{T}$ to D.

The direct algebraic inverse of $\mathcal{T}$ in (1.9), denoted as $\mathcal{V}_A$, is

$$\mathcal{V}_A(D) = \mathcal{T}^{-1}(D) = \left(\frac{D}{2}\right)^2 - \frac{3}{8}. \tag{1.10}$$

Due to the nonlinearity of $\mathcal{T}$, we generally have

$$\mathrm{E}\{\mathcal{T}(\mathbf{f})|\mathbf{u}\} \neq \mathcal{T}(\mathrm{E}\{\mathbf{f}|\mathbf{u}\}). \tag{1.11}$$

Thus,

$$\mathcal{T}^{-1}(\mathrm{E}\{\mathcal{T}(\mathbf{f})|\mathbf{u}\}) \neq \mathrm{E}\{\mathbf{f}|\mathbf{u}\}. \tag{1.12}$$

$$\Rightarrow \mathcal{T}^{-1}(D) \neq \mathbf{u}. \tag{1.13}$$

Therefore, $\mathcal{V}_A(D)$ is a biased estimator of $\mathbf{u}$.

Another inverse transformation that provides asymptotic unbiasedness for large counts is [Ans48]

$$\mathcal{V}_B(D) = \left(\frac{D}{2}\right)^2 - \frac{1}{8}. \tag{1.14}$$

However, applying this transformation to low-count data leads to a biased estimate [MF11]. Thus, there is a need for an exact unbiased inverse of $\mathcal{T}$.

In [MF11], Makitalo and Foi have defined that the exact unbiased inverse of the Anscombe transformation $\mathcal{T}$ is an inverse transformation $\mathcal{V}_C$ that maps the values $E\{\mathcal{T}(\mathbf{f})|\mathbf{u}\}$ to the desired values $E\{\mathbf{f}|\mathbf{u}\}$

$$\mathcal{V}_C \ : \ \mathrm{E}\{\mathcal{T}(\mathbf{f})|\mathbf{u}\} \longmapsto \mathrm{E}\{\mathbf{f}|\mathbf{u}\}. \tag{1.15}$$

Since $\mathrm{E}\{\mathbf{f}|\mathbf{u}\} = \mathbf{u}$ for any given $\mathbf{u}$, the problem of finding the inverse $\mathcal{V}_C$ reduces to computing the values $\mathrm{E}\{\mathcal{T}(\mathbf{f})|\mathbf{u}\}$. In the discrete case, $\mathrm{E}\{\mathcal{T}(\mathbf{f})|\mathbf{u}\}$ is computed as

$$\mathrm{E}\{\mathcal{T}(f)|u\} = \sum_{f=0}^{+\infty} \mathcal{T} p_F(f|u) = 2\sum_{f=0}^{+\infty} \left(\frac{u^f e^{-u}}{f!} \cdot \sqrt{f + \frac{3}{8}}\right). \tag{1.16}$$

In practice, to implement the exact unbiased inverse $\mathcal{V}_C$, equation (1.16) is computed for a limited set of values u. Thereby, for arbitrary values of u, $\mathcal{V}_C$ is determined by using linear interpolation on these computed values. For large values of u, $\mathcal{V}_C$ is approximated by $\mathcal{V}_B$.

1.3.3 Rician noise

In probability theory, the probability density of a random variable F that has a Rice distribution is

$$p_X(x|v,\sigma) = \frac{x}{\sigma^2} e^{-\frac{x^2+v^2}{2\sigma^2}} I_0\left(\frac{xv}{\sigma^2}\right) \tag{1.17}$$

where $x, v, \sigma > 0$, $I_0(\cdot)$ is the modified Bessel function of the first kind of order zero,

$$I_0(t) = \sum_{k=0}^{\infty} \frac{(t/2)^{2k}}{(k!)^2}. \tag{1.18}$$

If $R \sim \mathcal{N}(v\cos\theta, \sigma^2)$ and $Im \sim \mathcal{N}(v\sin\theta, \sigma^2)$ are statistically independent normal random variables for any $\theta \in \mathbb{R}$, then $X = \sqrt{R^2 + Im^2}$ has a Rician distribution and is denoted by $X \sim Rice(v, \sigma)$.

As it is shown in [GP95], Rician noise was found in MRI images. In magnetic resonance imaging, the signal is measured through a quadrature detector that gives the real and the imaginary signals. The noise in each signal is assumed to have a Gaussian distribution with standard deviation σ. The real (R) and the imaginary (Im) images are reconstructed from the acquired data by the complex Fourier transform. Because the Fourier transform is a linear and orthogonal transform, it will preserve the Gaussian characteristics of the noise. However, the final magnitude image is formed by calculating the magnitude ($\sqrt{R^2 + Im^2}$), pixel by pixel, from the real and the imaginary images. This is a nonlinear mapping and therefore the noise distribution is no longer Gaussian, it is known as the Rician noise.

If the image $\mathbf{f}$ is a version corrupted by Rician noise from the clean image $\mathbf{u}$, then

$$p_{F_x}(\mathbf{f}(x)|\mathbf{u}(x)) = \frac{\mathbf{f}(x)}{\sigma^2} e^{-\frac{\mathbf{f}(x)^2+\mathbf{u}(x)^2}{2\sigma^2}} I_0\left(\frac{\mathbf{u}(x)\mathbf{f}(x)}{\sigma^2}\right). \tag{1.19}$$

Therefore,

$$p_F(\mathbf{f}|\mathbf{u}) = \prod_{x\in\Omega} p_{F_x}(\mathbf{f}(x)|\mathbf{u}(x)) = e^{-\int_{x\in\Omega}\left[\frac{\mathbf{f}(x)^2+\mathbf{u}(x)^2}{2\sigma^2} - \ln I_0\left(\frac{\mathbf{u}(x)\mathbf{f}(x)}{\sigma^2}\right) - \ln\frac{\mathbf{f}(x)}{\sigma^2}\right]}. \tag{1.20}$$

where σ denotes the standard deviation of the Gaussian noise in the real image R and the imaginary image Im.

Notice that the Rice distribution tends to have a Rayleigh distribution

$$p(\mathbf{f}) = \frac{\mathbf{f}}{\sigma^2} e^{-\frac{\mathbf{f}^2}{2\sigma^2}}, \tag{1.21}$$

when the signal-to-noise ratio (SNR) goes to zero (i.e., when $\frac{\mathbf{u}}{\sigma} \to 0$) and approaches a Gaussian distribution at high SNR (i.e., when $\frac{\mathbf{f}}{\sigma} \to \infty$) [GP95]. It means that the Rician noise distribution is well approximated by a Gaussian in bright (high SNR) regions while a Rayleigh distribution is more appropriate in dark (low SNR) regions.

1.3.4 Speckle noise

Speckle noise is often found in Ultrasound imaging. As pointed out in [TSP88], when the scatterers density (the number of scatter per resolution cell) was more than 10, this speckle noise followed a Rayleigh distribution. The density function of the Rayleigh distribution with parameter $\sigma > 0$ is given by

$$p(x) = \frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}}. \tag{1.22}$$

However, as the original signal is usually pre-processed (e.g. filtered and compressed) to improve its visualization, this pre-processing modifies the distribution of the original signal. In [LMA89, TSP88, SS08], the authors show that the speckle noise in the displayed images no longer follows the Rayleigh distribution. One of the most successful model is the Loupas *et al.*'s model in [LMA89], where it is derived experimentally that a displayed ultrasonic image $\mathbf{f}$ can be modeled as being corrupted with a signal-dependent noise such that

$$\mathbf{f} = \mathbf{u} + \sqrt{\mathbf{u}}\mathbf{n} \tag{1.23}$$

where $\mathbf{n} \sim \mathcal{N}(0, \sigma^2)$ is a Gaussian noise. Thus, we obtain

$$p_F(\mathbf{f}|\mathbf{u}) = e^{-\int_{x\in\Omega}\left(\ln(\sigma\sqrt{2\pi})+\frac{1}{2\sigma^2}\frac{(\mathbf{f}(x)-\mathbf{u}(x))^2}{\mathbf{u}(x)}\right)dx}. \tag{1.24}$$

1.4 Image quality metrices

Typically, the performance of the image processing methods is often evaluated in both objective measures and subjective quality assessment protocols. The objective measure is determined through the image quality metrics. In the following, we present two full reference image quality metrics which are used in this thesis.

1.4.1 MSE - Mean Square Error

The mean square error (MSE) metric is one of the most popular and widely used fidelity measure today. This is mainly due to its simplicity, the low computation

time and because it can be easily integrated into an optimization process such as the minimization of a cost function or a mathematical model. It is given by:

$$MSE = \frac{1}{MN} \sum_{x=1}^{W} \sum_{y=1}^{H} I_r((x,y) - I(x,y))^2, \tag{1.25}$$

where $I_r(x,y)$ and $I(x,y)$ are the reference and the image to be evaluated, respectively. W and H represent the width and height of the image, respectively. From this definition, different variants have been proposed. The signal to noise ratio (SNR) and peak signal to noise ratio (PSNR) are the most used:

$$SNR = 10 \log_{10} \frac{\sum_{x=1}^{W} \sum_{y=1}^{H} (I_r(x,y))^2}{MSE}, \tag{1.26}$$

$$PSNR = 10 \log_{10} \frac{R^2}{MSE} \tag{1.27}$$

wherein R is the maximum fluctuation in the original image. The peak value R is set equal to 255 for images coded on 8 bits per pixel. The PSNR measures the intensity difference between two images. However, it is well-known that it can fail to describe the subjective quality of the image. Since there is no universal accepted image distortion measure, the PSNR is still widely used.

1.4.2 SSIM-Structural SIMilarity

To overcome the weakness of PSNR which is pixelwise, some authors have focused their research into quality metrics based on an analysis of local structures of the image content. Among the most popular and widely used metrics, we consider the SSIM (Structural SIMilarity) proposed by Wang *et al.* [WBSS04]. This metric is based on the assumption that the human visual system is more sensitive to structural information. The local structural attributes of the image are described by luminance, contrast and structure. Luminance, contrast, and structure are measured separately.

Given two vectorized images (or image blocks) $\mathbf{x} = \{x_k,\, k = 1, 2, \ldots, N\}$ and $\mathbf{y} = \{y_k,\, k = 1, 2, \ldots, N\}$ to be compared, the *luminance* term is estimated as the mean of each image

$$\mu_x = \frac{1}{N} \sum_{k=1}^{N} x_k, \tag{1.28}$$

the *contrast* term is estimated using standard deviation defined as

$$\sigma_x = \sqrt{\frac{1}{N-1} \sum_{k=1}^{N} (x_k - \mu_x)^2}, \tag{1.29}$$

and the *structure* term is estimated from the image vector $\mathbf{x}$ by subtraction of the mean μ_x and normalizing by the standard deviation σ_x

$$s_x = \frac{\mathbf{x} - \mu_x}{\sigma_x}. \tag{1.30}$$

Then, the measurements $\mu_x, \mu_y, \sigma_x, \sigma_y, s_x, s_y$ are combined using a luminance comparison function $l(\mathbf{x}, \mathbf{y})$, a contrast comparison function $c(\mathbf{x}, \mathbf{y})$, and a structure comparison function $s(\mathbf{x}, \mathbf{y})$ to give a composite measure of structural similarity:

$$SSIM = l(\mathbf{x}, \mathbf{y})^\alpha \cdot c(\mathbf{x}, \mathbf{y})^\beta \cdot s(\mathbf{x}, \mathbf{y})^\gamma, \tag{1.31}$$

where α, β, γ are positive constants used to weight each comparison function, the comparison functions are computed as:

$$l(\mathbf{x}, \mathbf{y}) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{1.32}$$

$$c(\mathbf{x}, \mathbf{y}) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{1.33}$$

$$s(\mathbf{x}, \mathbf{y}) = \frac{\sigma_{xy}\sigma_y + C_3}{\sigma_x\sigma_y + C_3} \tag{1.34}$$

In this thesis, we follow the example in [WBSS04] by setting $\alpha = \beta = \gamma = 1$, and $C_3 = C_2/2$ to get the specific SSIM quality metric

$$SSIM = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy}\sigma_y + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{1.35}$$

where $C_1 = (L \cdot K_1)^2$, $C_2 = (L \cdot K_2)^2$ with L is the dynamic range, $K1 = 0.01$ and $K2 = 0.03$ are two constants determined by the authors to avoid instability in homogeneous regions. The advantage of this metric is that no human visual system model is explicitly employed, but its performance is considerably better than PSNR. SSIM results vary from 0 to 1 where high value indicates high image quality. The Matlab implementation of SSIM is downloadable from `https://ece.uwaterloo.ca/~z70wang/research/ssim/`.

1.5 The state-of-the-art image denoising approaches

The objective of image denoising is to recover the original image $\mathbf{u}$ from the noisy version $\mathbf{f}$. Many effective denoising methods have been proposed and can be classified into two main categories: transform domain methods and spatial domain ones. In this section we only focus on state-of-the-art image denoising approaches in the spatial domain including the total variation-based methods, the non-local means-based methods, the data-adaptive kernel regression-based methods, the sparse dictionary-based methods, and the block matching 3D-based methods. In the scope of this work, we consider only 2D grayscale images.

1.5.1 Total Variation

First, we present some basic definition of the Total Variation (TV) and the TV-based denoising problem. Then, we present the TV-based denoising problems applied for Gaussian, Poisson, Rician and Speckle noises respectively. Finally, we present a new concept of the TV, namely the Total Generalized Variation (TGV) [BKP10].

Definition 1.5.1 (see for instance [Giu84])
Let Ω be a bounded open subset of $\mathbb{R}^2$. Given an image $\mathbf{u} \in L^1(\Omega)$, the total variation $\mathbf{u}$ in Ω is defined as

$$TV(\mathbf{u}) = \sup \left\{ \int_\Omega \mathbf{u}(x) \mathrm{div} v(x) dx \;\middle|\; v \in \mathcal{C}^1(\Omega, \mathbb{R}^2), |v(x)| \leq 1 \forall\, x \in \Omega \right\}, \tag{1.36}$$

where $\mathcal{C}^1(\Omega, \mathbb{R}^2)$ is the set of continuously differentiable vector functions on Ω.

The space of functions of bounded variation is defined by

$$BV(\Omega) = \{\mathbf{u} \in L^1(\Omega) \;:\; TV(\mathbf{u}) < \infty\}, \tag{1.37}$$

where $L^1(\Omega)$ is the space of integrable functions on Ω.

If $\mathbf{u}$ has a gradient $\nabla \mathbf{u} \in L^1(\Omega)$, then

$$\int_\Omega \mathbf{u} \mathrm{div} v dx = -\int_\Omega \nabla \mathbf{u} \cdot v dx.$$

Thus,

$$TV(\mathbf{u}) = \int_\Omega \|\nabla \mathbf{u}(x)\| dx, \tag{1.38}$$

here $\nabla \mathbf{u} = (u_1, u_2)$ denotes the gradient of $\mathbf{u}$ and $\|\mathbf{y}\| = \sqrt{y_1^2 + y_2^2}$ for every $\mathbf{y} = (y_1, y_2) \in \mathbb{R}^2$.

1.5.1.1 Total Variation-based denoising

Denoising is an ill-posed problem. A good prior is necessary to obtain reasonable solutions. This prior is formulated as a regularization term in an ennery function to be minimized. Among regularizations, the total variation of a function is a popular one due to its capacity of reconstruction with discontinuity (e.g. sharp edges) preserving.

In variational framework, we start by the *maximum a posteriori* probability (MAP) model for estimating $\mathbf{u}$ from $\mathbf{f}$:

$$\hat{\mathbf{u}} = \arg\max_{\mathbf{u}} \; p_F(\mathbf{u}|\mathbf{f}). \tag{1.39}$$

Applying Bayes's theorem we obtain

$$\max_{\mathbf{u}} \; p_F(\mathbf{u}|\mathbf{f}) \Leftrightarrow \max_{\mathbf{u}} \; p_F(\mathbf{f}|\mathbf{u}) p_F(\mathbf{u}) \tag{1.40}$$

$$\Leftrightarrow \min_{\mathbf{u}} \; (-\ln p_F(\mathbf{f}|\mathbf{u}) - \ln p_F(\mathbf{u})). \tag{1.41}$$

In (1.41), the first term, $-\ln p_F(\mathbf{f}|\mathbf{u})$, describes the degradation process from $\mathbf{u}$ to $\mathbf{f}$. Thus, it depends on the model of noise in the image. The second term in (1.41), $-\ln p_F(\mathbf{u})$, is called the *prior* on $\mathbf{u}$. The choice of prior will play an important role in restoring a satisfactory image.

In the context of image reconstruction, it was first proposed by Rudin, Osher and Fatemi in [ROF92] to consider the Total Variation (TV) as regularization functional for the prior,

$$-\ln p_F(\mathbf{u}) = TV(\mathbf{u}) = \int_\Omega \|\nabla \mathbf{u}(x)\| dx. \tag{1.42}$$

The denoising problem (1.41), accordingly, becomes

$$\min_{\mathbf{u}} \int_\Omega \|\nabla \mathbf{u}(x)\| dx + \lambda \mathcal{F}(\mathbf{u}, \mathbf{f}), \tag{1.43}$$

where, λ is is a positive constant which controls the trade-off between two terms. $\mathcal{F}(\mathbf{u}, \mathbf{f})$ is called fidelity term which satisfies

$$\min_{\mathbf{u}} \mathcal{F}(\mathbf{u}, \mathbf{f}) = -\min_{\mathbf{u}} \ln p_F(\mathbf{f}|\mathbf{u}). \tag{1.44}$$

From (1.43) and the probability densities given in Section 1.3, we obtain the TV-based denoising models that will be described in the next subsections.

1.5.1.1.1 TV-based Gaussian denoising

From (1.4), we have

$$-\ln p_F(\mathbf{f}|\mathbf{u}) = \int_{x\in\Omega} \left(\ln(\sigma\sqrt{2\pi}) + \frac{1}{2\sigma^2}(\mathbf{f}(x) - \mathbf{u}(x))^2 \right) dx. \tag{1.45}$$

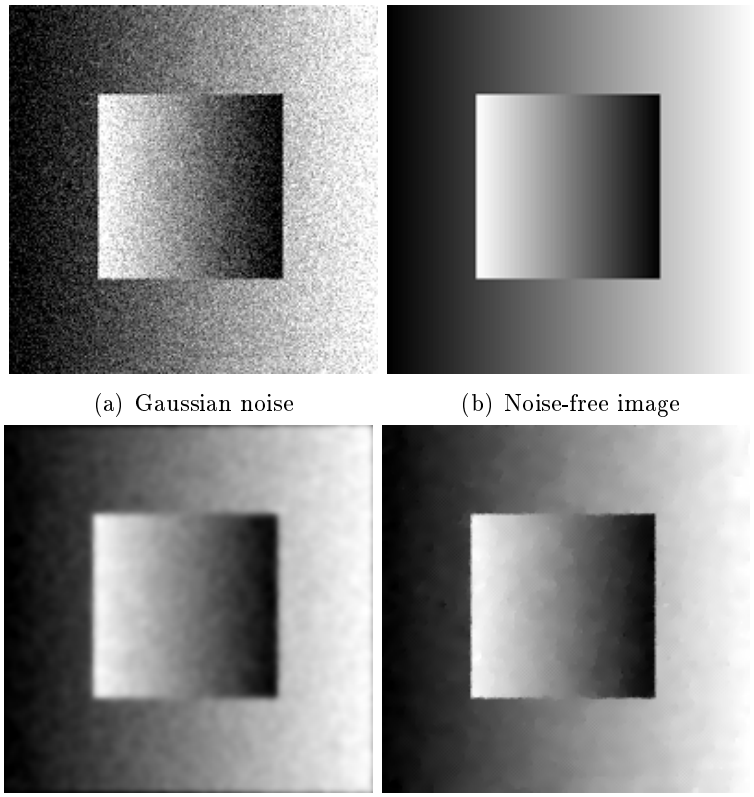
$$\Rightarrow \mathcal{F}(\mathbf{u}, \mathbf{f}) = \int_{x\in\Omega} (\mathbf{f}(x) - \mathbf{u}(x))^2 dx. \tag{1.46}$$

Therefore, the TV model for Gaussian denoising is

$$\min_{\mathbf{u}} \int_\Omega \|\nabla \mathbf{u}(x)\| dx + \lambda \int_\Omega (\mathbf{u}(x) - \mathbf{f}(x))^2 dx. \tag{1.47}$$

A lot of authors have studied the total minimization problem (1.47) and different approaches have been proposed, the readers may see for instance [ROF92, DK00, Cha04, CDV10].

Figure 1.2 and Figure 1.3 present some experimental results on a synthetic image and a CT image of pelvis, respectively. In particular, model (1.47) is solved



(a) Gaussian noise (b) Noise-free image

(c) Anisotropic diffusion filter (ADF) [PM90] (d) TV denoising [DK00]

Figure 1.2: Experiment on a synthetic image: (a) image corrupted by Gaussian noise $\sigma = 30$, (b) noise-free image, (c) Denoised image using Anisotropic Diffusion Filter (ADF) [PM90], (d) Denoised image using TV [DK00].

by the algorithm of Dibos and Koepfler [DK00]. Compared with the Anisotropic Diffusion Filter (ADF), in these figures TV gives better results. TV effectively denoises by smoothing flat regions while well preserving edges. However, we can see the staircasing effect in regions with gradual image variations.

1.5.1.1.2 TV-based Poisson denoising

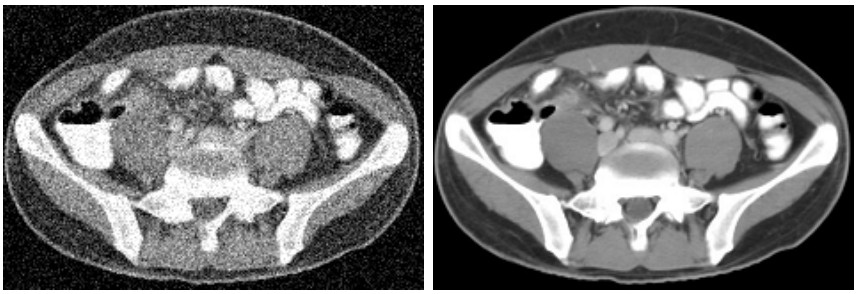
From (1.6), we have

$$-\ln p_F(\mathbf{f}|\mathbf{u}) = \int_\Omega (\mathbf{u}(x) - \mathbf{f}(x)\ln \mathbf{u}(x) + \ln(\mathbf{f}(x)!))\, dx \tag{1.48}$$

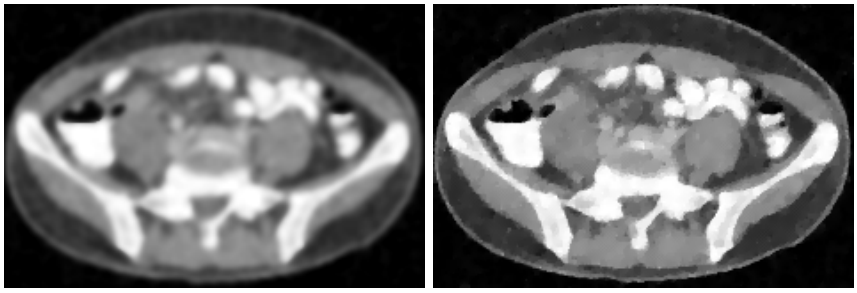
$$\Rightarrow \mathcal{F}(\mathbf{u}, \mathbf{f}) = \int_{x\in\Omega} (\mathbf{u}(x) - \mathbf{f}(x)\ln \mathbf{u}(x))dx. \tag{1.49}$$

Therefore, the TV model for Poisson denoising is

$$\min_{\mathbf{u}} \int_\Omega \|\nabla \mathbf{u}(x)\|dx + \lambda \int_\Omega (\mathbf{u}(x) - \mathbf{f}(x)\ln(\mathbf{u}(x)))dx. \tag{1.50}$$



(a) Gaussian noise $\sigma = 30$ (b) Noise-free image



(c) ADF [PM90] (d) TV [DK00]

Figure 1.3: Experiment on a CT image of Pelvis: (a) noisy image corrupted by Gaussian noise $\sigma = 25$, (b) original image (source: http://radiopaedia.org/), (c) result of the Anisotropic Diffusion Filter (ADF) [PM90], (d) result of TV [DK00].

This model can be found in [LCA07, SBMB09]. The performance of the model is illustrated in the Figure 1.4 and Figure 1.5. Although noise is effectively removed, we can still see the appearance of staircasing effects in the denoised images.

1.5.1.1.3 TV-based Rician denoising

From (1.20), we have

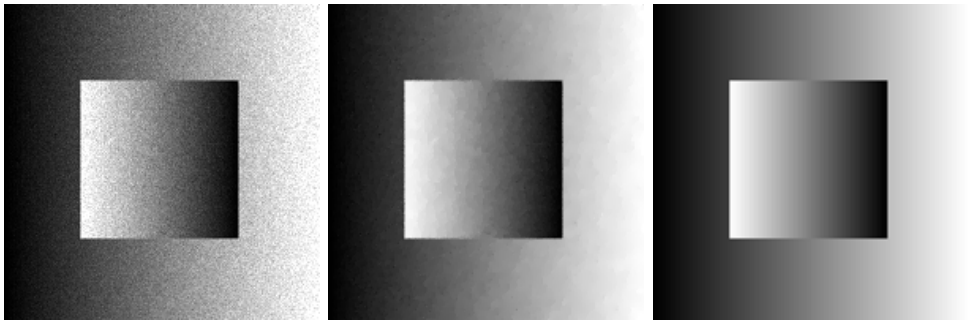
$$-\ln p_F(\mathbf{f}|\mathbf{u}) = \int_{x\in\Omega}\left(\frac{\mathbf{f}^2(x)+\mathbf{u}^2(x)}{2\sigma^2} - \ln I_0\left(\frac{\mathbf{u}(x)\mathbf{f}(x)}{\sigma^2}\right) - \ln\frac{\mathbf{f}(x)}{\sigma^2}\right)dx. \quad (1.51)$$

$$\Rightarrow \mathcal{F}(\mathbf{u},\mathbf{f}) = \int_{x\in\Omega}\left(\frac{\mathbf{u}^2(x)}{2\sigma^2} - \ln I_0\left(\frac{\mathbf{u}(x)\mathbf{f}(x)}{\sigma^2}\right)\right)dx. \quad (1.52)$$

Then, we obtain the TV-based Rician denoising model that can be found in [MGS11, GTV11]:

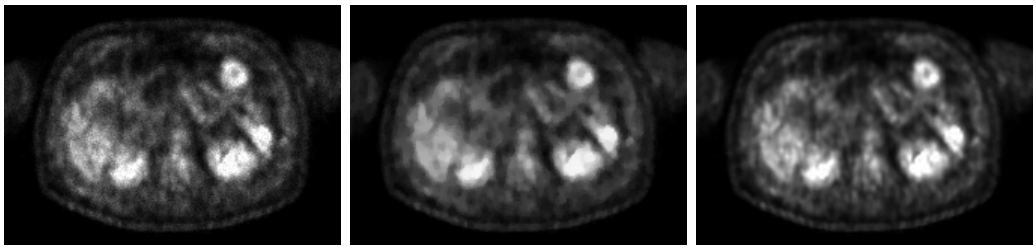
$$\min_{\mathbf{u}}\int_\Omega \|\nabla\mathbf{u}(x)\|dx + \lambda\int_{x\in\Omega}\left(\frac{\mathbf{u}^2(x)}{2\sigma^2} - \ln I_0\left(\frac{\mathbf{u}(x)\mathbf{f}(x)}{\sigma^2}\right)\right)dx, \quad (1.53)$$

Figure 1.6 describes an example for Rician denoising on a synthetic image by using the TV-based method of Getreuer *et al.* in [GTV11]. Figure 1.7 shows experimental results of the method performed on a MRI image of pelvis. As can be seen



(a) Poisson noise (b) Denoised image by [LCA07] (c) Noise-free image

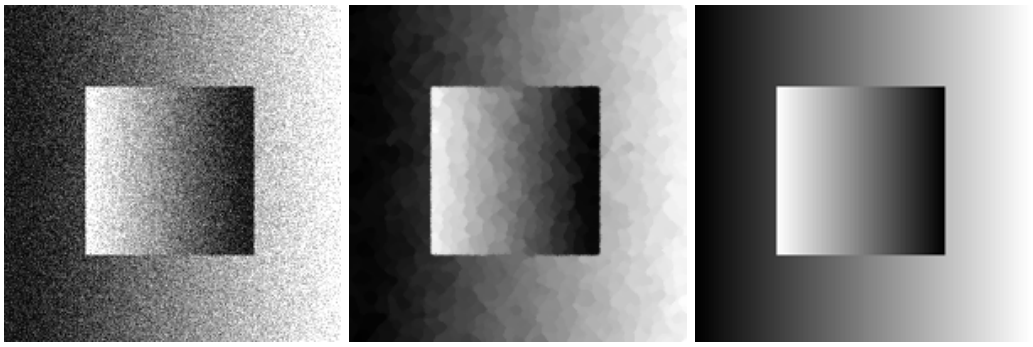
Figure 1.4: Poisson denoising results with TV [LCA07]: (a) noisy image, (b) denoised image, (c) noise-free image.



(a) Poisson noise (b) Denoised image by [LCA07] (c) Noise-free image

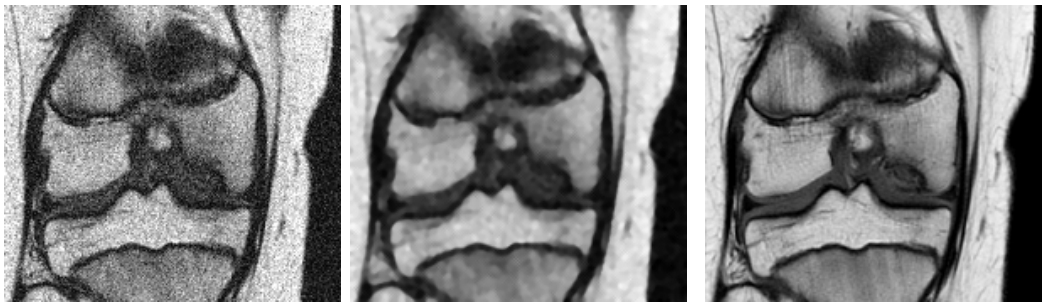
Figure 1.5: Poisson denoising results on a PET image by TV method [LCA07]: (a) noisy image, (b) denoised image, (c) noise-free image. Image courtesy of Dr. Jean-Marie Rocchisani (Avicenne University Hospital, Bobigny, France).

from Figure 1.7(b), the denoised result is very well, although some details are also oversmoothed. However, figure 1.6(b) shows that in the case of image with a gradual variation of contrast and brightness, the denoised image appears flat intensity regions and new contours.



(a) Rician noise (b) Denoised image by [GTV11] (c) Noise-free image

Figure 1.6: Rician denoising results with TV regularization on the synthetic image: (a) noisy image, (b) denoised image by [GTV11] and (c) noise-free image.



(a) Rician noise (b) TV [GTV11] (c) Noise-free image

Figure 1.7: Rician denoising results on a MRI image of knee with TV regularization: (a) noisy image, (b) TV-based denoise method [GTV11], (c) Original image (source: http://www.mr-tip.com/serv1.php).

1.5.1.1.4 TV-based Speckle denoising

From (1.24), we have

$$-\ln p_F(\mathbf{f}|\mathbf{u}) = \int_{x\in\Omega}\left(\ln(\sigma\sqrt{2\pi}) + \frac{1}{2\sigma^2}\frac{(\mathbf{f}(x)-\mathbf{u}(x))^2}{\mathbf{u}(x)}\right)dx \tag{1.54}$$

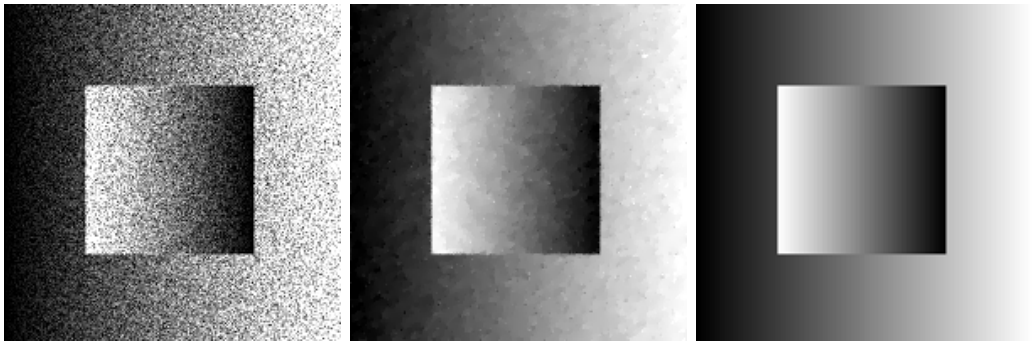
$$\Rightarrow \mathcal{F}(\mathbf{u},\mathbf{f}) = \int_{x\in\Omega}\frac{(\mathbf{f}(x)-\mathbf{u}(x))^2}{\mathbf{u}(x)}dx. \tag{1.55}$$

Therefore, the TV model for Speckle denoising is

$$\min_{\mathbf{u}>0} \int_\Omega \|\nabla \mathbf{u}(x)\| dx + \lambda \int_\Omega \frac{(\mathbf{f}(x) - \mathbf{u}(x))^2}{\mathbf{u}(x)} dx. \tag{1.56}$$

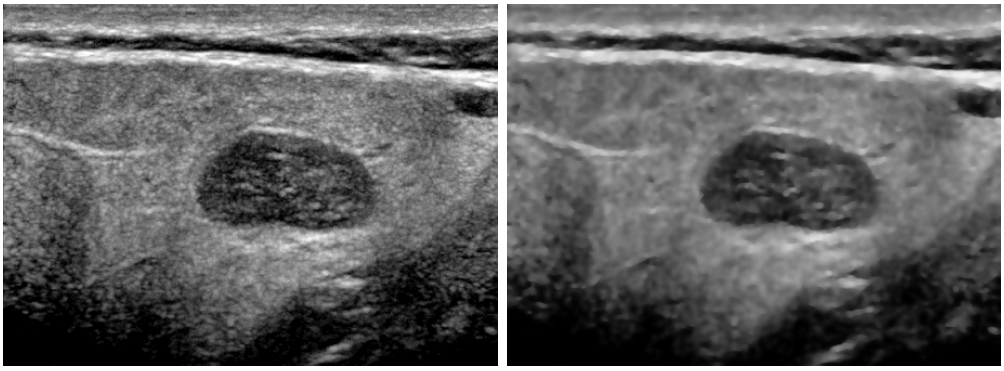
Note that in this case the image is assumed strictly positive.

This model can be found in [KKWV05, ZY11]. The performance of the TV-based method for speckle denoising is illustrated in Figure 1.8 and Figure 1.9. We can see in Figure 1.9(b) the performance of the TV in removing speckle noise on an ultrasound image of the parotid gland.



(a) Speckle noise (b) Denoised image by TV [ZY11] (c) Noise-free image

Figure 1.8: Speckle denoising result with TV on the synthetic image: (a) noisy image, (b) denoised image by [ZY11], (c) Original image.



(a) Speckle noise (b) Denoised image by TV [ZY11]

Figure 1.9: Speckle denoising results on an ultrasound image of the parotid gland with TV [ZY11]: (a) noisy image (source: https://sites.google.com/site/pierrickcoupe/), (b) denoised image.

1.5.1.1.5 Remarks

As it can be seen, the TV-based denoising methods provides excellent results

in edge preservation and smoothing of flat regions. However, it suffers from a well-known staircasing effect in regions with gradual image variations as can be seen in the experiments with the synthetic image. These artifacts have been observed by several authors, e.g., [Wei98, CMM00, CP04].To overcome this limitation, many methods focus on introducing higher order derivatives into the regularization term, such as in Chambolle and Lions [CL97], and in Chan *et al.* [CMM00], reducing effectively the staircase effect. However, as shown in [BV11], staircasing effect may still occur, usually in the neighborhood of edges. Recently, a new concept of total vatiation, namely the Total Generalized Variation (TGV) has been introduced in [BKP10] by Bredies, Kunisch and Pock, which is a generalization of the TV theory. It is shown that the application of TGV in image denoising is superior than the conventional TV-based methods. It not only is equivalent to TV in terms of edge preservation and noise removal but also eliminates the staircasing effect.

1.5.1.2 Total Generalized Variation

In the case of 2D image, the TGV is defined as follows:

Definition 1.5.2 ([BKP10])
Let Ω be a bounded open subset of $\mathbb{R}^2$. The TGV *of order k for $\mathbf{u} \in L^1(\Omega)$ is defined as*

$$\mathrm{TGV}_{\alpha}^{k}(\mathbf{u}) = \sup\left\{ \int_{\Omega} \mathbf{u}(x)\mathrm{div}^{k}v(x)dx \;\middle|\; v \in \mathcal{C}^{k}(\Omega, Sym^{k}(\mathbb{R}^2)), \right. \\ \left. \|\mathrm{div}^{l}v\|_{\infty} \leqslant \alpha_l,\ l = 0, \ldots, k-1 \right\} \tag{1.57}$$

where $\alpha = (\alpha_0, \alpha_1, \ldots, \alpha_{k-1}) > 0$ are fixed parameters, $\mathcal{C}_c^k(\Omega, Sym^k(\mathbb{R}^2))$ is the vector space of k-times continuously differentiable $Sym^k(\mathbb{R}^2)$-valued mappings on Ω, and $Sym^k(\mathbb{R}^2)$ denotes the space of symmetric tensors of order k with arguments in $\mathbb{R}^2$

$$Sym^{k}(\mathbb{R}^2) = \left\{ \xi : \underbrace{\mathbb{R}^2 \times \ldots \times \mathbb{R}^2}_{k\text{-times}} \longrightarrow \mathbb{R} \;\middle|\; \xi \text{ } k\text{-linear and symmetric} \right\}.$$

As shown in [BKP10], the case $k = 1$, $\mathrm{Sym}^1(\mathbb{R}^2) = \mathbb{R}^2$ [BKP10] and thus

$$\begin{aligned} \mathrm{TGV}_{\alpha}^{1}(\mathbf{u}) &= \alpha_0 \sup\left\{ \int_{\Omega} \mathbf{u}(x)\mathrm{div}v(x)dx \;\middle|\; v \in \mathcal{C}^{1}(\Omega, \mathbb{R}^2), \|v\|_{\infty} \leq 1 \right\} \\ &= \alpha_0 TV(\mathbf{u}). \end{aligned} \tag{1.58}$$

This is the reason why TGV can be seen as a generalization of the total variation.

The case $k = 2$, $\mathrm{Sym}^2(\mathbb{R}^2) = S^{2\times 2}$, i.e. the space of symmetric matrices [BKP10]. Then, the value

$$\mathrm{TGV}^2_\alpha(\mathbf{u}) = \sup\Big\{ \int_\Omega \mathbf{u}(x)\mathrm{div}^2 v(x)dx \;\Big|\; v \in \mathcal{C}^2(\Omega, S^{2\times 2}), \\ \|v\|_\infty \le \alpha_0,\ \|\mathrm{div} v\|_\infty \le \alpha_1 \Big\} \tag{1.59}$$

is called the total generalized variation of second order, where $\mathrm{div}v$ and $\mathrm{div}^2 v$ are defined by

$$(\mathrm{div}v)_i = \sum_{j=1}^{2} \frac{\partial v_{ij}}{\partial x_j}, \qquad (\mathrm{div}^2 v)_i = \sum_{j=1}^{2} \frac{\partial^2 v_{ij}}{\partial x_j^2} + 2\sum_{j<i} \frac{\partial v_{ij}}{\partial x_i \partial x_j}.$$

As it is proven in [BV11], if $\Omega \in \mathbb{R}^d$ is a bounded domain, $\mathbf{u} \in L^1(\Omega)$ then

$$\mathrm{TGV}^2_\alpha(\mathbf{u}) = \min_\vartheta\ \alpha_1 \int_\Omega \|\nabla \mathbf{u} - \vartheta\| dx + \alpha_0 \int_\Omega \|\mathcal{E}(\vartheta)\| dx, \tag{1.60}$$

where $\vartheta \in L^1(\Omega, \mathbb{R}^2)$ such that the distributional symmetrized derivative $\mathcal{E}(v) = \frac{1}{2}(\nabla v + \nabla v^T)$ is a measure.

By replacing $\mathrm{TV}(u)$ in (1.43) by $\mathrm{TGV}^2_\alpha(\mathbf{u})$, we obtain the TGV^2-based denoising model:

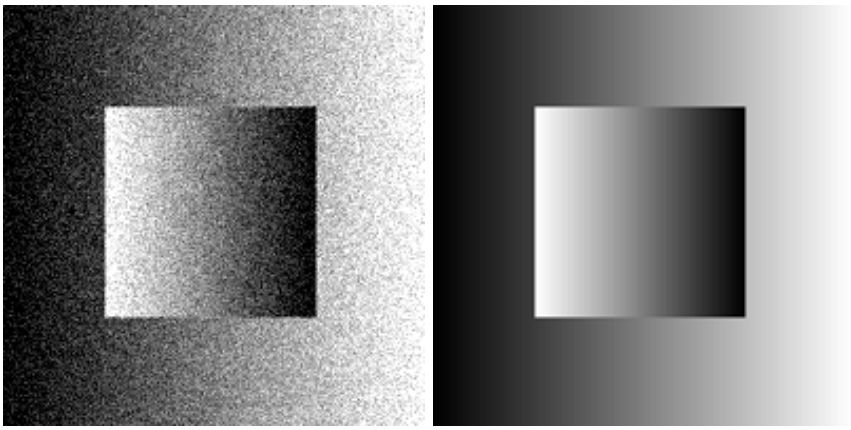
$$\min_{\mathbf{u}} \mathrm{TGV}^2_\alpha(\mathbf{u}) + \frac{\lambda}{2}\mathcal{F}(\mathbf{u}, \mathbf{f}). \tag{1.61}$$

Therefore, we can obtain the TGV-based denoising model for each type of noise by using the corresponding fidelity term $\mathcal{F}(\mathbf{u}, \mathbf{f})$ (see (1.46), (1.49), (1.52), (1.55)).

To see the effectiveness of the TGV, we show some experiments for Gaussian denoising using TGV^2_α regularization with the parameters $\alpha_1 = 1$ and $\alpha_0 = 2$. Figure 1.10 describes a comparison between TV-based Gaussian denoising method and TGV-based Gaussian denoising method on the synthetic image. It easy to see that unlike the TV's result (Figure 1.10(c)), the denoising result using TGV^2_α (Figure 1.10(d)), leads to an absence of the staircasing effect. Another experiment is shown in Figure 1.11, where a CT image of pelvis corrupted by Gaussian noise with standard deviation $\sigma = 25$ is denoised by both the TV and TGV methods. We can see that the image produced by the TGV (Figure 1.11(d)) looks smoother than the one produced by the TV (Figure 1.11(c)).

Figure 1.12 shows the results performed on a MRI image of pelvis. As can be seen, although noise is very effectively removed, both the TV and TGV-based denoising methods over-smooth many small details.

Unlike the TV (1.38) which only takes the first derivative into account, $\mathrm{TGV}^2_\alpha(\mathbf{u})$ proposes a way of balancing between the first and second derivative as can be seen from (1.60). Intuitively, for a given image $\mathbf{u}$, minimizing (1.60) requires that near



(a) Noisy image, $\sigma = 30$ (b) Noise-free image

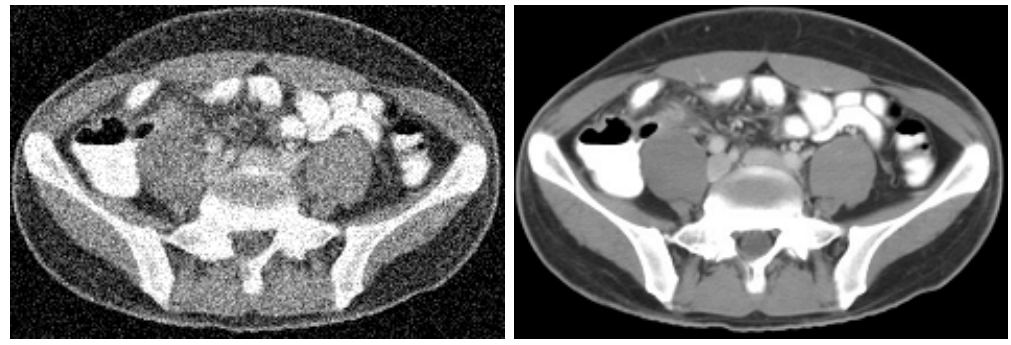
(c) TV denoising [DK00] (d) TGV^2_α denoising [BV11]

Figure 1.10: Gaussian denoising results on the synthetic image with TV model and TGV^2_α model: (a) image corrupted by Gaussian noise with the standard deviation$\sigma = 30$, (b) noise-free image, (c) denoised image by the TV [DK00] and (d) denoised image by the TGV^2_α [BV11]

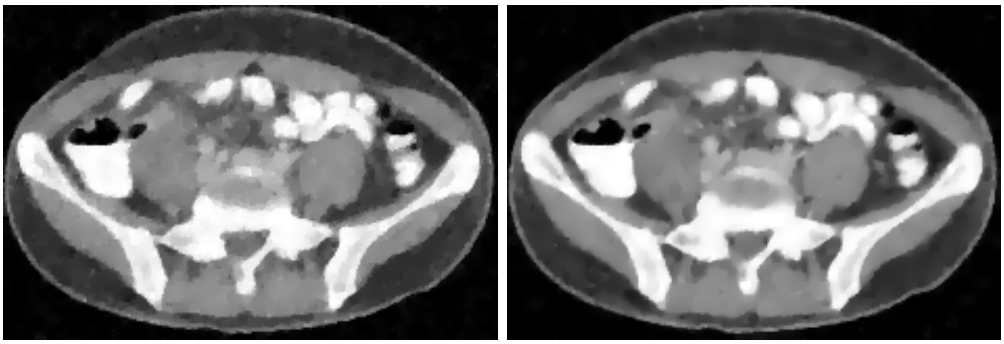
edges ($\nabla^2\mathbf{u}$ will be considerably larger than $\nabla\mathbf{u}$, it is beneficial for the minimization of (1.60) to choose locally $v = 0$), TGV^2_α behaves like the TV model, while in homogeneous regions (second derivative $\nabla^2\mathbf{u}$ is locally small and so, it is beneficial for the minimization of (1.60) to choose locally $v = \nabla\mathbf{u}$), TGV^2_α may takes more the second derivative into account. That is why this method leads to an absence of the staircasing effect which is often observed in TV regularization [KBPS11].

1.5.1.3 Conclusion

Although the TV and TGV can effectively reduce noise as well as well preserve edges, the small details and textures are often removed. On the other hand, an advantage of this variational approach is that it can determine exactly the contours of the



(a) Noisy image (b) Noise-free image [DK00]



(c) TV denoising (d) TGV^2_α denoising [BV11]

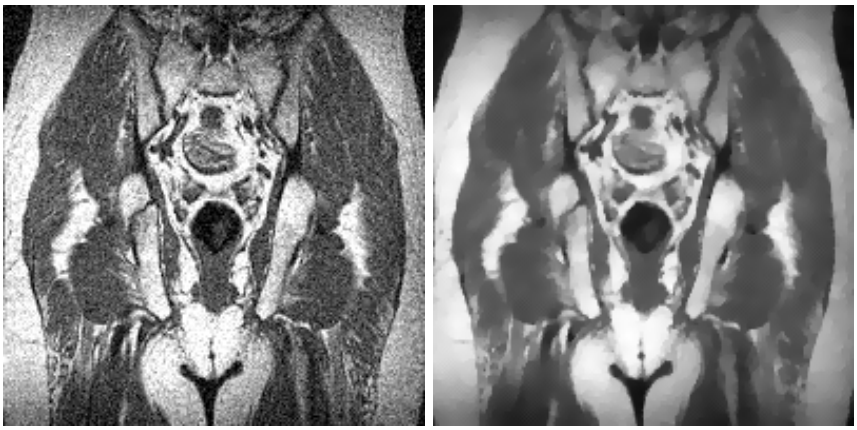
Figure 1.11: Experiment on a CT image of pelvis: (a) noisy image corrupted by Gaussian noise $\sigma = 25$, (b) Original image (source: `http://radiopaedia.org/`), (c) Denoising using TV [DK00], (d) Denoising using TGV^2_α [BV11].

subjects, so it may be used as a pre-processing step for segmentation to determine position or size of the interesting subjects. It is suited for applications that do not require photo-realistic images. Concerning the TGV, some of its applications for MRI imaging can be found in [KBPS11]. However, till now there only exists algorithm for solving the problem of Gaussian denoising by TGV-based model. The problem of finding effective algorithms for solving the other noise models is one of our future works.

1.5.2 Non-local means based methods

In this section, we present the non-local means (NLM) method proposed in [BCM05, BCM06] by Buades *et al.* and some of its applications for the reduction of common types of noise in medical images.

NLM method is based on the observation that patches in an image tend to redundantly recur many times inside the image. The method replaces the noisy value by a weighted average of all the pixels of the image. The weight of a pixel is computed based on the similarity between the image patch around it and the image



(a) Gaussian noise (b) Denoised image by TV [CDV10]

(c) Denoised image by TGV [KBPS11] (d) Noise-free image

Figure 1.12: Denoising results on a MRI image of pelvis with TV and TGV regularization. (a) Image with Gaussian noise ($\sigma = 20$), (b) Denoised image by TV [CDV10], (c) Denoised image by TGV [KBPS11], (d) Original image (source: http://www.mr-tip.com/).

patch around the reference pixel. The method is a well-known denoising one due to its simplicity and effectiveness. A comprehensive review can be found in [BCM06]. In the next subsections, we will present its applications for denoising images with respect to different types of noise. Let us begin with NLM for Gaussian noise.

1.5.2.1 Non-local means for Gaussian noise

Let us consider a discrete noisy image, $\mathbf{f} = \{\mathbf{f}(i) = \mathbf{u}(i) + \mathbf{n}(i), i \in \Omega\}$ (see (1.3)), in which $\mathbf{n}$ is additive white Gaussian noise with zero-means and standard deviation σ. The NLM filter is written,

$$\hat{\mathbf{u}}(i) = \frac{1}{W_i} \sum_{j \in \Omega_i} w(i,j)\mathbf{f}(j), \tag{1.62}$$

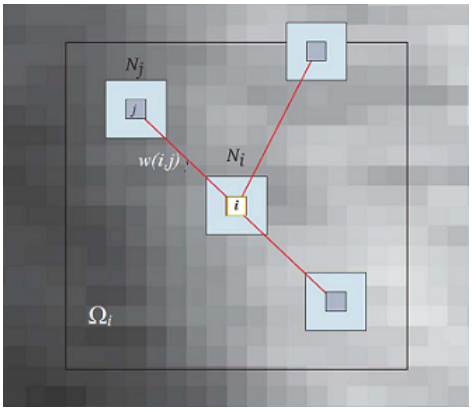


Figure 1.13: Two dimensional illustration of the NLM filter. The restored value at pixel i is the weighted average of all intensity values of pixels j in the search region Ω_i. The weights $w(i,j)$ are based on the similarity of the image patches $\mathbf{f}(N_i)$ and $\mathbf{f}(N_j)$.

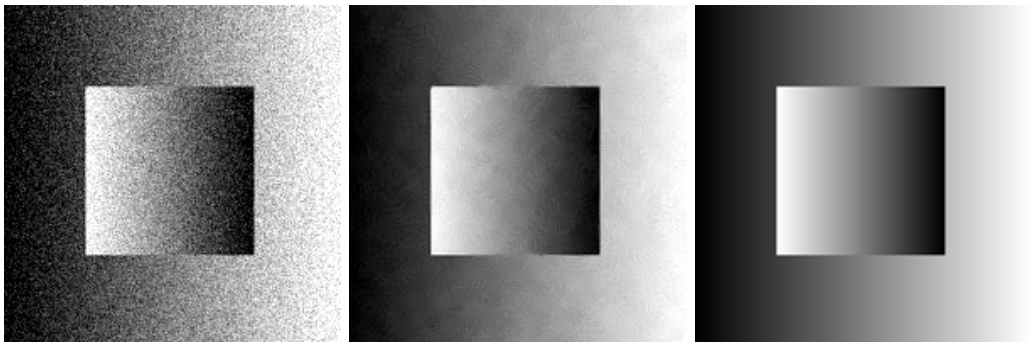
where Ω_i is called *search region* with respect to the pixel i, $W_i = \sum_{j\in\Omega_i} w(i,j)$ is the normalizing factor, and $w(i,j)$ is the weight assigned to value $\mathbf{f}(j)$ for restoring the pixel i. More precisely, the weight evaluates the similarity between the intensities $\mathbf{f}(N_i)$ and $\mathbf{f}(N_j)$ of the patches N_i and N_j centered on pixels i and j respectively (see Figure 1.13). The weight $w(i,j)$ is computed as follows:

$$w(i,j) = \exp\frac{-d(\mathbf{f}(N_i),\mathbf{f}(N_j))}{2h^2} = \exp\frac{-\|\mathbf{f}(N_i)-\mathbf{f}(N_j)\|^2_{2,a}}{2h^2}. \tag{1.63}$$

where $\|.\|^2_{2,a}$ is the Gaussian-weighted Euclidean distance. This distance is the traditional ℓ_2-norm convolved with a Gaussian kernel of standard deviation a. Here, h acts as a filtering parameter controlling the decay of the exponential function, it depends on the standard deviation σ of the noise. Choosing a very small value of h leads to results identical to the noisy input, while very large h gives an overly-smoothed image. A solution for selecting the optimal value of h can be found in [VK09] where Ville and Kocher use Stein's unbiased risk estimate (SURE) [Ste81] to monitor the mean squared error of the NLM algorithm.

The effectiveness of the NLM can be seen in Figure 1.14 as an example. We can see that the NLM method is very effective with Gaussian noise and well preserves edges. As proven in [BCM06], the NLM outperforms the TV-based method. This can be seen in Figure 1.15 and Figure 1.16. However, some small details can also be lost.

Many solutions have been proposed to improve the NLM method, such as [KB06, LWC+08, KBC07]. The applications in medical image denoising have



(a) Gaussian noise (b) NLM [BCM06] (c) Noise-free image

Figure 1.14: Denoising results of the NLM method on a synthetic image with white Gaussian noise, $\sigma = 25$: (a) Noisy image, (b) NLM's result, (c) Noise-free image

been largely exploited for reducing Rician noise in MRI images [MCCL+08, MCMB+10, WDPC+08, TK10], Poisson noise [DTD10], speckle noise in ultrasound images [CHKB09]. Unlike the Gaussian noise, the different types of noise such as Rician, Poisson and Speckle noises are image-dependent. Thus, this poses a greater challenge for denoising problem. In the following subsections, we briefly describe how to adapt the NLM to these types of noise.

1.5.2.2 Non-local means for Poisson noise

In medical imaging, the Poisson noise is often found in X-ray projection imaging, Fluoroscopy, Mammography, PET/SPECT, and Fluorescence Confocal Microscopy. The observation model with Poisson noise can be written as follows:

$$\mathbf{f}(x) = \mathcal{P}(\mathbf{u}(x)) \tag{1.64}$$

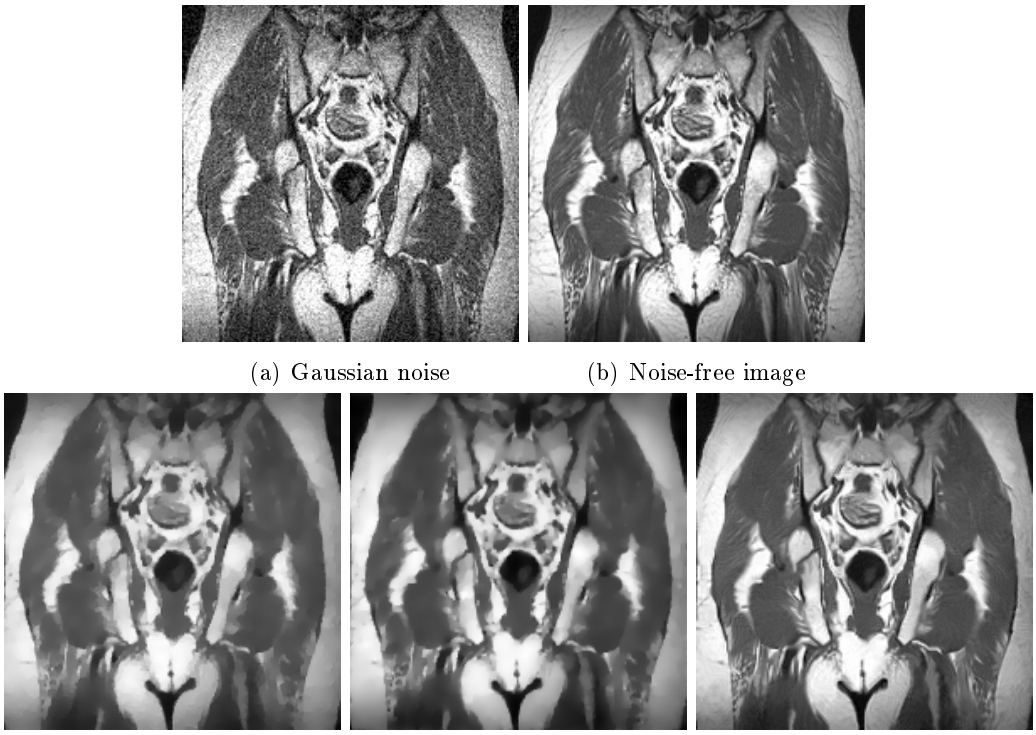
where $\mathcal{P}$ is a general noise distortion function which is dominated by Poisson noise, it means

$$p(\mathbf{f}(x)|\mathbf{u}(x)) = \frac{\mathbf{u}(x)^{\mathbf{f}(x)} e^{-\mathbf{u}(x)}}{\mathbf{f}(x)!}. \tag{1.65}$$

Unlike Gaussian noise, Poisson noise is intensity dependent, which makes the separation of image from noise very difficult. To reduce Poisson noise, the Anscombe root transformation [Ans48]:

$$\mathcal{T}(\mathbf{f}(x)) = 2\sqrt{\mathbf{f}(x) + \frac{3}{8}} \tag{1.66}$$

is usually used to convert approximately Poisson noise into Gaussian noise. Thus, the noise can be removed by applying a conventional method designed for additive Gaussian noise. Finally, an inverse transformation is applied to the denoised image



(a) Gaussian noise (b) Noise-free image

(c) TV [CDV10] (d) TGV [KBPS11] (e) NLM [BCM06]

Figure 1.15: Denoising results of the NLM method on a MRI image of Pelvis: (a) Image corrupted by white Gaussian noise with $\sigma = 20$, (b) Original image (source: http://www.mr-tip.com/), (c) Denoised image by TV [CDV10], (d) Denoised image by TGV [KBPS11], (e) Denoised image by NLM [BCM06].

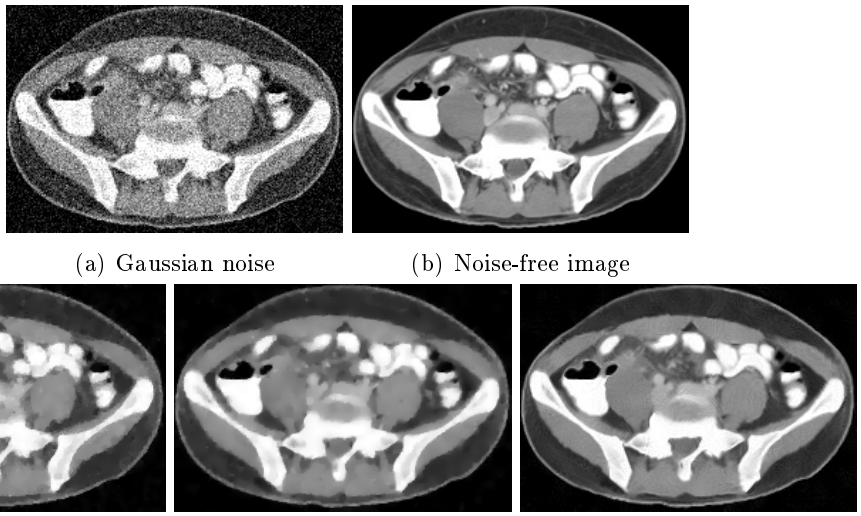
to obtain the final estimate $\hat{\mathbf{u}}$ (see Subsection 1.3.2). An experiment on the synthetic image is illustrated in Figure 1.17.

Another method, Deledalle *et al.* in [DTD10] have directly adapted the NLM to Poisson noise by using a pre-filtered image $\hat{\mathbf{f}}$ of the noisy image f and refining the weights $w(i,j)$ in (1.63). Denote $(\mathbf{f}^{(1)}(N_i), \ldots, \mathbf{f}^{(k)}(N_i), \ldots, \mathbf{f}^{(P)}(N_i))^T$ the vectorized image patch $\mathbf{f}(N_i)$, where P is the number of pixels in each patch. The NLM filter for Poisson noise is defined as in (1.62) with the weights redefined as

$$w(i,j) = \exp\left(-\frac{F_{i,j}}{\alpha} - \frac{G_{i,j}}{\beta}\right), \tag{1.67}$$

where F and G are similarity terms respectively to compare noisy image and pre-filtered image with

$$F_{i,j} = \sum_{k=1}^{P} \log \frac{\mathbf{f}^{(k)}(N_i)^{\mathbf{f}^{(k)}(N_i)} \mathbf{f}^{(k)}(N_j)^{\mathbf{f}^{(k)}(N_j)}}{\left[\frac{\mathbf{f}^{(k)}(N_i)+\mathbf{f}^{(k)}(N_j)}{2}\right]^{\mathbf{f}^{(k)}(N_i)+\mathbf{f}^{(k)}(N_j)}} \tag{1.68}$$



(a) Gaussian noise (b) Noise-free image

(c) TV [CDV10] (d) TGV [KBPS11] (e) NLM [BCM06]

Figure 1.16: Denoising results of the NLM method on a pelvis CT image: (a) Image corrupted by white Gaussian noise with $\sigma = 20$, (b) original image (source: `http://radiopaedia.org/`), (c) Results of TV [CDV10], (d) Result of TGV [KBPS11], and (e) Result of NLM [BCM06].

$$G_{i,j} = \sum_{k=1}^{P} (\hat{\mathbf{f}}^{(k)}(N_i) - \hat{\mathbf{f}}^{(k)}(N_j)) \log \frac{\hat{\mathbf{f}}^{(k)}(N_i)}{\hat{\mathbf{f}}^{(k)}(N_j)}. \tag{1.69}$$

In [DTD10], the parameters α and β are automatically determined based on the minimization of the Poisson unbiased risk estimator (PURE). We refer the reader to [DTD10] for additional detailed explanations. The effectiveness of this method can be seen in two experiments shown in Figure 1.18 and Figure 1.19.

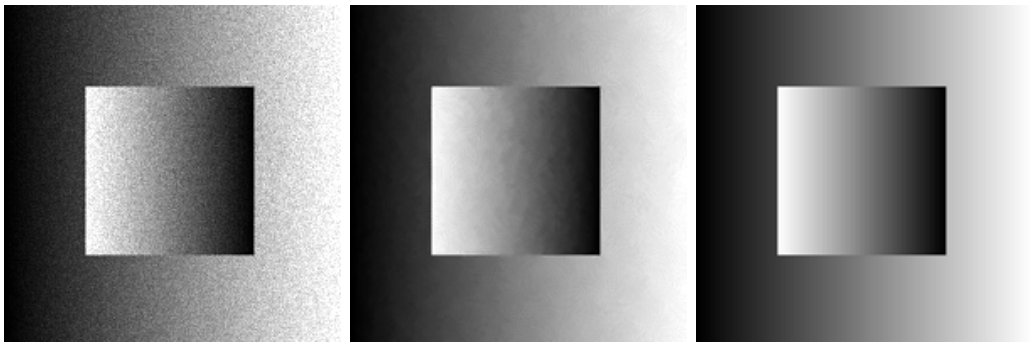
1.5.2.3 Non-local means for Rician noise

Normally, the MRI image $\mathbf{f} = \{\mathbf{f}(k), k \in \Omega\}$ is reconstructed by computing the inverse discrete Fourier transform of the measured signal components from real and imaginary channels [Now99] denoted by $\mathbf{f}_{Re}$ and $\mathbf{f}_{Im}$, respectively,

$$\mathbf{f} = |\mathbf{f}_{Re} + i\mathbf{f}_{Im}|, \tag{1.70}$$

where $\mathbf{f}_{Re} = \mathbf{u}\cos\theta + \mathbf{n}_1$ and $\mathbf{f}_{Im} = \mathbf{u}\sin\theta + \mathbf{n}_2$ are raw data which are corrupted by Gaussian noise $\mathbf{n}_1 = \{\mathbf{n}_1(k), k \in \Omega\}$ and $\mathbf{n}_2 = \{\mathbf{n}_2(k), k \in \Omega\}$ respectively, $\mathbf{n}_1 \sim \mathcal{N}(0, \sigma^2)$, $\mathbf{n}_2 \sim \mathcal{N}(0, \sigma^2)$, also $\mathbf{u} = \{\mathbf{u}(k), k \in \Omega\}$ is a true MRI image intensity and θ is a phase of real and imaginary channels, and i is the imaginary unit ($i^2 = -1$). From (1.70) we have

$$\mathbf{f} = \sqrt{\mathbf{f}_{Re}^2 + \mathbf{f}_{Im}^2} = \sqrt{(\mathbf{u}\cos\theta + \mathbf{n}_1)^2 + (\mathbf{u}\sin\theta + \mathbf{n}_2)^2}. \tag{1.71}$$



(a) Poisson noise (b) NLM using the Anscombe transformation (c) Noise-free image

Figure 1.17: Poisson denoising experiment on the synthetic image. (a) Image corrupted by Poisson noise, (b) Denoised image by the NLM method using the Anscombe transformation, (c) Noise-free image.

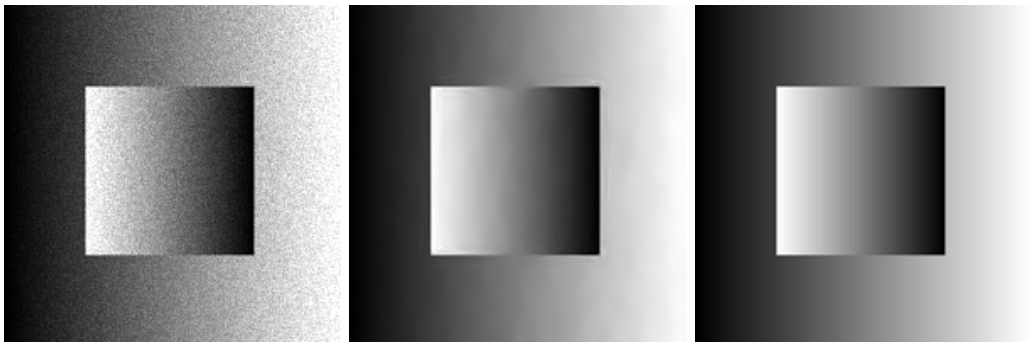
Thus, noise in $\mathbf{f}$ is often assumed to be Rician noise (see Section 1.3). Unlike Gaussian noise, Rician noise is not additive and its mean is signal-dependent. So, the NLM algorithm can not effectively remove Rician noise if it is applied directly.

We present here the method of Manjon *et al.* [MCCL+08] that relies on the second order moment of Rician noise. Let us consider the estimated image value $\hat{\mathbf{u}}^2(k)$ by the NLM method. With the same notations as in subsection 1.5.2.1 for W_k and w, we have

$$\begin{aligned}
\hat{\mathbf{u}}^2(k) &= \frac{1}{W_k}\sum_{i\in\Omega_k} w(k,i)\mathbf{f}^2(i) \approx E(\mathbf{f}^2) \\
&= \frac{1}{W_k}\sum_{i\in\Omega_k} w(k,i)[(\mathbf{u}(i)\cos\theta + \mathbf{n}_1(i))^2 + (\mathbf{u}(i)\sin\theta + \mathbf{n}_2(i))^2] \\
&= \frac{1}{W_k}\left(\sum_{i\in\Omega_k} w(k,i)\mathbf{u}^2(i) + 2\cos\theta\sum_{i\in\Omega_k} w(k,i)\mathbf{u}(i)\mathbf{n}_1(i) + \right. \\
&\quad \left. +2\sin\theta\sum_{i\in\Omega_k}\mathbf{u}(i)w(k,i)\mathbf{n}_2(i) + \sum_{i\in\Omega_k} w(k,i)\mathbf{n}_1^2(i) + \sum_{i\in\Omega_k} w(k,i)\mathbf{n}_2^2(i)\right) \\
&\approx E(\mathbf{u}^2) + 2a_k E(\mathbf{n}_1)\cos\theta + 2a_k E(\mathbf{n}_2)\sin\theta + E(\mathbf{n}_1^2) + E(\mathbf{n}_2^2) \\
&= E(\mathbf{u}^2) + E(\mathbf{n}_1^2) + E(\mathbf{n}_2^2) = E(\mathbf{u}^2) + 2\sigma^2, \qquad (1.72)
\end{aligned}$$

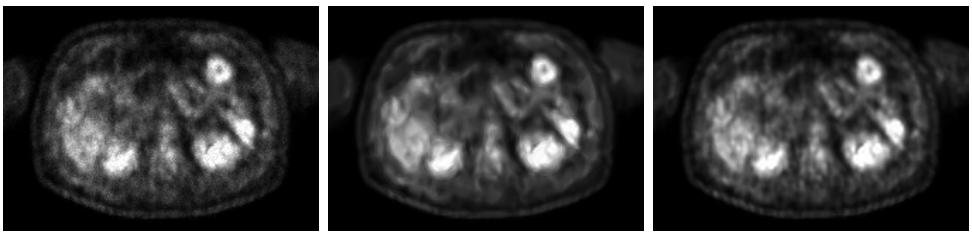
where $a_k = \frac{\sum_{i\in\Omega_k}\mathbf{u}(i)w(k,i)}{W_k}$. Thus, $E(\mathbf{f}^2) \approx E(\mathbf{u}^2)+2\sigma^2$. According to this results, in the squared magnitude image, the noise bias is $2\sigma^2$ and no longer signal-dependent. Thus, it seems natural to restore $\mathbf{u}(k)$ as:

$$\hat{\mathbf{u}}(k) = \sqrt{\frac{1}{W_k}\sum_{j\in\Omega_k} w(k,j)\mathbf{f}^2(j) - 2\sigma^2}. \qquad (1.73)$$



(a) Poisson noise (b) Denoised image by [DTD10] (c) Noise-free image

Figure 1.18: Poisson denoising experiment on the synthetic image. (a) Image corrupted by Poisson noise, (b) Denoised image by the NLM method for Poisson noise [DTD10], (c) Noise-free image.

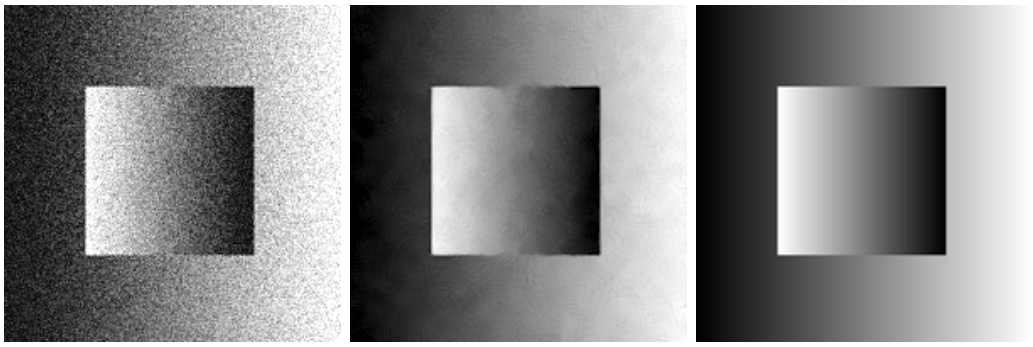


(a) Poisson noise (b) Denoised image by [DTD10] (c) Noise-free image

Figure 1.19: Poisson denoising experiment on a PET image of chest with NLM. (a) Image corrupted by Poisson noise, (b) Denoised image by the NLM method [DTD10], (c) Noise-free image. Image courtesy of Dr. Jean-Marie Rocchisani (Avicenne University Hospital, Bobigny, France)

By that way, we obtain the unbiased non-local means (UNLM) method in [MCCL+08] for Rician noise.

Figure 1.20 shows an experimental result of Rician denoising on the synthetic image by UNLM [MCCL+08]. Another example is shown in Figure 1.21. In this example, the UNLM performs Rician denoising on a Coronal MRI image of a knee. As can be seen in Figure 1.21(c), the intensity in the denoised image by conventional NLM algorithm [BCM06] seems to have changed compared with the original image. This is not the case for the result by the UNLM [MCCL+08]. It is clear to see that, compared with the results in Figure 1.21, the result in Figure 1.21(d) is closer to the ground truth image.



(a) Rician noise (b) Result of UNLM [MCCL+08] (c) Noise-free image

Figure 1.20: Rician denoising result on the synthetic image with UNLM. (a) Image corrupted by Rician noise with $\sigma = 25$, (b) Denoised image by the UNLM method in [MCCL+08], (c) Noise-free image.

1.5.2.4 Non-local means for Speckle noise

In ultrasound imaging, the image quality is often affected by Speckle noise. This noise is known to be tissue-dependent and is not easy to model. So, unlike the conventional denoising methods, ultrasound imaging requires specific filters. Here, we present briefly the Nonlocal means-based speckle filter proposed in [CHKB09] by Coupé *et al.*. In [CHKB09], the image with speckle noise is modeled as

$$\mathbf{f}(x) = \mathbf{u}(x) + \mathbf{u}^s(x)\mathbf{n}(x) \tag{1.74}$$

where $\mathbf{u}(x)$ is the original image, $\mathbf{f}(x) : \Omega \to \mathbb{R}$ is the observed image, $\mathbf{n}(x) \sim \mathcal{N}(0, \sigma^2)$ is a zero-mean Gaussian noise, the factor s depends on ultrasound devices and additional processing related to image formation. The authors use a Bayesian formation of the NLM filter proposed in [KBC07], where the nonlocal means expression (1.62) is generalized as (see [KBC07] for more details)

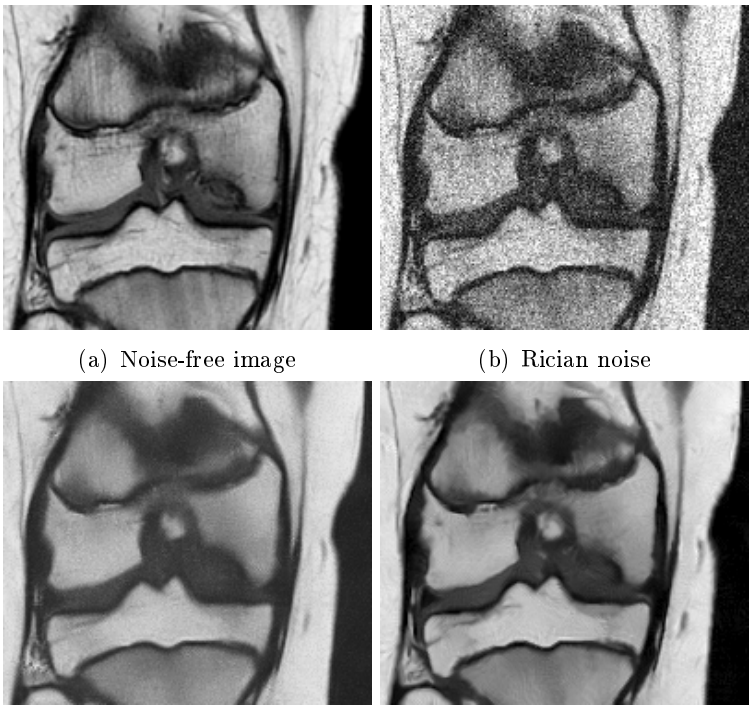
$$\hat{\mathbf{u}}(i) = \frac{\sum_{j\in\Omega_i} p(\mathbf{f}(N_i)|\mathbf{f}(N_j))\mathbf{f}(j)}{\sum_{j\in\Omega_i} p(\mathbf{f}(N_i)|\mathbf{f}(N_j))} \tag{1.75}$$

where $p(\mathbf{f}(N_i)|\mathbf{f}(N_j))$ denotes the probability density function (pdf) of $\mathbf{f}(N_i)$ conditionally to $\mathbf{f}(N_j)$. In order to adapt this formulation to speckle denoise, the authors assumed that, at each pixel,

$$\mathbf{f}(x)|\mathbf{u}(x) \sim \mathcal{N}(\mathbf{u}(x), \mathbf{u}(x)^{2s}\sigma^2). \tag{1.76}$$

This yields

$$p(\mathbf{f}(x)|\mathbf{u}(x)) \propto \exp -\frac{(\mathbf{f}(x) - \mathbf{u}(x))^2}{2\mathbf{u}(x)^{2s}\sigma^2} \tag{1.77}$$



(a) Noise-free image (b) Rician noise

(c) NLM [BCM06] (d) UNLM [MCCL+08]

Figure 1.21: Example of NLM algorithms under Rician noise. (a) Coronal MRI image of a knee (noise-free image) (source: http://www.mr-tip.com/serv1.php), (b) Noisy image, (c) Result of the normal NLM method in [BCM06], (d) Result of the UNLM method in [MCCL+08].

where $p(\mathbf{f}(x)|\mathbf{u}(x))$ denotes the pdf of $\mathbf{f}(x)$ conditionally to $\mathbf{u}(x)$. Thus,

$$p(\mathbf{f}(N_i)|\mathbf{f}(N_j)) = \prod_{k=1}^{P} p(\mathbf{f}^{(k)}(N_i)|\mathbf{f}^{(k)}(N_j)) \propto \exp - \sum_{k=1}^{P} \frac{(\mathbf{f}^{(k)}(N_i) - \mathbf{f}^{(k)}(N_j))^2}{2(\mathbf{f}^{(k)}(N_j))^{2s}\sigma^2}, \tag{1.78}$$

where P is the number of pixel in each image block. Accordingly, the distance $d(\mathbf{f}(N_i), \mathbf{f}(N_j))$ in (1.63) is substituted by

$$d_P(\mathbf{f}(N_i), \mathbf{f}(N_j)) = \sum_{p=1}^{P} \frac{(\mathbf{f}^{(p)}(N_i) - \mathbf{f}^{(p)}(N_j))^2}{(\mathbf{f}^{(p)}(N_j))^{2s}}. \tag{1.79}$$

The speckle denoising method is therefore implemented by a block-wise approach which consists of three main steps:

i) Divide the volume Ω into blocks N_i centered at pixel i with overlap supports, i.e, $\Omega = \cup_i N_i$.

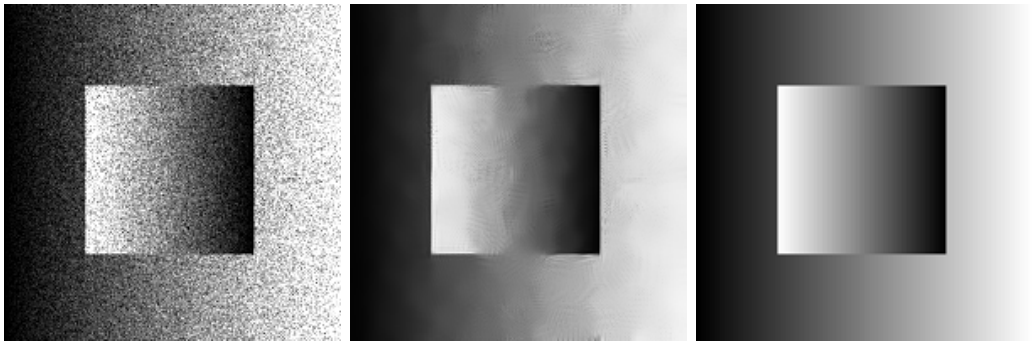
ii) Each block N_i is restored as follows:

$$\hat{\mathbf{u}}(N_i) = \frac{\sum_{j\in\Omega_i} w(N_i, N_j)\mathbf{f}(N_j)}{\sum_{j\in\Omega_i} w(N_i, N_j)}, \tag{1.80}$$

with $\mathbf{f}(N_j) = (\mathbf{f}^{(1)}(N_j), \ldots, \mathbf{f}^{(P)}(N_i))^T$ is a vectorized image patch, the weights $w(N_i, N_j)$ are computed as

$$w(N_i, N_j) = \exp -\frac{d_P(\mathbf{f}(N_i), \mathbf{f}(N_j))}{2h^2} = \exp -\sum_{k=1}^{P} \frac{(\mathbf{f}^{(k)}(N_i) - \mathbf{f}^{(k)}(N_j))^2}{2h^2(\mathbf{f}^{(p)}(N_j))^{2s}} \tag{1.81}$$

iii) Restore the pixel intensities from the restored blocks by averaging in overlap regions to get the final image.



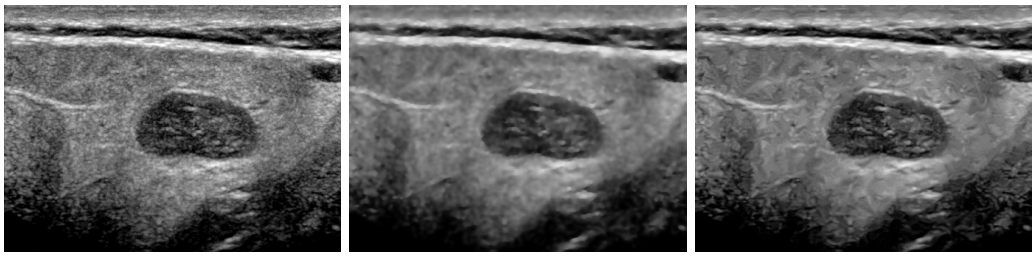
(a) Speckle noise (b) NLM by [CHKB09] (c) Noise-free image

Figure 1.22: Speckle denoising experiment on a synthetic image: (a) Noisy mage, (b) Denoised image by the NLM method in [CHKB09], (c) Noise-free image.

An example for speckle denoising is shown in Figure 1.22. In this experiment, the noisy image is generated by using the model (1.74) with $s = 2$. We can see the performance of the NLM method for speckle noise [CHKB09] in the Figure 1.22(b). Furthermore, Fig 1.23 illustrates a speckle denoising experiment on an ultrasound image of a parotid gland. Visually, the result of the NLM-based method (Figure 1.23(c)) seems to have better contrast compared with the result of the TV method [ZY11] (Figure 1.23(b)).

1.5.2.5 Conclusion

We have presented through this sub-section the Non-local means approach and its application for removing some common types of noise in medical image. As shown above, the NLM method has been adapted to most of types of noise. The denoising results produced by the NLM-based methods are often more effective than that the TV-based methods. Even so, some of small details can also be lost.



(a) Speckle noise (b) TV by [ZY11] (c) NLM by [CHKB09]

Figure 1.23: Speckle denoising experiment on an ultrasound image of the parotid gland: (a) Noisy mage (source: https://sites.google.com/site/pierrickcoupe/), (b) Denoised image by the TV method in [ZY11], (c) Denoised image by the NLM method in [CHKB09].

1.5.3 Data-adaptive Kernel Regression for Image Denoising

In this section, we give descriptions of another patch-based denoising method, namely the "Data-adaptive Kernel Regression" method of Takeda *et al.* [TFM07]. In particular, the method has beeb proven to be effective in removing Gaussian noise.

Let us recall the problem of denoising additive Gaussian noise in 2-D image: Given a noisy image $\mathbf{f}$ corrupted by additive white Gaussian noise η, we need to estimate the noise-free image $\mathbf{u}$ from $\mathbf{f}$ ($\mathbf{f} = \mathbf{u} + \eta$). The main idea in [TFM07] is to pose the denoising problem in a kernel regression framework. The authors consider the noisy image $\mathbf{f}$ as a set of M samples $\{f_i\}_{i=1}^{M}$ in 2-D, with the assumption

$$f_i = u(\mathbf{x}_i) + \eta_i, \quad \mathbf{x}_i = [x_{1i}, x_{2i}]^T, \tag{1.82}$$

where f_i is a noisy sample at $\mathbf{x}_i$ (the value of i-th pixel in $\mathbf{f}$), η_i is the independent and identically distributed zero-mean noise. In regression framework, $u(\cdot)$ is considered as *the regression function* corresponding to the set of samples $\{f_i\}_{i=1}^{M}$. Thus, denoising in $\mathbf{f}$ leads to estimating the regression function $u(\cdot)$.

Notice that the form of $u(\cdot)$ may remain unspecified. Thus, in order to estimate the value of $u(\cdot)$ at any point $\mathbf{x}$, the authors relied on a generic local expansion of the function about a sampling point $\mathbf{x}_i$. Specifically, if the function is locally smooth to some order N and $\mathbf{x}$ is near the sample at $\mathbf{x}_i$, we have the N-th order Taylor series

$$\begin{aligned} u(\mathbf{x}_i) &\approx u(\mathbf{x}) + \{\nabla u(\mathbf{x})\}^T(\mathbf{x}_i - \mathbf{x}) + \frac{1}{2!}(\mathbf{x}_i - \mathbf{x})^T\{\mathcal{H}u(\mathbf{x})\}(\mathbf{x}_i - \mathbf{x}) + \dots \\ &= u(\mathbf{x}) + \{\nabla u(\mathbf{x})\}^T(\mathbf{x}_i - \mathbf{x}) + \frac{1}{2}\text{vech}^T\{\mathcal{H}u(\mathbf{x})\}\text{vech}\left\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T\right\} + \dots \end{aligned} \tag{1.83}$$

where ∇ and $\mathcal{H}$ respectively are the gradient and Hessian operators, and vech$(\cdot)$ is the vectorization operator which lexicographically orders the lower-triangular portion of a symmetric matrix, e.g.,

$$\text{vech}\left(\begin{bmatrix} a & b \\ b & c \end{bmatrix}\right) = [a\ b\ c]^T$$
$$\text{vech}\left(\begin{bmatrix} a & b & c \\ b & d & e \\ c & e & f \end{bmatrix}\right) = [a\ b\ c\ d\ e\ f]^T. \tag{1.84}$$

Equation (1.83) can be written as the form

$$u(\mathbf{x}_i) \approx \beta_0 + \beta_1^T(\mathbf{x}_i - \mathbf{x}) + \beta_2^T \text{vech}\left\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T\right\} + \ldots \tag{1.85}$$

where β_0 is $u(\mathbf{x})$, which is the desired pixel value, and the vectors β_1 and β_2 are

$$\beta_1 = \nabla u(\mathbf{x}) = \left[\frac{\partial u(\mathbf{x})}{\partial x_1}\Big|_{\mathbf{x}=\mathbf{x}_i} \quad \frac{\partial u(\mathbf{x})}{\partial x_2}\Big|_{\mathbf{x}=\mathbf{x}_i}\right]^T, \tag{1.86}$$

$$\beta_2 = \frac{1}{2}\text{vech}\{\mathcal{H}u(\mathbf{x})\} = \frac{1}{2}\left[\frac{\partial^2 u(\mathbf{x})}{\partial x_1^2}\Big|_{\mathbf{x}=\mathbf{x}_i} \quad 2\frac{\partial^2 u(\mathbf{x})}{\partial x_1 \partial x_2}\Big|_{\mathbf{x}=\mathbf{x}_i} \quad \frac{\partial^2 u(\mathbf{x})}{\partial x_2^2}\Big|_{\mathbf{x}=\mathbf{x}_i}\right]^T \tag{1.87}$$

To estimate the parameters $\beta_n,\ n = 0, \ldots, N$, a subset of samples $\{f_i\}_{i=1}^P$ in a neighborhood (window) of $\mathbf{x}$ is considered. These parameters are computed from the following optimization problem:

$$\min_{\{\beta_n\}_{n=1}^N} \sum_{i=1}^{P} \left[f_i - \beta_0 - \beta_1^T(\mathbf{x}_i - \mathbf{x}) - \beta_2^T \text{vech}\left\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T\right\} - \ldots\right]^2 \cdot \cdot \mathcal{K}_{adapt}(\mathbf{x}_i - \mathbf{x}, f_i - f), \tag{1.88}$$

where N is the regression order ($N = 2$ typically), and $\mathcal{K}_{adapt}(\mathbf{x}_i - \mathbf{x}, f_i - f)$ is the data-adaptive kernel function. In [TFM07], Takeda *et al.* proposed steering kernel functions taken the form

$$\mathcal{K}_{adapt}(\mathbf{x}_i - \mathbf{x}, f_i - f) = \mathcal{K}_{\mathbf{H}_i}(\mathbf{x}_i - \mathbf{x}) \tag{1.89}$$

with

$$\mathcal{K}_{\mathbf{H}_i}(\mathbf{z}) = \frac{1}{\det(\mathbf{H}_i)}\mathcal{K}(\mathbf{H}_i{}^{-1}\mathbf{z}) \tag{1.90}$$

where $\mathbf{H}_i$ is the data-dependent full 2×2 matrix which is called *steering matrix*, $\det(\mathbf{H}_i)$ is the determinant of $\mathbf{H}_i$ and $\mathcal{K}$ is the 2-D realization of a kernel function such as the Gaussian kernel

$$\mathcal{K}(\mathbf{t}) = \frac{1}{2\pi}\exp\left(-\frac{\mathbf{t}^T\mathbf{t}}{2}\right). \tag{1.91}$$

In [TFM07], $\mathbf{H}_i$ is defined as

$$\mathbf{H}_i = h_i \mathbf{C}_i^{-\frac{1}{2}} \tag{1.92}$$

with $\mathbf{C}_i$s are (symmetric) covariance matrices based on differences in the local gray-values. With such steering matrices, for instance, if we use a Gaussian kernel, the steering kernel is represented as

$$K_{\mathbf{H}_i}(\mathbf{x}_i - \mathbf{x}) = \frac{\sqrt{\det(\mathbf{C}_i)}}{2\pi h_i^2} \exp\left\{-\frac{(\mathbf{x}_i - \mathbf{x})^T \mathbf{C}_i (\mathbf{x}_i - \mathbf{x})}{2h_i^2}\right\}. \tag{1.93}$$

The matrices $\mathbf{C}_i$s are computed based on *local orientation estimate* [FM02]. More specifically, $\mathbf{C}_i$ has the form

$$\mathbf{C}_i = \gamma_i \mathbf{U}_{\theta_i} \Lambda_i \mathbf{U}_{\theta_i}^T, \tag{1.94}$$

where

$$\mathbf{U}_{\theta_i} = \begin{bmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{bmatrix}, \quad \Lambda_i = \begin{bmatrix} \rho_i & 0 \\ 0 & \rho_i^{-1}. \end{bmatrix} \tag{1.95}$$

Here, $\mathbf{U}_{\theta_i}$ is a rotation matrix and Λ_i is the elongation matrix.

As we can see, the steering matrix $\mathbf{H}_i$ contains four parameters h_i, γ_i, ρ_i and θ_i. The first one is the global smoothing parameter controlling the smoothness of an entire resulting image. The remaining parameters respectively are the scaling (γ_i), elongation (ρ_i), and orientation angle (θ) parameters which capture local image structures. Figure 1.24 schematically describes how the parameters in $\mathbf{C}_i$ affects the spreading of kernels. First, the elongation matrix Λ_i elongates the circular kernel,

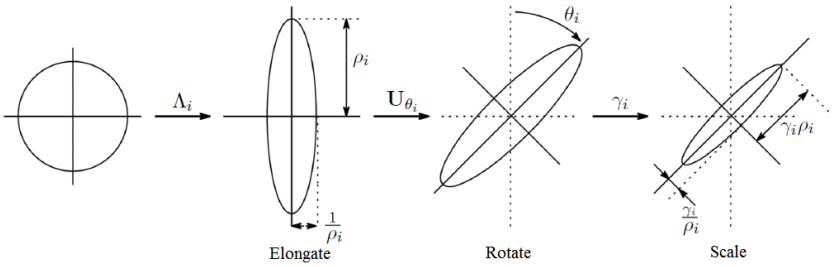


Figure 1.24: Schematic representation illustrating the effects of the steering matrix and its component ($\mathbf{C}_i = \gamma_i \mathbf{U}_{\theta_i} \Lambda_i \mathbf{U}_{\theta_i}^T$) on the size and shape of the regression kernel [TFM07].

and its semi-minor axis and semi-major axis are given by ρ_i. Second, the elongated kernel is rotated by the rotation matrix $\mathbf{U}_{\theta_i}$. Finally, the kernel is scaled by the scaling parameter γ_i.

The scaling, elongation, and rotation parameters in the covariance matrices $\{\mathbf{C}_i\}_{i=1}^P$ are estimated by applying singular value decomposition (SVD) to a collection of estimated gradient vectors in a neighborhood around every sampling position

of interest. More specifically, to estimate the parameters of $\mathbf{C}_i$, the gradient values of a neighborhood of ℓ samples around the position $\mathbf{x}_i$ is first grouped into a matrix $\mathbf{G}_i$ of size $\ell \times 2$,

$$\mathbf{G}_i = \begin{bmatrix} \vdots & \vdots \\ \frac{\partial u(\mathbf{x}_j)}{\partial x_{1j}} & \frac{\partial u(\mathbf{x}_j)}{\partial x_{2j}} \\ \vdots & \vdots \end{bmatrix}. \tag{1.96}$$

Then, $\mathbf{G}_i$ is decomposed using the SVD as

$$\mathbf{G}_i = \mathbf{U}_i \mathbf{S}_i \mathbf{V}_i^T, \tag{1.97}$$

where $\mathbf{U}_i$ is a $\ell \times 2$ matrix with orthonormal columns ($\mathbf{U}_i^T \mathbf{U}_i = \mathbf{I}$), while $\mathbf{V}_i$ is a 2×2 orthonormal matrix, and $\mathbf{S}_i$ is a 2×2 diagonal matrix with the singular values s_1, s_2 on the diagonal. Note that the singular vector corresponding to the smallest non-zero singular value of $\mathbf{G}_i$ represents the dominant orientation of the local gradient field. Thus, the second column of $\mathbf{V}_i$, $\mathbf{v}_2 = [v_1, v_2]^T$, gives the dominant orientation angle θ_i:

$$\theta_i = \arctan\left(\frac{v_1}{v_2}\right). \tag{1.98}$$

The elongation parameter ρ_i is selected corresponding to the energy of the dominant gradient direction

$$\rho_i = \frac{s_1 + \varepsilon}{s_2 + \varepsilon}, \tag{1.99}$$

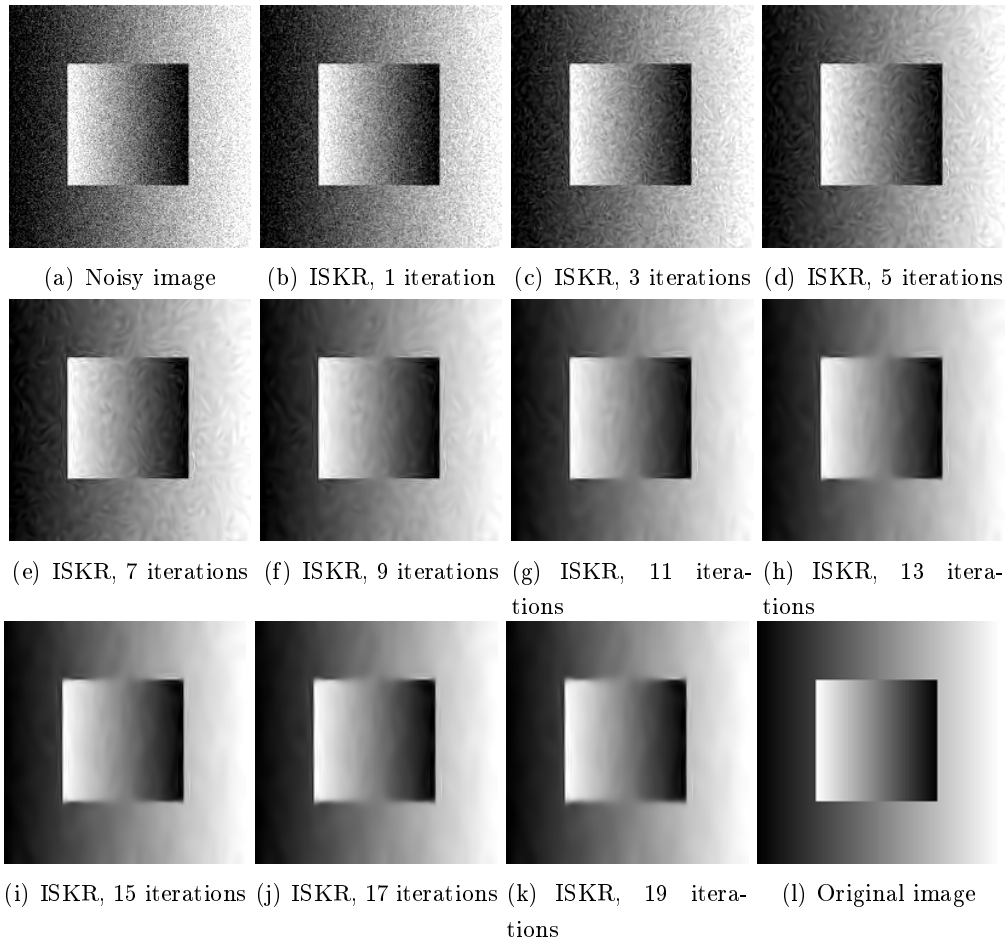
where $\varepsilon > 0$ is a regularization parameter for the kernel elongation, which restricts the ratio from becoming degenerate. Finally, the scaling parameter γ_i is defined by

$$\gamma_i = \sqrt{\frac{s_1 s_2 + \zeta}{\ell}} \tag{1.100}$$

where ζ is again a regularization parameter which keeps γ_i from becoming zero.

Due to the estimated steering matrices $\mathbf{H}_i$ of the method are data dependent, and thus sensitive to the noise in the input image. In order to further enhance the performance of the method, an iterative procedure is performed. First, the input noisy data is used to create the initial estimate of the output image. In the next iteration, the (less noisy) outcome of the previous step is used to recalculate a more reliable estimate of the gradient, and this process continues for a few more iterations. This iterative procedure was named *iterative steering kernel regression* (ISKR). Matlab code of the method available at: `http://users.soe.ucsc.edu/~milanfar/research/kernel-regression.html`.

To demonstrate the performance of the ISKR method, we carry out a test on two images, a synthetic image (Figure 1.25(l)) and a CT image of pelvis (Figure 1.26(b)). Here, we choose the regression order $N = 2$. The noisy images are generated by



(a) Noisy image (b) ISKR, 1 iteration (c) ISKR, 3 iterations (d) ISKR, 5 iterations

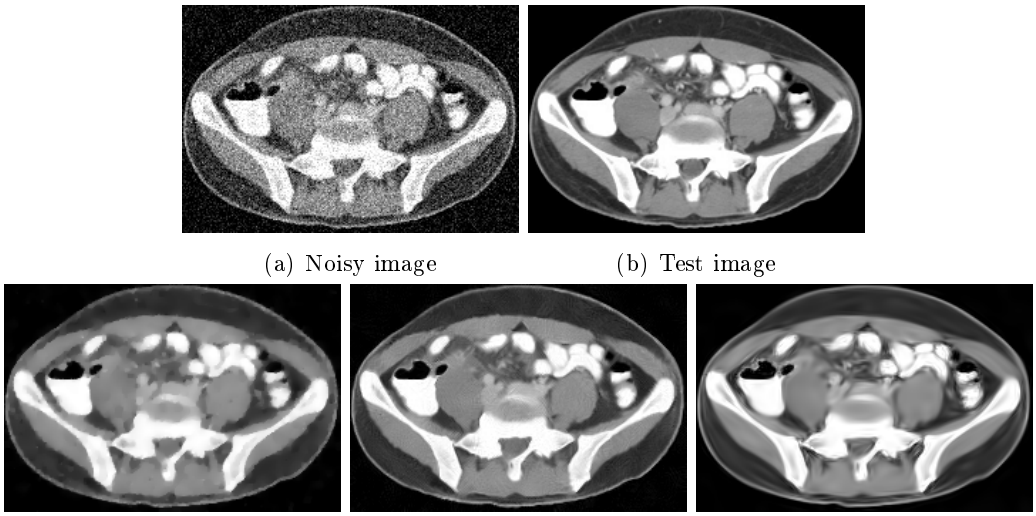
(e) ISKR, 7 iterations (f) ISKR, 9 iterations (g) ISKR, 11 iterations (h) ISKR, 13 iterations

(i) ISKR, 15 iterations (j) ISKR, 17 iterations (k) ISKR, 19 iterations (l) Original image

Figure 1.25: Experimental results of ISKR [TFM07] on a synthetic image corrupted by Gaussian noise with standard deviation of $\sigma = 25$. From left to right, from top to bottom: (a) noisy image, (b)-(k) the results of the algorithm after $1, 3, 5, \ldots, 19$ iterations, and the test image (noise-free).

adding white Gaussian noise to the test images. For the synthetic image, we perform a test with noise level $\sigma = 25$. Figure 1.25 reports the results of the ISKR after $1, 3, 5, \ldots, 19$ iterations. As can be seen, the number of iterations chosen has a significant effect on denoising. With low number of iterations, denoised result is not enough effective, and ISKR also introduces unreliable textures in homogeneous zones (see Figure 1.25(d)). With a high number of iterations, although noise in homogeneous zones is very-well removed, the estimated image would be over-smoothed at some edges (see Figure 1.25(k)).

Figure 1.26 shows the denoised results by the ISKR, the TGV [KBPS11] and the NLM [BCM06] on a pelvis CT image with noise level of $\sigma = 20$. Visually, the ISKR method of Takeda *et al.* gives better denoised result than the other methods



(a) Noisy image (b) Test image

(c) Denoised image by TGV [KBPS11] (d) Denoised image by NLM [BCM06] (e) Denoised image by ISKR [TFM07]

Figure 1.26: Gaussian denoising experiment on a CT image of pelvis: (a) Noisy mage ($\sigma = 20$), (b) original image (source: http://radiopaedia.org/), (c) denoised image by TGV [KBPS11], (d) denoised image by NLM [BCM06], and (e) denoised image by ISKR [TFM07].

but oversmooths some details.

Above we have presented the main content of the ISKR method [TFM07] and demonstrate the performance of the method though some experiments with additive Gaussian noise. The experimental results shown that this method is very good for Gaussian denoising. Even so, up to now, this method has not yet been extended for the other types of noise.

1.5.4 Image Denoising Via Learned Dictionaries and Sparse representation

Image denoising via learned dictionaries and sparse representation was first introduced in [EA06] by Elad and Aharon. Here, noise is assumed to be zero-mean white and homogeneous Gaussian additive noise as in (1.3). Sparse representation approach is studied extensively in recent years and it is also known as one of the most effective denoising methods. Before going into more details, we recall some basis notations.

The ℓ_p-norm of vector x is $\|x\|_p = (\sum_i |x_i|^p)^{1/p}$, and $\|x\|_0 = \lim_{p\to 0} \|x\|_p^p$ is the ℓ_0 pseudo-norm, i.e. the number of nonzero components. An atom is an elementary signal-representing template. A dictionary $\Phi = [\phi_1, \phi_2, \ldots, \phi_L]$ defines a $n \times L$ matrix whose columns are unit ℓ_2-norm atoms $\phi_i \in \mathbb{R}^n$. When the dictionary has

more columns than rows, it is called overcomplete or redundant. A vector $x \in \mathbb{R}^n$ has a sparse representation over the dictionary Φ if $x = \Phi\alpha$ with a small $\|\alpha\|_0$ ($\|\alpha\|_0 \ll n$). This means that x can be represented as a linear combination of few atoms from the dictionary Φ.

Mathematically, the problem of finding the sparse representation of x over Φ is

$$\alpha_x = \arg\min_{\alpha} \|\alpha\|_p \qquad \text{subject to} \qquad \|x - \Phi\alpha\| \leq \varepsilon, \tag{1.101}$$

where $\|\alpha\|_p$ is the penalty quantifying sparsity (p is often satisfy $0 \leq p \leq 1$), and ε is a small constant balancing the sparsity and the approximation error. In practice, one needs to define how deep is the required sparsity, such as $\|\alpha\|_p \leq L$ with L is a scalar controlling the sparsity.

Problem (1.101) is not easy to solve in general. In the case of $p = 0$, a greedy algorithm such as Orthogonal Matching Pursuit [PRK93] is usually used to obtain a approximate solution. In the case of $p = 1$, the problem (1.101) is convex and can be effectively solved with the Least Angle Regression (LARS) algorithm [EHJT04] or the Iterative-Shrinkage algorithm [DDD04, CDS01]. We point the interested reader to the book of Elad [Ela10] for more details.

In the denoising problem, x is not available to code and what we have is the observed version of it y. Suppose that x is corrupted by additive white Gaussian noise,

$$y = x + \eta, \tag{1.102}$$

where $\eta \sim \mathcal{N}(0, \sigma^2)$. In order to restore x from y, y is first sparsely coded over Φ by solving the following sparse decomposition problem:

$$\alpha_y = \arg\min_{\alpha} \|\alpha\|_p \qquad \text{subject to} \qquad \|y - \Phi\alpha\|^2 \leq \tau. \tag{1.103}$$

where threshold τ depends on ε and σ. Lagrange multipliers offer an equivalent formulation

$$\alpha_y = \arg\min_{\alpha} \lambda\|\alpha\|_p + \frac{1}{2}\|y - \Phi\alpha\|^2, \tag{1.104}$$

where the parameter λ balances sparsity of the solution and fidelity of the approximation to y. Denote $\hat{x}$ the denoised version of x. In order to $\hat{x}$ can be close to the true image x, α_y is expected to be very close to α_x in (1.101). Therefore, the desired image can be computed as $\hat{x} = \Phi\alpha_y$.

Now we come back to our problem of denoising in the image $\mathbf{f}$ under sparsity prior. In this approach, an image with size $\sqrt{N} \times \sqrt{N}$ is divided to M overlapping blocks (patches) $\mathbf{f}_i$ of size $\sqrt{n} \times \sqrt{n}$ ($n \ll N$) and denoising is then performed blockwise. These blocks are often taken overlap for two reason: first, to avoid blockiness and second for a better denoising process. Accordingly, the problem can

be classified into two main phases: *training phase* and *denoising phase*. The first phase is to determine an optimal overcomplete dictionary Φ adapted for the noisy image $\mathbf{f}$. In the second one, denoising is performed based on sparse coding each image patch of $\mathbf{f}$ over the dictionary Φ. The principal content of these phases can be briefly represented as follows:

Training phase: The dictionary $\Phi = [\phi_1, \phi_2, \ldots, \phi_L] \in \mathbb{R}^{n\times L}$ is learned from the following optimization problem

$$\min_{\Phi\in\mathcal{C},\{\alpha_i\}_{i=1}^M} \sum_{i=1}^{M} \|\alpha_i\|_p \quad \text{subject to} \quad \|\mathbf{f}_i - \Phi\alpha_i\|_2^2 \le \tau,\ 1 \le i \le M, \tag{1.105}$$

where $\mathcal{C}$ is the family of matrices in $\mathbb{R}^{n\times K}$ with unit ℓ_2-norm columns. In practice, the sparsity constraint $\|\alpha\|_p \le L$ is usually used, and the problem of jointly optimizing the dictionary Φ and the representation coefficient matrix of representation vectors α_i, $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_M]$, is formulated as

$$\Phi, \{\hat{\alpha}_i\}_{i=1}^M = \arg\min_{\Phi\in\mathcal{C},\{\alpha_i\}_{i=1}^M} \sum_{i=1}^{M} \|\mathbf{f}_i - \Phi\alpha_i\|_2^2 \quad \text{subject to} \quad \|\alpha_i\|_p \le L,\ 1 \le i \le M. \tag{1.106}$$

The problem (1.106) is a joint optimization problem of Φ and $\boldsymbol{\alpha}$, and it can be effectively solved by approximation algorithms such as the *method of optimal directions* (MOD) [EAH00] or K-SVD [AEB06] (see [Ela10] for more details).

Denoising phase: Suppose that we have obtained all $\hat{\alpha}_i$ and updated dictionary Φ. Then, the final denoised image $\hat{\mathbf{u}}$ is the solution of the following problem

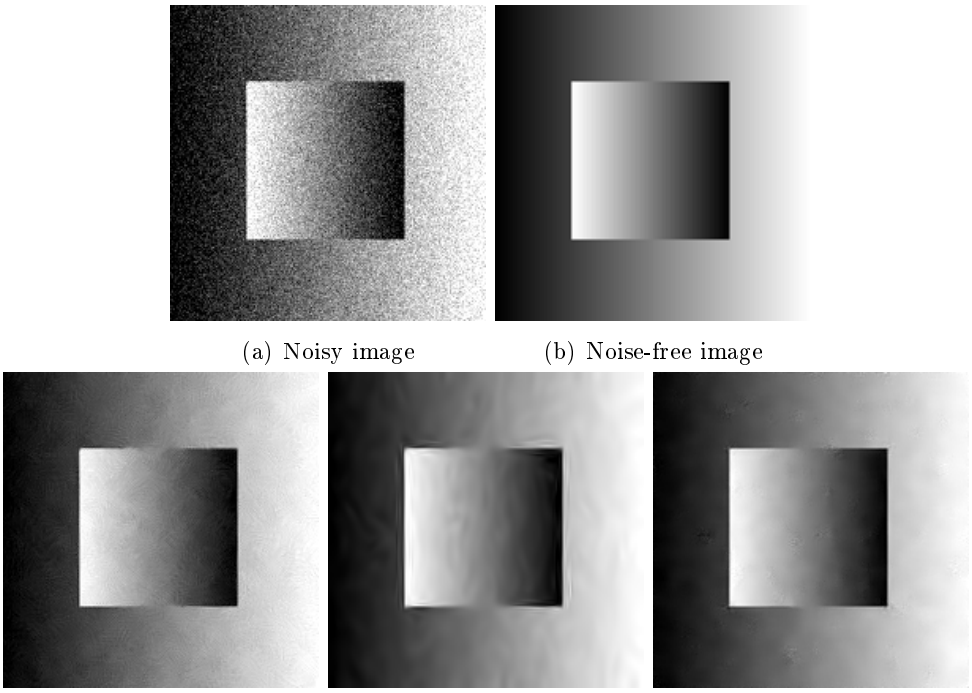
$$\hat{\mathbf{u}} = \arg\min_{\mathbf{x}} \lambda\|\mathbf{f} - \mathbf{x}\|_2^2 + \sum_i \|\Phi\hat{\alpha}_i - \mathbf{R}_i\mathbf{x}\|_2^2, \tag{1.107}$$

where $\mathbf{R}_i$ is the operator that extracts a patch $\mathbf{x}_i$ at location i of $\mathbf{x}$. This is a simple quadratic problem that has a closed-form solution of the form

$$\hat{\mathbf{u}} = \left(\lambda\mathbf{I} + \sum_i \mathbf{R}_i^T\mathbf{R}_i\right)^{-1}\left(\lambda\mathbf{f} + \sum_i \mathbf{R}_i^T\Phi\hat{\alpha}_i\right), \tag{1.108}$$

where $\mathbf{R}_i^T$ is the transpose of matrix $\mathbf{R}_i$.

To illustrate the performance of the K-SVD method with the additive white Gaussian noise, a subjective comparison with the NLM method [BCM06] and the ISKR method [TFM07] is shown in Figure 1.27 and Figure 1.28. Figure 1.27 shows the experimental results on a synthetic image with noise level $\sigma = 25$. Figure 1.28 shows the results on a CT image of pelvis with noise level $\sigma = 20$. As it can be seen,



(a) Noisy image (b) Noise-free image

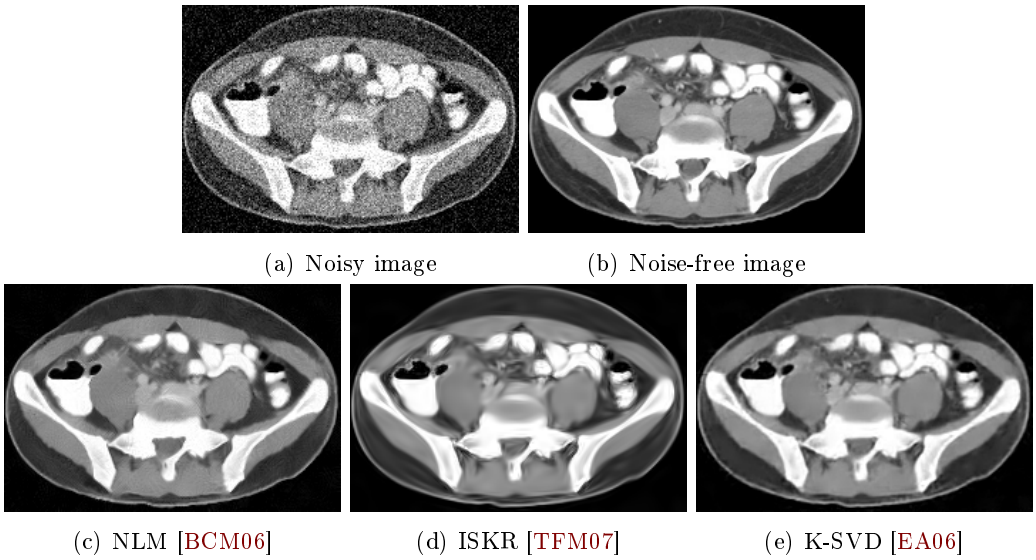
(c) NLM [BCM06] (d) ISKR [TFM07] (e) K-SVD [EA06]

Figure 1.27: Gaussian denoising experiment on a synthetic image: (a) Noisy mage ($\sigma = 25$), (b) Noise-free image, (c) Result of the NLM method in [BCM06], (d) Result of the ISKR method in [TFM07], and (e) Result of the K-SVD method [EA06].

the K-SVD method outperforms the NLM method [BCM06] and the ISKR method [TFM07] in both experiments. In the denoised results by K-SVD (Figure 1.27(e) and Figure 1.28(e)), noise is effectively removed while preserving better the edges and the small details compared with the other methods.

1.5.4.1 Conclusion

Till now, many advances of the sparse representation-based method have been proposed such as [MBP+09, DlZS11]. However, most of them focus on improving the performance of the method by modifying the original model or finding a new algorithm for the model with the assumption that the image is corrupted by Gaussian white noise. To deal with the other types of noise such as Poisson, Rician or Speckle noise, finding a corresponding denoising model via sparse representation is still an open problem.



(a) Noisy image (b) Noise-free image

(c) NLM [BCM06] (d) ISKR [TFM07] (e) K-SVD [EA06]

Figure 1.28: Gaussian denoising experiment on a CT image of pelvis: (a) Noisy image ($\sigma = 20$), (b) Noise-free image (source: http://radiopaedia.org/), (c) Denoised image by NLM [BCM06], (d) Denoised image by ISKR [TFM07], (e) Denoised image by K-SVD [EA06].

1.5.5 Block matching and 3D collaborative filtering

The transform-domain based approach is a very popular ones in signal denoising. It is based on assumption that in the transform domain, the true signal can be well approximated by a linear combination of few basis elements (sparse representation). That means most of the energy of the input image concentrates in a few high-magnitude coefficients. Hence, by keeping only a number of these high-magnitude coefficients (leading to the shrinkage) while discarding the rest of the transform coefficients which are mainly due to noise, the true signal can be effectively recovered. Thus, the effectiveness of the denoising depends on the sparsity of the true signal, i.e. the true signal can be better separated from the noise when its energy is compactly represented in the transform domain. However, the sparsity of the representation depends on both the transform and the true signal's properties [DFKE07]. In the case of 2D-signal, 2D transforms cannot always achieve good sparsity for all kinds of images due to their huge varieties, and so the noise reduction is limited. To overcome this limitation, Dabov *et al.* in [DFKE07] introduced a novel strategy named Block Matching 3D filter (BM3D) designed for normally distributed noise removal in 2D image. Up to now, it can be considered as one of the best methods in image denoising. The main idea is to stack similar 2D image fragments (e.g. blocks) together in 3-D arrays, and then performs denoising through transform-

domain shrinkage of the 3-D arrays. The basic operating principle behind the success of the BM3D is the fact that the similarity between the grouped blocks leads to a high correlation of the 3D arrays in all three dimensions, implying that the 3D transformation can reach a highly sparse representation of the true signal, allowing hence separation of the noise by shrinkage.

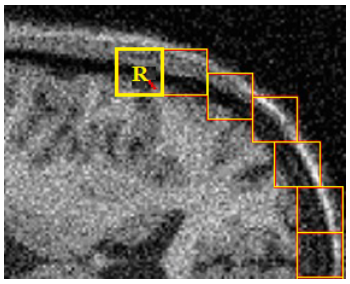


Figure 1.29: Block Matching

1.5.5.1 BM3D filter for Gaussian noise

Let us consider the case where the input noisy image $\mathbf{f}$ is assumed to be corrupted by an additive white Gaussian noise $\mathbf{n}$ with variance σ^2 and zero mean,

$$\mathbf{f}(x) = \mathbf{u}(x) + \mathbf{n}(x),\ x \in \Omega.$$

To apply the BM3D, the noisy image $\mathbf{f}$ is divided into fixed size squared blocks B_{x_R} denominated as "reference blocks", where x_R is the coordinate of the top-left corner of the block. The algorithm consists of two main steps including a basic estimate and a final one which are described briefly as follows:

1. **Basic estimate**
   The purpose of this step is to find a coarse estimate $\mathbf{u}^{basic}$ for $\mathbf{u}$. Finding $\mathbf{u}^{basic}$ is performed in three sub-steps as follows:

   - *Grouping by block matching:* For each block B_{x_R} in the noisy image, apply the Block Matching (BM) technique is applied to find similar blocks which are stacked to form a 3D array, i.e. a group. Block-matching allows finding image blocks similar to a given reference one (see Figure 1.29). This is performed by pairwise comparing the similarity between the reference block B_{x_R} and candidate blocks B_x based on a measure of similarity $d(B_{x_R}, B_x)$ between the blocks. In order to reduce mismatch due to the noise, the authors proposed a new measure of similarity using a coarse prefiltering. This prefiltering is realized by applying a normalized 2D

linear transform to all blocks and then hard-thresholding. The measure $d(B_{x_R}, B_x)$ is defined as

$$d(B_{x_R}, B_x) = \frac{\|\Upsilon'(\mathcal{T}_{2D}^{ht}(B_{x_R})) - \Upsilon'(\mathcal{T}_{2D}^{ht}(B_x))\|_2^2}{(N^{ht})^2}, \tag{1.109}$$

where Υ' is the hard-thresholding operator, $\mathcal{T}_{2D}^{ht}$ is the normalized 2D linear transform and N^{ht} is the number of pixels in a row of the block and ht stands for hard-thresholding. The result of the BM operation is the set

$$S_{x_R}^{ht} = \{x \in \Omega \ : \ d(B_{x_R}, B_x) \leq \tau_{match}^{ht}\} \tag{1.110}$$

of coordinates of all blocks τ_{match}^{ht}-similar to the reference one, with τ_{match}^{ht} is a preset threshold.

After obtaining $S_{x_R}^{ht}$, the matched noisy blocks are stacked to form a group, i.e. a 3D array $\mathbf{B}_{S_{x_R}^{ht}}$ of size $N^{ht} \times N^{ht} \times |S_{x_R}^{ht}|$, where $|S_{x_R}^{ht}|$ denotes the cardinality of $S_{x_R}^{ht}$.

- *Collaborative hard thresolding:* A transform domain-based collaborative filtering of the high correlation group $\mathbf{B}_{S_{x_R}^{ht}}$ is then realized to attenuate the noise by hard-thresholding of the transform coefficients in 3D transform domain, yielding a 3D array of blockwise estimates:

$$\hat{\mathbf{U}}_{S_{x_R}^{ht}}^{ht} = \mathcal{T}_{3D}^{ht\,-1}\left(\Upsilon\left(\mathcal{T}_{3D}^{ht}\left(\mathbf{B}_{S_{x_R}^{ht}}\right)\right)\right), \tag{1.111}$$

where $\mathcal{T}_{3D}^{ht}$ (its inverse transform $\mathcal{T}_{3D}^{ht\,-1}$) is a normalized 3D linear transform, Υ is a hard-threshold operator. The array $\hat{\mathbf{U}}_{S_{x_R}^{ht}}^{ht}$ includes $|S_{x_R}^{ht}|$ estimates of true image blocks stacked which are denoted by $\hat{U}_x^{ht,x_R}$ with $x \in S_{x_R}^{ht}$, where x_R indicates the reference block. We obtain the group of these estimates,

$$\hat{\mathbf{U}}_{S_{x_R}^{ht}}^{ht} = \{\hat{U}_x^{ht,x_R} \ : \ x \in S_{x_R}^{ht}\}. \tag{1.112}$$

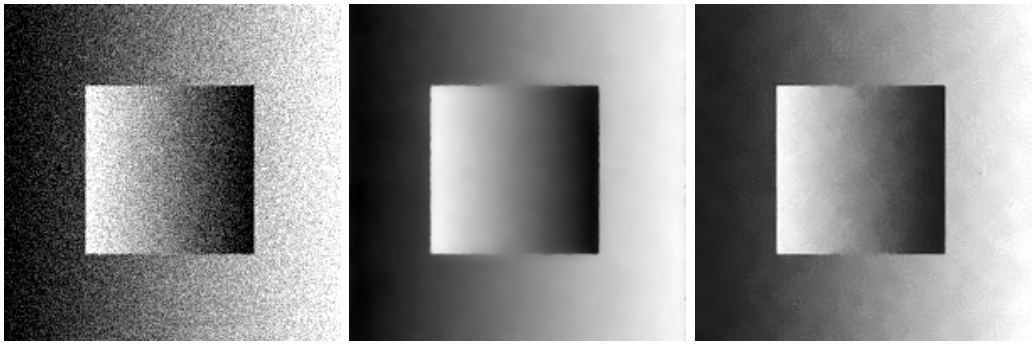
- *Aggregation by weighted average:* This step constructs the basic estimate, denoted by $\hat{\mathbf{u}}^{basic}$, of the true-image by repositioning the estimated blocks to their original positions, then performing weighted averaging to take into account multiple blocks overlap. Accordingly, for each $x_R \in \Omega$ the group of estimated blocks $\hat{U}_{S_{x_R}^{ht}}^{ht,x_R}$ is assigned the weight

$$w_{x_R}^{ht} = \begin{cases} \frac{1}{\sigma^2 N_{har}^{x_R}}, & \text{if } N_{har}^{x_R} \geq 1 \\ 1, & \text{otherwise} \end{cases} \tag{1.113}$$

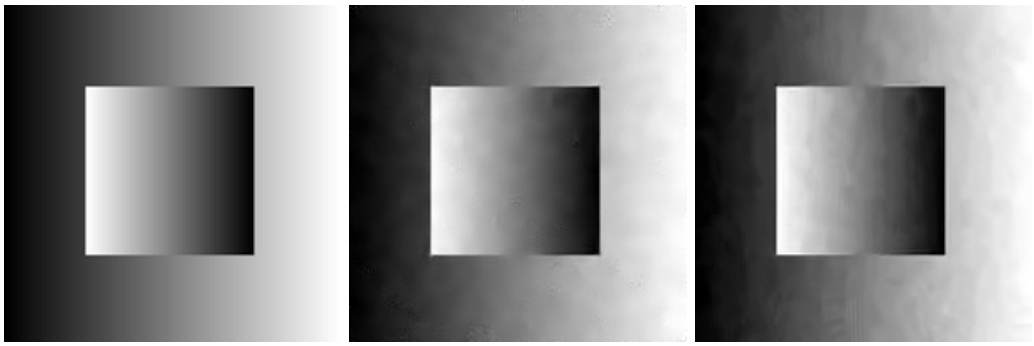
where $N_{har}^{x_R}$ is the number of non-zero coefficients in the transform domain after hard-thresholding. The global basic estimate $\hat{\mathbf{f}}^{basic}$ is constructed by

$$\hat{\mathbf{u}}^{basic}(x) = \frac{\sum_{x_R \in \Omega} \sum_{x_i \in S_{x_R}^{ht}} w_{x_R}^{ht} \hat{U}_{x_i}^{ht,x_R}(x)}{\sum_{x_R \in \Omega} \sum_{x_i \in S_{x_R}^{ht}} w_{x_R}^{ht} \chi_{x_i}(x)}, \forall x \in \Omega \tag{1.114}$$

with $\chi_{x_i} : \Omega \to \{0, 1\}$ is the characteristic function of the square support of a block located at $x_i \in \Omega$, and the estimates $\hat{U}_{x_i}^{ht,x_R}$ are zero-padded outside of their support.



(a) Noisy image (b) TGV [KBPS11] (PSNR=39.02, SSIM=**0.973**) (c) NLM [BCM06] (PSNR=41.09, SSIM=0.823)



(d) Noise-free image (e) K-SVD [EA06] (PSNR=**41.37**, SSIM=0.959) (f) BM3D [DFKE07] (PSNR=40.77, SSIM=0.952)

Figure 1.30: Gaussian denoising experiment on a synthetic image: (a) Noisy mage ($\sigma = 25$), (b) TGV's result [KBPS11], (c) NLM's result [BCM06], (d) Noise-free image, (e) K-SVD's result [EA06], and (f) BM3D's [DFKE07].

2. **Final estimate**

   Given the basic estimate of the true image obtained in Step 1, the denoising can be improved in Step 2 by performing stacking within this basic estimate and collaborative empirical Wiener filtering. In Step 2, each sub-step is a modified version of the ones in the Step 1, as follows

- *Grouping by block matching:* BM is here applied to the basic estimate $\hat{\mathbf{u}}^{basic}$ in order to obtain higher accuracy. Since the noise in $\hat{\mathbf{u}}^{basic}$ is significantly reduced, the d-distance in (1.109) is replaced with the normalized squared ℓ^2-distance.Therefore, the result of BM is

$$S_{x_R}^{wie} = \left\{ x \in \Omega \ : \ \frac{\left\| \hat{B}_{x_R}^{basic} - \hat{B}_{x}^{basic} \right\|_2^2}{(N^{wie})^2} \leq \tau_{match}^{wie} \right\} \tag{1.115}$$

where $\hat{B}_{x_R}^{basic}$ and $\hat{B}_{x}^{basic}$ are the reference block and the compared one, respectively, in $\hat{\mathbf{u}}^{basic}$, and N^{wie} is the width of the blocks and *wie* stands for Wiener filtering. Then, the sets $\{\hat{B}_{x}^{basic} \in \hat{\mathbf{u}}^{basic} \ : \ x \in S_{x_R}^{wie}\}$ and $\{B_x \in \mathbf{f} \ : \ x \in S_{x_R}^{wie}\}$ are grouped to 3D forms yielding $\hat{\mathbf{B}}_{S_{x_R}^{wie}}^{basic}$ and $\mathbf{B}_{S_{x_R}^{wie}}$.

- *Collaborative Wiener filtering:* By applying a 3D transforms $\mathcal{T}_{3D}^{wie}$ to both 3D arrays, the empirical Wiener shrinkage coefficients from the energy of the 3-D transform coefficients of the group $\hat{\mathbf{B}}_{S_{x_R}^{wie}}^{basic}$ can be expressed as

$$\mathbf{W}_{S_{x_R}^{wie}} = \frac{\left| \mathcal{T}_{3D}^{wie} \left(\hat{\mathbf{B}}_{S_{x_R}^{wie}}^{basic} \right) \right|^2}{\left| \mathcal{T}_{3D}^{wie} \left(\hat{\mathbf{B}}_{S_{x_R}^{wie}}^{basic} \right) \right|^2 + \sigma^2}. \tag{1.116}$$

Then, the collaborative Wiener filtering of $\mathbf{B}_{S_{x_R}^{wie}}$ is realized (in transform domain) by performing element-by-element multiplication the Wiener shrinkage coefficients with the 3D transform coefficients of the noisy group $\mathbf{B}_{S_{x_R}^{wie}}$. Afterwards, the 3D transform of the noisy image blocks gives the final image blocks.

$$\hat{\mathbf{U}}_{S_{x_R}^{wie}}^{wie} = \mathcal{T}_{3D}^{wie^{-1}} \left(\mathbf{W}_{S_{x_R}^{wie}} \mathcal{T}_{3D}^{wie} \left(\mathbf{B}_{S_{x_R}^{wie}} \right) \right) \tag{1.117}$$

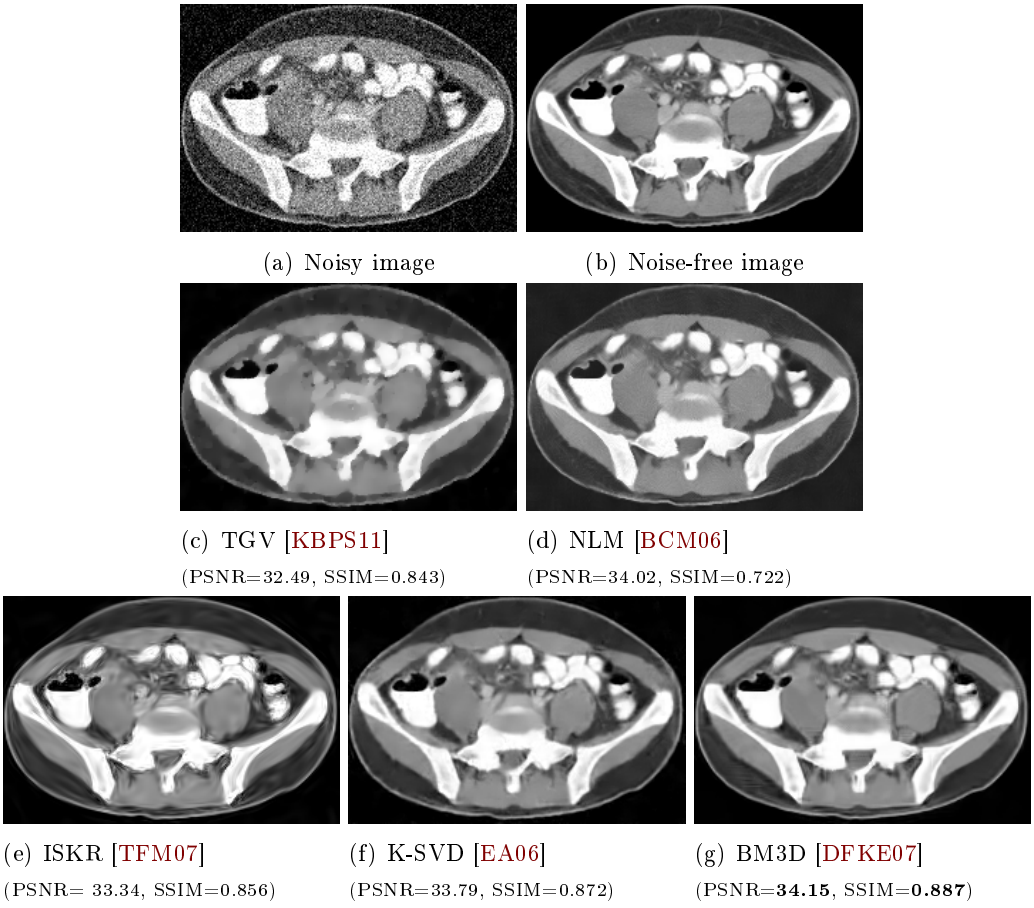
This 3D group comprises of the block-wise estimates $\hat{U}_{x}^{wie,x_R}$. Hence, $\hat{\mathbf{U}}_{S_{x_R}^{wie}}^{wie,x_R} = \{\hat{U}_{x}^{wie,x_R} \ : \ x \in S_{x_R}^{wie}\}$.

- *Aggregation by weighted average:* The global final estimate $\hat{\mathbf{u}}^{final}$ is computed by (1.114), where the weight $w_{x_R}^{ht}$ is replaced by

$$w_{x_R}^{wie} = \sigma^{-2} \left\| \mathbf{W}_{S_{x_R}^{wie}} \right\|_2^{-2}, \tag{1.118}$$

and $\hat{\mathbf{u}}^{basic}$, $\hat{U}_{x_i}^{ht,x_R}$, $S^{ht}x_R$ are replaced by $\hat{\mathbf{u}}^{final}$, $\hat{U}_{x_i}^{wie,x_R}$, $S_{x_R}^{wie}$, respectively.

Here, we show the Gaussian denoising results performed on a synthetic image and a pelvis CT image by different methods including the TGV [KBPS11], the NLM [BCM06], the K-SVD [EA06] and the BM3D [DFKE07]. Visually, in the case of the synthetic image Figure 1.30, the TGV method gives the best results. Regarding the experimental results with CT image in Figure 1.31, the BM3D outperforms the rest both objectively and subjectively. In Figure 1.31(g), we can see that the result of the BM3D method not only effectively removes noise but also very well preserves the image details.



(a) Noisy image (b) Noise-free image

(c) TGV [KBPS11] (PSNR=32.49, SSIM=0.843) (d) NLM [BCM06] (PSNR=34.02, SSIM=0.722)

(e) ISKR [TFM07] (PSNR= 33.34, SSIM=0.856) (f) K-SVD [EA06] (PSNR=33.79, SSIM=0.872) (g) BM3D [DFKE07] (PSNR=**34.15**, SSIM=**0.887**)

Figure 1.31: Gaussian denoising experiment on a CT image of pelvis: (a) noisy mage ($\sigma = 25$), (b) noise-free image (source: `http://radiopaedia.org/`), (c) TGV's result [KBPS11], (d) NLM's result [BCM06], (e) ISKR's result [TFM07], (f) K-SVD's result [EA06], and (g) BM3D's result [DFKE07].

1.5.5.2 BM3D filter for other noise models

Some applications of this method for removing noise in medical images have been proposed such as BM3D for Poisson removing [MF11], BM3D for Rician denoising in MRI images [Foi11] (see http://www.cs.tut.fi/~foi/GCF-BM3D/).

The application of the BM3D filter for removing the other type of signal-dependent noise is often performed in three steps. First, the noise variance is stabilized by applying a variance-stabilizing transformation $\mathcal{T}$ to the noisy image. This produces an image whose noise is asymptotically additive Gaussian noise. Second, the noise in the transformed image is removed using a conventional denoising algorithm for additive noise. Third, an inverse transformation of $\mathcal{T}$ is applied to the denoised image, obtaining the estimate of the final image of interest.

Figure 1.32 shows an experiment on the synthetic image of the Poisson denoising method of Makitalo and Foi [MF11]. In this method, the authors used the Anscombe transformation and its exact unbiased inverse (see Subsection 1.3.2). As a comparison, the result of the TV-based method for Poisson noise [LCA07] has staircasing artifact, while the NLM-based method [DTD10] very effectively denoises but seems to have artifacts near edges. As for the BM3D filter [MF11] is not only effective in removing noise but also very good in preserving edges and without introducing artifacts. Similar observation can be seen in Figure 1.33.

1.5.5.3 Conclusion

As presented above, the BM3D filter is very effective in image denoising. Moreover, it also well preserves the image details such as edges and textures. The extensions of the BM3D for the signal-dependent noise as Poisson noise [DTD10] and Rician noise [Foi11] are performed based on the variance-stabilizing transformation and its inverse. The construction of a direct strategy for the signal-dependent noises without using variance-stabilizing solution is still an open problem. Similarly, the adaptation of BM3D to Speckle noise is also still unresolved.

1.6 Summary

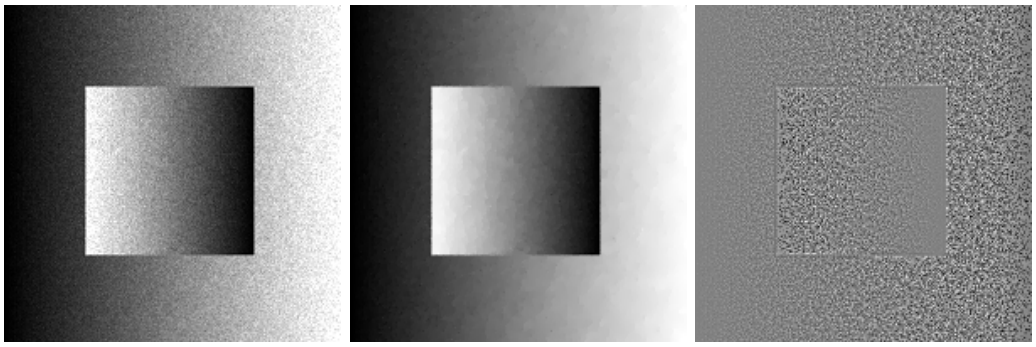
In this chapter, we briefly presented a survey of the state-of-the-art image denoising approaches including the TV-based methods, the NLM-based methods, the Data-adaptive Kernel Regression-based method, the Sparse representation-based methods and the BM3D filter-based methods. For a better understanding, different common noise models are first described followed by the description of two quality metrics used for the performance evaluation of the denoising methods. We have then inves-

tigated our effort on leading state-of-the-art denoising methods. In particular, we have also presented some recent advances of these methods for removing different common types of noise found in medical images such as Gaussian noise, Rician noise, Poisson noise, and Speckle noise. These methods, however, are designed for dealing with a particular type of noise. Furthermore, this study shows that Gaussian noise can be handled effectively, and it is still a high challenge to deal with other types of noise. This motivates our research on approaches that can deal with as large as posible types of noise.

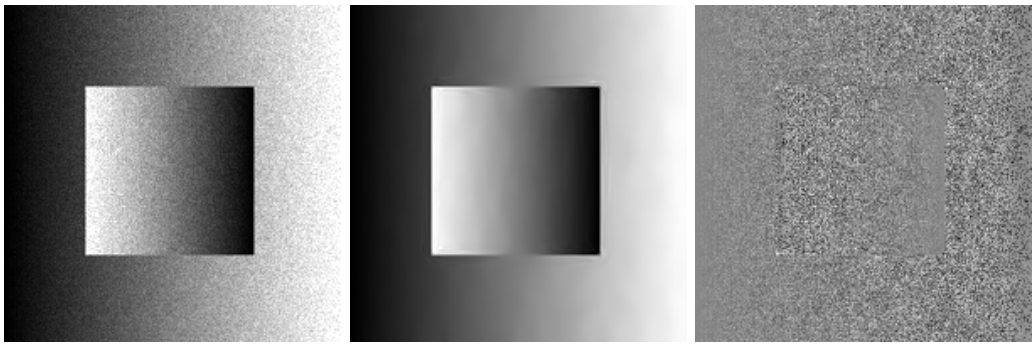
Among this state-of-the-art study, we also observed that example-based approach, while it has been well developing for super-resolution with very promising result, it not enough investigated for denoising. This would be interesting to develop learning and example-based approaches denoising and see in what extend these approaches can be applied for different types of noise in medical images. This will be presented in the next chapter.

In which, we also present some recent advances of the approaches for removing some common types of noise found in medical images. As it can be seen, in these approaches, noise component is directly removed from noisy image. While many methods to general images are significant, the special nature of medical images still requires specific solutions.

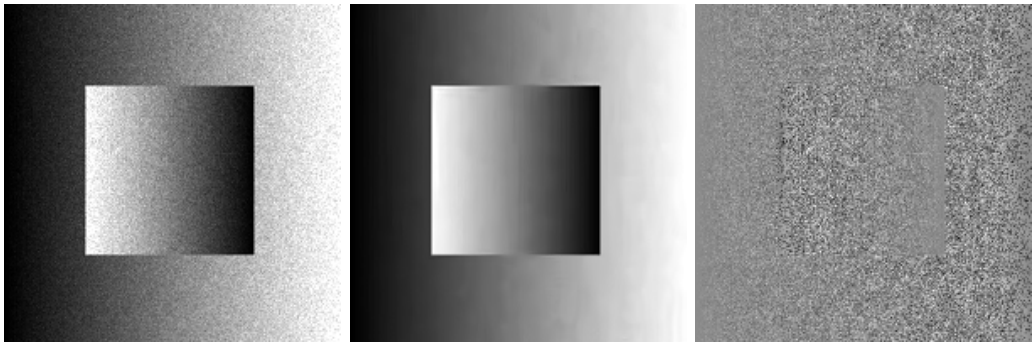
In the next chapter, we will introduce three example-based denoising methods. Unlike the above methods, in the proposed methods, denoising is performed indirectly based on learning from a given database of standard images. In the first method, we propose to apply the kernel ridge regression for denoising of medical image corrupted by Gaussian noise and Rician noise. In the second method, we introduce another example-based learning method for reducing Gaussian noise and Poisson noise which often appear on medical imaging modalities using ionizing radiation. In the third method, an optimization model for denoising on image corrupted by Gaussian noise is proposed. In this method, denoising is performed patch-wise based on finding the sparse positive linear representation of the noisy patch over the database of noise-free patches. The proposed methods are competitive with the existing state-of-the-art denoising approaches.



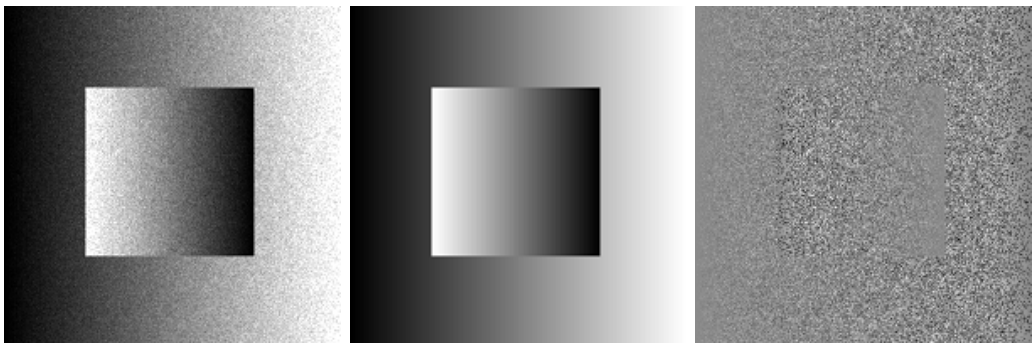
(a) Noisy image (b) TV [LCA07] (PSNR=41.09, SSIM=0.943) (c) TV's residual



(d) Noisy image (e) NLM [DTD10] (PSNR=42.14, SSIM=**0.983**) (f) NLM's residual

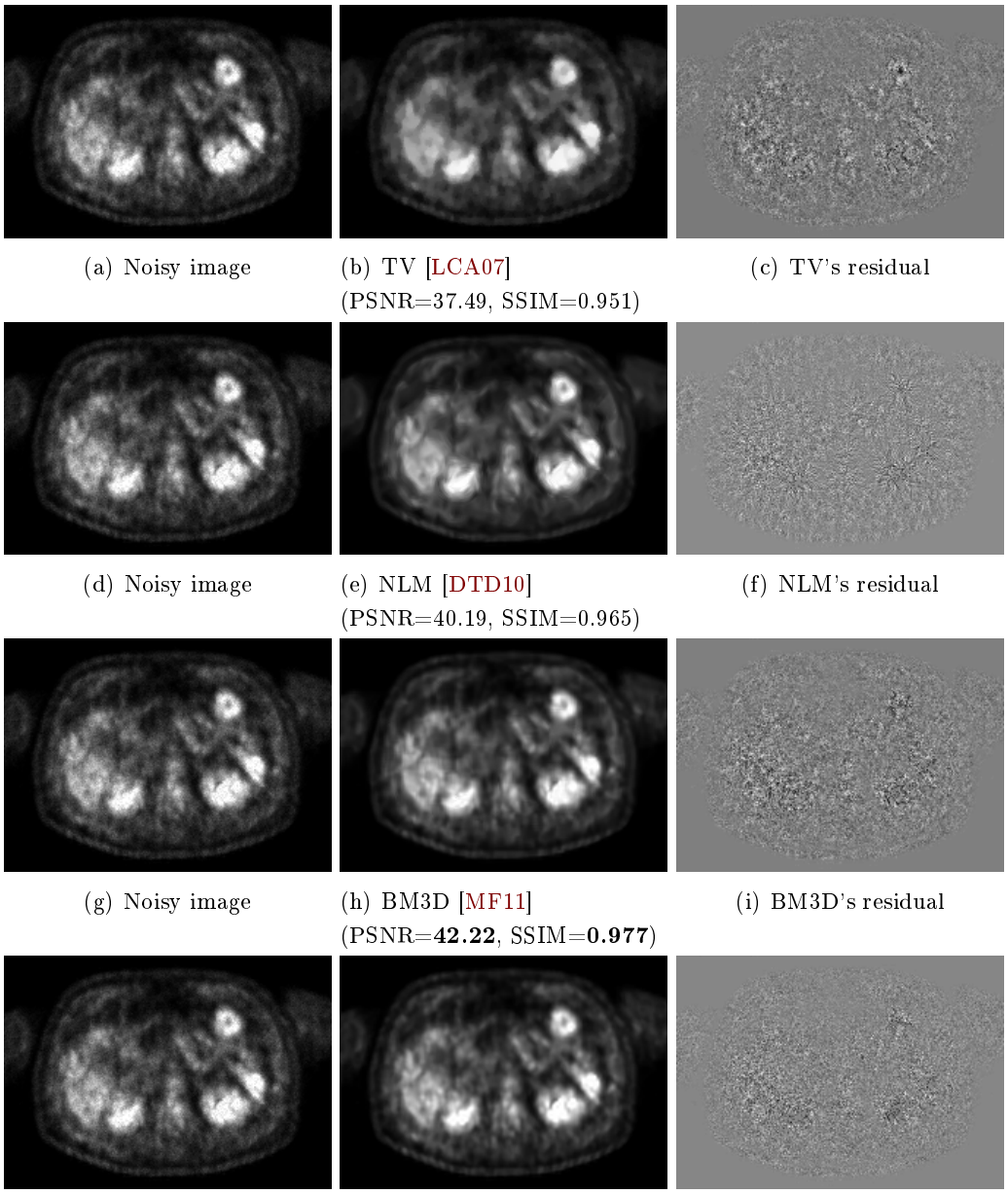


(g) Noisy image (h) BM3D[MF11] (PSNR=**46.42**, SSIM=0.981) (i) BM3D's residual



(j) Noisy image (k) Noise-free image (l) Poisson noise

Figure 1.32: Poisson denoising results of the TV-based method [LCA07], the NLM-based method [DTD10] and the BM3D method [MF11] on a synthetic image.



(a) Noisy image (b) TV [LCA07] (PSNR=37.49, SSIM=0.951) (c) TV's residual

(d) Noisy image (e) NLM [DTD10] (PSNR=40.19, SSIM=0.965) (f) NLM's residual

(g) Noisy image (h) BM3D [MF11] (PSNR=**42.22**, SSIM=**0.977**) (i) BM3D's residual

(j) Noisy image (k) Noise-free image (l) Poisson noise

Figure 1.33: Poisson denoising results of the TV-based method [LCA07], the NLM-based method [DTD10] on a PET image of chest and the BM3D method [MF11]. Image courtesy of Dr. Jean-Marie Rocchisani (Avicenne University Hospital, Bobigny, France).

Chapter 2

The proposed methods for image denoising

Contents

2.1 Introduction

In the previous chapter we have presented some main approaches in the image denoising domain including the total variation-based approach, the non-local means-based approach, the sparse representation-based approach, the Data-adaptive Kernel Regression-based approach, and the block-matching 3D approach. Beside, we also presented these approaches that have been adapted to medical images. In this chapter, we present another denoising approach based on example and learning for medical image denoising and introduce our contributions within this approach.

Medical images are often affected by random noise, resulting in a loss in image quality and a reduction of the visibility of image features especially in low contrast regions. Such effects can thereby compromise the accuracy and the reliability of pathological diagnosis or surgery purposes. Thus, denoising is one of the major challenges in medical imaging. As an example, noise in CT imaging can be decreased by increasing the X-ray dose. However, the disadvantage of increasing radiation dose is that high X-Ray doses may be harmful to patients. As shown in [LHLL01, YLL+09], low radiation imaging is often associated with a number of quality-degrading artifacts, the most prominent of them being noise. Therefore, if noise can be removed by a robust image denoising technique, lower radiation scans become possible and thus bringing less damage to the patient. Regarding the specific nature of medical images, denoising is not an easy task, and the difficulty is almost to preserve subtle details. Hence, denoising of medical images requires specific treatment. Among various directions explored in studying this problem, learning-based methods although have not been well developed, seem to be a promising direction for investigating the problem of medical image denoising.

In medical imaging, we observe the interesting fact that many images are taken from the same type of subject at approximately the same location, and some of them can be considered as standard images (high quality or proven as noise-free images by experts or passed a preprocessing step). Thus, it would be really useful if we can use such standard images as examples to denoise another noisy image. Based on this observation, we propose in this chapter three example and learning-based denoising approaches to effectively remove noises in medical images.

Unlike the denoising approaches presented in chapter 1 where noise component is directly removed from noisy image, in our proposed methods, denoising is performed indirectly through learning from a training set constructed from a given set of standard images. This is the main difference between the proposed methods and the methods presented in the previous chapter. The idea of using noise-free images for denoising have been proposed before such as in

[LSM07, Li09, vGM06, XM11, MHFZ11]. The denoising methods based on using the noise-free images are often called the example-based denoising methods. Li *et al.* in [LSM07, Li09] proposed a denoising method using support vector regression. In this method, the authors formulate image denoising as a regression problem and use support vector regression in solving the problem, where the regression function is determined from a given set of noise-free images. Although the results are still far from satisfactory, the idea is interesting. In [vGM06], [XM11] and [MHFZ11] present the example-based methods for denoising in low-dose CT images. In these methods, to denoise a noisy low-dose CT image, a high-dose CT image (noiseless images) at the same location is required. This high-dose CT image is used as a reference image in the denoising process.

In this chapter, we introduce three example-based learning methods for denoising some common types of noise in medical images. The first one, namely KRRD, is an improved method based on the idea in [LSM07, Li09]. This method is based on learning of a high data . This proposed method can be used to reduce Gaussian noise and Rician noise. The second method, namely kNND, is proposed to outperform the first method and designed to reduce Gaussian noise and Poisson noise. The third method denoted by SWMGD is proposed to effectively remove Gaussian noise. In this method, denoising is realized patch-wise via a sparse weight model. All the proposed methods use the assumption that with a given noisy image, there exists a set of standard images taken from the same type of subject at similar locations. The details the proposed methods are presented in the next sections.

This chapter is organized as follows: Section 2.2 introduces the first proposed method, followed by the second method in section 2.3. The third method is presented in section 2.4. Section 2.5 presents a comparison of the three proposed methods. Finally, the conclusion is given in section 2.6.

2.2 The first proposed method: Kernel Ridge Regression-based Denoising (KRRD)

2.2.1 Introduction

We propose an example-based learning method using kernel ridge regression for denoising. So, we refer this method as KRRD which stands for Kernel Ridge Regression-based Denoising. This method is performed pixel-wise with the help a training data set established from a given set of standard images.

Let us start with the main idea of the machine learning approach. Suppose that we are given ℓ independent observations $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\}$ (called *training set*),

each of observation consists of a pair: a vector $\mathbf{x}_i \in \mathbb{R}^d$ (called *feature vector*) and the associated target value $y_i \in \mathbb{R}$ given to us by a trusted source. It implies that we have a discrete function from $A = \{\mathbf{x}_i\}_{i=1}^{\ell}$ to $B = \{y_i\}_{i=1}^{\ell}$:

$$\begin{array}{rcl} y : A & \longrightarrow & B \\ \mathbf{x}_i & \mapsto & y_i. \end{array} \tag{2.1}$$

The function y describes the relationship between the feature vector and its target value in the training set. The main idea of the machine learning approach is to learn this function by finding a function $f : \mathbb{R}^d \to \mathbb{R}$ such that $f|_A \approx y$. From there, with a new feature vector $\mathbf{x}$, we can estimate its target value $\hat{y} = f(\mathbf{x})$.

In [LSM07, Li09], Li proposed a method using support vector regression for denoising in natural image and text document image. This method consists of two independent phases: training phase and denoising phase.

- The purpose of the training phase is to construct a training set from a given set of standard images and then training a regression function which will be used to denoise in the denoising phase. In the training phase, the first step is to generate a training set $\{(\mathbf{x}_i, y_i)\}$ where $\mathbf{x}_i$ is a patch centered at poisition i in a noisy image and y_i is the true value of pixel at i in the associated noise-free image. For this purpose, the noise component is simulated and then added to the standard images to generate the corresponding noisy images. Finally, once the training set is obtained, the regression mapping f is estimated by using support vector regression function.

- In the denoising phase, denoising is performed pixel-wise by using the regression function trained in the training phase.

Although the idea is very interesting, the weakness of this method is that normally a very large training set is required to ensure its effectiveness. Thus, it not easy to obtain a good enough estimate for the regression function. Moreover, it would require a large amounts of memory and long computation time. That is why the results of this method are still far from satisfactory.

Unlike natural image, in medical imaging, the images may be given from different patients, and there exists the repetition of local structures between images taken at nearby location, due to the similarity in the physical and biological structures of the human body. Thus, from a set of standard images we can construct a good training set for the training phase. Moreover, unlike the existing methods [LSM07, Li09], the problem of determining the regression function from the large training set can be solved by classifying it into many smaller subset.

In this section, we propose an example-based learning solution for denoising Gaussian noise and Rician noise in medical images. The proposed solution also consists of two independent phases: training phase and denoising phase. In the training phase, we first perform classifying the training set into subsets according to some image features, namely homogeneous zones, edges/textures zones and luminance. For each of subsets, a regression function is then determined. In general, the training set is not linearly distributed. Thus, a conventional linear regression is not sufficient. In such cases, a nonlinear regression function is often required for a better estimation. Here, we use the kernel regression technique namely Kernel Ridge Regression (KRR). In the denoising phase, the KRR functions trained in the training phase are then used for denoising selectively, according to the feature of pixel in noisy images. Experimental results show that our method not only effectively removes noise but also well preserves subtle details.

Before going into the details of the proposed method in section 2.2.3, let us recall the basics of the kernel ridge regression. Then, we present the proposed method as well as the denoising machine model for medical images.

2.2.2 Kernel Ridge Regression

Assume that we are given a training set $\{(\mathrm{x}_1, y_1), \ldots, (\mathrm{x}_\ell, y_\ell)\}$ with $\mathrm{x}_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$. The goal of the regression problem is to estimate a function f which minimizes some measure of discrepancy between the estimation $\hat{y}_i = f(\mathbf{x}_i)$ and the observation value y_i. Kernel ridge regression is a well-known statistical technique to solve non linear regression. Its basic idea relies on a mapping $\phi(\mathrm{x})$ to map the training data set into a feature space $\mathcal{F}$ and then find in $\mathcal{F}$ a linear regression function according to the new training data set $\{(\phi(\mathbf{x}_1), y_1), \ldots, (\phi(\mathbf{x}_\ell), y_\ell)\}$. The linear regression function in $\mathcal{F}$ will then correspond to a nonlinear regression function in the original input space. Before giving further details, let us begin by recalling the basic concept of kernels.

2.2.2.1 The concept of kernels

Let $\mathcal{X}$ and $\mathcal{F}$ be two Euclidean spaces. A kernel is defined as a function $\mathcal{K} : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$, such that there exist a mapping $\phi : \mathcal{X} \mapsto \mathcal{F}$, where the following inner-product relation holds

$$\mathcal{K}(\mathbf{x}_1, \mathbf{x}_2) = \langle \phi(\mathbf{x}_1), \phi(\mathbf{x}_2) \rangle_{\mathcal{F}}, \tag{2.2}$$

where $\langle \cdot, \cdot \rangle_{\mathcal{F}}$ denotes the inner-product in $\mathcal{F}$ which is often referred to as the feature space. The literature on kernel methods usually refers to the space $\mathcal{F}$ as a Hilbert

space. However, in this thesis we only work on Euclidean spaces with classical *dot product* (scalar product).

The question that arises now is, which functions correspond to a dot product in some feature space. The answer is given by Mercer's condition (Vapnik 1995 [Vap95]). For a function to constitute a valid kernel, Mercer's condition must be fulfilled, which most importantly states that the kernel function $\mathcal{K}$ must be symmetric and positive definite.

- *Mercer's condition* (see [Vap95], [Bur98])
  For a given symmetric function $\mathcal{K} : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$, there exists a mapping ϕ and an expansion
  $$\mathcal{K}(\mathbf{x}, \mathbf{y}) = \sum_i \langle \phi(\mathbf{x})_i, \phi(\mathbf{y})_i \rangle \tag{2.3}$$
  if only if, for any $g(\mathbf{x})$ such that
  $$\int g(\mathbf{x})^2 d\mathbf{x} \quad \text{is finite} \tag{2.4}$$
  then
  $$\int \mathcal{K}(\mathbf{x}, \mathbf{y}) g(\mathbf{x}) g(\mathbf{y}) d\mathbf{x} d\mathbf{x} \geq 0. \tag{2.5}$$

Mercer's condition tells us whether or not a prospective kernel is actually a dot product in some space, but it does not tell us how to construct ϕ or even what $\mathcal{F}$ is. However, we can explicitly construct the mapping for some kernels, e.g. with the polynomial kernel (2.6), the corresponding space $\mathcal{F}$ is a Euclidean space of dimension $\binom{d+p-1}{p}$, where $d = \dim(\mathcal{X})$ [Bur98]. In the following, we list some of the most commonly used kernels:

- Homogeneous polynomial kernels:
  $$\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = (\langle \mathbf{x}_i, \mathbf{x}_j \rangle)^p, \quad p \in \mathbb{N} \tag{2.6}$$
- Inhomogeneous polynomial kernels:
  $$\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = (\langle \mathbf{x}_i, \mathbf{x}_j \rangle + c)^p, \tag{2.7}$$
  where $p \in \mathbb{N}$, $c \geq 0$.
- Sigmoid kernels:
  $$\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \tanh(a \langle \mathbf{x}_i, \mathbf{x}_j \rangle - r) \tag{2.8}$$

- Radial Basis Function (RBF) kernels (or Gaussian kernels):

$$\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2h^2}\right) \tag{2.9}$$

2.2.2.2 Kernel Ridge Regression

We present here the ridge regression algorithm in dual variables that was introduced by Saunders *et al.* in [SGV98].

Suppose that the training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^{\ell}$ was mapped into the feature space $\mathcal{F}$ by a mapping ϕ. The problem is to find a linear regression function $y = \langle \mathbf{w}, \phi(\mathbf{x})\rangle_{\mathcal{F}}$ in the feature space $\mathcal{F}$ with the new training set $\{(\phi(\mathbf{x}_i), y_i)\}_{i=1}^{\ell}$. The ridge regression model solves the following optimization problem:

$$\min_{\mathbf{w}} \left[\sum_{i=1}^{\ell}(y_i - \langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle_{\mathcal{F}})^2 + \lambda\|\mathbf{w}\|^2\right] \tag{2.10}$$

where λ is a fixed positive number used as regularization parameter to control the trade-off between the bias and the variance of the estimate.

To solve this problem, we start with re-expressing (2.10) as the form:

$$\begin{aligned} &\min_{\mathbf{w},\xi} \ \lambda\|\mathbf{w}\|^2 + \sum_{i=1}^{\ell}\xi_i^2 \\ &\text{subject to } y_i - \langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle_{\mathcal{F}} = \xi_i,\ i = 1, 2, \ldots, \ell. \end{aligned} \tag{2.11}$$

This gives Lagrangian:

$$L_P \equiv \lambda\|\mathbf{w}\|^2 + \sum_{i=1}^{\ell}\xi_i^2 + \sum_{i=1}^{\ell}\alpha_i(y_i - \langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle_{\mathcal{F}} - \xi_i), \tag{2.12}$$

where α_i, $i = 1, 2, \ldots, \ell$ are Lagrange multipliers. Then, we can replace the constrained optimization problem (2.11) by the problem of finding the saddle point of the Lagrange function L_P. It follows from the saddle point condition that the partial derivatives of L_P with respect to the primal variables $\mathbf{w}, \xi_i$ have to vanish for optimality.

$$\partial_{\mathbf{w}} L_P = 2\lambda\mathbf{w} - \sum_{i=1}^{\ell}\alpha_i\phi(\mathbf{x}_i) = 0 \tag{2.13}$$

$$\partial_{\xi_i} L_P = 2\xi_i - \alpha_i = 0,\ \forall i = 1, 2, \ldots, \ell. \tag{2.14}$$

Substituting (2.13) and (2.14) into (2.12) yields the dual optimization problem:

$$\max_{\alpha=(\alpha_1,\ldots,\alpha_\ell)} L_D = -\frac{1}{4\lambda}\sum_{i,j=1}^{\ell}\alpha_i\alpha_j\langle\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\rangle_{\mathcal{F}} - \frac{1}{4}\sum_{i=1}^{\ell}\alpha_i^2 + \sum_{i=1}^{\ell}y_i\alpha_i. \tag{2.15}$$

Denoting $\mathbf{K}$ as the $\ell \times \ell$ matrix of scalar products

$$K_{ij} = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{F}}, \tag{2.16}$$

and $\mathbf{y} = (y_1, \ldots, y_\ell)^T$, problem (2.15) can be rewritten as

$$\min_{\alpha} \frac{1}{4\lambda} \alpha^T \mathbf{K} \alpha + \frac{1}{4} \|\alpha\|^2 - \langle \mathbf{y}, \alpha \rangle. \tag{2.17}$$

Differentiating (2.17) in α, we have

$$\frac{1}{2\lambda} \mathbf{K} \alpha + \frac{1}{2} \alpha - \mathbf{y} = 0. \tag{2.18}$$

Therefore, we obtain the optimal α:

$$\alpha = 2\lambda (\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{y}, \tag{2.19}$$

where $\mathbf{I}$ is the identity matrix.

From (2.13) and (2.19), we obtain then the prediction function $y(\mathbf{x})$ given by the ridge regression procedure on the new undefined example $\mathbf{x}$ as

$$\begin{aligned} y(\mathbf{x}) &= \langle \mathbf{w}, \phi(\mathbf{x}) \rangle_{\mathcal{F}} = \left\langle \frac{1}{2\lambda} \sum_{i=1}^{\ell} \alpha_i \phi(\mathbf{x}_i), \phi(\mathbf{x}) \right\rangle_{\mathcal{F}} = \frac{1}{2\lambda} \sum_{i=1}^{\ell} \alpha_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle_{\mathcal{F}} \\ &= \frac{1}{2\lambda} \langle \alpha, \kappa \rangle = \mathbf{y}^T (\mathbf{K} + \lambda \mathbf{I})^{-1} \kappa \end{aligned} \tag{2.20}$$

where $\kappa = (\langle \phi(\mathbf{x}_1), \phi(\mathbf{x}) \rangle_{\mathcal{F}}, \ldots, \langle \phi(\mathbf{x}_\ell), \phi(\mathbf{x}) \rangle_{\mathcal{F}})^T$.

As it can be seen, the prediction function $y(\mathbf{x})$ only depends on the data through scalar products in $\mathcal{F}$, i.e. on the function of the form $\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{F}}$. Therefore, the kernel trick can be used in this case. It means that if there is a kernel function $\mathcal{K}$ such that $\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{F}}$, the regression function can be rewritten as follows:

$$y(\mathbf{x}) = a_1 \mathcal{K}(\mathbf{x}_1, \mathbf{x}) + a_2 \mathcal{K}(\mathbf{x}_2, \mathbf{x}) + \ldots + a_\ell \mathcal{K}(\mathbf{x}_\ell, \mathbf{x}) \tag{2.21}$$

where $(a_1, a_2, \ldots, a_\ell) = \mathbf{y}^T (\mathbf{K} + \lambda \mathbf{I})^{-1}$. Here, we can also see that, the direct mapping $\phi(\mathbf{x})$ is not used. Hence, it is not necessary to explicitly obtain $\phi(\mathbf{x})$ as long as we can access the kernel function.

2.2.3 The proposed KRRD method

In this section, we present our proposed denoising method. Suppose that $\mathbf{X}$ is a noisy image that needs to be denoised and $\{A_k\}$ is a set of the standard images that are taken at nearly the same location as $\mathbf{X}$. There are three main phase: *noise estimation phase*, *training phase* and *denoising phase*.

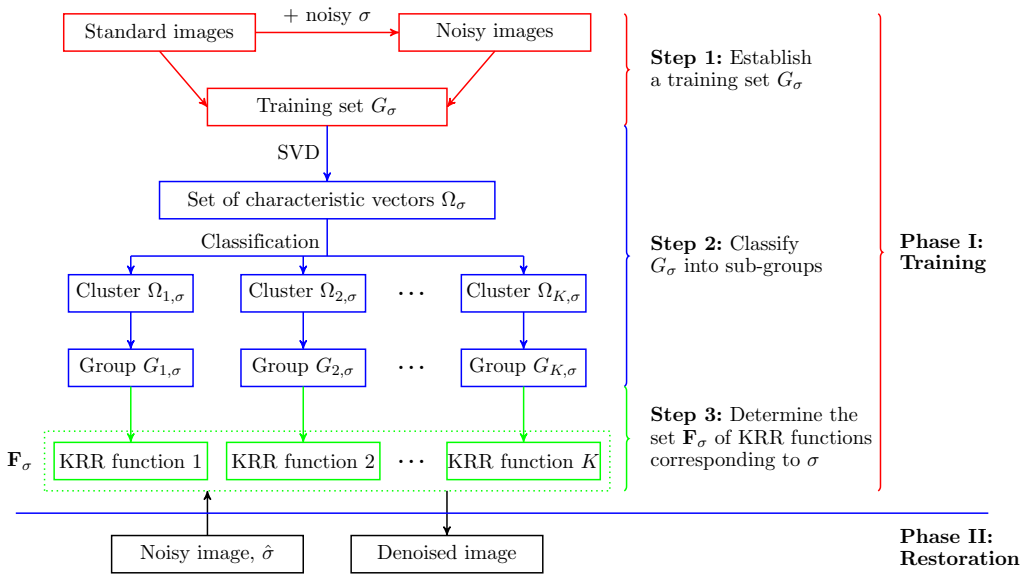


Figure 2.1: Overview of the denoising method using Kernel Ridge Regression (KRR).

- The aim of the noise estimation phase is to estimate the noise component from the input noisy image.
- In the training phase, a training set is first constructed from the given standard images. This training set is then classified into some training subsets before determining the regression functions for each of them.
- In the denoising phase, the regression functions which are trained in the training phase will be used to estimate the output image (denoised image).

An overview of the proposed method is illustrated in Figure 2.1. In remaining of the method, we consider $\sigma \approx \hat{\sigma}$, where σ is preset noise level used in the trainning phase and $\hat{\sigma}$ is the noise level estimated from the noisy image $\mathbf{X}$.

In the following, we will describe in more details each phase.

2.2.3.1 Noise estimation

To obtain an effective denoising solution for $\mathbf{X}$, it is important to determine the type of noise and the level of the noise in the image. In fact, it is not possible to determine precisely the nature of noise. However, it is possible to estimate its statistical properties such as the mean value, the variance or the probability density function. Normally, the type of the noise often depends on the type of image.

For example, if $\mathbf{X}$ is CT an image then noise in $\mathbf{X}$ is often Gaussian noise. If $\mathbf{X}$ is a MRI image then noise in $\mathbf{X}$ is often Rician noise. Regarding the problem of noise estimation, there are many effective methods that have been proposed [LFSK06, SdDA+98, CMG+10, RPJS10, LHLC12]. In the scope of this work, we directly use the available methods to implement estimating the noise level in $\mathbf{X}$ such as the method of Rajan *et al.* [RPJS10] which is used to estimate the standard deviation of Rician noise, the method of Muresan and Parks [MP03] which is used to estimate the standard deviation of the Gaussian noise.

2.2.3.2 Training phase

The aim of the training phase is to construct a set of regression functions which exploit the relationship between noisy image and noise-free image. To this end, the training phase is realized in three steps: Training set generation, training set classification and determine regression functions.

2.2.3.2.1 Training set generation

In order to construct the training set $G_\sigma = \{(\mathbf{x}_i, y_i)\}$ from a given set of standard images $\{A_k\}$ and a given noise level σ, a set of corresponding noisy images $\{B_k\}$ is first generated. Here, B_k is obtained from A_k by simulating the noise component in the noisy image $\mathbf{X}$. In general, we can establish the training set for different given noise levels σ. As specified above, we use $\sigma = \hat{\sigma}$, where $\hat{\sigma}$ is the noise level estimated from the input noisy image. Then, for each pixel i of the noisy image B_k, a pair $(\mathbf{x}_i, y_i)$ is determined, where $\mathbf{x}_i \in \mathbb{R}^n$ is the vector corresponding to a square patch of fixed size $\sqrt{n} \times \sqrt{n}$ and centered at pixel i of B_k, while y_i is the value of pixel i of the corresponding standard image A_k. From such the set of all couples of observation $(\mathbf{x}_i, y_i)$ we obtain the training set G (see Figure 2.2).

2.2.3.2.2 Training set classification

Let us denote by ℓ the number of observation examples in the training set. Normally, the number of observation examples ℓ is often very large. Moreover, due to the random distribution of the samples in the training set, it is not easy to achieve an effective regression function that is a good fit for all the training examples. In addition, it is difficult to obtain a regression function for a large training set. Considering the kernel ridge regression for instance, the matrix $(\mathbf{K} + \lambda\mathbf{I})$ in equation (2.21) is a non-sparse matrix and of size $\ell \times \ell$, it is thus not easy to compute its inverse with large ℓ. To perform the training step in an easier way, the training set needs to be classified into smaller groups. In this work, classifying the training set is realized in two steps as follows:

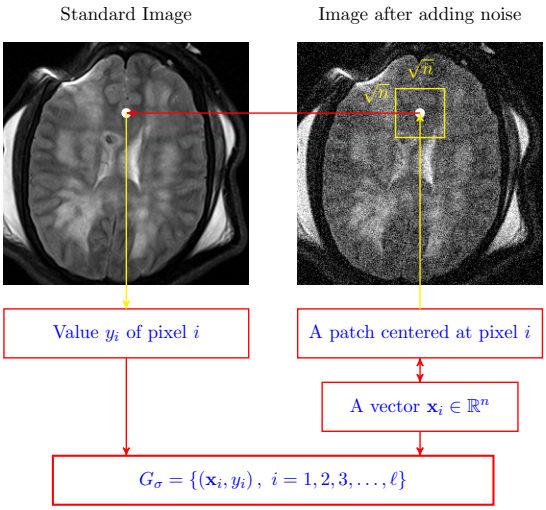


Figure 2.2: Illustration of the training set generation $G_\sigma = \{(\mathbf{x}_i, y_i)\}$. Standard original image (left) and its noised version (right). $\mathbf{x}_i$ corresponds to a patch centered at pixel i in noisy image, y_i is the value of pixel i in the standard image

- First, for each of patch $\mathbf{x}_i$ in the training set, we define an associated characteristic vector $\mathbf{v}_i$.
- Then, the training set $G_\sigma = \{(\mathbf{x}_i, y_i)\}_{i=1}^{\ell}$ is classified according to the classification of corresponding set of characteristic vectors $\{\mathbf{v}_i\}_{i=1}^{\ell}$. Consequently, each training group contains observations such that their corresponding patches have some similar features.

Let us first define the characteristic vector.

Determination of characteristic vector: There are many ways to define a characteristic vector for a given image patch $\mathbf{x}_i$. In the proposed method, the characteristic vector $\mathbf{v}_i$ is simply defined as a two dimensional vector that can express some image features of $\mathbf{x}_i$ such as homogeneous zone, texture/edge zone and luminance. In other words, we can characterize a patch $\mathbf{x}_i$ by its intensity and whether it is situated in a homogeneous zone or a texture/edge zone. More specifically, the average of pixel values in the patch is used to quantify its luminance. This is the first component of $\mathbf{v}_i$. In the other hand, as shown in [FM02], by applying the Single Value Decomposition (SVD) method to the gradient field of $\mathbf{x}_i$, we can quantify its edgeness. Indeed, for a homogeneous region, there is no dominant direction and all eigenvalues are small. For an oriented edge/texture region, there is a dominant direction and the

corresponding eigenvalue is significantly larger than the others. Therefore, the second component of the characteristic vector $\mathbf{v}_i$ is determined as follows. First, we group gradient values of the patch $\mathbf{x}_i$ into matrix D of size $n \times 2$

$$D = \left[\nabla\mathbf{x}(1)^T \nabla\mathbf{x}(2)^T \dots \nabla\mathbf{x}(n)^T\right]^T, \tag{2.22}$$

where $\nabla\mathbf{x}(j) = \left[\frac{\partial\mathbf{x}(j)}{\partial x} \frac{\partial\mathbf{x}(j)}{\partial y}\right]^T$ is the gradient of $\mathbf{x}$ at pixel j. Then, SVD of the matrix D is computed via the following formula:

$$D = USV^T, \tag{2.23}$$

where U is an $n \times n$ orthogonal matrix, S is a diagonal matrix with elements λ_1, λ_2 which are called singular values and always in decreasing order $\lambda_1 \geq \lambda_2 \geq 0$, V is an 2×2 orthogonal matrix which describes the dominant orientation of the gradient field. We use $\lambda_1^i - \lambda_2^i$ to define the second component of $\mathbf{v}_i$.

In summary, for each of patch $\mathbf{x}_i$ in the training set we define a characteristic vector

$$\mathbf{v}_i = (\mu_i, \lambda_1^i - \lambda_2^i) \in \mathbb{R}^2 \tag{2.24}$$

with λ_1^i, λ_2^i are singular values and μ_i is the mean of pixel values in $\mathbf{x}_i$. The use of characteristic vectors $\mathbf{v}_i$ for classifying the zones of an image is simulated in Figure 2.3.

(a) Noisy image (b) $\lambda_1 - \lambda_2$ (c) $(\mu, \lambda_1 - \lambda_2)$

Figure 2.3: Classification simulation based on using characteristic vectors $\mathbf{v} = (\mu, \lambda_1 - \lambda_2)$.

Classification of the characteristic vectors: This step concerns the classification of the training set into K groups ($2 \leq K \leq \ell$). It can be done by classifying the set of characteristic vectors $\Omega_\sigma = \{\mathbf{v}_i, i = 1, \dots, \ell\}$ into K clusters $\{\Omega_{c,\sigma}\}_{c=1}^K$. For an effective classification, some well-known clustering

techniques such as the K-means clustering or the fuzzy K-means clustering [Bez81] can be used.

- **K-means clustering:**
  The K-means clustering aims to partition the characteristic set Ω_σ into K $(2 \leq K < \ell)$ clusters $\{\Omega_{1,\sigma}, \Omega_{2,\sigma}, \ldots, \Omega_{K,\sigma}\}$ while minimizing the following optimization problem:

$$\min_{\{\Omega_{c,\sigma}\},\{\nu_c\}} \sum_{c=1}^{K} \sum_{\mathbf{v}_i \in \Omega_{c,\sigma}} \|\mathbf{v}_i - \nu_c\|^2, \tag{2.25}$$

  where ν_c is the mean of elements in cluster $\Omega_{c,\sigma}$. An example of the K-means clustering algorithm is illustrated in Figure 2.4.

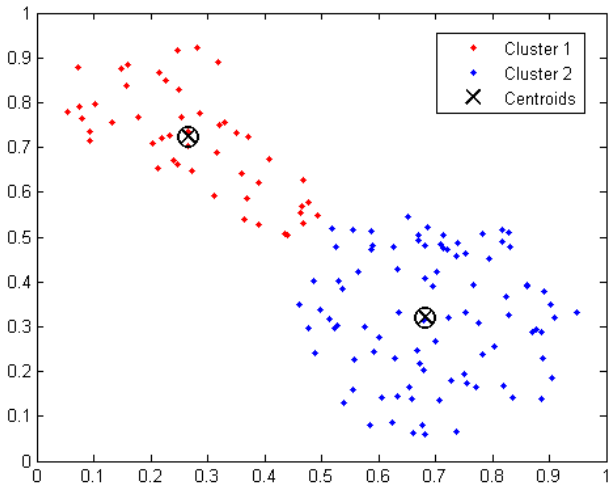


Figure 2.4: Demo of K-means clustering in case of $K = 2$.

- **Fuzzy K-Means clustering:**
  Unlike the K-means clustering, fuzzy K-means algorithm perform partition Ω_σ into K clusters by solving the following optimization problem:

$$\min_{U,\nu} J(U,\nu) = \sum_{c=1}^{K} \sum_{i=1}^{\ell} u_{ci}^2 \|\mathbf{v}_i - \nu_c\|^2, \tag{2.26}$$

  with the condition

$$u_{ci} \in [0,1]\ 1 \leq c \leq K; 1 \leq i \leq \ell,\ \sum_{c=1}^{K} u_{ci} = 1, \forall i,\ \text{and}\ \sum_{i=1}^{\ell} u_{ci} > 0\ \forall c. \tag{2.27}$$

  In (2.26), $\nu = (\nu_1, \nu_2, \ldots, \nu_K)^T \in \mathbb{R}^{K2}$, $\nu_c \in \mathbb{R}^2$ is the prototype for cluster $\Omega_{c,\sigma}$ and $U = [u_{ci}] \in \mathbb{R}^{K\ell}$ represents a non-degenerate fuzzy

K-partition of Ω_σ. When the solution U, ν of the problem (2.26) with condition (2.27) is obtained, classifying Ω_σ is determined as

$$\mathbf{v}_i \in \Omega_{c,\sigma} \text{ if } u_{ci} = \max\{u_{ki},\, k = 1, \ldots, K\}. \tag{2.28}$$

It is easy to see that in the case where the fuzzy coefficients $u_{ci} = 1\, \forall c, i$, the fuzzy K-means problem (2.26) becomes the K-means clustering. Thus, fuzzy K-means clustering can be considered as a soft version of K-means, where each data point has a fuzzy degree of belonging to each cluster.

Classification of the training set: Suppose that the set of characteristic vectors Ω_σ have been classified into K clusters. Then, classifying the training set G_σ into K groups $\{G_{c,\sigma}\}_{c=1}^{K}$ is determined as

$$(\mathbf{x}_i, y_i) \in G_{c,\sigma} \Leftrightarrow \mathbf{v}_i \in \Omega_{c,\sigma}. \tag{2.29}$$

2.2.3.2.3 Regression functions

After performing classification, we obtain the training set $G_\sigma = G_{1,\sigma} \cup G_{2,\sigma} \cup \ldots \cup G_{K,\sigma}$, where each group $G_{k,\sigma}$ has a characteristic vector $\nu_k \in \mathbb{R}^2$. Consequently, the regression functions $f_1, f_2, \ldots, f_K$ according to the groups $G_{1,\sigma}, G_{2,\sigma}, \ldots, G_{K,\sigma}$ are determined, respectively. These regression function will be used in the denoising phase. It is worth noting that the training phase can be performed off-line.

2.2.3.3 Denoising phase

In this phase, denoising an input image $\mathbf{X}$ is realized pixel-wise by using the trained regression functions in the previous phase. There are four steps in this phase:

- Step 1: For each of pixel i in $\mathbf{X}$, consider image patch of size $\sqrt{n} \times \sqrt{n}$ and centered at i an then order lexicographically as vector $\mathbf{x}$ in $\mathbb{R}^n$ (as in section 2.2.3.2.1).
- Step 2: Compute the characteristic vector $\mathbf{v} = (\mu_x, \lambda_{1,x} - \lambda_{2,x}) \in \mathbb{R}^2$ of $\mathbf{x}$.
- Step 3: Determine which the regression function to be used for estimating the true value of pixel i. The function f_k is selected if

$$k = \min_c \{\|\mathbf{v} - \nu_c\|^2, c = 1, \ldots, K\}, \tag{2.30}$$

  where ν_c is the characteristic vector of the group $G_{c,\sigma}$ (ν_c is determined from (2.26)).
- Step 4: True value of pixel i is estimated as

$$\hat{X}(i) = f_k(\mathbf{x}). \tag{2.31}$$

From (2.21), we see that the KRR function f_k in (2.31) has the form

$$f_k(\mathbf{x}) = \sum_j w_j \mathcal{K}(\mathbf{x}_j, \mathbf{x}). \tag{2.32}$$

In the proposed method, the Gaussian kernel $\mathcal{K}(\mathbf{x}_j, \mathbf{x}) = \exp(-\|\mathbf{x}_j - \mathbf{x}\|^2)/(2h^2)$ is chosen, where h is the decay parameter. The estimate of each pixel in the noisy image only depends on the training group, which includes patches that have some characteristics as the patch defined for the pixel under consideration. The Gaussian function can be seen as a measurement of similarity between two image patches. Therefore, in equation (2.32), $\mathcal{K}(\mathbf{x}_j, \mathbf{x})$ may be viewed as weights. Where, the more $\mathbf{x}_j$ is similar to $\mathbf{x}$, the higher is the weight. This shows the adaptiveness of the proposed method.

2.2.4 Denoising machine model

As presented above, the denoising using the regression functions learned from a given training set is a very promising solution. Thereby, we can construct an automatic denoising machine model for medical images.

2.2.4.1 Machine

The denoising machine is designed with many stacks, each of them is designed for one type of image. Moreover, each stack includes many sub-stacks. Each sub-stack is designed to image denoising at a certain location in the body such as brain, neck, knee, etc. For each sub-stack, there are several options corresponding to different levels of the noise. Each option has many regression functions trained from the given standard images as presented above.

2.2.4.2 Principle of Operation of the Machine

Let $\mathbf{X}$ be the image to be denoised. Then, the machine performs the following steps:

1. Determine the type of the image $\mathbf{X}$ (CT, MRI) and the position in the body of $\mathbf{X}$.

2. Estimate noise on $\mathbf{X}$: The distribution of the noise may be determined according to the opinion of experts or some statistical properties of noise found on the types of medical image (see Section 1.3). Then, noise level on $\mathbf{X}$ can be estimated.

3. Select the suitable option based on the type of image, the position in the body of $\mathbf{X}$ as well as the estimated noise level. Then, the machine performs denoising based on the regression functions of the selected option.

2.2.5 Performance evaluation

In this section, we provide some tests of the proposed KRRD method and comparisons to some other state-of-the-art denoising methods with experiments on CT images and MRI images (with two noise models, namely Gaussian noise and Rician noise, respectively). Two image quality metrics namely the PSNR and the SSIM [WBSS04] (see Section 1.4) are used to compare objectively the performance of the methods.

We first start with parameter settings. Then, we present the experiments of denoising Gaussian noise in CT images followed by the experiments of removing Rician noise in MRI images.

2.2.5.1 Parameter settings

The parameters of the method are the number of clusters K, the ridge regression parameter λ in (2.10), the patch size, the decay parameter h of the kernel function. The choices of their value are described in what follows:

1. Parameter K:
   This is a difficult problem because it is not easy to propose an optimal criteria for choosing K. In the proposed method, K is determined based on the histogram of the standard images used to establish the training set. More precisely, K is automatically determined in the three following steps:

   - **Step 1:** Merge the standard images into one larger image and then compute its histogram, $his(n)$. The histogram $his(n)$ is considered as a function of n, where $n \in [0, 1, \ldots, 255]$ for images coded on 8 bits per pixel.
   - **Step 2:** Use a filter to smooth the function $his(n)$. Let us denote by $his_{smooth} = Filter(h)$ the smoothed histogram function. In practice, we use the mean filter for smoothing the histogram function.
   - **Step 3:** Determine number of extreme points of the discrete function $his_{smooth}(n)$. Here, the point $(n, his_{smooth}(n))$ is consider as an extreme point if the following criteria is satisfied:

   $$\big(his_{smooth}(n) - his_{smooth}(n-1)\big)\big(his_{smooth}(n) - his_{smooth}(n+1)\big) > 0. \tag{2.33}$$

   Assume that the number of extreme points of $his_{smooth}(n)$ is N. Then, the number of subsets K is set to $N + 1$.

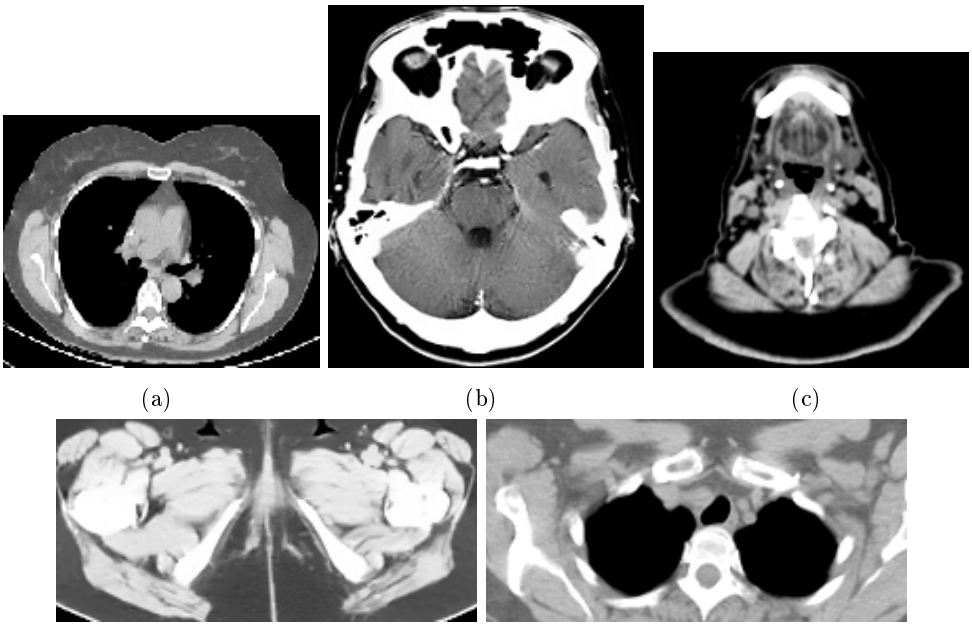


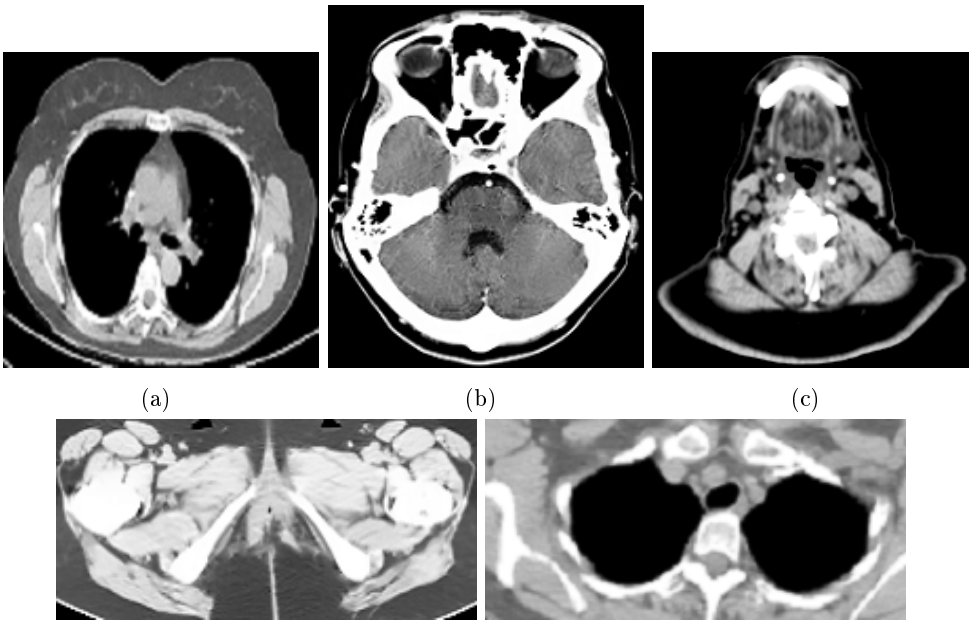
Figure 2.5: Original images used to create test noisy images: (a) CT of thorax, (b) CT of head, (c) CT of neck, (d) CT of pelvis, and (e) CT of chest. Image courtesy of Dr. Jean-Marie Rocchisani (Avicenne University Hospital, Bobigny, France).

2. The other parameters:
   In all experiment, the ridge parameter λ in (2.10) is set to 0.01, the size of the patch is set to 5×5. We use the Gaussian kernel $\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \exp -\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2h^2}$ where h is set to $\hat{\sigma}$ (the standard deviation of noise estimated from the input noisy image).

In the following subsections, we report some experiments on CT images with Gaussian noise and MRI images with Rician noise.

2.2.5.2 Denoising Gaussian noise on CT image

The experimental tests are carried out on five CT images of thorax, head, neck, pelvis, and chest (see Figure 2.5). For each test image, a training set is established using two standard images (not necessarily identical to the test image). In Figure 2.6, only one standard image is illustrated for each example to better see that the standard image is different from the test image. In these experiments, a test images is obtained from an original and high quality (noise-free) image by adding an additive, independent Gaussian noise with three standard deviation values: $\sigma = 10$, 20 and 30. The standard deviation of Gaussian noise in test images, $\hat{\sigma}$, is estimated by using method in [MP03]. In the training phase, we use $\sigma = 10$, 20 and 30 instead



(a) (b) (c)

(d) (e)

Figure 2.6: Standard CT images used to establish the training sets. Image courtesy of Dr. Jean-Marie Rocchisani (Avicenne University Hospital, Bobigny, France).

of estimated noise level $\hat{\sigma}$. Indeed, we can see in Tables 2.1 and 2.2 that σ and $\hat{\sigma}$ are very close to each other.

The performance of our KRRD method is compared with the state-of-the-art methods for denoising of images corrupted by Gaussian noise, namely the TGV [BKP10], the NLM [BCM06], the ISKR [TFM07], the K-SVD [EA06], and the BM3D [DFKE07]. The objective results are reported in Table 2.1 and Table 2.2. The best quality indices are represented in bold characters. As can be seen, the PSNR and SSIM indices of the proposed method (KRRD) are higher than that of the TGV and NLM. The quality indices of the BM3D method are the best in most cases. Compared with the K-SVD, the K-SVD has more better PSNR results than the KRRD. However, for SSIM, the KRRD has three best SSIM indices and an equivalent performance with the K-SVD.

Figure 2.7 shows the results of different methods on the CT image of thorax (Fig. 2.5(a)). We can see that the results of the ISKR, TGV, NLM, K-SVD and BM3D denoise very effectively. However, certain subtle details are lost compared to our method where noise is effectively removed in the smooth region while better preserving edges around the object contours. Moreover, residual images of CT images are also shown to illustrate the out-performance of our denoising method, as our residual image is close to the Gaussian noise and contains nearly no structural

Table 2.1: PSNR comparison of Gaussian denoising methods on CT images

Test image	σ	$\hat{\sigma}$	PSNR					
			Image denoising algorithms					
			TGV	NLM	ISKR	K-SVD	BM3D	KRRD
(a)	10	9	29.44	31.17	31.15	32.29	**32.73**	32.08
	20	21	27.32	27.37	26.86	27.78	**28.30**	27.43
	30	31	24.34	25.55	24.81	25.49	**25.92**	25.61
(b)	10	12	34.18	34.33	34.96	35.54	**36.13**	35.22
	20	20	30.37	31.25	30.94	31.82	**32.38**	31.90
	30	31	27.89	28.57	28.40	29.32	**30.12**	29.19
(c)	10	10	34.02	34.99	35.52	36.21	**36.58**	35.87
	20	19	30.19	31.32	31.45	31.94	**32.62**	31.61
	30	31	27.85	28.63	28.88	29.49	**30.27**	29.78
(d)	10	12	31.27	31.37	32.11	32.48	**32.95**	32.14
	20	21	27.94	28.37	27.98	28.73	**29.19**	28.68
	30	31	25.28	26.02	25.65	26.83	**27.12**	27.01
(e)	10	10	32.09	32.49	33.05	33.09	**33.40**	33.28
	20	18	28.76	29.51	29.29	29.58	**29.82**	29.60
	30	31	26.54	27.20	27.03	27.56	**27.82**	27.50

Table 2.2: SSIM comparision of Gaussian denoising methods on CT images

Test image	σ	$\hat{\sigma}$	SSIM					
			Image denoising algorithms					
			ISKR	TGV	NLM	K-SVD	BM3D	KRRD
(a)	10	9	0.894	0.911	0.908	0.917	**0.940**	0.911
	20	21	0.826	0.831	0.829	0.834	**0.877**	0.873
	30	31	0.766	0.788	0.769	0.802	**0.820**	0.809
(b)	10	12	0.934	0.924	0.927	0.940	**0.961**	0.930
	20	20	0.882	0.843	0.847	0.893	**0.924**	0.854
	30	31	0.833	0.823	0.784	0.847	**0.888**	0.801
(c)	10	10	0.945	0.928	0.926	0.950	**0.971**	0.938
	20	19	0.875	0.829	0.812	0.879	**0.923**	0.825
	30	31	0.831	0.788	0.731	0.819	**0.869**	0.791
(d)	10	12	0.936	0.929	0.931	0.940	**0.946**	0.929
	20	21	0.880	0.874	0.867	0.892	**0.899**	0.877
	30	31	0.828	0.811	0.791	0.849	**0.859**	0.852
(e)	10	10	0.921	0.914	0.925	0.930	0.935	**0.946**
	20	18	0.867	0.884	0.846	0.864	0.877	**0.883**
	30	31	0.816	0.789	0.769	0.816	0.827	**0.828**

element.

2.2.5.3 Denoising Rician noise on MRI image

The proposed algorithm is also implemented and tested on various MRI images. Its performance is compared with three state-of-the-art denoising methods designed for Rician noise including the TV-based method[1] of Getreuer *et al.*[GTV11], the wavelet based method[2] of Pizurica *et al.* [PPLA03] namely VWNF, and the unbiased non-local means (UNLM) method[3] of Manjón *et al.* [MCCL+08]. Here, four examples of original MRI images are shown in Figure 2.8. For each test image, a training set is established by using two standard images of three same category. However, for the sake of clarity, Figure 2.9 only shows one of those standard images for each example. A test noisy image used in these experiments is obtained by adding Rician on an original high quality (noise-free) image. More precisely, two independent Gaussian noises with zero-mean and standard deviation σ (σ = 20 and 30 in these experiments) are added to the real and imaginary part of image data, respectively.

Table 2.3: PSNR Comparison of Rician Denoising methods on MRI

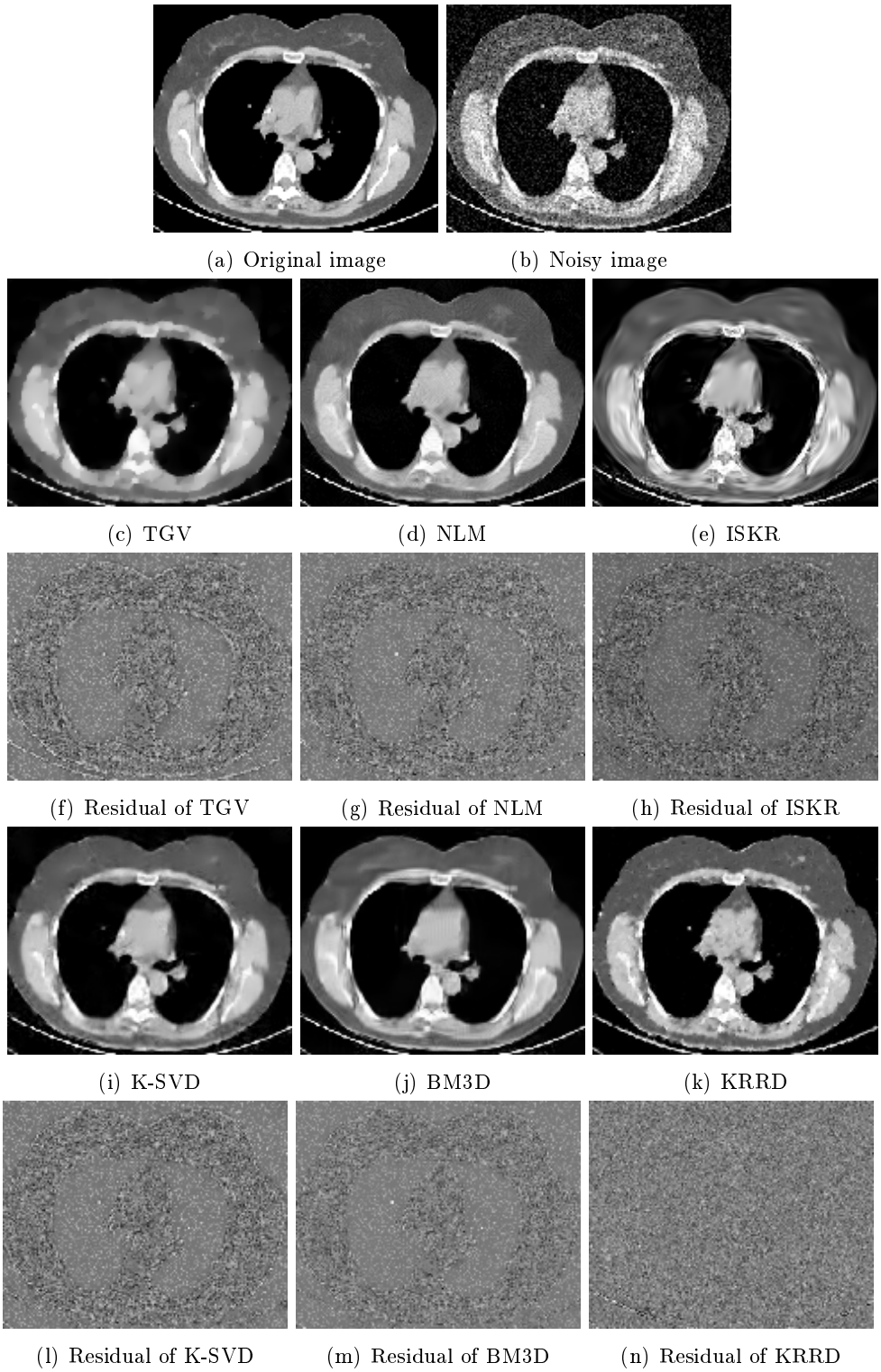
Test Image	σ	$\hat{\sigma}$	PSNR			
			Image denoising algorithms			
			WL	TV	UNLM	KRRD
(a)	20	20	19.14	19.71	21.99	**22.35**
	30	31	18.09	16.28	19.01	**20.82**
(b)	20	22	18.95	22.61	20.49	**23.26**
	30	29	17.71	16.65	19.30	**20.90**
(c)	20	20	18.64	20.54	18.32	**22.05**
	30	31	17.31	18.32	17.49	**18.15**
(d)	20	18	17.45	17.35	19.04	**19.73**
	30	29	16.43	14.13	16.22	**18.86**

The quality results are reported in Tables 2.3 and 2.4. The results of denoised images are reported in Figure 2.10, Figure 2.11 and Figure 2.12, for visual comparison. Figure 2.11 illustrates zoom-in images of a desired region of interest (DROI) in Figure 2.10. Residual image defined as the difference between a noisy image and the denoised image is also reported. As can be seen, residual image of our method contains nearly no structure or texture while other methods and especially

[1]Matlab program available at http://www.getreuer.info/home/tvreg

[2]Matlab program can be downloaded via http://telin.ugent.be/~sanja/Sanja_files/WaveletDEN.htm

[3]Matlab program available at http://personales.upv.es/jmanjon/



(a) Original image (b) Noisy image

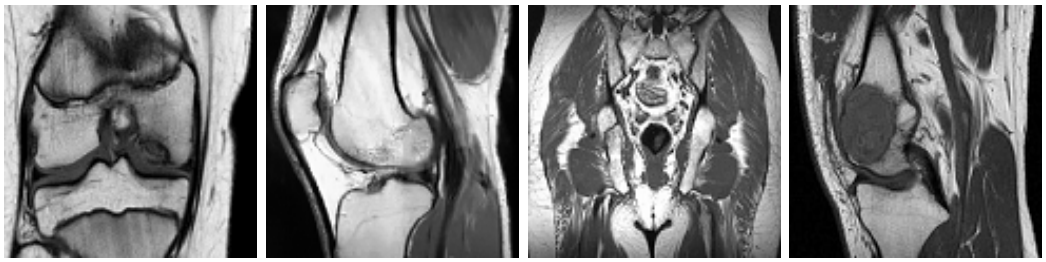
(c) TGV (d) NLM (e) ISKR

(f) Residual of TGV (g) Residual of NLM (h) Residual of ISKR

(i) K-SVD (j) BM3D (k) KRRD

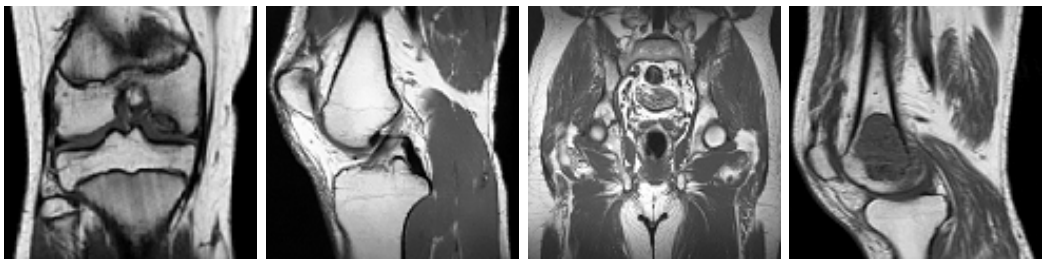
(l) Residual of K-SVD (m) Residual of BM3D (n) Residual of KRRD

Figure 2.7: Results of a CT image of thorax (Fig. 2.5(a)): (a) Original image, (b) Noisy image with $\sigma = 30$, (c) TGV [BKP10], (d) NLM [BCM06], (e) ISKR [TFM07]; (f), (g) and (h) are the residual image of the TGV and NLM and ISKR, respectively; (i) K-SVD [EA06], (j) BM3D [DFKE07], (k) KRRD, and (h), (m), (n) are the residual image of the K-SVD, BM3D and KRRD, respectively.



(a) (b) (c) (d)

Figure 2.8: Original MRI images used to generate the test noisy images (source: http://www.mr-tip.com/serv1.php).

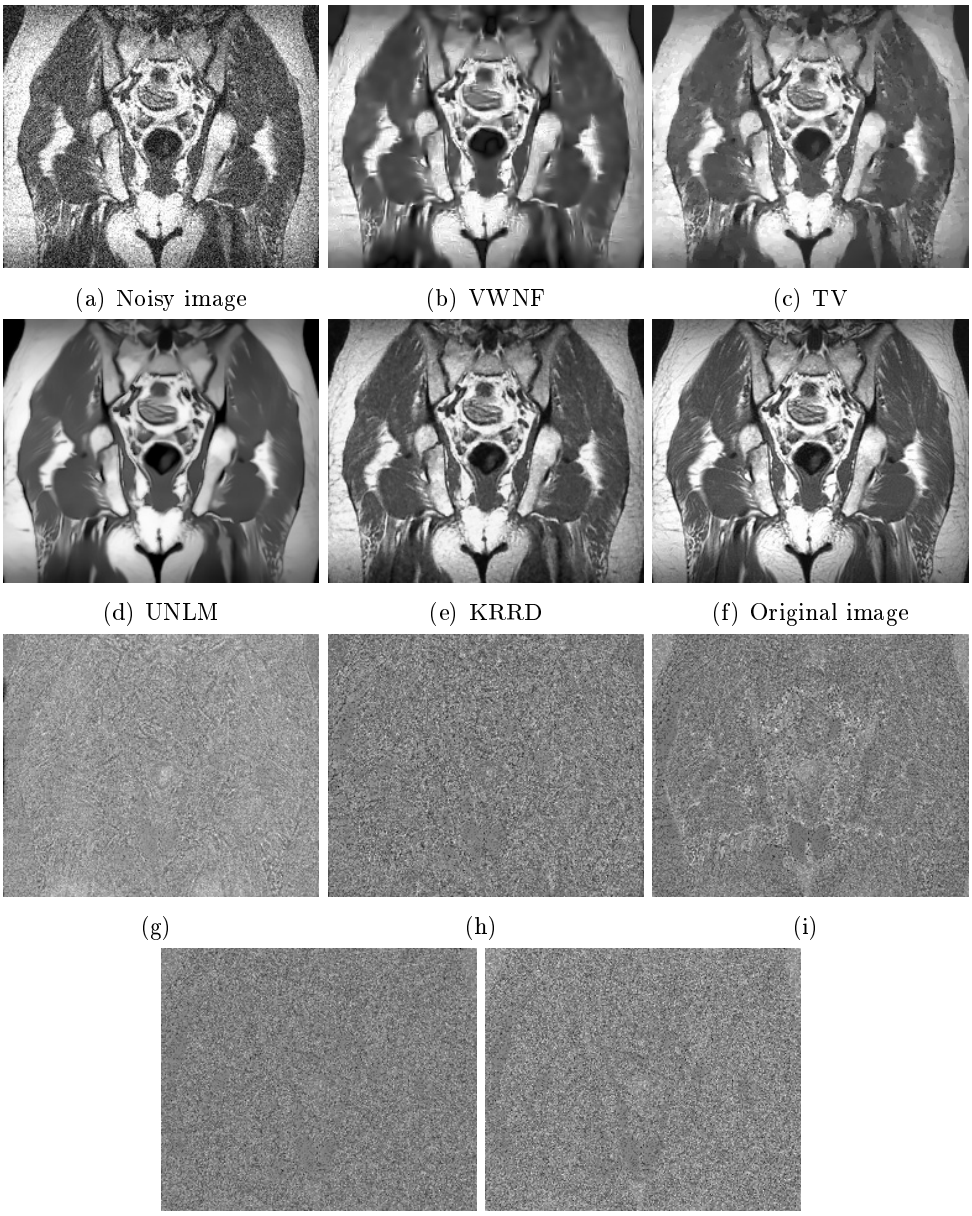


(a) (b) (c) (d)

Figure 2.9: Standard images used for training.

Table 2.4: SSIM Comparison of Rician Denoising methods on MRI

Test Image	σ	$\hat{\sigma}$	SSIM			
			Image denoising algorithms			
			WL	TV	UNLM	KRRD
(a)	20	20	0.8193	0.8333	0.8907	**0.9072**
	30	31	0.7700	0.7499	0.8356	**0.8756**
(b)	20	22	0.8443	0.9095	0.8991	**0.9265**
	30	29	0.7944	0.7967	0.8704	**0.8828**
(c)	20	20	0.8253	0.8653	0.8448	**0.9229**
	30	31	0.7748	0.8210	0.8059	**0.8406**
(d)	20	18	0.8171	0.8265	0.8855	**0.8906**
	30	29	0.7645	0.7535	0.8278	**0.8571**



(a) Noisy image (b) VWNF (c) TV

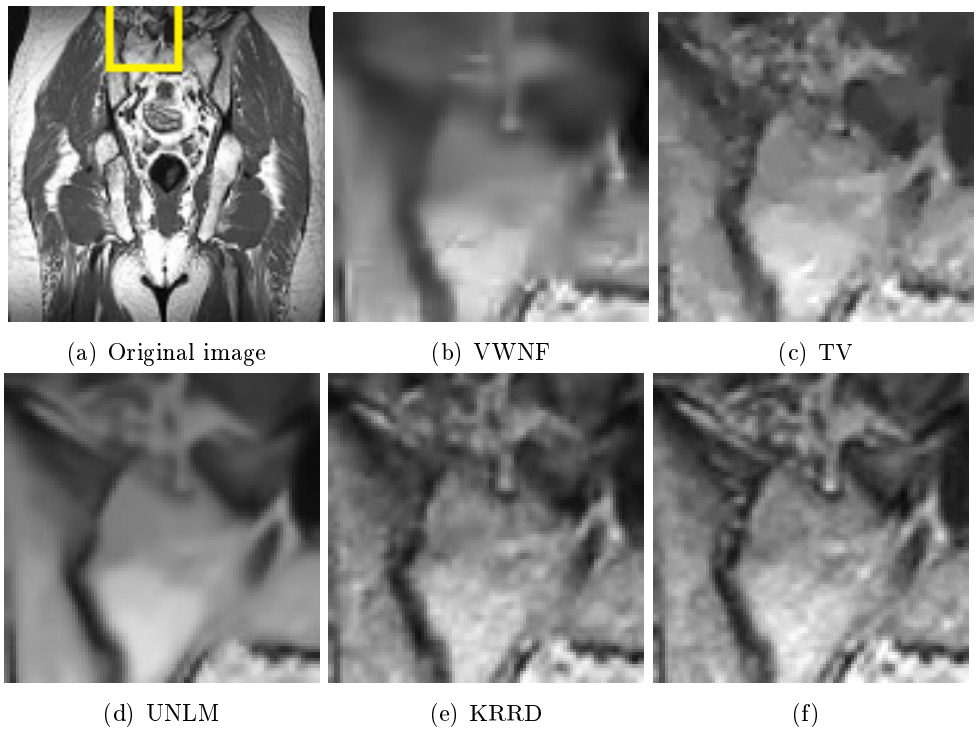
(d) UNLM (e) KRRD (f) Original image

(g) (h) (i)

(j) (k) Rician noise

Figure 2.10: Visual quality comparison of MRI image (Figure 2.8(c)) from various denoised images. (a) MRI image of the pelvis corrupted by Rician noise with standard deviation ($\sigma = 20$). From (b) to (e): denoised images by the VWNF [PPLA03], TV [GTV11], UNLM [MCCL+08] and KRRD methods, respectively. (f) is original image. From (g) to (k): residual images of (b), (c), (d) and (e), respectively. (k) Noise component in the noisy image, determined by subtracting the noisy image from the original image.

the UNLM contain many structure or edges that have been unfortunately removed by these denoising methods. Figure 2.10 clearly shows that the results of VWNF and UNLM are very blurred, while the TV-based method [GTV11] is less blurred and preserve well details and edges as to the KRRD. We can see that our proposed KRRD method effectively removes Rician noise while better preserving the subtle details and the textures. In terms of fidelity, as shown in Table 2.3 and Table 2.4, the proposed method yields significant PSNR and SSIM gap over the other methods. This confirms the visual observations.



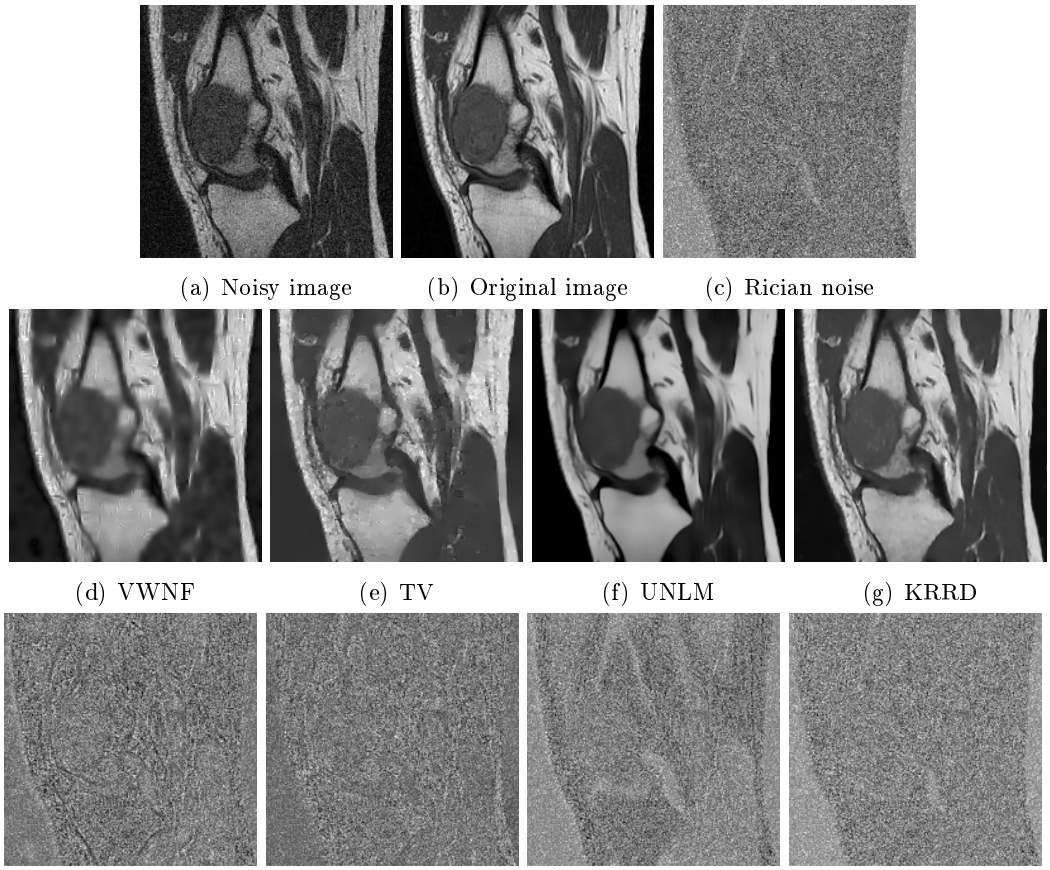
(a) Original image (b) VWNF (c) TV

(d) UNLM (e) KRRD (f)

Figure 2.11: Comparison of denoised (zoomed-in) MRI images by various methods in Figure 2.10: (a) Original image with DROI; (b)-(e) results of the VWNF [PPLA03], TV [GTV11], UNLM [MCCL+08] and KRRD methods, respectively; (f) zoomed in portion of the image (a).

2.2.6 Conclusion

We have proposed a novel denoising method namely KRRD for CT and MRI images using the kernel ridge regression. This is an example-based learning method, where denoising is performed by using the regression functions trained from a given set of standard images. Experimental results demonstrated the performances of the proposed method over some well known techniques in denoising in MRI images. For Gaussian noise, although the PSNR and SSIM results of the KRRD are inferior



(a) Noisy image (b) Original image (c) Rician noise

(d) VWNF (e) TV (f) UNLM (g) KRRD

(h) Residual of VWNF (i) Residual of TV (j) Residual of UNLM (k) Residual of KRRD

Figure 2.12: Comparison of various methods for MRI image of the knee (sagittal view). The first row shows the test noisy image ($\sigma = 20$), the original image, and the Rician noise component in the noisy image, respectively. The second row contains the results of VWNF [PPLA03], TV [GTV11], UNLM [MCCL$^+$08] and our KRRD method. The third row shows the residual images (difference between the noisy image and the denoised one) corresponding to the denoised images in the second row. The residual image illustrates the noise component that was removed by a denoising method.

than the BM3D or ever equivalent with the K-SVD, our visual results outperform those of these methods. Denoising machine model based on learning from a given database seems to be a promising solution for denoising of medical images. With an effective training set, we believe that this model may be quite useful. Although we achieved some initial success with the proposed solution, there are still some drawbacks which need to be improved such as:

- Noise simulation in the training phase: The training set is constructed based on simulating noise from the input noisy image. However, it is in fact difficult to simulate exactly the noise component in a given noisy image. Thus, the training set need to be established more effective.
- Parameters of the method: It is difficult to obtain the optimal value for the parameters of the method such as the regression parameter λ, the number K of the subset of the training set needs to be classified, the kernel parameter h. We have observed that in practice, for denoising medical images, optimal settings for parameters vary per application, and even per image and within images. This is due to the complexity and variability of shapes, structures and the quality of medical images.
- In the case where the input vector lies on the boundary of the clusters, the use of only one regression function as in (2.30) may not enough effective.
- Run-time: Although the training phase can be performed off-line, computational time of the algorithm is also expensive.

2.3 The second proposed method: k-nearest neighbor-based Denoising (kNND)

2.3.1 Introduction

In the previous section, we have presented an example-based learning solution for denoising medical images corrupted by Gaussian noise and Rician noise. This section introduces another example-based learning solution for the problem of denoising Gaussian noise and Poisson noise which often appear on medical imaging modalities using ionizing radiation. In the proposed method, denoising is performed pixel-wise with the help a training data set constructed from a given set of standard images. Unlike the first method where learning via regression function is performed off-line through the training phase, in this method learning by regression function and denoising for a given input are performed online using the k-nearest neighbors among the training examples.

Many effective denoising methods based on using the regression function of k-nearest neighbors have been proposed such as in [BCM06, CHKB09, vGM06, XM11]. In these approaches, the non-parametric regression techniques are often used. To determine the regression value $f(\mathbf{x})$ for an input $\mathbf{x}$, the non-parametric regression techniques use a subset $\{(\mathbf{x}_j, y_j)\}_{j=1}^k$ of a set of training examples $\{(\mathbf{x}_i, y_i), i \in \Omega\}$, in which $\{\mathbf{x}_j\}_{j=1}^k$ is the set of k nearest neighbors of $\mathbf{x}$. Nadaraya [Nad64] and Watson [Wat64] proposed to estimate $f(\mathbf{x})$ as a locally weighted average

$$f(\mathbf{x}) = \frac{\sum_{j=1}^k y_j \exp \frac{-\|\mathbf{x}-\mathbf{x}_j\|^2}{2h^2}}{\sum_{j=1}^k \exp \frac{-\|\mathbf{x}-\mathbf{x}_j\|^2}{2h^2}}. \tag{2.34}$$

The form of the Nadaraya-Watson estimator gives us an relationship with the non-local means (NLM) method [BCM06] presented in chapter 1 (see section 1.5.2). Indeed, from a noisy image $\mathbf{X} = \{\mathbf{X}(i),\ i \in \Omega\}$ we can generate the corresponding sample dataset $\{(\mathbf{x}_i, y_i), i \in \Omega\}$ as follows:

- $\mathbf{x}_i$ is the vectorized image patch centered at pixel i of $\mathbf{X}$.
- y_i is assigned by the value of pixel i in the noisy image $\mathbf{X}$, that is

$$y_i = \mathbf{X}(i). \tag{2.35}$$

Then, it is easy to see that the value $f(\mathbf{x}_i)$ of the Nadaraya-Watson regression function (2.34) exactly equals to $\hat{\mathbf{u}}(i)$ in (1.62) with the weights determined by (1.63) of the NLM method [BCM06]. The success of the NLM method proves the potential application of the k-nearest neighbor-based approach in image denoising.

According to the point of view of machine learning theory, a regression function can be considered as a machine whose task is to learn the mapping $\mathbf{x}_i \mapsto y_i$ from the training data set. Thus, the regression function highly depends on the reliability of the training data set and the regression model to be used. However, we can see that the training data set of the NLM method is directly established from the noisy image. Thus, it affects the reliability of the training data set. This is a motivation for us to propose another method where the training set is constructed more reliably.

In [XM11] Xu *et al.* proposed a method namely RNLM (Reference-based Non-Local Means) for denoising low-dose CT image. This method modified the Non-local means method [BCM06]. Unlike the NLM method where the set of training examples is established directly from the input noisy image $\mathbf{X}$ (low-dose image), the training set $\{(\mathbf{x}_i, y_i)\}$ in the RNLM method is constructed from a given set of artifacts-free reference (standard) CT images and their converted images which contain similar artifact types and distribution than the image $\mathbf{X}$. In the training set, $\mathbf{x}_i$ is an image patch in at position i in a converted image and y_i is the value of pixel i in the

corresponding standard image. It is shown that the RNLM method outperforms the NLM method. However, we can see that generating the converted images which contain similar artifact types and distribution than the input noisy image from the standard images is a difficult problem.

We can see that in the k-nearest neighbor-based methods, there are three main points that need to be considered:

1. How to establish a training set that is reliable and consistent with a specific problem?

2. What are the criteria for finding the k-nearest neighbors?

3. What is the suitable regression model to use?

In [vGM06], Ginneken and Mendrik proposed to use k-Nearest Neighbor Regression (kNNR) for denoising noise on low-dose CT images. In this method:

- The training set is constructed by using the standard CT images A_n (high dose images which have very little noise). For each high dose scan A_n, a corresponding low dose data B_n is simulated by adding physically realistic noise to the raw CT scanner data before reconstruction. Then, the noisy image B_n is filtered by different filters $\mathbf{F}_1, \mathbf{F}_2, \ldots, \mathbf{F}_d$ such as the standard noise reduction filters (Gausian filter, median filter, anisotropic diffussion filter, etc.). Denoted by $C_n^{(1)}, C_n^{(2)}, \ldots, C_n^{(d)}$ the filtering results,

$$C_n^{(t)} = \mathbf{F}_t(B_n), \forall t = 1, 2, \ldots, d. \tag{2.36}$$

  The training set $\{(\mathbf{x}_i, y_i)\}$ is established,

$$\mathbf{x}_i = \left(B_n(i), C_n^{(1)}(i), C_n^{(2)}(i), \ldots, C_j^{(d)}(i)\right) \quad \text{and} \quad y_i = A_n(i), \tag{2.37}$$

  where i is the pixel index i-th of the images $A_n, B_n, C_n^{(1)}, C_n^{(2)}, \ldots, C_n^{(d)}$.

- The Euclidean distance metric was used to find the k-nearest neighbors.

- Denoising is performed by using the k-NNR technique. More specifically, for a noisy image $\mathbf{X}$ to be denoised, the first step is to filter $\mathbf{X}$ by the filters $\mathbf{F}_1, \mathbf{F}_2, \ldots, \mathbf{F}_d$. Then, to denoise each pixel p of $\mathbf{X}$, compute its corresponding feature vector $\mathbf{x}_p = (\mathbf{X}(p), \mathbf{F}_1(\mathbf{X})(p), \mathbf{F}_2(\mathbf{X})(p), \ldots, \mathbf{F}_d(\mathbf{X})(p))$ and the set Ω_i of the k nearest neighbors of $\mathbf{x}_p$ are computed. Finally, the denoised pixel value is the average output y_i of these neighbors.

In fact, this method combines the outputs of different standard noise reduction algorithms using supervised technique. It was shown that the performance of standard noise reduction technique can be improved through the use of this technique. However, this technique requires prohibitive amounts of computation time and memory [vGM06].

In the subsection 2.3.2, we will present in details our example-based denoising method. This method can overcome some limitations of the previous example-learning-based methods such as the KRRD method, the methods in [LSM07, Li09, vGM06]. For instance, in constructing the training set we avoid simulating noise as in the previous methods. We also propose an effective criterion based on the statistical property of noise in the input noisy image for finding the k-nearest neighbors. The performance of the proposed method demonstrates that it is competitive with other well-known methods.

2.3.2 The proposed kNND method

The k-nearest neighbors-based denoising (kNND) is an example-based learning method which is performed pixel-wise, always with the help of a set of standard images. Suppose that $\mathbf{X}$ is a noisy image that needs to be denoised, $\mathbf{Y}$ is the ground-truth of $\mathbf{X}$, and there exists a set of standard images taken at nearly the same location (which can be considered as high quality or noise-free images) as the noisy image $\mathbf{X}$. By considering an image as an arranged set of pixels, the images $\mathbf{X}$ and $\mathbf{Y}$ can be respectively presented as $\{x_p\}_{p=1}^N$ and $\{y_p\}_{p=1}^N$, where N is the number of pixels in the images, x_p is the value of the pixel p in $\mathbf{X}$ and y_p is the value of the corresponding pixel p in $\mathbf{Y}$. The purpose of the denoising problem is to estimate y_p from x_p for all $p = 1, 2, \ldots, N$ with the help of the standard images.

The proposed method is performed in two phases: *the generation of the training set* and *the denoising phase*.

- In the first phase, instead of using examples from simulated noises as proposed in the first method (KRRD), we only use examples from standard images to generate a training set Ω. The training set include the couples $(\mathbf{u}_i, y_i) \in \mathbb{R}^n \times \mathbb{R}$, where $\mathbf{u}_i$ is the vector corresponding to an image patch of size $\sqrt{n} \times \sqrt{n}$ (n is an odd square number) in the standard image and y_i is the value of the center pixel of the vectorized patch $\mathbf{u}_i$. Here, the image vectorized patches $\mathbf{u}_i$ in the training set are randomly extracted from the standard images. So, we do not have to perform simulating noise in constructing the training set Ω.

- In the second phase, for each pixel to denoise, we first search for its nearest neighbors in the standards patches of the training set, and then we use these

patches to learn relationships between them and their associated true values in the training set. The main idea of the second phase can be summarized as follows: for denoising each pixel p in the input noisy image $\mathbf{X}$

1. A patch of size $\sqrt{n}\times\sqrt{n}$ and centered at p is first extracted and vectorized. This vector is denoted by $\mathbf{x}_p \in \mathbb{R}^n$.
2. Find a subset Ω_p of k nearest neighbors of the input $\mathbf{x}_p$ from the training set.
3. Determine a regression function f_{Ω_p} on Ω_p.
4. Perform denoising for the pixel p by using the trained regression function f_{Ω_p}.

More details on the steps are presented in the following subsections.

2.3.2.1 Nearest Neighbors Search

In this work, the k nearest neighbors of $\mathbf{x}_p$ is found in two steps:

1. The first step: we use a fast algorithm namely k-NN (k-Nearest Neighbors) which is based on kd-trees structure [FBF77] and priority search [AMN+94] to find the set of K elements which have the shortest Euclidean distance to $\mathbf{x}_p$. Let us denote this set by $\overline{\Omega}_p$. The C++ code source of this algorithm is available in [MA06]. A Matlab wrapper provided by Dahua Lin can be downloaded via this link `http://people.csail.mit.edu/dhlin/softwares.html`.

2. The second step: from the set $\overline{\Omega}_p$ we select a subset of k ($k < K$) elements by using a criterion based on the statistical properties of the noise in the noisy patch $\mathbf{x}_p$.

Here, we present the criteria for searching the k nearest neighbors for two cases depending on whether: 1-the noise in patch $\mathbf{x}_p$ is an additive white Gaussian noise; 2-the noise in patch $\mathbf{x}_p$ is a Poisson noise.

- Case of Gaussian noise:
  Suppose that

$$\mathbf{x}_p = \mathbf{y}_p + \eta_p, \tag{2.38}$$

  where $\mathbf{y}_p$ is the noise-free patch, η_p is the noise component assumed to be additive white Gaussian noise, $\eta_p \sim \mathcal{N}(0, \sigma_p^2)$. Here, an element $\mathbf{u}_i$ in the set

$\overline{\Omega}_p$ is called the nearest neighbor of $\mathbf{x}_p$ if it is as similar as possible to $\mathbf{y}_p$. To determine the k nearest neighbors, let us consider equation (2.38). We have,

$$\mathbf{x}_p - \mathbf{y}_p = \eta_p \sim \mathcal{N}(0, \sigma_p^2). \tag{2.39}$$

Thus, $\mathbf{u}_i \approx \mathbf{y}_p$ if

$$\mathbf{x}_p - \mathbf{u}_i \sim \mathcal{N}(0, \sigma_p^2). \tag{2.40}$$

It can be inferred that,

$$\begin{cases} \mathrm{E}(\mathbf{x}_p - \mathbf{u}_i) & \simeq 0 \\ \mathrm{Var}(\mathbf{x}_p - \mathbf{u}_i) - \sigma_p^2 & \simeq 0. \end{cases}$$

Therefore,

$$\left|\mathrm{E}(\mathbf{x}_p - \mathbf{u}_i)\right| + \left|\mathrm{Var}(\mathbf{x}_p - \mathbf{u}_i) - \sigma_p^2\right| \simeq 0. \tag{2.41}$$

So, determining the subset of the k nearest neighbors of $\mathbf{x}_p$ from the set $\overline{\Omega}_p$ is performed as follows: first, for each of element $\mathbf{u}_i$ in $\overline{\Omega}_p$ $(i = 1, 2, \ldots, K)$, we define a 2-dimension vector ϑ_i such that

$$\vartheta_i = \left(\mathrm{E}(\mathbf{x}_p - \mathbf{u}_i), \mathrm{Var}(\mathbf{x}_p - \mathbf{u}_i) - \sigma_p^2\right). \tag{2.42}$$

Let us refer to ϑ_i as the characteristic vector of $\mathbf{u}_i$ with respect to $\mathbf{x}_p$. The characteristic vector ϑ_i allows us to evaluate the statistical property of noise in the residual patch. Finally, we select from $\overline{\Omega}_p$ k elements $\mathbf{u}_i$ that have the smallest $\|\vartheta_i\|_1$. These k elements determine the set of k nearest neighbors $\Omega_p = \{\mathbf{u}_i, y_i\}_{i=1}^k$.

- Case of Poisson noise:
  Suppose that noise in the patch $\mathbf{x}_p$ has the Poisson distribution. In this case, the k nearest neighbors search is realized in two steps: first, we use the Anscombe root transformation [Ans48] (see section 1.3.2) to convert approximately Poisson noise into standard Gaussian noise. Then, we apply the criterion for Gaussian noise as in (2.41) to determine the set of the k nearest neighbors of $\mathbf{x}_p$.

  In the first step, the Anscombe transformation, $\mathcal{T}(\mathbf{z}) = 2\sqrt{\mathbf{z} + \frac{3}{8}}$, is applied on $\mathbf{x}_p$ and on the patches $\mathbf{u}_i$ $(i = 1, 2, \ldots, K)$ of the set $\overline{\Omega}_p$. Then, noise in the transformed patch denoted by $\mathbf{x}_p^{\mathcal{T}}$,

  $$\mathbf{x}_p^{\mathcal{T}} = \mathcal{T}(\mathbf{x}_p) \tag{2.43}$$

  is asymptotically additive standard normal, $\mathcal{N}(0, 1)$. Denote by $\overline{\Omega}_p^{\mathcal{T}}$ the set of the transformed example patches, we have

  $$\overline{\Omega}_p^{\mathcal{T}} = \mathcal{T}(\overline{\Omega}_p) = \left\{\mathbf{u}_i^{\mathcal{T}} = \mathcal{T}(\mathbf{u}_i)\right\}_{i=1}^K. \tag{2.44}$$

In the second step, we define for each $\mathbf{u}_i^{\mathcal{T}}$ in $\overline{\Omega}_p^{\mathcal{T}}$ a characteristic vector ϑ_i as in equation (2.42),

$$\vartheta_i = \left(\mathrm{E}(\mathbf{x}_p^{\mathcal{T}} - \mathbf{u}_i^{\mathcal{T}}), \mathrm{Var}(\mathbf{x}_p^{\mathcal{T}} - \mathbf{u}_i^{\mathcal{T}}) - 1\right). \tag{2.45}$$

Finally, $\Omega_p = \{\mathbf{u}_i, y_i\}_{i=1}^k$ is determined in which $\{\mathbf{u}_i\}_{i=1}^k$ is a subset of $\overline{\Omega}_p$ such that the corresponding characteristic vectors, ϑ_i, have the smallest ℓ_1-norm.

In the next subsection, we will present the regression model learned from the Ω_p.

2.3.2.2 Regression model trained with the set of the k nearest neighbors

Suppose that for a given input $\mathbf{x}_p$ we obtain a corresponding training subset $\Omega_p = \{\mathbf{u}_i, y_i\}_{i=1}^k$. Now, the problem is to determine a regression function f_{Ω_p} on Ω_p. From the KRRD method (see section 2.2.2.2), we can extract some the following observations.

- As shown in (2.32), the KRR has the form:

$$f(\mathbf{x}) = \sum_i a_i \mathcal{K}(\mathbf{u}_i, \mathbf{x}), \tag{2.46}$$

  where $\mathcal{K}(\mathbf{u}_i, \mathbf{x})$ is the kernel function, and a_i are the regression coefficients determined according to the corresponding regression model. Note that, there are no any constraints on these coefficients.

- Normally, the regression function $f(\mathbf{x})$ is used to estimate noise-free pixel value. Thus, $f(\mathbf{x})$ is often a nonnegative value. On the other hand, in the KRRD method, we used the Gaussian kernel

$$\mathcal{K}(\mathbf{u}_i, \mathbf{x}) = \exp -\frac{\|\mathbf{u}_i - \mathbf{x}\|_2^2}{2h^2} \tag{2.47}$$

  where h is the decay parameter. Note that with Gaussian kernel, we have $\mathcal{K}(\mathbf{u}_i, \mathbf{x}) > 0, \forall \mathbf{u}_i, \mathbf{x}$. Thus, we can assume that $a_i \geq 0, \forall i = 1, 2, \ldots, k$.

Therefore, we propose to find the regression function f_{Ω_p} in the form

$$f_{\Omega_p}(\mathbf{x}) = \sum_i a_i \mathcal{K}(\mathbf{u}_i, \mathbf{x}), \tag{2.48}$$

where the vector of the regression coefficients, $\mathbf{a} = (a_1, a_2, \ldots, a_k)^T$, is a nonnegative sparse vector which determines the solution of the following optimization problem:

$$\mathbf{a} = \arg\min_{\mathbf{a} \geq 0} \frac{1}{2} \sum_{i=1}^k \left(f(\mathbf{u}_i) - y_i\right)^2 + \lambda \|\mathbf{a}\|_1. \tag{2.49}$$

By setting $\mathbf{U} = (U_{ij})_{k\times k}$ as the $k \times k$ matrix, where $U_{ij} = \mathcal{K}(\mathbf{u}_i, \mathbf{u}_j)$. Problem (2.49) can be rewritten as

$$\mathbf{a} = \arg\min_{\mathbf{a}\geq 0} \frac{1}{2} \|\mathbf{U}\mathbf{a} - \mathbf{v}\|_2^2 + \lambda\|\mathbf{a}\|_1. \tag{2.50}$$

$$\Leftrightarrow \mathbf{a} = \arg\min_{\mathbf{a}\geq 0} \frac{1}{2}\mathbf{a}^T\mathbf{A}\mathbf{a} + \mathbf{b}^T\mathbf{a}, \tag{2.51}$$

where, $\mathbf{v} = (y_1, y_2, \ldots, y_k)^T$, $\mathbf{b} = (\lambda\mathbf{1} - \mathbf{U}^T\mathbf{v})$ in which $\mathbf{1}$ denotes an appropriately sized vector of ones, and $\mathbf{A} = \mathbf{U}^T\mathbf{U}$. In this case, $\mathbf{A}$ is a positive definite matrix.

As it can be seen, (2.51) is a Nonnegative Quadratic Programming (NPQ) which can be effectively solved by many algorithms (see [PPM+09]). To solve this problem, we use in this work the multiplicative updates algorithm proposed by Sha *et al.* in [SSL02]. This is a quite effective algorithm and has been used to solve interesting problems such as support vector machines [SSL02]. The updates from [SSL02] for the NQP are summarized as follows (see [SSL02] for details on derivation and convergence):

Algorithm 1 Multiplicative Updates Algorithm for NQP [SSL02]

Input: $\mathbf{a} = \mathbf{a}_0 > 0$

Updating:

$$\mathbf{a} \longleftarrow \mathbf{a}\left[\frac{-\mathbf{b} + \sqrt{\mathbf{b}.^*\mathbf{b} + 4(\mathbf{A}^+\mathbf{a}).^*(\mathbf{A}^-\mathbf{a})}}{2\mathbf{A}^+\mathbf{a}}\right]. \tag{2.52}$$

Output: $\mathbf{a}^* = \mathbf{a}$

In the algorithm, $\mathbf{A}^+$ and $\mathbf{A}^-$ are the nonnegative matrices defined by:

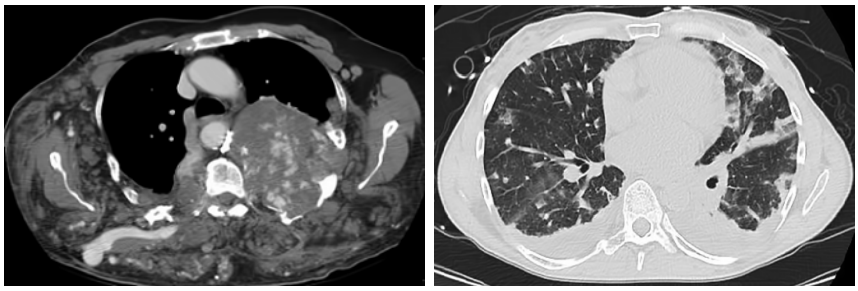
$$A_{ij}^+ = \begin{cases} A_{ij} & \text{if } A_{ij} > 0 \\ 0 & \text{otherwise,} \end{cases} \quad \textit{and} \quad A_{ij}^- = \begin{cases} |A_{ij}| & \text{if } A_{ij} < 0 \\ 0 & \text{otherwise.} \end{cases} \tag{2.53}$$

In (2.52), the operator $.^*$ represents element-wise multiplication. Division and square root are also element-wise.

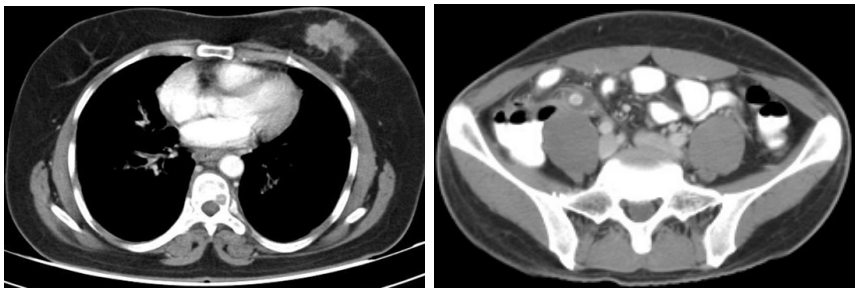
Suppose that we obtained the regression function f_{Ω_p}. We can see that the function f_{Ω_p} expresses the relationship between $\mathbf{u}_i$ and y_i in Ω_p, in which $\mathbf{u}_i$ is a vector of a standard patch. Thus, to obtain an effective estimate $f_{\Omega_p}(\mathbf{x})$ for an input $\mathbf{x}$, $\mathbf{x}$ should be pre-processed. Therefore, we propose to perform denoising of the pixel p in two following steps. The first step is to find a coarse estimate of the input $\mathbf{x}_p$ by averaging the k nearest neighbors:

$$\bar{\mathbf{x}}_p = \frac{1}{k}\sum_{i=1}^{k} \mathbf{u}_i. \tag{2.54}$$

Then, the true value of the pixel p is estimated by, $\mathbf{Y}(p) = f_{\Omega_p}(\bar{\mathbf{x}}_p)$.

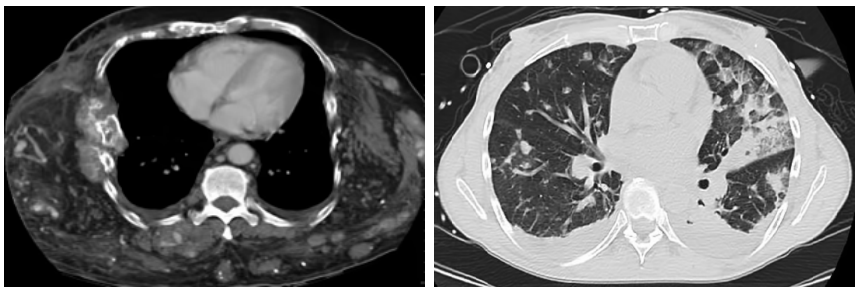


(a) CT scan of chest (b) CT scan of lung

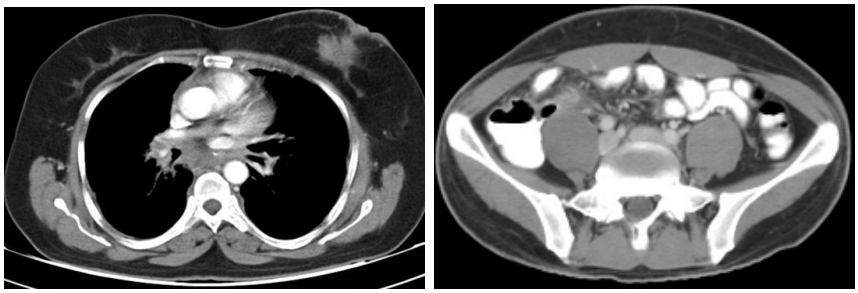


(c) CT scan of thorax (d) CT scan of pelvis

Figure 2.13: Original images: (a) CT scan of chest, (b) CT scan of lung, (c) CT scan of thorax, (d) CT scan of pelvis. (source:http://radiopaedia.org/cases/



(a) (b)



(c) (d)

Figure 2.14: Standard images used to establish the training set. (source: http://radiopaedia.org/cases/)

2.3.3 Performance evaluation of the kNND method

In this subsection, we present two experiments for denoising Gaussian noise and Poisson noise.

2.3.3.1 Performance evaluation with Gaussian noise

The experimental tests are performed using four original CT images: CT scans of chest, of lung, of thorax, and of pelvis (see Figure 2.13). These images contain different types of features: texture, sharp edges and smooth regions. They are used to create the noisy images by adding a white Gaussian noise with different standard deviation values: $\sigma = 10, 20$ and 30. The proposed method (kNND) is compared with the methods: Total Generalized Variation (TGV) [BKP10], Non-local means (NLM) [BCM05], ISKR [TFM07], K-SVD [EA06], BM3D [DFKE07], and RNLM [XM11]. For the example-based methods RNLM and kNND, we use another image taken at nearly the same location to the test image for establishing the training data set. These standard images are shown in Figure 2.14. In our experiments, the size of the training data set is of 20000 observations, the regression parameter λ in (2.49) is set to 1, and the parameter K in subsection 2.3.2.1 is set to $5k$. We will discus on value of parameter k in the experiments.

To objectively evaluate the results, we use two fidelity measures, PSNR and SSIM [WBSS04], as described in section 1.4. The best results of eight methods are reported in tables from Table 2.5 to Table 2.8. As can be seen, our kNND method yields significant PSNR and SSIM gains over the TGV method [BKP10], the NLM method [BCM06], the ISKR method [TFM07] and the RNLM method [XM11]. Compared to the K-SVD, the quality indices of the kNND are higher in the cases of $\sigma = 20$ and 30. However, for some cases of slight noise (CT scan of lung), the SSIM index of the K-SVD method are sometimes higher than those of the proposed method. It can also be seen that the BM3D method has higher PSNR and SSIM compared with the kNND method in the case of low noise (level $\sigma = 10$) except for the PSNR in CT scan of lung (Table 2.6). In contrast, the quality indices of the kNND method are nearly the best for noise levels $\sigma = 20$ and $\sigma = 30$. In general, the proposed method obtains better quality indices in case of heavy noise.

For subjective evaluation, we show in Figures 2.15, 2.16 and 2.17 the denoising results of three images: Thorax, Chest and Pelvis respectively, with three noise levels: $\sigma = 10$, 20 and 30. As it can be seen in Figure 2.15, in the case of slight noise ($\sigma = 10$), the performance of the proposed method is competitive with the methods NLM, RNLM, K-SVD and BM3D. For the cases of strong noise, Figure 2.16 and

Table 2.5: Objective comparison for the CT scan of chest (Figure 2.13(a))

Quality	σ	Image denoising algorithms						
		TGV [BKP10]	NLM [BCM06]	ISKR [TFM07]	K-SVD [EA06]	BM3D [DFKE07][XM11]	RNLM	kNND
PSNR	10	32.37	33.16	34.02	34.42	**34.83**	32.84	34.57
	20	29.22	29.83	29.92	30.27	30.96	29.89	**31.15**
	30	27.21	27.41	27.56	27.97	28.87	27.83	**29.08**
SSIM	10	0.895	0.920	0.922	0.930	**0.946**	0.907	0.932
	20	0.816	0.822	0.853	0.857	0.854	0.841	**0.874**
	30	0.779	0.769	0.791	0.776	0.819	0.752	**0.820**

Table 2.6: Objective comparison for the CT scan of lung (Figure 2.13(b))

Quality	σ	Image denoising algorithms						
		TGV [BKP10]	NLM [BCM06]	ISKR [TFM07]	K-SVD [EA06]	BM3D [DFKE07][XM11]	RNLM	kNND
PSNR	10	29.23	31.18	32.06	32.66	33.05	31.24	**33.38**
	20	27.04	28.28	28.02	28.86	29.19	27.63	**29.43**
	30	25.30	26.24	25.67	26.61	27.05	26.62	**27.88**
SSIM	10	0.855	0.907	0.910	0.918	**0.926**	0.878	0.915
	20	0.806	0.818	0.839	**0.876**	0.861	0.820	0.870
	30	0.749	0.734	0.777	0.792	0.808	0.795	**0.812**

Table 2.7: Objective comparison for the CT scan of thorax (Figure 2.13(c))

Quality	σ	Image denoising algorithms						
		TGV [BKP10]	NLM [BCM06]	ISKR [TFM07]	K-SVD [EA06]	BM3D [DFKE07][XM11]	RNLM	kNND
PSNR	10	31.51	33.53	34.24	34.42	**35.29**	33.61	34.50
	20	28.66	30.41	30.20	30.79	31.64	29.94	**32.10**
	30	26.42	27.94	27.78	28.38	29.40	28.14	**30.22**
SSIM	10	0.907	0.921	0.902	0.933	**0.949**	0.919	0.924
	20	0.856	0.849	0.864	0.882	0.903	0.852	**0.917**
	30	0.789	0.767	0.815	0.826	0.856	0.843	**0.876**

Table 2.8: Objective comparison for the CT scan of pelvis (Figure 2.13(d))

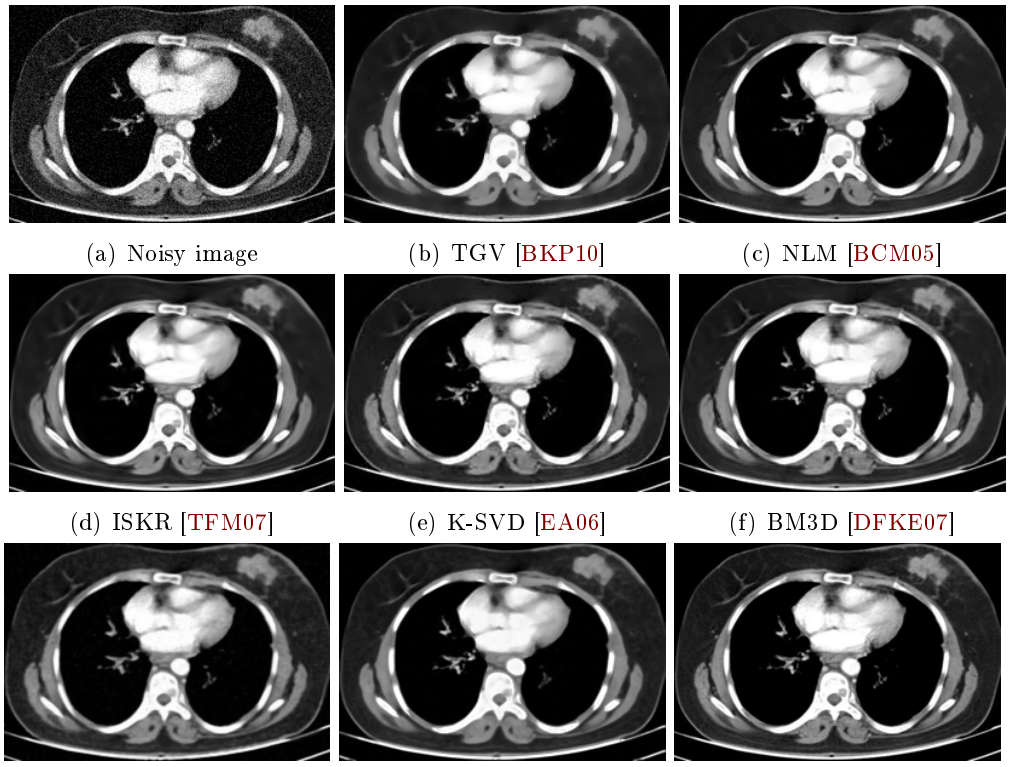
Quality	σ	Image denoising algorithms						
		TGV [BKP10]	NLM [BCM06]	ISKR [TFM07]	K-SVD [EA06]	BM3D [DFKE07][XM11]	RNLM	kNND
PSNR	10	31.38	33.39	33.79	33.94	**34.45**	30.96	34.31
	20	28.68	30.16	29.96	30.06	30.84	28.57	**31.20**
	30	26.64	27.75	27.52	28.10	28.74	27.39	**29.00**
SSIM	10	0.915	0.935	0.933	0.940	**0.945**	0.928	0.937
	20	0.872	0.866	0.888	0.895	**0.904**	0.892	**0.904**
	30	0.824	0.788	0.847	0.851	0.870	0.874	**0.880**

Figure 2.17 show that the kNND method visually outperforms the other methods. In particular, observation of zoom-in parts of a desired region of interest in Figure 2.16 shows that the image details in the denoised image by our kNND method are preserved better than the other methods. For the TGV method and the ISKR method, many details and edges have been smoothed out, especially in texture regions and along the edges.

In the next subsections, we will analyze the influences of the parameters in this method including:

- The decay parameter h of the kernel function
- The patch size
- The number of the nearest neighbors, k.

In order to objectively evaluate the impact of these parameters, we report results



(a) Noisy image (b) TGV [BKP10] (c) NLM [BCM05]

(d) ISKR [TFM07] (e) K-SVD [EA06] (f) BM3D [DFKE07]

(g) RNLM [XM11] (h) kNND (i) Original image

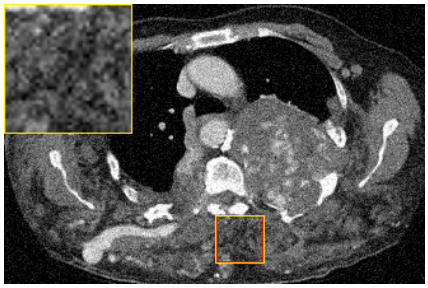
Figure 2.15: Results of CT image of thorax for $\sigma = 10$.

of two quality measures namely PSNR and SSIM. Recall that high value of PSNR, SSIM indicates high image quality.

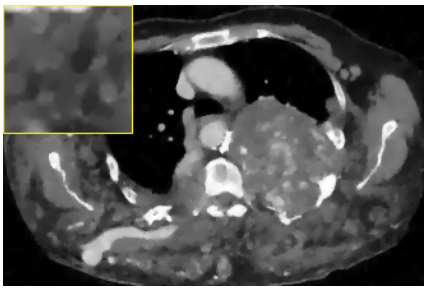
2.3.3.1.1 Effects of patch size and decay parameter of the kernel function

In order to validate the effects of the patch size and the decay parameter for denoising, we conduct some experiments on two CT images, chest image (Figure 2.13(a)) and pelvis image (Figure 2.13(d)), with different patch sizes of 3×3, 5×5, 7×7, 9×9, and 11×11. The decay parameter is often dependent on standard deviation level σ of the noise. Thus, we consider h in the form $h = n_h\sigma$. Here, we consider three noise levels $\sigma = 10$, 20 and 30, the decay parameter is tested with several values $n_h = [2 : 2 : 100]$ (here we use Matlab notation which means that n_h takes the values from 2 to 100 with step 2). The experimental results are shown in Figure 2.18 and Figure 2.19.

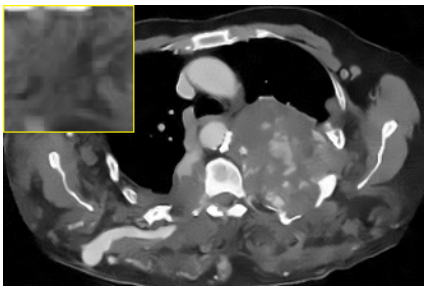
As it can be seen, the general and increasing form of these curves indicates that good results can be obtained for sufficiently large value of n_h for any patch size. Let us denote by n_{ht} a threshold value of nh for which the quality result is optimum.



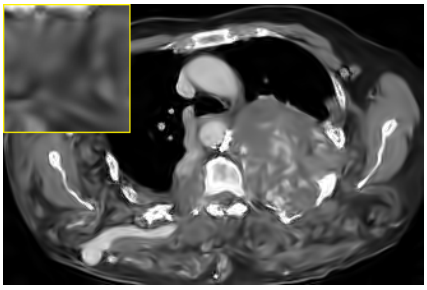
(a) Noisy image



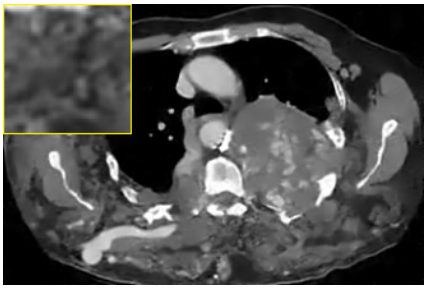
(b) TGV [BKP10]



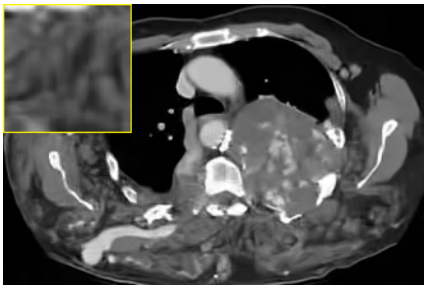
(c) NLM [BCM05]



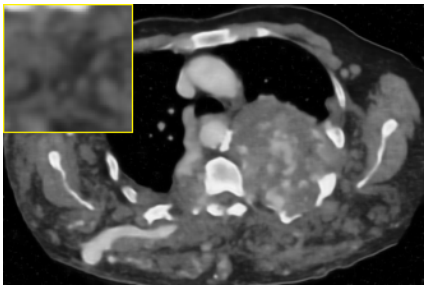
(d) ISKR [TFM07]



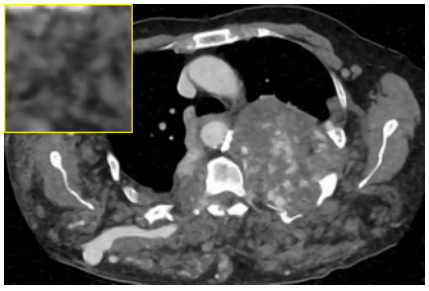
(e) K-SVD [EA06]



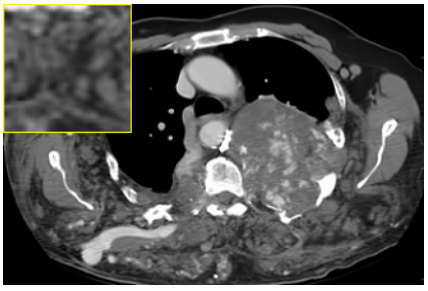
(f) BM3D [DFKE07]



(g) RNLM [XM11]



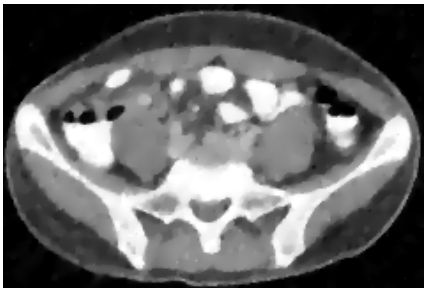
(h) kNND



(i) Original image

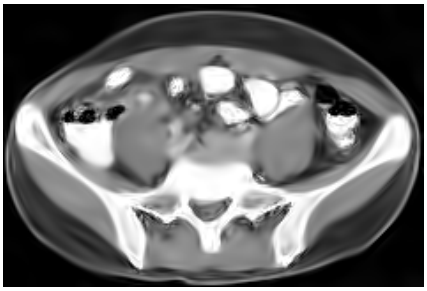
Figure 2.16: Results of CT image of chest for $\sigma = 20$.

(a) Noisy image

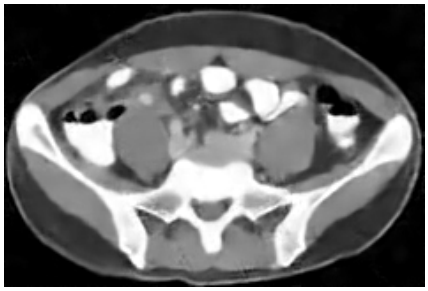


(b) TGV [BKP10]

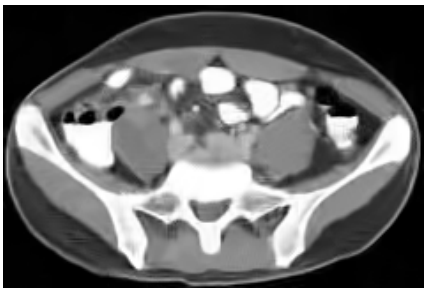
(c) NLM [BCM05]



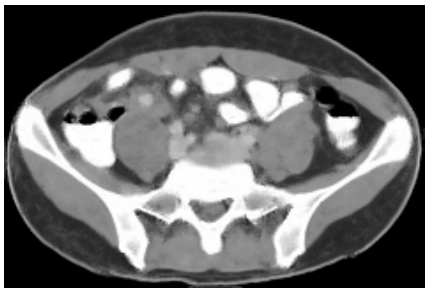
(d) ISKR [TFM07]



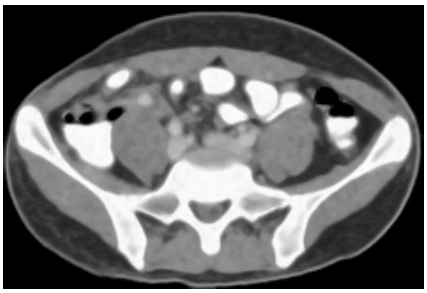
(e) K-SVD [EA06]



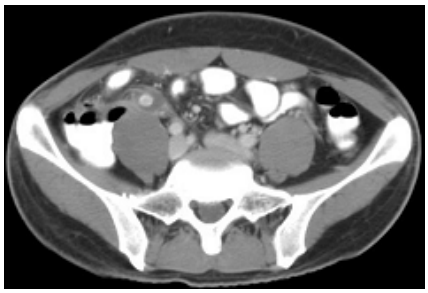
(f) BM3D [DFKE07]



(g) RNLM [XM11]

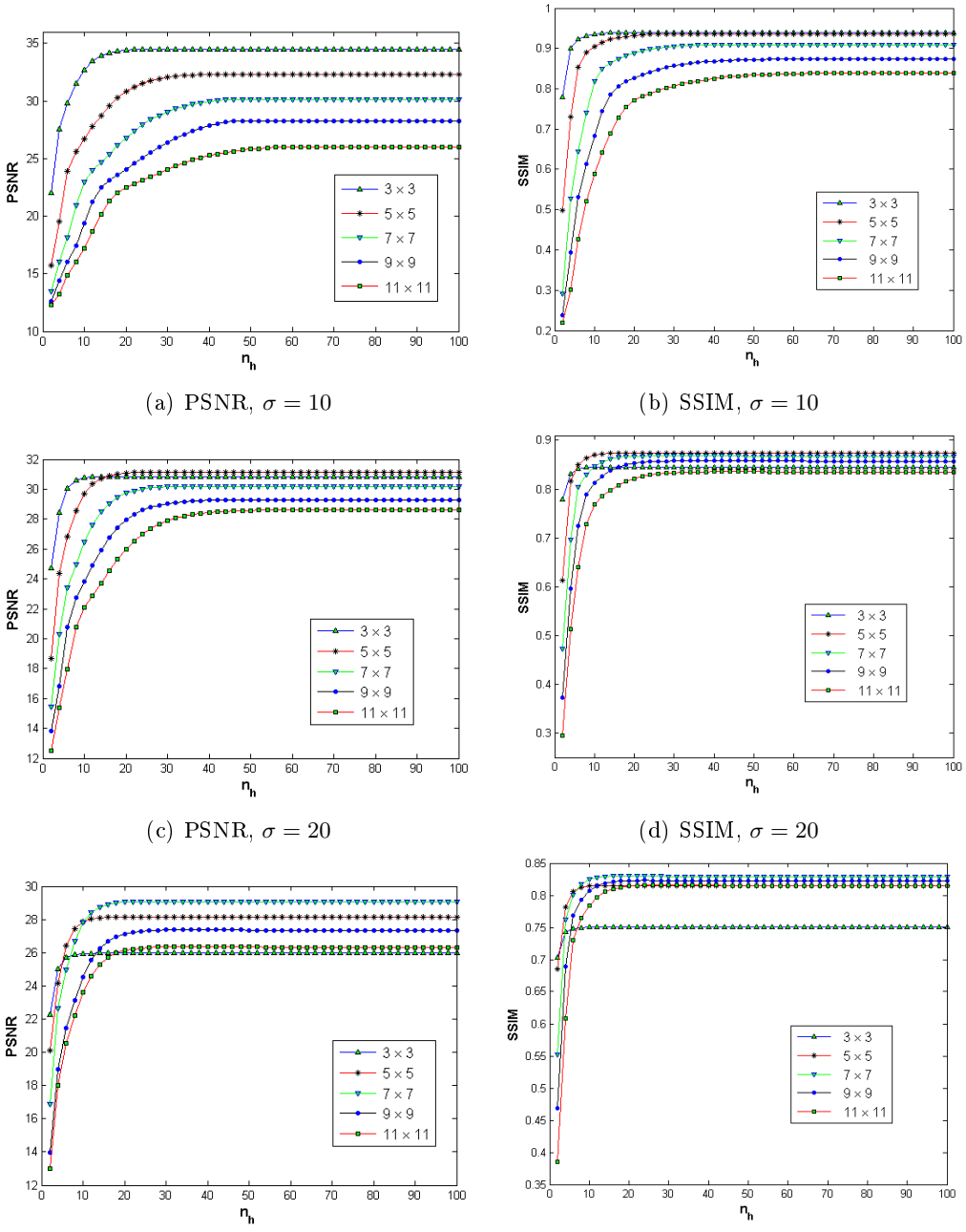


(h) kNND



(i) Original image

Figure 2.17: Results of CT image of pelvis for $\sigma = 30$.

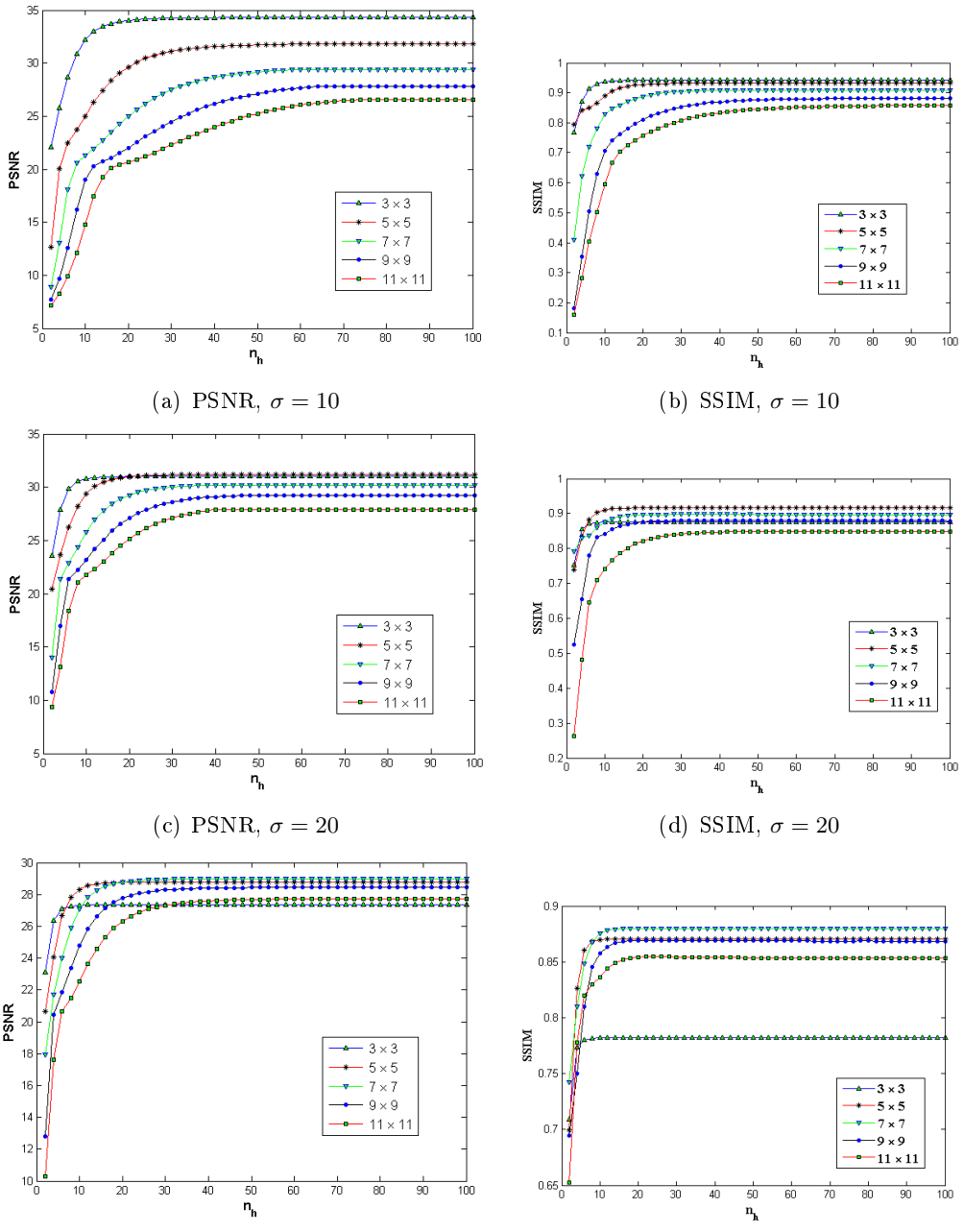


(a) PSNR, $\sigma = 10$ (b) SSIM, $\sigma = 10$

(c) PSNR, $\sigma = 20$ (d) SSIM, $\sigma = 20$

(e) PSNR, $\sigma = 30$ (f) SSIM, $\sigma = 30$

Figure 2.18: Objective image quality measures with respect to different patch sizes and different values of the decay parameter, $h = n_h\sigma$. PSNR and SSIM curves as the functions of parameter n_h. Experiment is performed on the CT image of chest in Figure 2.13(a).



(a) PSNR, $\sigma = 10$ (b) SSIM, $\sigma = 10$

(c) PSNR, $\sigma = 20$ (d) SSIM, $\sigma = 20$

(e) PSNR, $\sigma = 30$ (f) SSIM, $\sigma = 30$

Figure 2.19: Objective image quality measures with respect to different patch sizes and different values of the decay parameter, $h = n_h\sigma$. PSNR and SSIM curves as the functions of parameter n_h. Experiment is performed on the CT image of pelvis in Figure 2.13(d).

As can be seen, the threshold value n_{ht} varies with patch size. For example, a small patch size 3×3 should have a threshold value n_{ht} around 15 for small noise ($\sigma = 10$) while this value decreases for stronger noise levels. Regarding patch size, we observe that the stronger the noise, the larger patch size should be chosen to obtain good quality measures. For example, the quality measures are the best for 3×3 patch size with $\sigma = 10$. Likely, for 5×5 patch size, the best quality indices are obtained with $\sigma = 20$. As for the 7×7 patch size, the best quality results correspond to $\sigma = 30$.

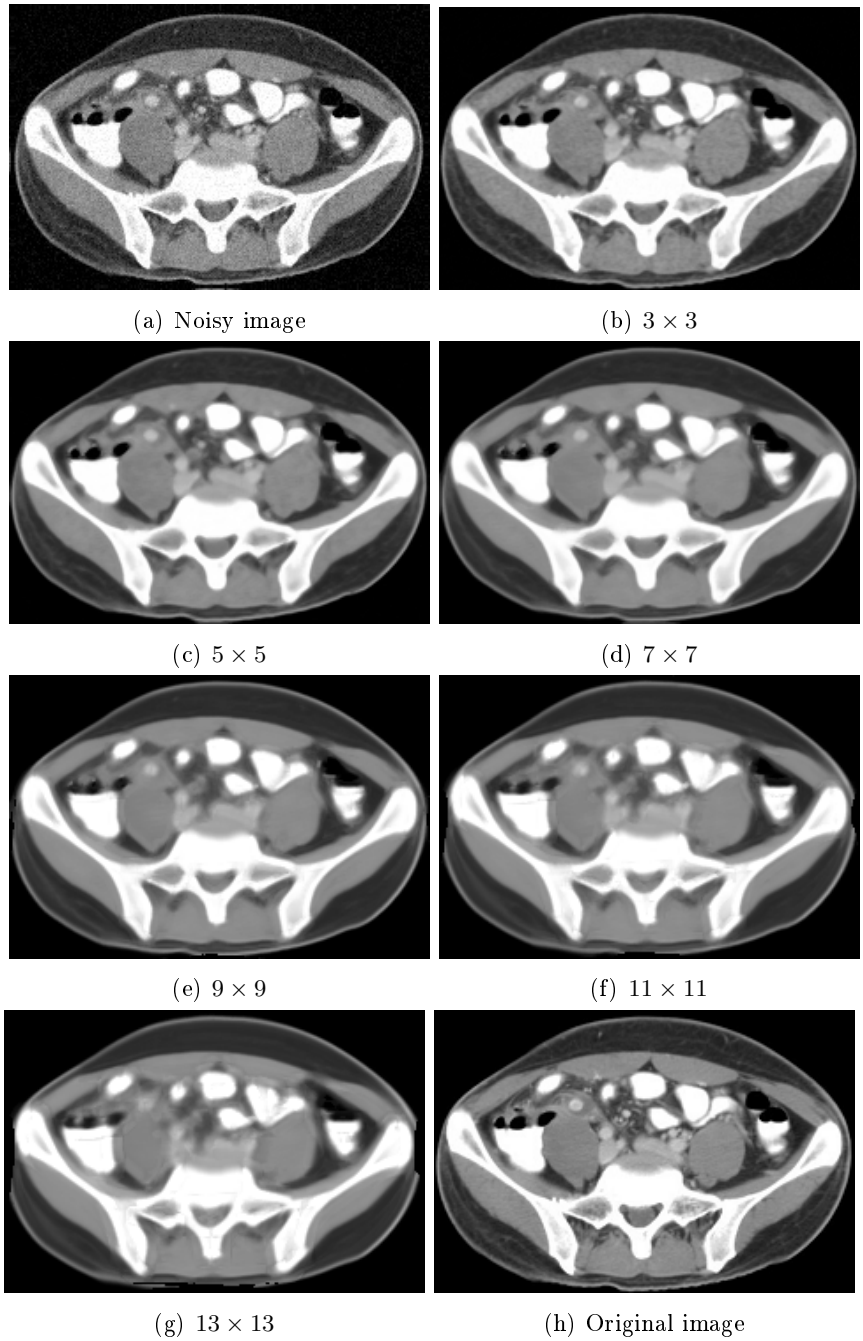
Visually, we present in Figure 2.20 the denoising results of the Pelvis image from different patch sizes. We can see that if the patch size is too large, some fine details cannot be properly recovered, and the denoising result will be smooth. Alternatively, the noise is not efficiently eliminated if the size of image patch is too small.

2.3.3.1.2 Effects of number of the nearest neighbors, k

For the proposed denoising approach, an important issue is the choice of the number of the nearest neighbors k. In our work, we test the impact of several values k =[1, 4, 7, 10, 14, 18, 22, 26, 30, 40, 50, 70, 100, 140, 200] on the quality measures, and analyse the evolution of PSNR and SSIM in function of k. We carry out a test on the images in Figure 2.13 for three noise levels $\sigma = 10$, 20 and 30. In our experiments, we use patch size 3×3 for $\sigma = 10$, 5×5 for $\sigma = 20$ and 7×7 for $\sigma = 30$, the decay parameter h is set to 50σ. These values are chosen from the observation of the above study to expect having good results. As it can be seen in Figure 2.21 and 2.22, the choice of k depends on the image and the noise level in the image. Generally, there is a range of values of k providing good quality indices. This "optimal" range of k is obtained from 4 to 15 for PSNR metric. However, for SSIM metric, these "optimal" values of k decrease with the noise level. For instance, the best SSIM of the Pelvis image is obtained for large range of k varying from $k = 50$ to 70 for small noise level ($\sigma = 10$), whereas this range narrows around $k = 5$ to 10 for higher noise levels ($\sigma = 20$ and 30).

2.3.3.2 Performance evaluation with Poisson noise

In this section we illustrate the performance of the proposed method for Poisson denoising through some experiments on synthetic and real data. The proposed method are compared with three denoising methods which were specifically designed for Poisson noise, including the Total Variation (TV) method in [LCA07], the Non-local Means (NLM) method in [DTD10] and the Block-Matching 3D filter in [MF11]. The parameters of the kNND methods are set as follows: the regression parameter $\lambda = 1$, the number of nearest neighbors $k = 7$, the parameter K in subsection 2.3.2.1



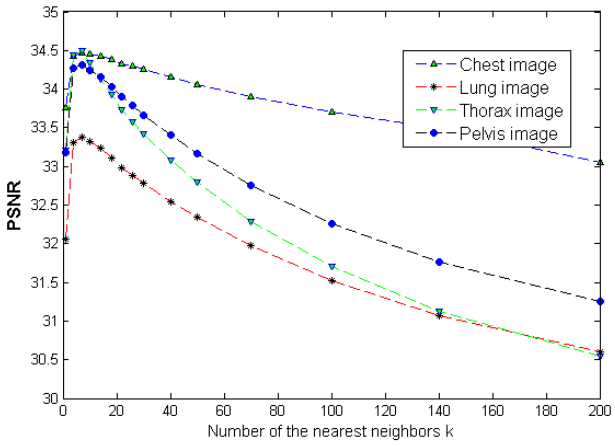
(a) Noisy image (b) 3×3

(c) 5×5 (d) 7×7

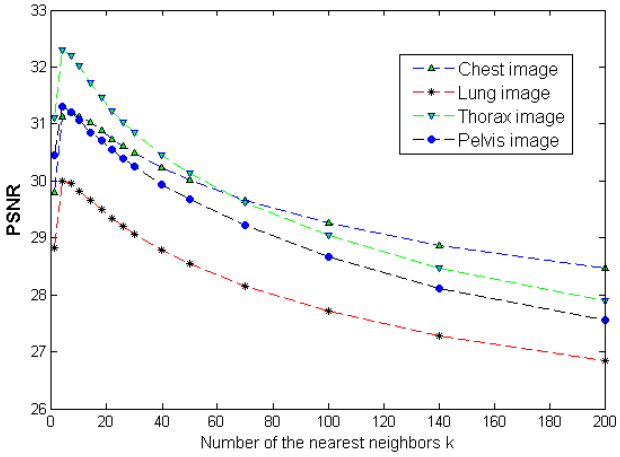
(e) 9×9 (f) 11×11

(g) 13×13 (h) Original image

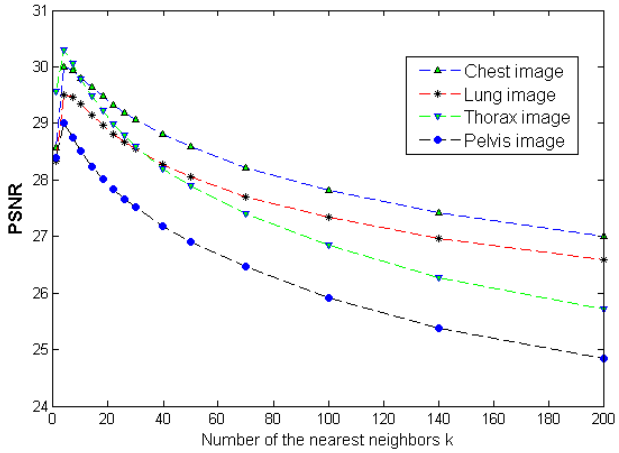
Figure 2.20: Comparison of denoising results on CT image of pelvis for $\sigma = 10$ with different patch sizes. (a) Noisy image, (b) 3×3 result, (c) 5×5 result, (d) 7×7 result, (e) 9×9 result, (f) 11×11 result, (g) 13×13 result, and (h) original image



(a) $\sigma = 10$

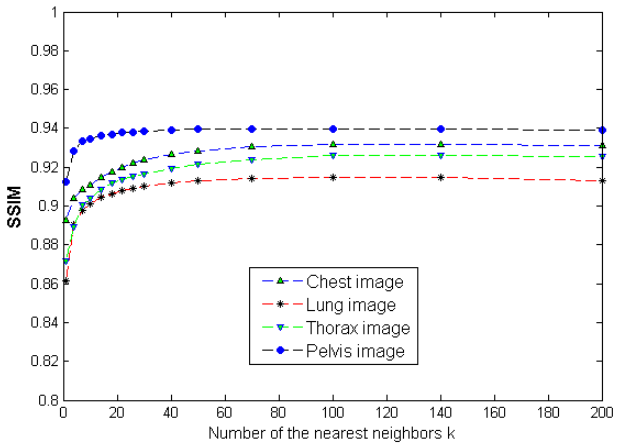


(b) $\sigma = 20$

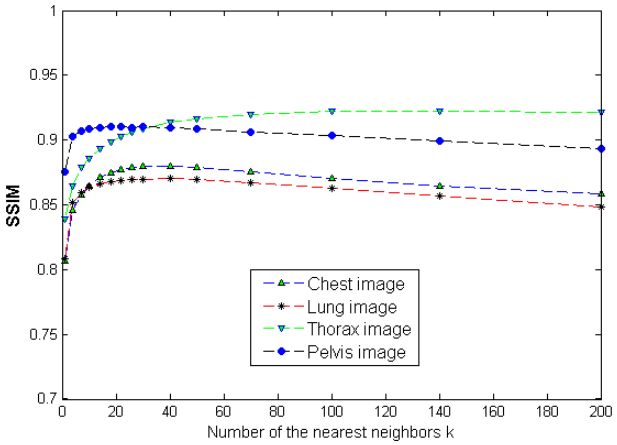


(c) $\sigma = 30$

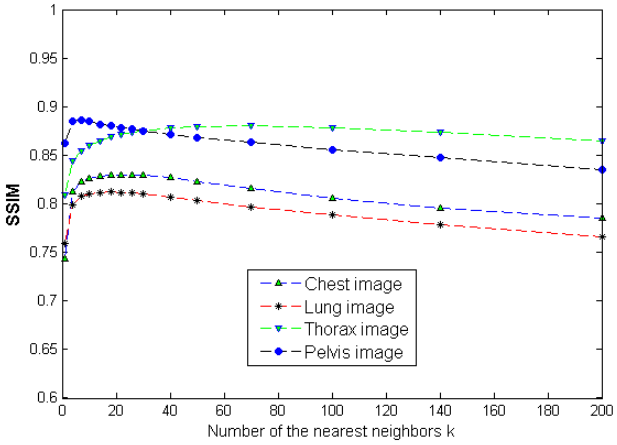
Figure 2.21: Effects of the number of the nearest neighbors k using the PSNR measure. Experiment is performed on the images in Figure 2.13 with three noise level $\sigma = 10$, 20 and 30.



(a) $\sigma = 10$

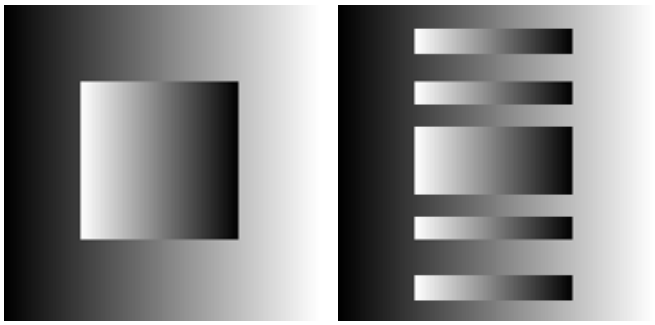


(b) $\sigma = 20$



(c) $\sigma = 30$

Figure 2.22: Effects of the number of the nearest neighbors k using the SSIM measure. Experiment is performed on images in Figure 2.13 with three noise level $\sigma = 10$, 20 and 30.



(a) Original image (b) Standard image

Figure 2.23: The images are used in the experiment with synthetic image: (a) original image and (b) standard image.

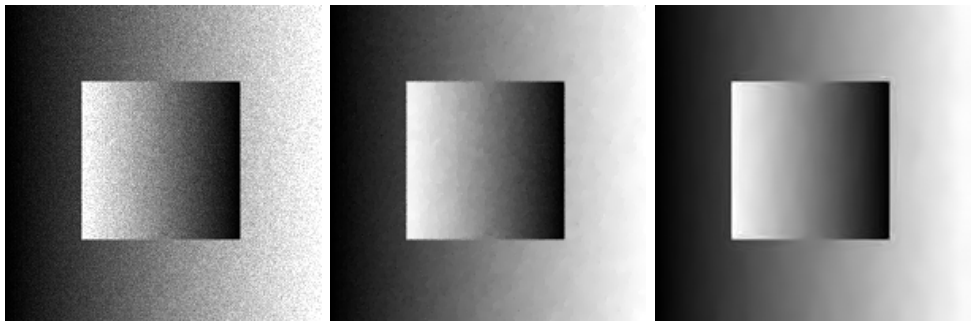
is set to $5k$, and the patch size is set to 7×7.

First, we report the experiment results on a synthetic image of size 200×200 pixels, in a gray scale ranging from 0 to 255 (Figure 2.23(a)). In our method, we use another image in Figure 2.23(b), to construct the training data set of 30000 observations.

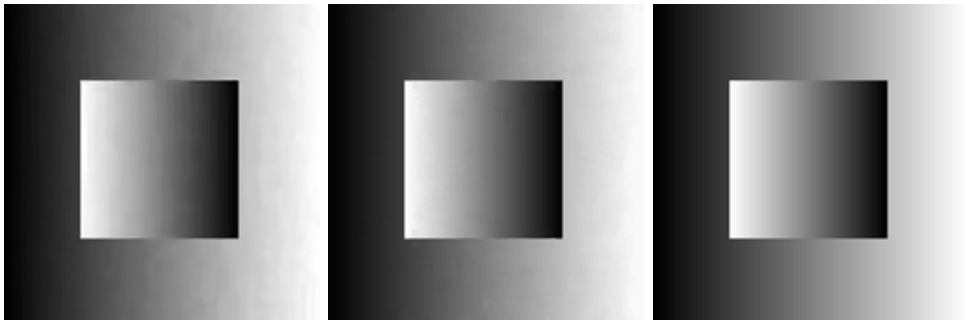
The original image in Figure 2.23 is corrupted by Poisson noise, providing a noisy image (Figure 2.24(a)). This corrupted image is then denoised by different denoising methods. The best results of the methods are presented in Figure 2.24. Objectively, the kNND method outperforms the TV [LCA07], NLM [DTD10] and BM3D [MF11]. As it can be seen in Figure 2.24(e), our kNND method effectively removes noise while excellently preserving the sharpness of image edges and corners.

Another experiment is realized on a PET image of chest (Figure 2.25(a)). This image is used as a test original image, meaning that it is corrupted by Poisson noise and then the Poisson denoising methods are applied to remove noise. Finally, the denoised results are compared with the original test image to evaluate the performance of the methods. In this experiment, we use another PET image taken at nearly the same location with the test image to construct the training data set for the kNND method (Figure 2.25(b)). The experimental results are shown in Figure 2.26. As it can be seen, the PSNR and SSIM of the denoised image by the kNND are considerably higher than of the images denoised by the TV in [LCA07] and the NLM in [DTD10], and slightly higher than those of the denoised image by BM3D in [MF11].

For subjective comparison, we can see the staircasing artifacts in the denoised image by the TV in [LCA07] (Figure 2.26(b)). It seems that there is no significant difference between the results of the three methods NLM, BM3D and kNND. To clearly see the difference in the results of these methods, Figure 2.27 respectively shows the residual images (defined as the difference between the noisy image and



(a) Noisy image (b) TV [LCA07] (PSNR=38.74, SSIM=0.943) (c) NLM [DTD10] (PSNR=42.32, SSIM=0.983)



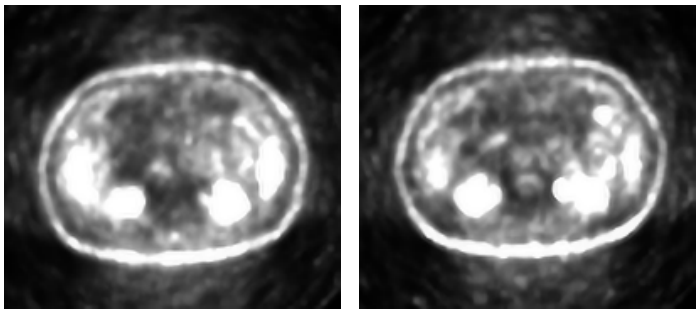
(d) BM3D [MF11] (PSNR=43.51, SSIM=0.981) (e) kNND (PSNR=**43.70**, SSIM=**0.985**) (f) Noise-free image

Figure 2.24: Experimental results on the synthetic image (Figure 2.23(a)) with Poisson noise: (a) noisy image, (b) denoised result by the TV method in [LCA07], (c) denoised result by the NLM method in [DTD10], (d) denoised result by the BM3D method in[MF11], (e) denoised result by the proposed kNND, and (f) original image (high quality).

the denoised image) of the TV [LCA07], NLM [DTD10] and BM3D [MF11]. These residual images are compared to the Poisson noise component in the noisy image in Figure 2.27(e). As can be seen, the residual image of the proposed kNND (Figure 2.27(d)) looks more similar to the Poisson noise image (Figure 2.27(e)) than of the other methods. This confirms the performance of the proposed method.

2.3.4 Conclusion

We presented in this section another learning-based method for denoising Gaussian and Poisson noise in medical image. This method is developed from the machine learning-based denoising method namely the KRRD proposed method in Section 2.2. Like the KRRD, the kNND method provides novel contribution in denoising approach for finding a regression function on a training set from a given set of standard images. However, unlike the KRRD method where the denoising phase is performed



(a) Original image (b) Standard image

Figure 2.25: Images used in the experiment with a PET image of abdomen: (a) test original image and (b) standard image used to construct the training data set. Image courtesy of Dr. Jean-Marie Rocchisani (Avicenne University Hospital, Bobigny, France).

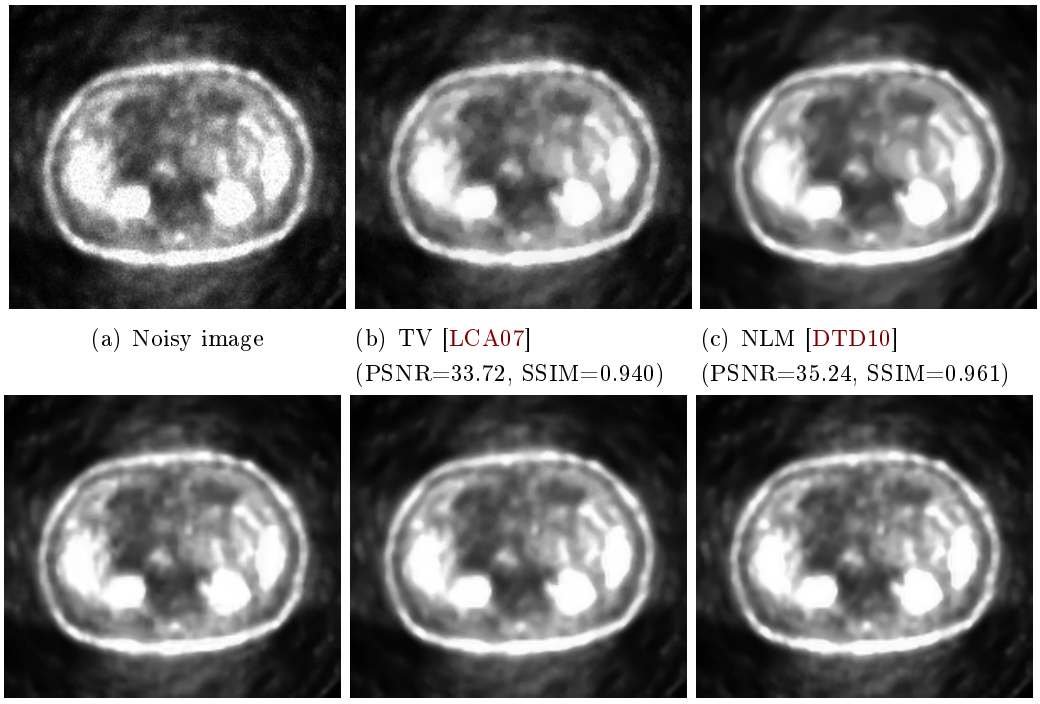
using the fixed regression functions trained off-line from the training phase, this second method performs the construction of regression function and denoising are performed simultaneously. Another contribution of the kNND method is to avoid using simulated noise in the construction of the training set. Experimental results carried out on CT, PET, and synthetic images demonstrated the performances of the proposed method over leading state-of-the-art techniques. We believe that with an effective training set, this technique may be quite useful and very promising.

2.4 The third proposed method: A Sparse Weight Model for Gaussian Denoising (SWMGD)

2.4.1 Introduction

In CT imaging, pixel noise can be decreased by increasing the X-ray dose. The disadvantage of increasing radiation dose is that high X-Ray doses may be harmful to patients. Thus, one of the challenges is to obtain a high quality image from a low-dose scan, which brings less damage to the patient. However, low radiation imaging is often associated with a number of quality-degrading artifacts, the most prominent of them being noise. Noise obscures diagnostically-valuable details; and if it can be removed by a robust image denoising technique, lower radiation scans become possible.

In this section, we focus on the problem of removing noise in low-dose CT images. It has been proven that the noise was often found to have a Gaussian distribution [LLHL02]. In fact, noise distribution in the CT image is usually unknown and noise variance in CT images is spatially changing (see Figure 2.28). Therefore, reducing



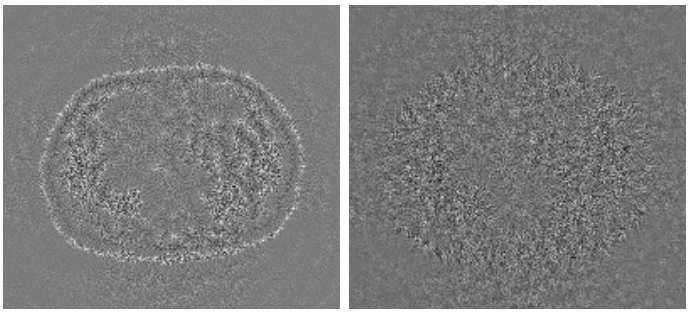
(a) Noisy image (b) TV [LCA07] (PSNR=33.72, SSIM=0.940) (c) NLM [DTD10] (PSNR=35.24, SSIM=0.961)

(d) BM3D [MF11] (PSNR=37.27, SSIM=0.974) (e) kNND (PSNR=**37.98**, SSIM=**0.976**) (f) Original test image

Figure 2.26: Comparison of the Poisson denoising methods on a PET image of abdomen: (a) noisy image, (b) denoised image by the TV method in [LCA07], (c) denoised image by the NLM method in [DTD10], (d) denoised image by the BM3D in [MF11], (e) denoised image by the kNND method, and (f) the original test image (considered as noise-free image).

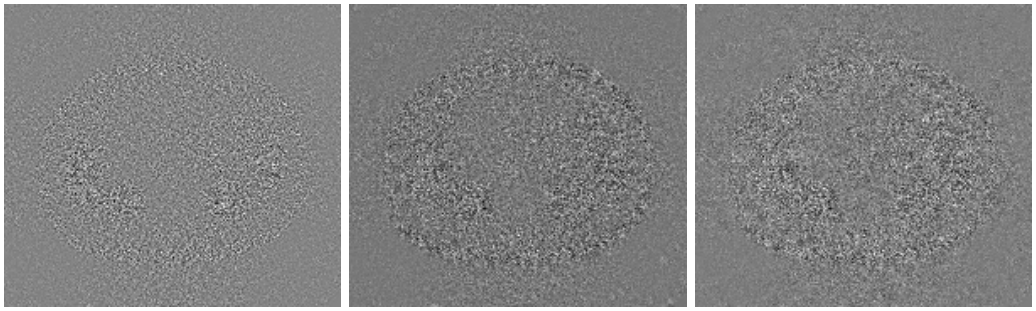
noise in reconstructed image becomes more difficult. In such cases, the denoising methods which are based on the assumption of independent identically distributed additive noise might not be efficient enough. These methods may adversely affect regions that have no noise or very slight noise and where there may be no need to denoise. In the proposed method, we assume that noise in low-dose CT image has locally Gaussian distribution. Thus, by considering that a noisy image is an arranged set of small patches, we can suppose that noise on each patch is an additive Gaussian noise with stable variance. Based on these assumptions, we can perform denoising on each patch. Note that noise level of different patches may be different.

Similar to the previously proposed methods, namely the KRRD and the kNND, we also make use of the standard images for denoising. However, unlike the previous method where the standard images are used to establish the training set of the observation examples $\{(\mathbf{x}_i\, y_i)\}$, in this second method the standard images are used to generate a database of image patches. This database of the standard image



(a) Residual image of TV [LCA07]

(b) Residual image of NLM [DTD10]



(c) Residual image of BM3D [MF11]

(d) Residual image of kNND

(e) Poisson noise

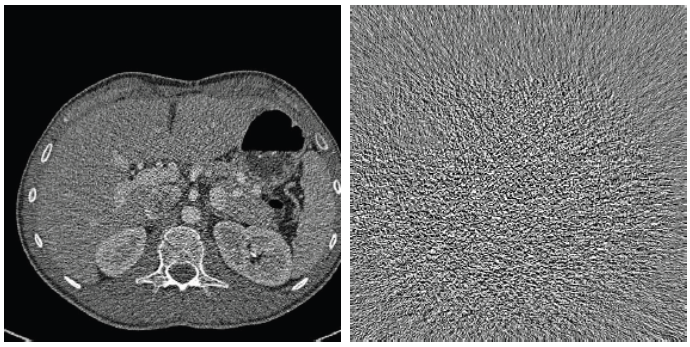
Figure 2.27: Comparison of the denoising methods on residual images: (a) residual image of TV [LCA07], (b) residual image of NLM [DTD10] and (c) residual image of BM3D [MF11], (d) residual image of kNDD, and (e) the Poisson noise component (defined as the difference between the noisy image and the test image) in the noisy image (Figure 2.26(a)).

patches is then directly used for patch-denoising.

To provide some motivation for our idea, we first consider the image given in Figure 2.29(a). In this image, we select two arbitrary 7×7 patches in the image. Then, for each patch, we find the 99 other patches in the image which are most similar to the given patch, the measure of similarity being the Euclidean distance. Then, we obtain 100 similar patches for each of two groups. Though the dimensionality of each patch is 49 (49 pixels in each patch), for each group of patches, 98% of the total variance[4] is contained in the top 15 eigenvectors (see Figure 2.29(b)). This indicates that the group of similar patches lies approximately on a 15-dimensional subspace. Therefore, a given patch can be sparsely represented by the similar patches.

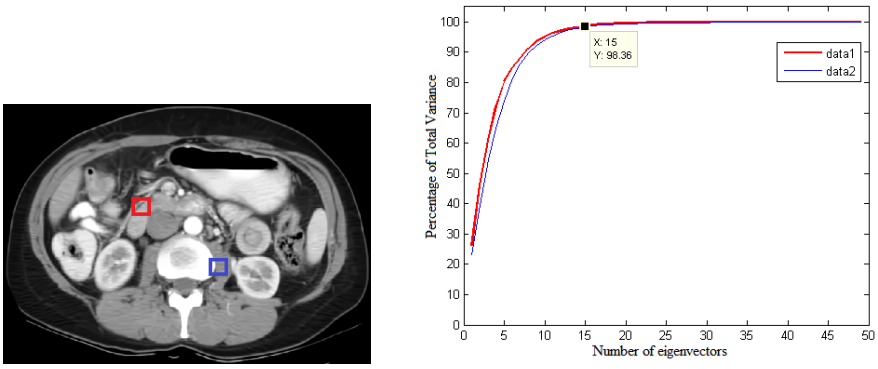
Now, tackle again the problem of denoising a noisy image patch with the help of a given database of standard patches. From the above observation, a simple

[4]The total variance of a data set is equal to the *trace* of the covariance matrix



(a) CT image (b) Noise

Figure 2.28: Real noise in a CT image.



(a) (b)

Figure 2.29: Distribution of the similar image patches

solution would be based on searching in the database, k-nearest neighbors of the noisy patch. However, this strategy highly depends on the parameter k for which it is not easy to find an optimal value. To overcome this limitation, we try to find a sparse positive linear representation of the noisy patch from standard patches. In this representation, the coefficients highly depend on the measure of similarity between patches. This idea is realized by formulating a constrained optimization formulation in which the similarity between patches is considered. Here, the measure of similarity between patches plays an important role in the behaviour of penaty function to enforce sparsity. In the remaining of this section, we refer to the method as SWMGD which stands for a Sparse Weight Model for Gaussian Denoising.

The contribution of this method is two fields. First, it is formulated as a sparse representation model where penalty function expressed in terms of dissimilarity measure is used to enforce sparsity, and the dissimilarity measure is proposed based on statistical properties of Gaussian noise for a better result. Second, the standard images are used as prior for the sparsity constraint in the optimization problem.

Unlike conventional sparse representation technique, this example-based method is designed to learn the prior from standard images without construction of a dictionary. The proposed method is detailed in the next section.

2.4.2 The proposed method

In this section, we present a sparse weight model for Gaussian denoising with the help of given set of standard image patches. Let $\Omega = \{\mathbf{u}_k \in \mathbb{R}^n, k \in \mathcal{I}\}$ denote the database of vectorized standard image patches, where $\mathcal{I}$ is the index set. Assume that Y is a noisy image that needs to be denoised. Our aim is to estimate the true image, denoted by X, from Y with the help of the database Ω. To this end, we consider Y as a set of small overlapping patches

$$Y = \{\mathbf{y}_i = \mathbf{R}_i Y, i = 1, \ldots, N\} \tag{2.55}$$

where $\mathbf{y}_i \in \mathbb{R}^n$ represent the vectorized $\sqrt{n} \times \sqrt{n}$ patch centered at location i, while $\mathbf{R}_i$ is a linear operator that extracts the patch at i. Overlap is often taken for these patches for two reasons: to avoid blockiness artifacts by averaging results and to take account for redundancy of estimates in denoising process. The proposed method is performed patch-wise with two main steps as follows.

- Step 1. Patch denoising:
  For each noisy patch $\mathbf{y}_i$, estimate the corresponding noise-free patch $\mathbf{x}_i$ of X. This is realized based on determining the best sparse positive representation of $\mathbf{y}_i$ over the database Ω via an optimization problem where a measure of similarity between patches is used as penalization function to enforce sparsity.
- Step 2. Aggregation:
  Construct the final estimate $\hat{X}$ for X using the result obtained from the first step.

The following subsections present these steps in details.

2.4.2.1 Step 1: Patch Denoising

Consider a noisy patch $\mathbf{y}_i$ in which the noise component η_i has normal distribution, $\eta_i \sim \mathcal{N}(0, \sigma_i^2)$ (mean 0 and standard deviation σ_i)

$$\mathbf{y}_i = \mathbf{x}_i + \eta_i. \tag{2.56}$$

Thanks to the repetition of local structures of images, we can believe that there exists a subset of similar patches (in Ω) which can be considered as the candidates

for each noise-free patch $\mathbf{x}_i$. Such patches will play an important role in finding an estimate $\hat{\mathbf{x}}_i$ of $\mathbf{x}_i$ from $\mathbf{y}_i$. In this work, $\mathbf{x}_i$ is assumed to be a weighted sum of the patches $\mathbf{u}_k$ in Ω,

$$\mathbf{x}_i = \sum_{k\in\mathcal{I}} \alpha_k \mathbf{u}_k, \qquad \alpha_k \geq 0, \forall k \in \mathcal{I}, \tag{2.57}$$

where α_k, with $k \in \mathcal{I}$ are the representation coefficients.

Indeed, to avoid the influence of non-candidate patches, it is better to involve only patches which are the good candidates for $\mathbf{x}_i$ in the estimation of $\hat{\mathbf{x}}_i$. Then, $\mathbf{x}_i$ can be seen as a sparse positive linear combination of the elements in Ω where most of zero coefficients α_k correspond to the elements $\mathbf{u}_k$ which are not good candidates for $\mathbf{x}_i$. That is why we try to estimate $\mathbf{x}_i$ based on a sparse positive linear representation over Ω with the weights α_k depending on the similarity between $\mathbf{u}_k$ and $\mathbf{y}_i$. Under these conditions, denoising a patch $\mathbf{y}_i$ implies solving the following sparse decomposition problem:

$$\begin{aligned} \alpha^* = \arg\min_{\alpha\geq 0} \|\alpha\|_0 + \sum_{k\in\mathcal{I}} \Phi_i\left(d(\mathbf{u}_k, \mathbf{y}_i)\right)\alpha_k \\ \text{subject to} \quad \left\|\mathbf{y}_i - \sum_{k\in\mathcal{I}} \alpha_k \mathbf{u}_k\right\|_2^2 \leq \varepsilon\sigma_i^2, \end{aligned} \tag{2.58}$$

where $\|\alpha\|_0$ stands for the ℓ_0-norm which counts the non-zero entries in α, ε is a positive constant, σ is the standard deviation of the noise on the patch $\mathbf{y}_i$, $\Phi_i : \mathbb{R} \to \mathbb{R}$ is a non-negative increasing function, and $d : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is a patch-dissimilarity measure. A high value of d, hence of Φ, forces the sparsity constraint.

In (2.58), the ℓ_0-norm assures that the solution α^* is a sparse one. In the second term, $\Phi_i\left(d(\mathbf{u}_k, \mathbf{y}_i)\right)$ may be viewed as the penalty coefficients in the sense that if the value of $\Phi_i\left(d(\mathbf{u}_k, \mathbf{y}_i)\right)$ is suitably large, the term $\Phi_i\left(d(\mathbf{u}_k, \mathbf{y}_i)\right)\alpha_k$ will be penalized a heavy cost if α_k is large. Thus, in the cases where $\mathbf{u}_k$ and $\mathbf{y}_i$ are very dissimilar (i.e. $\Phi_i\left(d(\mathbf{u}_k, \mathbf{y}_i)\right)$ is large), objective function in (2.58) can be minimized with weight α_k often very small or null. Therefore, for the sparse solution α^* of (2.58), the non-zero components often correspond to the small penalty coefficients. The constraint in (2.58) implies that the denoised patch $\hat{\mathbf{x}}_i$,

$$\hat{\mathbf{x}}_i = \sum_{k\in\mathcal{I}} \alpha_k^* \mathbf{u}_k \tag{2.59}$$

has to consistent with the noisy input $\mathbf{y}_i$. Therefore, the estimate $\hat{\mathbf{x}}_i$ of a desired patch $\mathbf{x}_i$ can be determined optimally from several standard candidate patches $\mathbf{u}_k \in \Omega$.

Normally, to measure the dissimilarity among the image patches, Euclidean distance is one of the most popular dissimilarity measure. However, in the case of

patches affected by noise this distance may not be effective enough. To obtain a better dissimilarity measure, we propose a novel dissimilarity criterion. For this, let us consider the residual patch, $\mathbf{y}_i - \mathbf{u}_k$. Since $\mathbf{u}_k$ in Ω is considered as noise-free while $\mathbf{y}_i$ is assumed to be corrupted by Gaussian white noise $\eta_i \sim \mathcal{N}(0, \sigma_i^2)$ (see (2.56)), the patch $\mathbf{u}_k$ is similar to $\mathbf{y}_i$ if $(\mathbf{y}_i - \mathbf{u}_k) \sim \mathcal{N}(0, \sigma_i^2)$. Then, we have

$$\begin{cases} \mathrm{E}(\mathbf{y}_i - \mathbf{u}_k) & \simeq 0 \\ \mathrm{Var}(\mathbf{y}_i - \mathbf{u}_k) - \sigma_i^2 & \simeq 0. \end{cases}$$

Therefore,

$$a_{i,k} = \big|\mathrm{E}(\mathbf{y}_i - \mathbf{u}_k)\big| + \big|\mathrm{Var}(\mathbf{y}_i - \mathbf{u}_k) - \sigma_i^2\big| \simeq 0. \tag{2.60}$$

The parameter $a_{i,k}$ allows us to evaluate the statistical property of noise in the residual patch. So, in this work, the dissimilarity measure is defined by

$$d(\mathbf{y}_i, \mathbf{u}_k) = \|\mathbf{y}_i - \mathbf{u}_k\|_2^2 + a_{i,k}. \tag{2.61}$$

It is easy to see that the objective in (2.58) is not a convex function, since ℓ_0-norm is not a true norm. This problem is too complex to solve in general. To avoid the above problem we replace ℓ_0-norm by ℓ_1-norm, and problem (2.58) is then convex and can be rewritten as:

$$\begin{aligned} \alpha^* = \underset{\alpha \geq 0}{\operatorname{argmin}} \ & \|\alpha\|_1 + \sum_{k \in \mathcal{I}} \Phi_i \left(d(\mathbf{y}_i, \mathbf{u}_k)\right) \alpha_k \\ \text{subject to} \quad & \Big\|\mathbf{y}_i - \sum_{k \in \mathcal{I}} \alpha_k \mathbf{u}_k\Big\|_2^2 \leq \varepsilon \sigma_i^2. \end{aligned} \tag{2.62}$$

Lagrange multipliers offer an equivalent formulation

$$\alpha^* = \underset{\alpha \geq 0}{\operatorname{argmin}} \ \frac{1}{2}\Big\|\mathbf{y}_i - \sum_{k \in \mathcal{I}} \alpha_k \mathbf{u}_k\Big\|_2^2 + \lambda \sum_{k \in \mathcal{I}} \big(1 + \Phi_i \left(d(\mathbf{y}_i, \mathbf{u}_k)\right)\big)\alpha_k, \tag{2.63}$$

where the positive parameter λ balances sparsity of the solution and fidelity of the approximation to $\mathbf{y}_i$.

Let us denote $S(\mathbf{y}_i) = \{k \in \mathcal{I} : \alpha_k^* > 0\}$ as the support set of $\mathbf{y}_i$. As analyzed above, $S(\mathbf{y}_i)$ involves $\mathbf{u}_k$ where the dissimilarity $d(\mathbf{y}_i, \mathbf{u}_k)$ is not large. Thus, with a suitable value of the threshold r_i, exists a subset $\mathcal{I}_i$ of $\mathcal{I}$,

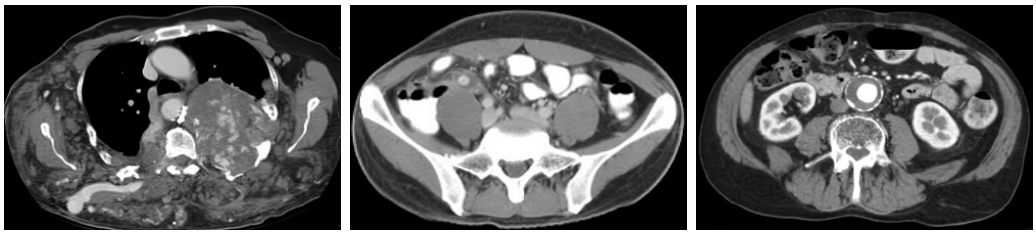
$$\mathcal{I}_i = \{j \in \mathcal{I} : d(\mathbf{y}_i, \mathbf{u}_k) \leq r_i\}, \tag{2.64}$$

such that

$$S(\mathbf{y}_i) \subseteq \mathcal{I}_i. \tag{2.65}$$

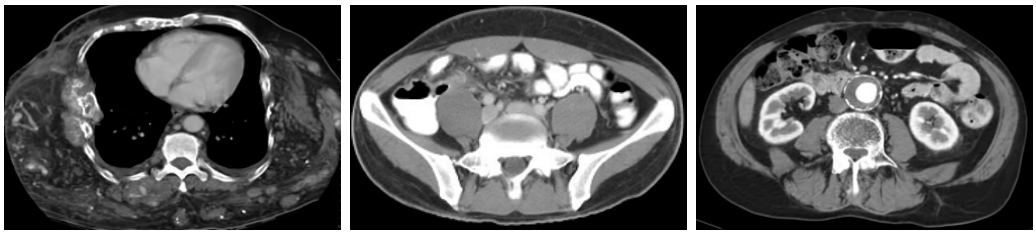
Thus, to save computing time, problem (2.63) should be considered on the subset $\mathcal{I}_i$,

$$\alpha^* = \underset{\alpha \geq 0}{\operatorname{argmin}} \ \frac{1}{2}\Big\|\mathbf{y}_i - \sum_{k \in \mathcal{I}_i} \alpha_k \mathbf{u}_k\Big\|_2^2 + \lambda \sum_{k \in \mathcal{I}_i} \big(1 + \Phi_i \left(d(\mathbf{y}_i, \mathbf{u}_k)\right)\big)\alpha_k \tag{2.66}$$



(a) (b) (c)

Figure 2.30: Test (noise-free) images for (a) CT of Chest, (b) CT of Pelvis, and (c) CT of Abdomen (source: http://radiopaedia.org/)



(a) (b) (c)

Figure 2.31: Standard images are used to construct the database of example patches (source: http://radiopaedia.org/).

It is easily to see that problem (2.66) can be rewritten as:

$$\begin{aligned} \alpha^* &= \underset{\alpha \geq 0}{\operatorname{argmin}} \frac{1}{2}\left\|\mathbf{y}_i - \mathbf{U}_i\alpha\right\|_2^2 + \lambda \mathbf{w}_i^T \alpha \\ &= \underset{\alpha \geq 0}{\operatorname{argmin}} \frac{1}{2}\alpha^T ({\mathbf{U}_i}^T \mathbf{U}_i)\alpha + \left(\lambda \mathbf{w}_i - {\mathbf{U}_i}^T \mathbf{y}_i\right)^T \alpha \end{aligned} \tag{2.67}$$

where $\mathbf{U}_i$ is the matrix whose columns are the vectors $\mathbf{u}_k$, $\mathbf{w}_i$ is the vector formed by concatenating all the numbers $\left(1 + \Phi_i\left(d(\mathbf{y}_i, \mathbf{u}_k)\right)\right)$, with $k \in \mathcal{I}_i$. We can see that (2.67) is a Nonnegative Quadratic Programming (NQP), which can be effortlessly solved by the multiplicative updates algorithm [SSL02] (see Subsection 2.3.2.2).

2.4.2.2 Step 2: Aggregation

Once the estimation $\hat{\mathbf{x}}_i$ of $\mathbf{x}_i$ from $\mathbf{y}_i$ for all $i = 1, 2, 3, \ldots, N$ are obtained, the final denoised image $\hat{\mathbf{X}}$ is then determined from the set $\{\hat{\mathbf{x}}_i,\ i = 1, 2, 3, \ldots, N\}$. We put $\hat{\mathbf{x}}_i$ in their proper locations and perform averaging in overlap regions to get the final image, enforcing then the consistency between neighboring patches. Indeed, since adjacent patches with overlap are often similar and provide redundant estimates, averaging in the overlap regions yields rather satisfactory result.

Table 2.9: PSNR comparison of denoised images

Image	σ	PSNR (dB)						
		Image denoising algorithms						
		TGV	NLM	ISKR	K-SVD	BM3D	RNLM	SWMGD
(a)	10	32.37	33.16	34.02	34.42	**34.83**	32.84	34.68
	20	29.22	29.83	29.92	30.27	30.96	29.89	**31.80**
	30	27.21	27.41	27.56	27.97	28.87	27.83	**29.32**
(b)	10	31.38	33.39	33.79	33.94	**34.45**	30.96	34.38
	20	28.68	30.16	29.96	30.06	30.84	28.57	**31.27**
	30	26.64	27.75	27.52	28.10	28.74	27.39	**29.10**
(c)	10	33.24	33.87	34.44	34.79	35.21	33.38	**35.72**
	20	29.33	30.26	30.57	30.81	33.52	29.61	**34.01**
	30	27.56	27.98	28.11	28.61	29.37	27.28	**31.35**

2.4.3 Performance evaluation

We have carried out several experimental results of the proposed method in the two cases: images with simulated noise and real noisy images. Furthermore, the proposed method is compared with some well-known denoising methods, including the ISKR method [TFM07], the TGV method [BKP10], the NLM method [BCM06], the K-SVD method [EA06], the BM3D method [DFKE07] and the RNLM method [XM11].

In our experiments, the noisy images are generated by adding Gaussian noise to the corresponding test noise-free image (Figure 2.30). The experiments are done on three test image, with $\sigma = 10$, 20 and 30. The database of 100000 standard patches is established from three standard images (images of nearly the same location as the test image) in Figure 2.31.

For objective evaluation, we use two image quality metrics, namely PSNR and SSIM. As can be seen from table 2.9 and table 2.10, the quality indices of our method (SWMGD) are higher than those of the ISKR, the TGV, the NLM, and the RNLM. Compared with K-SVD and BM3D, in some case of $\sigma = 10$, the quality indices of SWMGD are lower. However, SWMGD obtains the better quality indices than those of the K-SVD and the BM3D for high noise level, $\sigma = 20$, 30. Moreover, these quality indices are the best for strong noise levels.

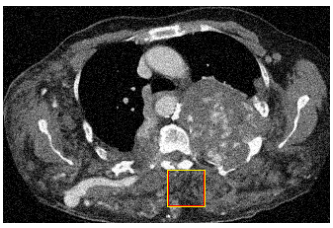
For subjective comparison, we show in Figure 2.32 the experimental results for CT image of chest (Fig. 2.30(a)) corrupted by Gaussian noise with the noise standard deviation $\sigma = 20$. Also, Figure 2.33 illustrates zoom-in images of a Desired

Table 2.10: SSIM comparison of denoised images

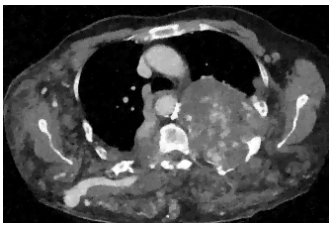
Image	σ	SSIM						
		Image denoising algorithms						
		TGV	NLM	ISKR	K-SVD	BM3D	RNLM	SWMGD
(a)	10	0.895	0.914	0.922	0.930	**0.946**	0.907	0.942
	20	0.816	0.807	0.853	0.857	0.854	0.841	**0.884**
	30	0.743	0.711	0.791	0.776	0.819	0.752	**0.830**
(b)	10	0.915	0.935	0.933	0.940	**0.945**	0.928	0.937
	20	0.872	0.866	0.888	0.895	**0.904**	0.892	**0.908**
	30	0.824	0.788	0.847	0.851	0.870	0.874	**0.888**
(c)	10	0.914	0.919	0.933	0.935	**0.945**	0.936	**0.945**
	20	0.835	0.831	0.874	0.863	0.880	0.882	**0.898**
	30	0.743	0.748	0.816	0.794	0.821	0.790	**0.859**

Region Of Interest (DROI) in Figure 2.32(b)-2.32(i), respectively. Visually, our SWMGD method effectively removes noise while better preserving the image structures compared to the other methods.

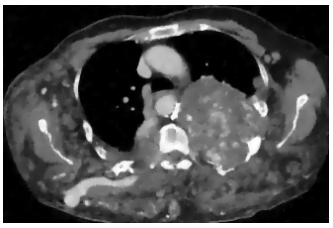
The proposed method is also performed on CT image with real noise. Figure 2.34 shows the experimental results on a noisy low-dose CT image of abdomen. As can be seen, the input low-dose image, Figure 2.34(a), is strongly corrupted by tomographic noise. The noise is assumed to have Gaussian distribution, with the standard deviation is estimated by the method in [MP03] $\hat{\sigma} = 27$. The proposed SWMGD method is compared to the TGV [BKP10], the NLM [BCM06], the ISKR [TFM07], the K-SVD [EA06], the BM3D [DFKE07] methods, and the example-based learning methods including the RNLM [XM11], the first proposed method KRRD, the second proposed kNDD method. For example-based learning methods, we use a standard image, Figure 2.34(b), to establish the training set for the KRRD and kNND method, and the database of standard patches for the SWMGD method. We use 7×7 patch size for the methods NLM, K-SVD, BM3D, and RNLM, kNND, and SWMGD. For the KRRD method, we use patch size in this case is 5×5. Denoised images by the methods are illustrated in Figure 2.34(c)-2.34(k). We can see in Figure 2.34(h)-2.34(k), the denoised results by the example-based denoising methods (RNLM, KRRD, kNND and SWMGD) look better than the results of the non-example-based methods (the TGV, NLM, ISKR, K-SVD and BM3D methods). The results of the methods TGV, NLM, ISKR, K-SVD and BM3D are blurred, e.g. bones in white. Compared to the results of the RNLM and the kNND, the results of SWMGD is less smoothed. Obviously, the SWMGD method better de-



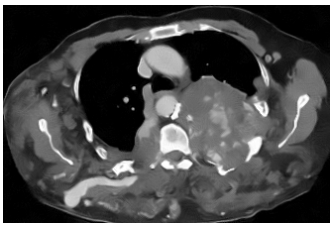
(a) Noisy image



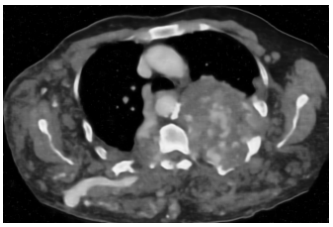
(b) Denoised image by ISKR [TFM07]



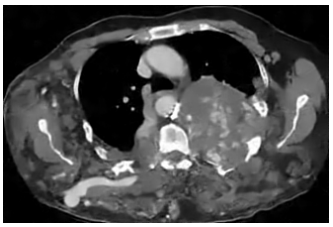
(c) Denoised image by TGV [BKP10]



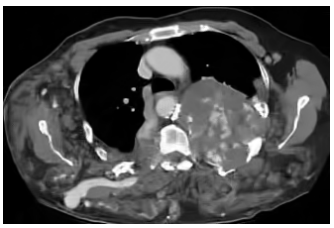
(d) Denoised image by NLM [BCM05]



(e) Denoised image by RNLM [XM11]



(f) Denoised image by K-SVD [EA06]



(g) Denoised image by BM3D [DFKE07]

(h) Denoised image by SWMGD

(i) Original image

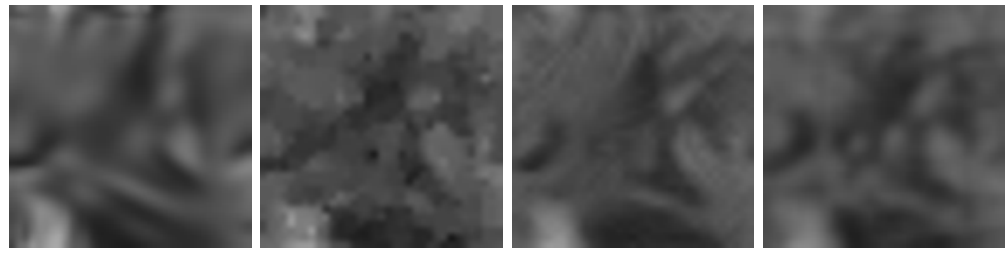
Figure 2.32: Results of the CT image of Chest in Figure 2.30(a): (a) Noisy image with Gaussian noise ($\sigma = 20$) in which the yellow square illustrates a desired region of interest, (b) result of ISKR [TFM07], (c) result of TGV [BKP10], (d) result of NLM [BCM05], (e) result of RNLM [XM11], (f) result of K-SVD [EA06], (g) result of BM3D [DFKE07], (h) result of SWMGD, and (i) original test image.

noises compared with the KRRD method. As it can be seen in Figure 2.34(k), by effectively denoising while slightly enhancing contrast, the result has its quality improved, although the denoised image by the SWMGD method has less contrast than the standard image (Figure 2.34(b)) due to the poor contrast of the noisy image.

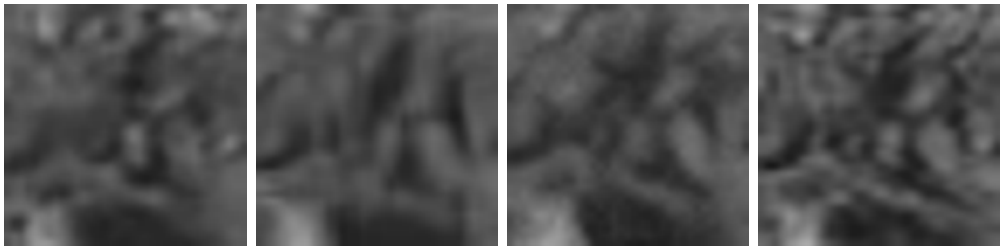
In the following, we present the effects of the patch size and regularization parameter in the proposed algorithm.

2.4.3.1 Effects of the patch size

Since SWMGD perform patch-wise denoising, the patch size thus is an important parameter of the algorithm. To see the impact of patch size, we test the proposed



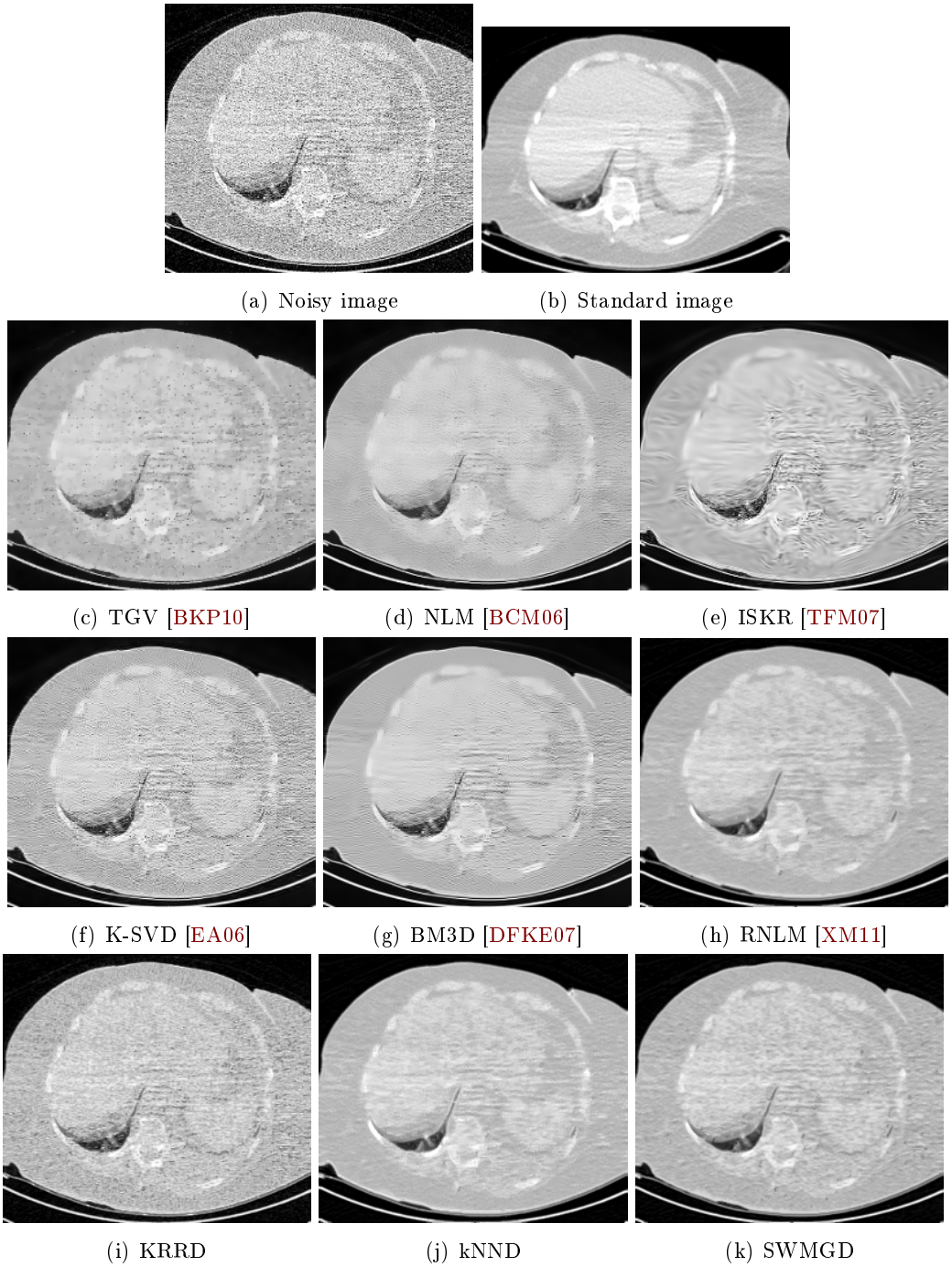
(a) ISKR [TFM07] (b) TGV [BKP10] (c) NLM [BCM05] (d) RNLM [XM11]



(e) K-SVD [EA06] (f) BM3D [DFKE07] (g) SWMGD (h) Original image

Figure 2.33: The zoom-in images of DROI in the denoised images in Figure 2.32. (a)-(g) are the DROI in the denoised results by the methods: ISKR [TFM07], TGV [BKP10], NLM [BCM05], RNLM [XM11], K-SVD [EA06], BM3D [DFKE07], and our SWMGD method, respectively. (h) The zoom-in image of DROI of the original image (Fig. 2.30(a)).

method on three images of Figure 2.30 with different patch size 3×3, 5×5, 7×7, 9×9, 11×11. The objective quality results in three cases of noise level, $\sigma = 10$, 20 and 30 are reported in Figure 2.35, 2.36 and 2.37, respectively. As it can be seen in those figures, the curves evolve with nearly the same form, with a maximum value of indices. The stronger the noise level, the larger size should be chosen to obtain the best quality indices. As it can be seen in Figure 2.35, with noise level $\sigma = 10$. The indices PSNR of the chest CT image (image (a)) and the pelvis CT image (image (b)) are the best with the patch size of 3. However, it is not valid for the case of abdomen CT image (image (c)) where the highest value of PSNR is obtained with patch size 5×5. With SSIM metric, we can see that in all three cases of images, the best results are obtained at the patch size 5×5. Figure 2.36 shows experimental results with $\sigma = 20$. Visually, the objective quality metrics of the images CT of chest and CT of pelvis increase when the patch size increases from 3×3 to 5×5 then reduce from 5×5 to 11×11. For the abdomen image, the best result is obtained at the patch size 7×7. In the case of noise level $\sigma = 30$, Figure 2.37 shows that with 7×7 patch size the highest quality are obtained for the chest image and the pelvis image while the best result for abdomen image is obtained with 9×9 patch size.



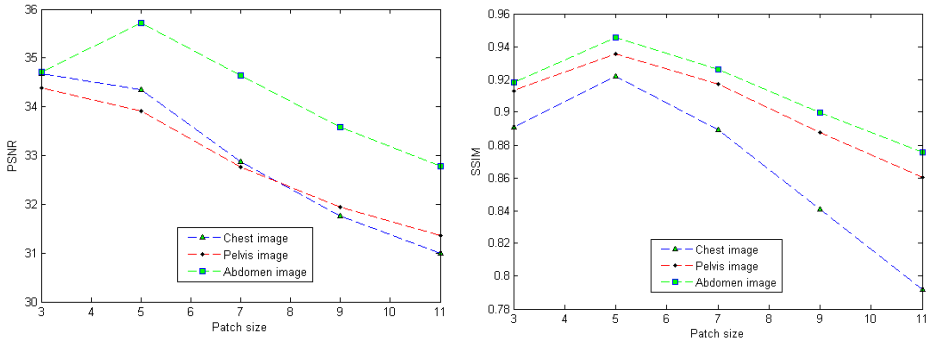
(a) Noisy image (b) Standard image

(c) TGV [BKP10] (d) NLM [BCM06] (e) ISKR [TFM07]

(f) K-SVD [EA06] (g) BM3D [DFKE07] (h) RNLM [XM11]

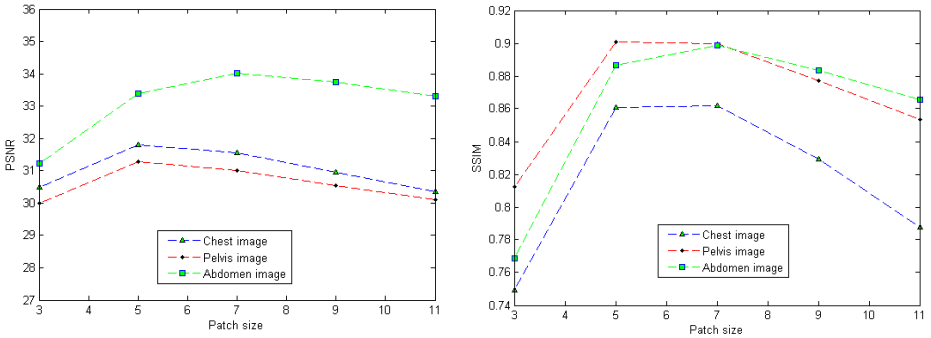
(i) KRRD (j) kNND (k) SWMGD

Figure 2.34: Experimental results on CT image of abdomen with real noise. (a) Noisy image, (b) Standard image, from (c) to (h): Denoised images by the TGV [BKP10], the NLM [BCM06], the ISKR [TFM07], the K-SVD [EA06], the BM3D [DFKE07], and the RNLM [XM11], respectively. From (i) to (k): Denoised images by the three proposed methods KRRD, kNND, and SWMGD, respectively. Image courtesy of Dr. Jean-Marie Rocchisani (Avicenne University Hospital, Bobigny, France).



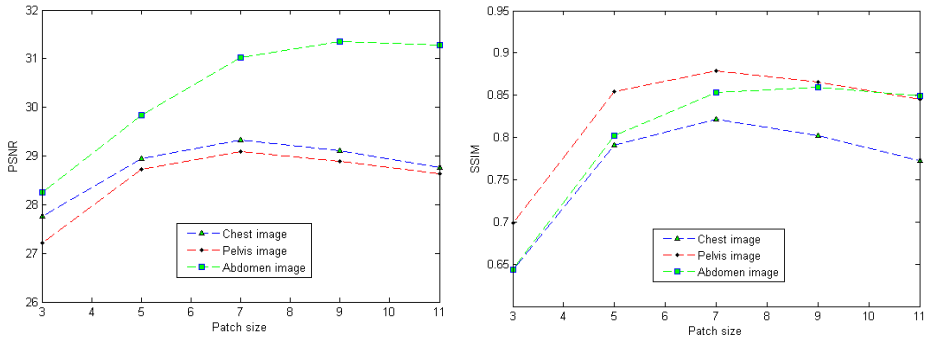
(a) PSNR (b) SSIM

Figure 2.35: Objective image quality measures with respect to the patch sizes for $\sigma = 10$.



(a) PSNR (b) SSIM

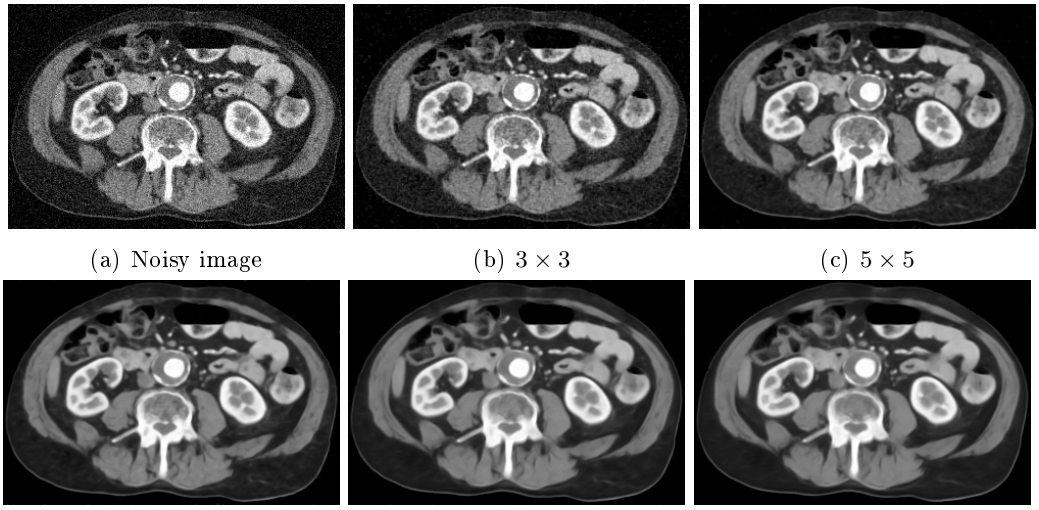
Figure 2.36: Objective image quality measures with respect to the patch sizes for $\sigma = 20$.



(a) PSNR (b) SSIM

Figure 2.37: Objective image quality measures with respect to the patch sizes for $\sigma = 30$.

To visually see the influence of the patch size, we show in Figure 2.38 the denoised images of the abdomen image (Figure 2.30(c)) corrupted by Gaussian noise with standard deviation $\sigma = 20$. It is observed that the result obtained from 7×7 patch



(a) Noisy image (b) 3×3 (c) 5×5

(d) 7×7 (e) 9×9 (f) 11×11

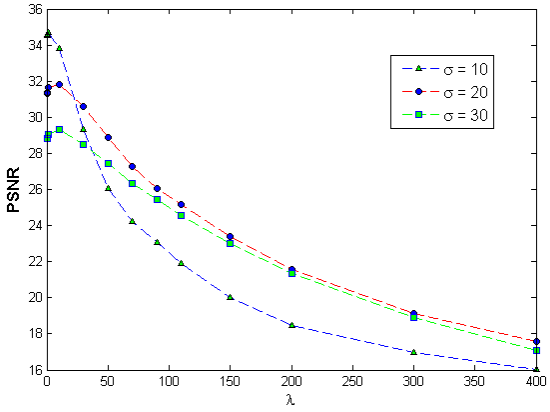
Figure 2.38: Comparison of denoising results on the Abdomen image (Figure 2.30(c)) corrupted by Gaussian noise with standard deviation $\sigma = 20$ with different patch sizes. From left to right, from top to bottom: (a) Noisy image, (b) 3×3 result, (c) 5×5 result, (d) 7×7 result, (e) 9×9 result, and (f) 11×11 result, respectively.

size in Figure 2.38(d) is the best. Obviously, with a small patch size (e.g., 3×3), the noise is not effectively removed. In contrast, with a larger patch size (e.g., 11×11), noise is very well removed but the result is oversmoothed. Therefore, the choice of patch size depends on noise level. We can conclude that the patch size is proportional to the noise level.

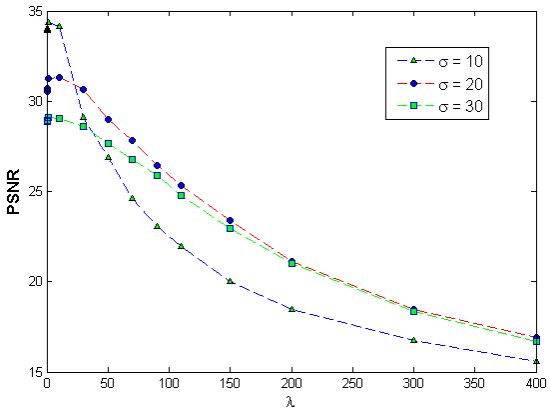
2.4.3.2 Impact of regularization parameter λ

In this subsection, we analyze the influence of the parameter λ in equation (2.67). For this purpose, we perform experiments on three images of Figure 2.30 with three noise levels $\sigma = 10$, 20 and 30. In all experiments, we choose the patch size 5×5 for noise levels $\sigma = 10$ and 20, and 7×7 for noise level $\sigma = 30$. The parameter λ is tested with several values $\lambda = [0\ 0.001\ 0.01\ 0.1\ 1\ 10\ 30\ 50\ 70\ 90\ 110\ 150\ 200\ 300\ 400]$. The evolution of PSNR and SSIM versus λ is analyzed.

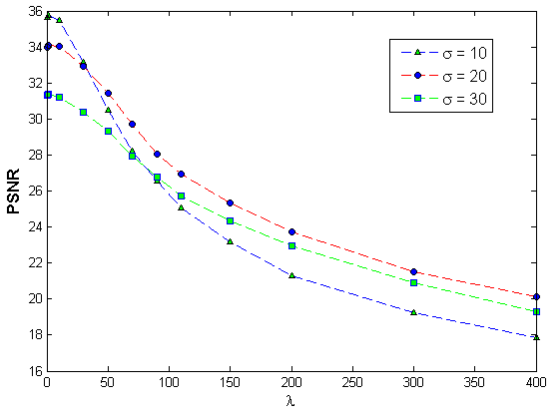
The PSNR and SSIM curves are illustrated respectively in Figure 2.39 and Figure 2.40. As it can be seen in Figure 2.39 and Figure 2.40, the PSNR, and SSIM curves increase when λ increases in some first steps (around from 0 to 10), and



(a) CT image of Chest

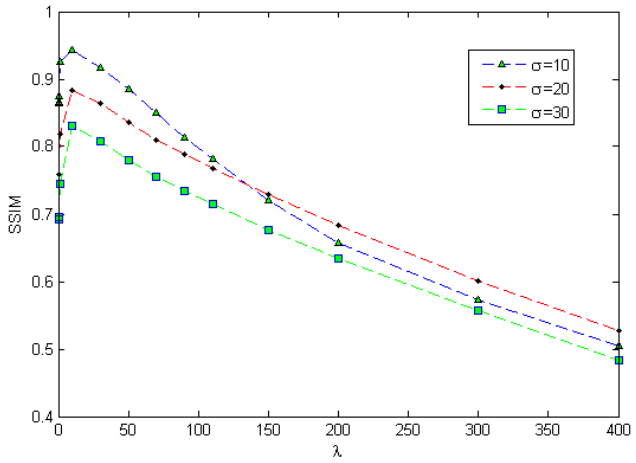


(b) CT image of Pelvis

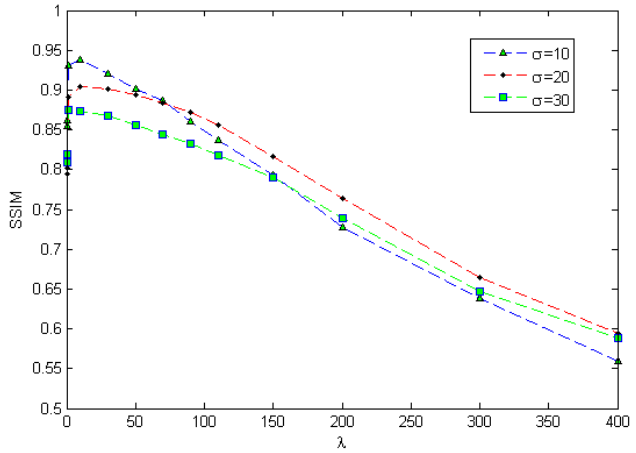


(c) CT image of Abdomen

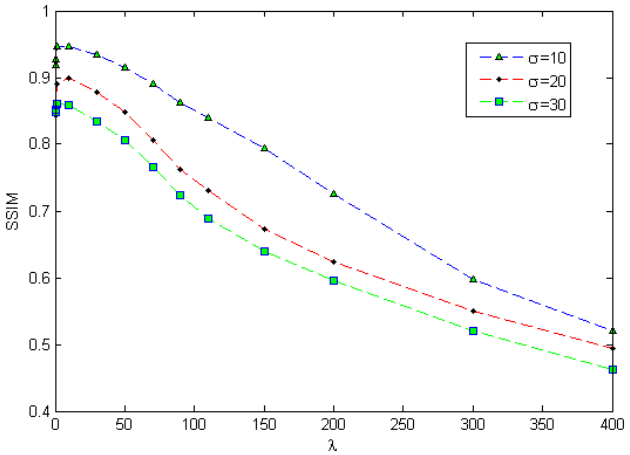
Figure 2.39: PSNR curves as a function of parameter λ.



(a) CT image of Chest



(b) CT image of Pelvis



(c) CT image of Abdomen

Figure 2.40: SSIM curves as a function of parameter λ.

then noticeably reduce when λ increases from 10 to 400. The optimal range of λ is usually found from 1 to 10. Figure 2.40

2.4.4 Conclusion

In this section, a novel example-based method has been proposed. The method is based on finding a sparse positive linear representation of the input noisy image patch over a database of standard image patches. The proposed method can effectively remove instable noise. Furthermore, the proposed method offers a simple formulation from the denoising problem within the sparse representation framework into a common nonnegative quadratic programming problem. The experimental results have demonstrated the effectiveness of the method on medical images. In the future works, we are going to study optimal solutions for establishing the database. Moreover, we will extend this model to other types of noise such as Poisson noise, Rician noise, Speckle noise.

2.5 Comparison of the proposed methods

In this section we present a comparison of the proposed methods for denoising Gaussian noise. First of all, let us outline the difference on the algorithms of the three methods KRRD, kNND and SWMGD. Considering the algorithms of the first method (KRRD) and the second method (kNND), the kNND has some improvements compared to the KRRD as follows:

1. Unlike the KRRD, the training set of the kNND method is constructed directly from the standard images. Thus, it avoids the problem of noise simulation as in the KRRD method.

2. By using the k-nearest neighbors we avoid the problem of finding the number of subsets need to be classified from the training set.

3. By using a small set Ω_k of the k nearest neighbors (see Section 2.3.2.1), the determining the regression function on Ω_k becomes easier.

4. In the denoising phase, for an input patch its output value is computed using the regression function which is determined from a subset of the k nearest neighbors of the input. Therefore, the second method overcomes the limitation when the noisy input lies on boundary of the groups as in the first method (see in subsection 2.2.6).

As it has been shown, kNND highly depends on the choice of parameter k. In order to reduce the influence of this parameter, the third method has been proposed.

Table 2.11: The types of noise and the proposed denoising methods

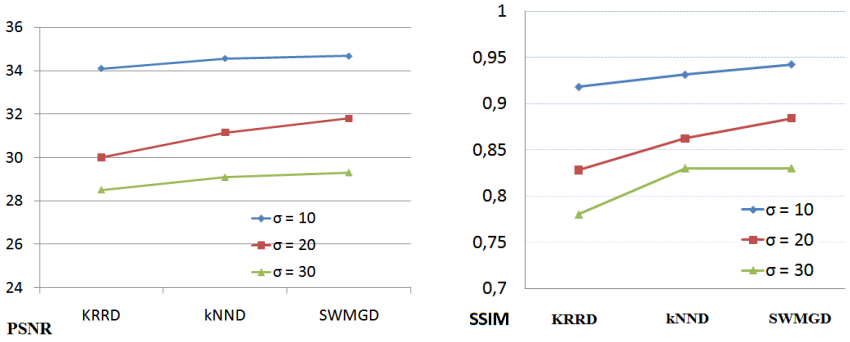
Method	**Type of noise**			
	Gaussian	Poisson	Rician	Speckle
KRRD	*Yes*		*Yes*	
kNND	*Yes*	*Yes*		
SWMGD	*Yes*			

Unlike the two first methods which try to find the nonlinear relationship between the examples in the training set, the third method (SWMGD) performs denoising based on a linear relationship between image patches. The sparse weight model in the SWMGD method allows the algorithm to be independent of the number of nearest neighbors.

On the application point of view, the KRRD method can reduce Gaussian noise and Rician noise, the kNND method is consistent with Gaussian noise and Poisson noise, while the SWMGD is designed for reducing Gaussian noise on CT images. This is summarized in Table 2.11. As it can be seen, all the proposed methods can remove Gaussian noise. So, we need to compare the performance of the three methods on CT images corrupted by Gaussian noise by carrying out some experiments.

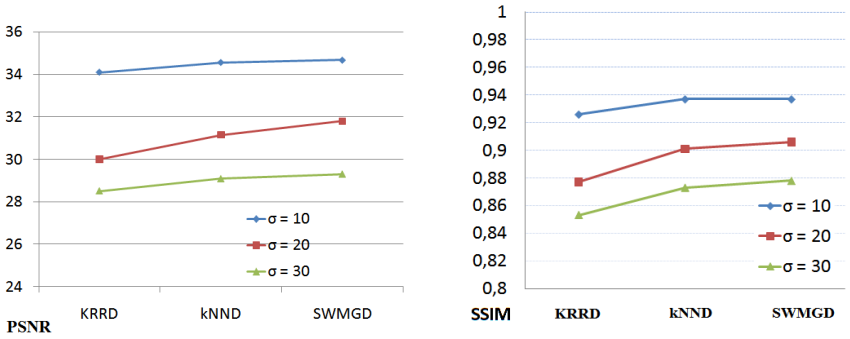
As in the experiments of the SWMGD method, we use the images in Figure 2.30 as the test noise-free images, and the images in Figure 2.31 are used as the standard images. Three cases are considered when the test image is corrupted by additive Gaussian noise with zero-mean and standard deviation $\sigma = 10, 20$ and 30. The quality metrics PSNR and SSIM are used to evaluate the performance of the denoising method. The experimental results are reported in Figures 2.41, 2.42 and 2.43. As it can be observed in all the PSNR and SSIM curves, the SWMGD has the best result, followed by the kNND then by the KRRD. This confirms that the SWMGD method outperforms the kNND and the kNND methods in reducing Gaussian noise from CT image.

From the above comparison and the characteristics of noise found in the types of medical imaging, e.g. the Gaussian noise usually appears in Computed Tomography (CT) [LS92, LLHL02] and sometimes in high intensity MRI [BZ03], the Poisson noise in systems involving counting procedures like X-ray Projection Radiography (X-PR), Fluoroscopy (FC), Mammography (MG), PET/SPECT [VSK85, OF97], the Rician noise in MRI images [GP95], the speckle noise in Ultrasound (US) imaging. Thus, the field of applications of the proposed methods for medical image denoising can be summarized in Table 2.12. Here, we use notation Yes^* for SWMGD to indicate that it is a good method for CT image denoising compared with the other two methods.



(a) (b)

Figure 2.41: Objective comparison of the methods KRRD, kNND and SWMGD on the CT image of chest (Figure 2.30(a)): (a) PSNR curves, (b) SSIM curves.

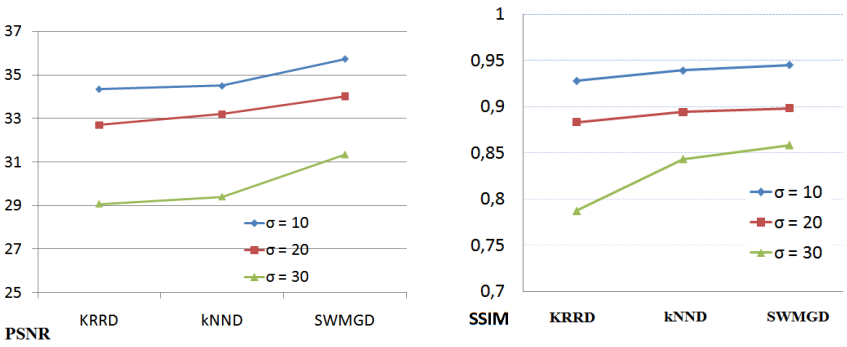


(a) (b)

Figure 2.42: Objective comparison of the methods KRRD, kNND and SWMGD on the CT image of pelvis (Figure 2.30(b)): (a) PSNR curves, (b) SSIM curves.

2.6 Conclusion

We have presented in this chapter three learning-based denoising methods. The main idea of the proposed methods is to use a given set of standard images as prior for denoising. The first method namely KRRD is a machine-learning-based approach designed for removing Gaussian noise and Rician noise, where we try to construct an automatic denoising machine model for medical images using the kernel ridge regression technique. The denoising machine is a set of many different regression functions which are trained from a training set constructed from the given standard images. Each of them corresponds to a type of noise with a certain noise level around a certain position in the body. Then, denoising can be performed using the functions of the machine. The noisy image is the input, while its output is the denoised image. This method not only effectively removes noise but also well preserves small details. The second method namely kNND is a k-nearest neighbor-based one which



(a) (b)

Figure 2.43: Objective comparison of the methods KRRD, kNND and SWMGD on the CT image of pelvis (Figure 2.30(c)): (a) PSNR curves, (b) SSIM curves.

Table 2.12: The applications of the proposed methods

Method	**Type of medical imaging**						
	CT	X-PR	FC	MG	PET/SPECT	MRI	US
KRRD	*Yes*					*Yes*	
kNND	*Yes*	*Yes*	*Yes*	*Yes*	*Yes*		
SWMGD	*Yes**						

is developed from the first method. However, unlike the previous method where training phase and denoising phase are performed independently (training phase is perform off-line), in this method they are realized in parallel. Moreover, we have also developed a novel model for determining the regression function. The experimental results have demonstrated the effectiveness of this method for removing Gaussian noise and Poisson noise. On the other hand, this method highly depends on the parameter k that is not easy to find its optimal value. To overcome this limitation, the SWMGD method has been proposed. Denoising in this method is performed via a nonnegative sparse representation model. This method is very consistent images corrupted by Gaussian noise such as CT images. The experimental results shown the performance of the proposed methods. In particular, the SWMGD outperforms the other denoising methods in denoising Gaussian noise in CT images.

In the future works, we will improve the proposed methods in order to reduce the dependency between the example images and the input noisy image (images taken at different locations and various characteristics). We will also extend the proposed methods to the other types of noise such as speckle noise and mixed noise.

Chapter 3

Medical Image Super-Resolution by Example-based Methods

Contents

3.1 Introduction

In this chapter, we propose two novel Super-Resolution (SR) methods, namely the Super-Resolution by Projection onto Convex Hull (SPOCH) and the Super-Resolution by Optimal Weights (SROW). These methods are based on examples using a given set of standard images, as in the denoising framework in the previous chapter. These methods are designed for enhancing spatial resolution of medical images [TLR+12a], [TLR+12b].

In medical imaging, image resolution is often limited by a number of factors such as hardware of the imaging system, time constraints, and so forth. A high quality and enhanced resolution imagery is quite useful to improve the clinical diagnosis [RCL+04, MCB+10, GEW10]. Resolution enhancement methods traditionally involve interpolations techniques that have occupied an important role from image generation through post-processing and visualization. In medical imaging, the interpolation methods have found widespread applications in medical imaging [LGS99]. However, the main drawback of these methods is their inefficiency when the given low-resolution (LR) image is corrupted by noise. Furthermore, these methods may also introduce blurring effect, ringing artifacts, as well as aliasing effect along edges [PPK03]. Thus, efforts have been devoted to develop new techniques. In particular, super-resolution methods which are known as effective techniques for improving image resolution.

Super-resolution (SR) image reconstruction is the problem of generating a high-resolution (HR) image from one or more LR images. Generally, SR designates methods that exploit the information contained in the LR images to produce an image with a higher spatial resolution than that of the original LR images. Up to now, a large number of SR approaches have been developed. A good overview of the resolution enhancement methods can be found in [PPK03, Ouw06, LGS99, FREM04a]. Broadly, these methods can be divided into two main categories: multi-image super-resolution methods and single-image super-resolution ones.

3.1.1 Multi-images super-resolution

Multi-images methods perform super-resolution by using information extracted from a few LR images of the same scene [IP91, TKG95, HBBW98, EF99, TH99, ABHY00, FREM04b, SLZZ04]. The classic model of multi-images super-resolution in the spatial domain assumes that a sequence of K low-resolution images represent different snapshots of the same scene. The real scene to be estimated is represented by a vectorized single high-resolution reference image $\mathbf{X}$ (of size $N^2 \times 1$). Each LR frame, $\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_K$, is a noisy, downsampled version of the reference image that is subjected to various imaging conditions such as optical, sensor and atmospheric blur, motion effects, and geometric warping. The size of each LR frame is $M^2 \times 1$, $M < N$. The LR observations are related with the HR scene $\mathbf{X}$ by

$$\mathbf{Y}_k = D_k H_k F_k \mathbf{X} + V_k,\ k = 1, 2, \ldots, K, \tag{3.1}$$

where F_k is the $N^2 \times N^2$ matrix encoding the motion information for the k-th frame, H_k (size-$N^2 \times N^2$) models the blurring effects, D_k is the down-sampling operator of size $M^2 \times N^2$, and V_k ($M^2 \times 1$) is the noise term. These linear equations can be

rearranged into a large linear system

$$\begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \vdots \\ \mathbf{Y}_K \end{bmatrix} = \begin{bmatrix} D_1 H_1 F_1 \\ D_2 H_2 F_2 \\ \vdots \\ D_K H_K F_K \end{bmatrix} \mathbf{X} + \underline{V} \tag{3.2}$$

or equivalently

$$\underline{\mathbf{Y}} = F\mathbf{X} + \underline{V}. \tag{3.3}$$

Most of the existing methods recover the HR image $\mathbf{X}$ based on the assumption that the matrices D_k, H_k, F_k, V_k are exactly known. A typical solution for super-resolution from an image sequence involves three sub-tasks: registration, fusion and deblurring (see Figure 3.1). The first and most important task of these methods is

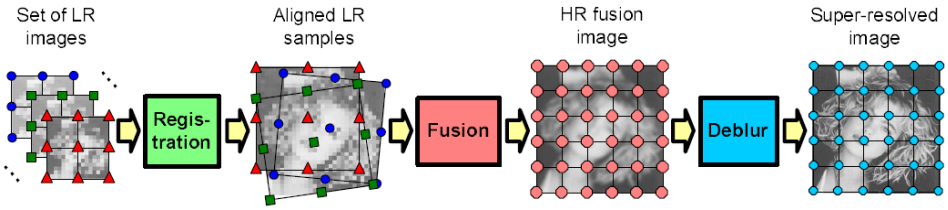


Figure 3.1: A three-step algorithm for super-resolution of an image sequence: image registration, image fusion and image deblurring.

motion estimation, or registration between LR images because the precision of the estimation is crucial for the success of the whole method [PPK03]. However, such assumption is only feasible when considering a static scene. In general, it is not easy to estimate accurately the motion between LR images, because it may involve not only horizontal and vertical shifts, but also rotation angles, etc.

In practical applications, the linear system (3.2) is typically ill-posed. Furthermore, in real imaging systems, the matrices D_k, H_k, F_k, V_k are unknown and need to be estimated, leaving the problem even more ill-conditioned. As a result, the performance of the multi-image-based SR methods is not ready for practical applications.

3.1.2 Single-image super-resolution (Example-based super-resolution)

The single image SR methods, also known as example-learning-based methods, have been successful in recent years. The focus of single-image super resolution is to estimate a high-resolution (HR) image with just a single low-resolution image. These methods recover missing high frequency details based on learning the

mapping between low and high-resolution image patches from a given database. Many methods have been proposed for learning with demonstrated promising results [FJP02, CYX04, NN07, SSXS08, TLBL10, YHY10, KK08, YWHM08, GBI09, YWHM10, ZEP10].

In the example-learning-based SR methods, an image is considered as a set of image patches and SR is performed on each patch. Some of proposed methods are based on the nearest neighbors search (e.g. [FJP02, CYX04, KK08]). Basically, each patch of the LR image is compared to the LR patches stored in the database. The HR patches corresponding to the nearest LR patches are then used to estimate the output via many approaches. In [FP00, FJP02], Freeman *et al.* used a Markov network to probabilistically model relationships between HR and paired LR patches, and between neighboring HR patches, and an approximate solution is proposed using the belief propagation. In [CYX04], Chang *et al.* proposed to determine the HR patch by a linear combination of the nearest neighbors such that it is closest to the input LR patch. Then, the output HR patch is defined by replacing LR patches with the associated HR patches in the linear combination. In [KK08], Kim *et al.* exploited the relationship between patch pairs based on finding a regression function. Despite the success of the nearest neighbor-based methods, the drawback of these methods is that they highly depend on the number of the nearest neighbors. More recently, some methods based on learning the sparse association between image patches have been proposed [YWHM08, YWHM10, ZEP10, YWL+12, Ela10] and offer promising performance. In these approaches, LR (HR) patches can be sparsely represented by the atoms of a dictionary $\mathbf{D}_L$ ($\mathbf{D}_H$ respectively) which is learned from the database of low and high-resolution image patch pairs. Then, super-resolution is performed based on the assumption that the sparse-representation coefficients for representing the LR patch by $\mathbf{D}_L$ are the same as those for representing the HR patch by $\mathbf{D}_H$. Unlike the K-nearest neighbor-based methods, the advantage of this method is that it is independent of K and needs not compute the similarity between image patches is not computed. These methods offer promising performances. However, the main drawback of these methods is related to learning the dictionary from the database which is often time-consuming. Another difficulty is the questionable performance of these methods when dealing with noisy images.

Conventionally, the database of low and high-resolution patch pairs for example-learning-based methods is often established from extrinsic large dataset (e.g. [FJP02, CYX04, KK08, YWHM08, YWHM10, Ela10]). Some novel single image SR methods have been proposed where the database can be extracted directly from the input LR image, particularly when it is considered as a noise-free image (e.g. [GBI09], [YHY10]). In [GBI09], Glasner *et al.* proposed a SR method that exploits

the patch redundancy among in-scale and cross-scale images in an image pyramid to enforce constraints for reconstructing the unknown HR image [YHY10]. However, the challenge remains when the LR input image is noisy.

The advantage of the single-image SR methods is that they do not require LR images of the same scene as well as registration. However, the main limitation of these methods is that their effectiveness highly depends on the supporting database.

3.1.3 Our choice

In medical imaging, it is difficult to obtain a sequence of images represent different snapshots of the same scene. On the other hand, we observe the interesting fact that lots of images are taken from the same type of subject or anatomical structures at nearly the same location, and many of them can be considered by experts as standard (high quality or proven as noise-free by experts) images. Therefore, we can use these available standard images as examples to establish the database. That is why we follow the example-learning-based approach to design our methods for improving spatial resolution of medical images.

The rest of this chapter is organized as follows. In section 3.2, an overview of example-based super-resolution approach is presented. We notably focus on the nearest neighbor-based SR methods [FP00, FJP02, CYX04] and the sparse-coding SR methods [YWHM08, YWHM10, ZEP10]. Afterward, two novel example-learning-based methods for single image super-resolution ([TLR+12a],[TLR+12b]) are proposed and presented in section 3.3.

3.2 Survey of example-learning-based super-resolution methods

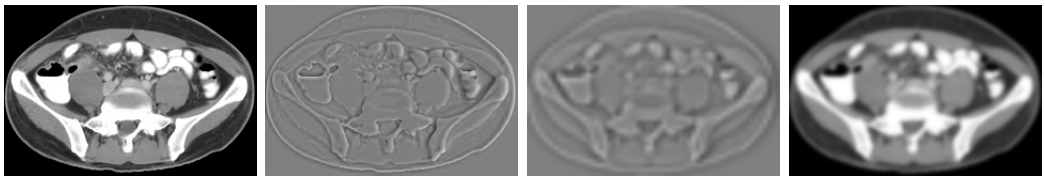
Let us recall the problem of example-learning-based super-resolution. Given a LR image $\mathbf{Y}$ and a set of example images (high quality images) $\{\mathbf{A}_h\}$, find a HR version $\mathbf{X}$ of $\mathbf{Y}$ with magnification factor s. The LR observation $\mathbf{Y}$ is related with the HR one $\mathbf{X}$ by

$$\mathbf{Y} = D_s H \mathbf{X} + V, \tag{3.4}$$

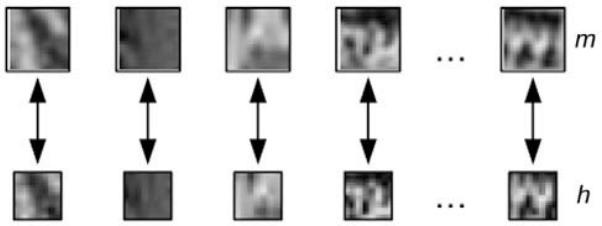
where, H is a blur operator, D_s is a decimation operator with factor s, and V is the additive noise component.

In example-learning-based SR methods, a common assumption is that a HR image consists of three frequency layers, specifically the high frequency layer (denoted as $\mathbf{h}$), the middle frequency layer (denoted as $\mathbf{m}$), and the low frequency layer (denoted as $\mathbf{l}$) [WZG10] (an illustration is shown in Figure 3.2). It is also assumed that

the LR image is obtained from the original HR version by discarding the high frequency components. Thus, the super-resolution goal of this approach is to estimate missing high-frequency detail that is not present in the original LR image. To this end, the example-learning-based approach solves the SR problem by maximizing the conditional probability of $\mathbf{h}$, given $\mathbf{m}$and $\mathbf{l}$, $Pr(\mathbf{h}|\mathbf{m},\mathbf{l})$ for any input LR image. In addition, the high frequency component $\mathbf{h}$ is independent of $\mathbf{l}$, hence it is only required to maximize $Pr(\mathbf{h}|\mathbf{m})$.



(a) HR image (b) $\mathbf{h}$ layer (c) $\mathbf{m}$ layer (d) $\mathbf{l}$ layer



(e) co-occurrent patches

Figure 3.2: Image co-occurrence patterns.

A typical example-learning-based SR method consists of two main phases: the *database construction* phase and the *Super-resolution reconstruction* phase.

- **Database construction:** From the example images, a database of patch pairs $(\mathbf{P}_m, \mathbf{P}_h) = \{(u_i^m, u_i^h), i \in \mathcal{I}\}$ ($\mathcal{I}$ is the index set) are extracted from the middle frequency layer $\mathbf{m}$ and the corresponding high frequency layer $\mathbf{h}$, respectively. Both u_i^m and u_i^h are the vectors corresponding to the square image patches, $u_i^m \in \mathbb{R}^{d_1}$ and $u_i^h \in \mathbb{R}^{d_2}$.

- **Super-resolution reconstruction:** For a LR input image $\mathbf{Y}$, N middle frequency patches are extracted in a similar way as in the database construction phase, and denoted as $\{y_i^m, i = 1, 2, \ldots, N\}$. The missing high frequency components $\{y_i^h, i = 1, 2, \ldots, N\}$ are then estimated with the help of the patch pairs in the training data $(\mathbf{P}_m, \mathbf{P}_h)$. The final HR image is computed from the estimated high frequency components.

Let us now describe some principal methods of the example-learning-based SR approach. Let us start with the Markov random field-based SR, then, the K-nearest

neighbor-based SR methods (SR via Neighbor Embedding), followed by the SR method via sparse representation.

3.2.1 Markov random field based super-resolution

In this subsection we present in details the method proposed by Freeman *et al.* [FP00, FJP02] (cited more than 850 times since 2000), namely the Markov Network-based method (MN). We begin with the database construction phase, followed by the SR reconstruction phase.

3.2.1.1 Database construction phase

The database of high and low-resolution patch pairs is constructed from a given set of the example HR image $\mathbf{A}_h$. Before generating the patch pairs, these training images are pre-processed through some steps:

1. Perform blurring and subsampling of the HR example image $\mathbf{A}_h$ to create the corresponding low-resolution image $\mathbf{A}_l$ ($\mathbf{A}_l = D_s H \mathbf{A}_h$), where D_s is the dow-sampling operator and H models the blurring effects.

2. After creating the low-resolution image, an initial analytic interpolation Q, such as cubic spline, is applied to the low-resolution image $\mathbf{A}_l$. This step generates an image $\mathbf{B}_l$ of the same size as $\mathbf{A}_h$, $\mathbf{B}_l = Q(\mathbf{A}_l)$. $\mathbf{B}_l$ is called the *upsampled low-resolution image*.

3. A low-pass filter is applied to the upsampled low-resolution image $\mathbf{B}_l$. The output of the low-pass filter is called the low-band, $\mathbf{a}_l$. The image $\mathbf{a}_m = \mathbf{B}_l - \mathbf{a}_l$ is called the mid-frequency band. The difference between the upsampled low-resolution image, $\mathbf{B}_l$, and the original image, $\mathbf{A}_h$, is the high-frequency band, $\mathbf{a}_h$ ($\mathbf{a}_h = \mathbf{A}_h - \mathbf{B}_l$).

4. The authors assumed that the relationship of the mid-frequency band, $\mathbf{a}_m$, to the high-frequency band, $\mathbf{a}_h$, data is independent of the local contrast level. So, the contrast of the mid- and high-band is normalized in the following way:

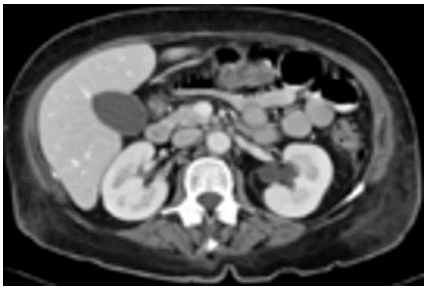
$$\hat{\mathbf{a}}_m = \frac{\mathbf{a}_m}{std(\mathbf{a}_m) + \varepsilon} \tag{3.5}$$

$$\hat{\mathbf{a}}_h = \frac{\mathbf{a}_h}{std(\mathbf{a}_m) + \varepsilon} \tag{3.6}$$

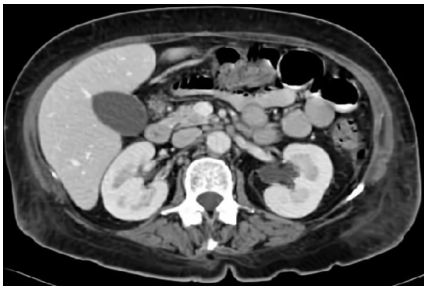
where $std(\cdot)$ is a standard deviation operator, and ε is a small value added to avoid the denominator becoming zero at very low contrasts.

The training database $(\mathbf{P}_m, \mathbf{P}_h)$ stores the vectorized patch pairs (u_i^m, u_i^h) in which u_i^m, u_i^h correspond to the patches at pixel i of $\hat{\mathbf{a}}_m$ and $\hat{\mathbf{a}}_h$, respectively. The processing steps for the database construction phase are illustrated in Figure 3.3.

(a) LR image ($\mathbf{A}_l$)

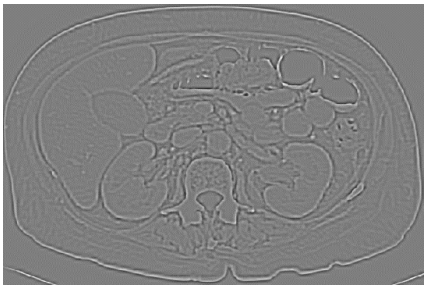


(b) Initial interpolation ($\mathbf{B}_l$)



(c) HR image ($\mathbf{A}_h$)

(d) Band-pass filtered and contrast normalized version ($\hat{\mathbf{a}}_m$)



(e) High-pass filtered version ($\mathbf{a}_h$)

Figure 3.3: The processing steps for database construction phase. From a LR image, (a), and its corresopnding HR source, (c). The LR image (a) is interpolated to the size of the original image (c), (b) is the interpolated image. (d) Band-pass filtered and contrast normalized version of (b). (e) High-pass filtered version of (c). In the training set, corresponding pairs of patches from (d) and contrast normalized version of (e) are stored.

3.2.1.2 Super-resolution reconstruction phase

For a given LR image $\mathbf{Y}$ (see (3.4)), estimating the corresponding HR image X is performed in tree steps as follows:

- *Step 1. Pre-processing:* The LR input $\mathbf{Y}$ is first pre-processed exactly as in the database construction phase. First, $\mathbf{Y}$ is expanded to have the same size

as the desired HR image $\mathbf{X}$ using the interpolation algorithm Q, this image is denoted by $\mathbf{Y}_{up}$,

$$\mathbf{Y}_{up} = Q(\mathbf{Y}). \tag{3.7}$$

Next, the interpolated image is passed through a bandpass filter and a contrast normalization step, resulting in a mid-frequency image with the contrast normalized. Denote by $\mathbf{y}^m$ the pre-processed image, $\mathbf{y}^m$ is considered as the middle frequency layer of $\mathbf{X}$.

The main aim of the super-resolution reconstruction phase is to estimate the missing high-frequency layer $\mathbf{x}^h$ of $\mathbf{X}$ from $\mathbf{y}^m$ with the help of the training data $(\mathbf{P}_m, \mathbf{P}_h)$. Here, the high-frequency layer $\mathbf{x}^h$ is found based on maximizing the prior probability $Pr(\mathbf{x}^h|\mathbf{y}^m)$.

- *Step 2. Super-resolution on patch:* By considering $\mathbf{y}^m$ as a set of overlapping patches $\{y_i^m\}_{i=1}^N$, the problem of estimating the high-frequency layer $\mathbf{x}^h$ becomes the problem of estimating the set of the corresponding high-frequency patches $\{x_i^h\}_{i=1}^N$ of $\mathbf{x}^h$.

  For an input patch y_i^m, a naive solution for estimating the corresponding patch y_i^h is to find in the database the best candidate (only one candidate) for x_i^h. To this end, a nearest neighbor (1-NN) of y_i^m in the database $\mathbf{P}_m$ is first found,

$$\mu_i^m = \underset{u_k^m \in \mathbf{P}_m}{\arg\min}\, d\left(y_i^m, u_k^m\right), \tag{3.8}$$

  where $d\left(y_i^m, u_k^m\right)$ is a criterion for determining the nearest neighbor such as $d\left(y_i^m, u_k^m\right) = \|y_i^m - u_k^m\|_2^2$. Then, with the co-occurrence prior, the corresponding patch $\mu_i^h \in \mathbf{P}_h$ is considered as the best estimation of x_i^h, which is used as the missing x_i^h for reconstruction. This is a simple and straightforward solution. However, the performance of this 1-NN solution heavily depends on the example images as well as the criteria for determining the nearest neighbor. Moreover, this simple solution will produce disturbing artifacts due to noise and the ill-posed nature of super-resolution [ED07].

  To improve the robustness of 1-NN strategy, Freeman *et al.* [FP00, FJP02] proposed to use the Markov Network (MN) to model the relationships between neighboring patches as well as patches from different frequency layers (see Figure 3.4). More specifically, the Markov network model is used to select the best high-resolution patch found by K-nearest neighbors that has best compatibility with adjacent patches. The network is drawn as nodes connected by lines, which indicate statistical dependencies (see Figure 3.4). In this model, one node of the Markov network is assigned to each image patch. For this

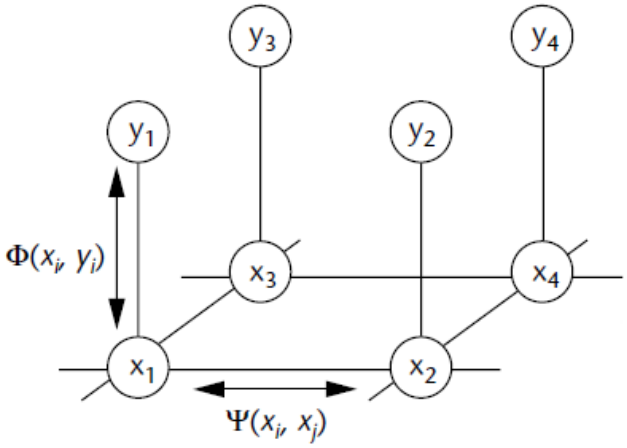


Figure 3.4: Markov network model for super-resolution problem. The low-resolution patches at each node y_i are the observed input. The high-resolution patch at each node x_i is the quantity we want to estimate. Lines in the graph indicate statistical dependencies between nodes.

Markov network, the joint probability has the factorized form [FP00],

$$Pr(\mathbf{x}^h|\mathbf{y}^m) = \frac{1}{Z}\prod_{(i,j)\in E}\psi(x_i^h, x_j^h)\prod_i \phi(x_i^h, y_i^m), \tag{3.9}$$

where Z is a normalization constant such that the probabilities sum to one, E is the set of edges in the Markov network denoted by the neighboring nodes, x_i^h and x_j^h, ψ and ϕ are potential functions. In [FP00], these functions are referred to as compatibility functions which are learned from training data.

The main idea of the Markov network model of Freeman *et al.*'s [FP00] is to estimate high-frequency patches $\{x_i^h\}_{i=1}^N$ by determining a subset of N high-frequency patches $\{\mu_i^h \in \mathbf{P}_h\}_{i=1}^N$ of $\mathbf{P}_h$ such that

$$\{\mu_i^h\}_{i=1}^N = \underset{\{u_i^h\}_{i=1}^N \subset \mathbf{P}_h}{\arg\max} \prod_{(i,j)\in E}\psi(u_i^h, u_j^h)\prod_i \phi(u_i^h, y_i^m). \tag{3.10}$$

In this method, the compatibility function, $\psi(u_i^h, u_j^h)$ is computed by

$$\psi(u_i^h, u_j^h) = \exp -\frac{\|O_{ij}(u_i^h) - O_{ji}(u_j^h)\|_2^2}{2\beta_1^2} \tag{3.11}$$

where β_1 is a positive parameter, O_{ij} is an operator which extracts a vector consisting of the pixels of patch u_i^h in the overlap region between patches u_i^h and u_j^h (see Figure 3.5). On the other hand, the authors assumed that an example patch u_i^h is compatible with an observed image patch y_i^m if its

Figure 3.5: The patch-patch compatibility function is computed from the sum of squared pixel differences in the overlap region.

associating patch $u_i^m \in \mathbf{P}_m$, matches y_i^m. Thereby, the function $\phi(u_i^h, y_i^m)$ is specified by

$$\phi(u_i^h, y_i^m) = \exp -\frac{\|y_i^m - u_i^m\|_2^2}{2\beta_2^2}, \tag{3.12}$$

where β_2 is a positive constant.

We can see that (3.10) minimizes the following objective function for the whole network

$$\{\mu_i^h\}_{i=1}^N = \underset{\{u_i^h\}_{i=1}^N \in \mathbf{P}_h}{\arg\min} \sum_i \left(\|y_i^m - u_i^m\|_2^2 + \lambda \sum_{j:(i,j)\in E} \|O_{ij}(u_i^h) - O_{ji}(u_j^h)\|^2 \right), \tag{3.13}$$

where (i, j) denotes an edge in the set E of edges in the Markov network. By comparing with equation (3.8), it can be seen that (3.13) obtains a MAP (Maximum A Priori) estimation of μ_i^h. Hence the result can be more robust.

As shown in [WZG10], exact inference of (3.10) is computationally intractable. In [FP00], the authors proposed to find one feasible solution in two steps which can be summarized as follows. For each of input patches y_i^m $(i = 1, 2, \ldots, N)$, K-nearest neighbors $\{u_k^m\}_{k=1}^K$ of y_i^m are first selected from the training data $\mathbf{P}_m$. The set of K corresponding high-frequency patch $\Omega_i = \{u_k^h\}_{k=1}^K \in \mathbf{P}_h$ are used as the set of candidates for estimating x_i^h at the hidden node of the Markov network. Then, the estimates $\{\mu_i^h\}_{i=1}^N$ of the desired patches $\{x_i^h\}_{i=1}^N$ are determined by,

$$\{\mu_i^h\}_{i=1}^N = \underset{\{u_i^h\}_{i=1}^N \in \mathbf{P}_h,\ u_i^h \in \Omega_i}{\arg\min} \sum_i \left(\|y_i^m - u_i^m\|_2^2 + \lambda \sum_{j:(i,j)\in E} \|O_{ij}(u_i^h) - O_{ji}(u_j^h)\|^2 \right). \tag{3.14}$$

In [FP00], the approximate solution of this problem is found by using a fast, iterative algorithm called *belief propagation* (see [FP00] for more details). The estimated patches μ_i^h are then used as the final x_i^h for reconstruction.

- *Step 3. Reconstruction of the entire HR image:* After step 2, we obtain an estimate $\hat{\mathbf{x}}^h$ for the high-frequency layer $\mathbf{x}^h$ of $\mathbf{X}$. The estimated high frequency image $\hat{\mathbf{x}}^h$ is then applied to the inverse of the contrast normalization that we used in the pre-processing step. The result is added to the interpolated image $\mathbf{Y}_{up}$ to obtain the final estimate of the desired HR image $\mathbf{X}$.

Figure 3.6 shows an experiment of the Freeman *et al.*'s SR method. In this experiment, the LR image (Figure 3.6(b)) is generated from the original image, Figure 3.6(f). Indeed, the original image is blurred by a low-pass filter which is a 7×7 pixel Gaussian filter, normalized to have unit sum, of standard deviation 1 pixel. Then, the result is subsampled by a factor of four in each dimension, to obtain the observed low-resolution. As can be seen, the SR result of the MN-based method (Figure 3.6(e)) is better than that of the 1-NN method (Figure 3.6(d)). However, the processed image generates some artifacts.

3.2.2 Super-resolution via Neighbor Embedding (NE)

The MN-based method of Freeman *et al.* [FP00] proves that using K-NN candidates provide more the robustness than the 1-NN approach. However, instead of selecting one of the candidates from $\Omega_i = \{u_k^h\}_{k=1}^K$ as winner for each of desired patch x_i^h (see (3.13)), an alternative is to find x_i^h as the form

$$x_i^h = \sum_{k=1}^{K} w_k u_k^h \tag{3.15}$$

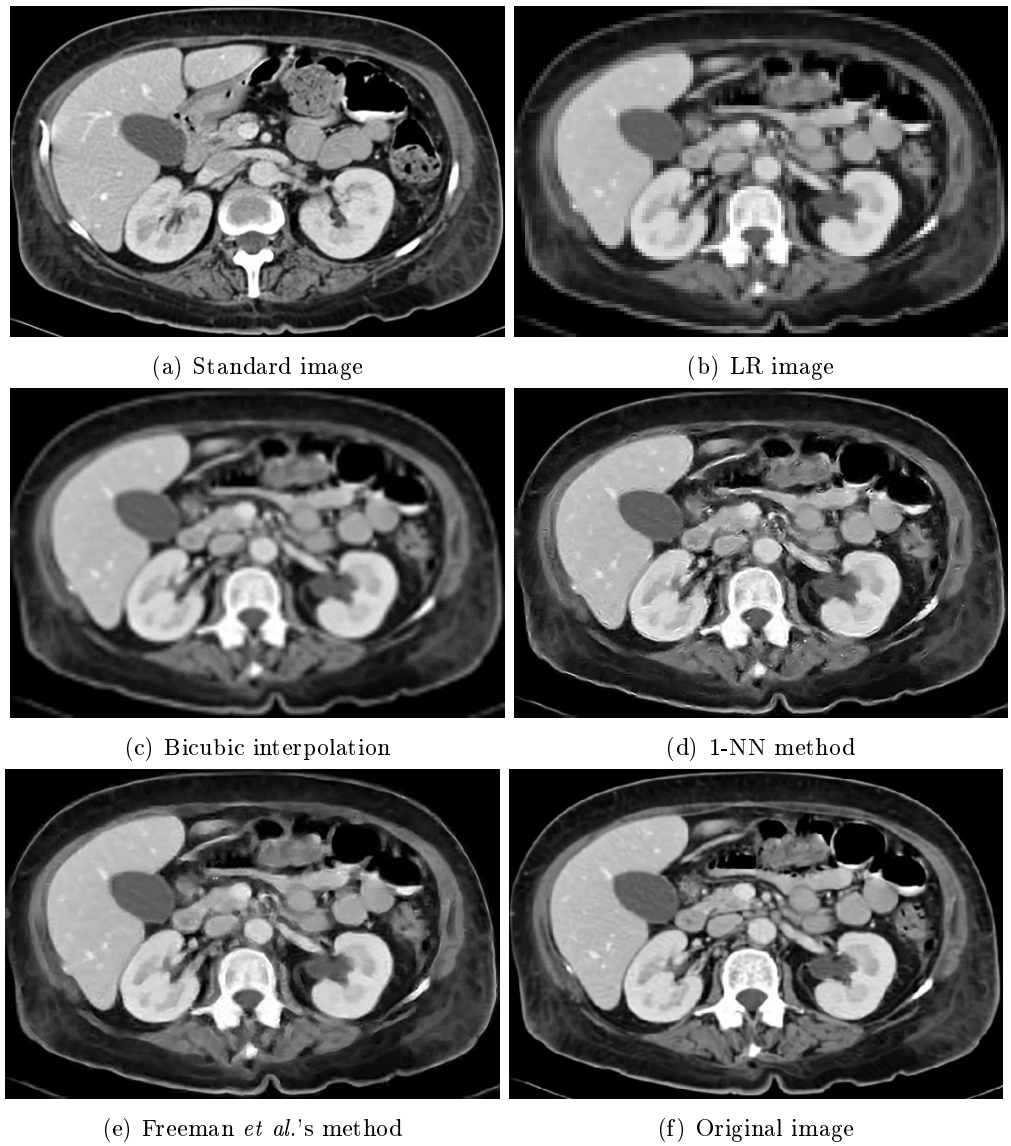
which is a weighted combination of the K candidates. Here, $\{w_k\}$ can be learned using manifold learning techniques. For instance, the Neighbor Embedding (NE) algorithm of Chang *et al.* in [CYX04] adopted the locally linear embedding algorithm to estimate x_i^h.

The NE-based method performs in two independent processes to synthesize HR image patches. In the first process, for each LR input y_i^m, K nearest neighbors $\{u_k^m\}_{k=1}^K$ were selected from the training data using the Euclidean distance, and thus having K candidates $\{u_k^h\}_{k=1}^K$ for the desired output x_i^h. In the second process, the optimal x_i^h weights are obtained by solving a constrained least-squares problem,

$$\{w_k^*\} = \arg\min_{\{w_k\}} \left\| y_i^m - \sum_{k=1}^{K} w_k u_k^m \right\|_2^2, \quad \text{subject to} \quad \sum_{k=1}^{K} w_i = 1. \tag{3.16}$$

The learned weights by (3.16) is then applied to estimate x_i^h by

$$\hat{x}_i^h = \sum_{k=1}^{K} w_k^* u_k^h. \tag{3.17}$$



(a) Standard image (b) LR image

(c) Bicubic interpolation (d) 1-NN method

(e) Freeman *et al.*'s method (f) Original image

Figure 3.6: An experiment of the method of Freeman *et al.* on a CT image of abdomen with the magnification factor of 4. (a) is a training image used to construct the database $(\mathbf{P}_m, \mathbf{P}_h)$, (b) Input LR image, of resolution 135×90 (shown with nearest neighbor interpolation), (c) Bicubic interpolation, (d) 1-NN method, (e) Method of Freeman *et al.*, (f) Original image.

Similar to the Freeman *et al.*'s method above, in the NE method [CYX04], both the LR and the HR patches are represented by feature vectors. However, in the NE method, the feature vectors are defined in another way. More specifically, the training data $(\mathbf{P}_m, \mathbf{P}_h)$ is constructed as follows.

1. The example HR images $\{\mathbf{A}_h\}$ are blurred and subsampled to create the corresponding low-resolution images $\{\mathbf{A}_l\}$ $(\mathbf{A}_l = D_s H \mathbf{A}_h)$.

2. For each of training image pairs $(\mathbf{A}_l, \mathbf{A}_h)$, a set of LR and HR patch pairs $\{(\mathbf{p}_i^l, \mathbf{p}_i^h)\}$ is first extracted. Here, $\mathbf{p}_i^l$ is a $d_1 \times d_1$ patch in $\mathbf{A}_l$ and $\mathbf{p}_i^h$ is the corresponding $d_2 \times d_2$ patch in $\mathbf{A}_h$, where $d_2 = sd_1$ with s is the magnification factor between the LR input $\mathbf{Y}$ and the desired HR image $\mathbf{X}$ (see (3.4)). Then, a corresponding feature vector pair (u_i^m, u_i^h) is defined on each pair $(\mathbf{p}_i^l, \mathbf{p}_i^h)$. Here, u_i^m is defined from the LR patch $\mathbf{p}_i^l$ by using the first-order and second-order gradients of the luminance. It means that we obtain four features for each pixel in $\mathbf{p}_i^l$, and thus u_i^m is a $4d_1^2$ vector which is the concatenation of the features for all pixels within the patch $\mathbf{p}_i^l$. For the HR patch $\mathbf{p}_i^h$, the feature vector u_i^h is defined by

$$u_i^h = \mathbf{p}_i^h - \mathrm{E}(\mathbf{p}_i^h), \tag{3.18}$$

   where $\mathrm{E}(\mathbf{p}_i^h)$ is the mean luminance value of the pixels in $\mathbf{p}_i^h$.

3. The training database $(\mathbf{P}_m, \mathbf{P}_h)$ stores the feature vector pairs (u_i^m, u_i^h) defined from the example HR images.

Now, the NE algorithm for estimating HR image $\mathbf{X}$ can be summarized as follows:

1. The LR input $\mathbf{Y}$ is separated into N overlapping $n_1 \times n_1$ patches $\mathbf{p}_i^l$. The aim is to estimate the N corresponding HR patches $\mathbf{p}_i^h$ in $\mathbf{X}$.

2. For each $\mathbf{p}_i^l$, define its feature vector, y_i^m, in the same way as the vector u_k^m in the database $\mathbf{P}_m$.

3. Estimate the feature vector x_i^h of $\mathbf{p}_i^h$ by (3.17) and (3.16).

4. The HR patch $\mathbf{p}_i^h$ is estimated by

$$\hat{\mathbf{p}}_i^h = \hat{x}_i^h + \mathrm{E}(\mathbf{p}_i^l) \tag{3.19}$$

5. When all HR patches are estimated, the patches $\hat{\mathbf{p}}_i^h$ are put in their proper locations and averaging in overlap regions is performed to get the final HR image.

To date, the NE-based method has been cited more than 350 times thanks to its efficiency and simplicity. Many variant improvements over the NE-based SR method have been proposed [CZ06, CZPH09, FY07]. For example, Chan and Zhang [CZ06] utilized histogram matching to choose more reasonable training images. Considering the importance of both edges and neighborhood size for reconstruction, Chan *et al.* [CZPH09] elaborated on edge detection and feature selection to mitigate the adverse effect of the representation of LR patches. However, this solution highly depends on edges detection as well as neighborhood size selection. In [FY07], a primitive prior is introduced to establish a refined training database. In addition, residual errors related to the reconstructed image patches were estimated to offset the information lost in the local averaging process.

In general, the performance of the NE-embedding methods are limited by the parameter K and the quality of the K candidates. Fixing K for each low resolution patch may result in overfitting or underfitting. On the other hand, the gradient informations are used to define the feature vectors u_i^m and y_i^m which represent geometric structure of the image patches. Unfortunately, this is one of the reasons why the NE method are often less effective in the case LR image is a noisy one. Indeed, gradient of a noise-free image and its noisy version is very different. Thus, there exists a significant difference between the feature vectors of the training database u_i^m and the feature vector of the noisy patch y_i^m. Consequently, this affects the quality of K searched nearest neighbors and hence the quality of the output image.

3.2.3 Super-resolution via sparse representation

As explained in the previous section, the use of a few candidate patches limits the performance of example-based SR method. The NE method is a promising idea except that it carries out two independent processes to synthesize high-resolution (HR) image patches: Searching for K candidates from the database, and estimating the best combinations of the K candidates. Hence, a better idea is to address the two phases simultaneously. This idea have been realized very successful in the sparse-coding (SC)-based methods [YWHM08, YWHM10, YWL+12, WZG10, ZEP10].

The goal of sparse coding is to represent an input vector approximately as a weighted linear combination of a small number of basis vectors called basis atoms. Suppose that the matrix $\mathbf{D} \in \mathbb{R}^{d\times M}$ $(d < M)$ is an over-complete dictionary, in which each column vector is a $d-$dimension atom. Given a vector $y \in \mathbb{R}^d$, its sparse representation can be seen as finding a sparse solution $\alpha = [\alpha_1, \ldots, \alpha_M]^T \in \mathbb{R}^M$ of the following optimization problem:

$$\min_{\alpha} \|\alpha\|_p \quad \text{subject to} \quad \|y - \mathbf{D}\alpha\|_2^2 \leq \varepsilon^2, \tag{3.20}$$

where $\|\alpha\|_p$ is ℓ_p-norm (p is often satisfy $0 \leq p \leq 1$), that is $\|\alpha\|_p = (\sum_i |\alpha_i|)^{\frac{1}{p}}$ with $p > 0$, and $\|\alpha\|_0 = \lim_{p\to 0} \|\alpha\|_p^p$ is the ℓ_0 pseudo-norm which counts the non-zero entries in α.

The dictionary needs to be learned from training data. Given a training data $\{y_i,\ i = 1, \ldots, N\}$, the problem of learning a dictionary for sparse-coding is to solve the following optimization problem:

$$\min_{\mathbf{D},\{\alpha^i\}_{i=1}^N} \sum_{i=1}^{N} \|y_i - \mathbf{D}\alpha^i\|_2^2 + \lambda\|\alpha^i\|_p, \quad \text{s.t.} \ \ \|\mathbf{D}(:,k)\|_2 \leq 1,\ \forall k = 1, \ldots, M, \quad (3.21)$$

where λ is a parameter controlling the sparsity penalty and representation fidelity, and $\mathbf{D}(:,k)$ denotes k^{th} column of $\mathbf{D}$.

The effectiveness of the sparse representation-based methods for image denoising was presented in section 1.5.4. In this section, we present the principal content of the single image SR approach via sparse representation [YWHM08, WZG10, YWHM10, ZEP10, YWL+12].

The SR algorithm via sparse representation is often performed in two phases: *training phase* and *SR phase*.

- *Training phase:* This phase includes the following steps:

  1. Database Construction: Similar to the example-based method described above, a database $(\mathbf{P}_m, \mathbf{P}_h) = \{(u_i^m, u_i^h)\}$ is first constructed from the example HR images $\{\mathbf{A}_h\}$. Each of these patch-pairs undergoes a pre-processing stage that removes the low-frequencies from u_i^h and extracts features from u_i^m.
  2. Dictionary Training: From the database of example vector pairs, $(\mathbf{P}_m, \mathbf{P}_h)$, a coupled dictionary pair $\mathbf{D}_m$ and $\mathbf{D}_h$ is trained such that the sparse representation of $u_i^m \in \mathbf{P}_m$ in terms of $\mathbf{D}_m$ is the same as that of $u_i^h \in \mathbf{P}_h$ in terms of $\mathbf{D}_h$.

  The above training phase is done off-line, producing the two dictionaries, $\mathbf{D}_m$ and $\mathbf{D}_h$, to be used in the super-resolution reconstruction.

- *Super-resolution phase:* Given a test LR image $\mathbf{Y}$ to be super-resolved, we first extract pre-processed patches y_i^m from each location i, and then sparse-code it using the trained dictionary $\mathbf{D}_m$. The found representation α^i is then used to recover the high-resolution patch by multiplying it with $\mathbf{D}_h$, $x_i^h \approx \mathbf{D}_h\alpha^i$. We will describe all the above process is more details.

3.2.3.1 Training Phase

The training phase starts by constructing database $(\mathbf{P}_m, \mathbf{P}_h)$ from the example HR images $\{\mathbf{A}_h\}$. These training images are first pre-processed through some steps:

1. Each of these images is blurred and down-scaled by a factor s to create the corresponding low-resolution image $\mathbf{A}_l$ ($\mathbf{A}_l = D_s H \mathbf{A}_h$).

2. The LR images $\{\mathbf{A}_l\}$ are then scaled up back to the original size by using an interpolation algorithm Q, resulting in the set $\{\mathbf{B}_l\}$ ($\mathbf{B}_l = Q(\mathbf{A}_l) = Q D_s H \mathbf{A}_h$).

3. A pre-processing is applied on the high-resolution images to remove their low-frequencies. This is done by computing the difference images $\mathbf{a}_h = \mathbf{A}_h - \mathbf{B}_l$ (exactly as in the Markov network-based method [FP00]). As for the pre-processing of the low-resolution images, these are filtered using K high-pass filters $f_k, k = 1, 2, \ldots, K$, in order to extract local features that correspond to their high-frequency content. Thus, each LR image $\mathbf{B}_l$ leads to a set of K filtered images, $f_k \otimes \mathbf{B}_l$, for $k = 1, 2, \ldots, K$ ($\otimes$ stands for a convolution). For instance, the first-order and second-order gradients of the luminance used in [YWHM08, YWHM10, ZEP10], mean that four high-pass filters used to extract the derivatives are:

$$f_1 = [-1, 1], \quad f_2 = f_1^T,$$
$$f_3 = [1, -2, 1], \quad f_4 = f_3^T,$$

   where the superscript "T" means transpose.

After the two pre-processing steps described above, the database $(\mathbf{P}_m, \mathbf{P}_h) = \{(u_i^m, u_i^h), i \in \mathcal{I}\}$ is ready to be established. Considering only locations $i \in \mathcal{I}$, $u_i^h \in \mathbb{R}^d$ is a vectorized patch of size $\sqrt{d} \times \sqrt{d}$ pixels from the high-frequency image $\mathbf{a}_h$. To establish the associating vector u_i^m, the corresponding patches are first extracted from the same locations in the K filtered images $f_k \otimes \mathbf{B}_l$ and using the same size $\sqrt{d} \times \sqrt{d}$. Then, the K vectors corresponding to K such patches are merged into one vector u_i^m (u_i^m of length dK).

In [YWHM08], [WZG10], the training sets $\mathbf{P}_m$ and $\mathbf{P}_h$ are directly used as two dictionaries. In the later versions [YWHM10], [YWL+12] and [ZEP10], this approach is further extended by learning a coupled dictionary $(\mathbf{D}_m, \mathbf{D}_h)$ instead of using the training sets, allowing the algorithm to be much more efficient. The dictionary pair $\mathbf{D}_m$, $\mathbf{D}_h$ are trained such that the sparse representation of u_i^m in terms of $\mathbf{D}_m$ is the same as that of u_i^h in terms of $\mathbf{D}_h$.

The dictionary-learning is started by training the dictionary $\mathbf{D}^m$ from the set of LR training patches $\mathbf{P}_m = \{u_i^m, i \in \mathcal{I}\}$ using the sparse-coding model (3.21). As

a side product of this training, we obtain the sparse representation vector α^i that corresponds to the training patch u_i^m. After constructing $\mathbf{D}_m$, the HR dictionary construction $\mathbf{D}_h$ is proceeded. Specifically, $\mathbf{D}_h$ is defined by solving the following problem

$$\mathbf{D}_h = \arg\max_{\mathbf{D}_h} \sum_{i \in \mathcal{I}} \|u_i^h - \mathbf{D}_h \alpha^i\|_2^2. \tag{3.22}$$

3.2.3.2 Super-resolution Phase

We are given a test low-resolution image $\mathbf{Y} = D_s H \mathbf{X} + V$ to be magnified. The high-frequency larger $\mathbf{x}_h$ that is represented as the set of $\sqrt{d} \times \sqrt{d}$ patches $\{x_i^h\}_{i=1}^N$ of $\mathbf{X}$ is estimated by the following steps:

1. The LR image $\mathbf{Y}$ is scaled up by a factor of s using the interpolation algorithm Q, resulting with $\mathbf{Y}_{up}$.

2. The image $\mathbf{Y}_{up}$ is then filtered by using the same K high-pass filters which were used for feature extraction in the training, and obtain $\{f_k \otimes \mathbf{Y}_{up}\}_{k=1}^K$.

3. To estimate HR patch x_i^h at location i, first, extract from each image $f_k \otimes \mathbf{Y}_{up}$ a patch of size $\sqrt{d} \times \sqrt{d}$ at the same location i, resulting in a set of K patches which define the features of the input LR patch. These K patches are to be concatenated to form a patch vector y_i^m.

4. The sparse representation vectors α^i of y_i^m in terms of $\mathbf{D}_m$ is found by solving the sparse-coding problem:

$$\alpha^i = \arg\min_{\alpha} \ \|y_i^m - \mathbf{D}_m \alpha\|_2^2 + \lambda \|\alpha\|_p. \tag{3.23}$$

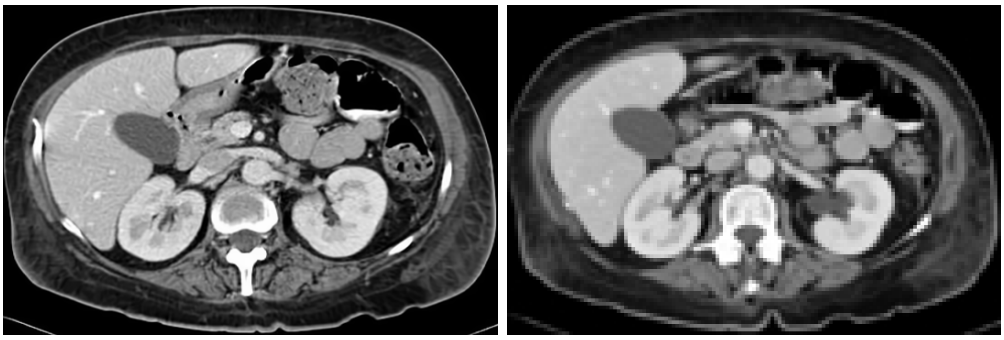
5. Once α^i is obtained, the corresponding output can be reconstructed as $\hat{x}_i^h = \mathbf{D}_h \alpha^i$.

6. The high-frequency layer $\mathbf{x}_h$ is constructed by putting $\hat{x}_i^h$ to their proper location and averaging in overlap regions.

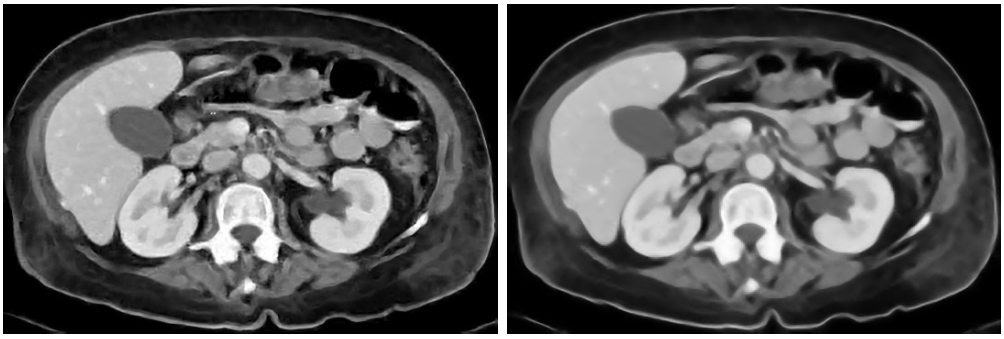
Finally, the HR estimate of $\mathbf{X}$ is obtained by adding $\mathbf{x}_h$ to $\mathbf{Y}_{up}$,

$$\hat{\mathbf{X}} = \mathbf{Y}_{up} + \mathbf{x}_h. \tag{3.24}$$

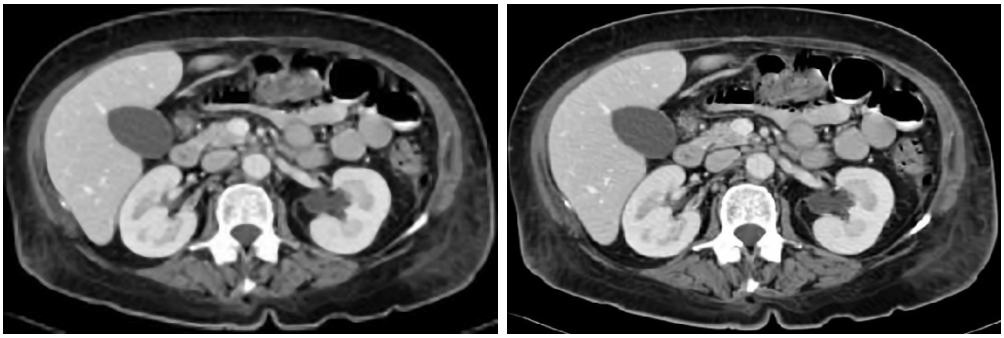
Figure 3.7 shows the SR results of different methods which are performed on a CT image of abdomen with the magnification factor of $s = 4$, for subjective comparison. As can be seen, the SC-based method [YWL+12] (Figure 3.7(e)) gives the better result compared with the others.



(a) Standard image (b) LR image



(c) MN-based method [FP00] (d) NE-based method [CYX04]



(e) SC-based method [YWL+12] (f) Original image

Figure 3.7: Comparison of SR results on CT image of abdomen with the magnification factor of $s = 4$ by different methods. (a) Example HR image used to construct the database $(\mathbf{P}_m, \mathbf{P}_h)$, (b) Input LR image of resolution 135×90 (shown with nearest neighbor interpolation), (c) MN-based method [FP00], (d) NE-based method [CYX04], (e) SC-based method [YWL+12], (f) Original image. Image courtesy of Dr. Jean-Marie Rocchisani (Avicenne University Hospital, Bobigny, France).

3.2.4 Conclusion

We have presented through this section a review of example-learning-based super-resolution methods. Here, we only present three principal methods including the Markov Network-based (MN) method of Freeman *et al.* [FP00, FJP02] (this is also the first example-based SR method proposed), the Neighbor Embedding-based (MN) method [CYX04], and the Sparse-coding-based methods [YWHM08, YWHM10, ZEP10, YWL+12]. The MN-based method determines the output HR patch by selecting only one winner from the K HR candidates such that the pixel difference in overlapped regions is minimized. Thus, this method is often unstable, due to over- or under-fitting. With inappropriate examples, the processed image generates many artifacts [WZG10]. The NE-based method estimates the HR patch as an affine combination of the K HR candidates. Thus, compared to the MN-based method, the artifacts of the NE-based method may be reduced. However, using a fixed number K of neighbors for reconstruction often results in blurring effects (see Figure 3.7(d)). Compared to the NE-based method with fixed K neighbors, SC-based methods adaptively chooses the fewest necessary supports for reconstruction, avoiding hence overfitting.

As shown in [YWHM10], most example-learning-based super-resolution algorithms assume that the input images are clean and free of noise, an assumption which is likely to be violated in real applications such as medical imaging. To deal with noisy data, previous algorithms usually divide the recovery process into two disjoint steps: first denoising and then super-resolution. However, the results of such a strategy depend on the specific denoising technique, and any artifacts during denoising on the low-resolution image will be kept or even magnified in the latter super-resolution process. In [YWHM10], it is shown that the SC-based methods can be applied with noisy data. However, logically, we can see that the NE-based and SC-based methods use the first- and second-order derivatives as the feature for the low-resolution patch. This is not an effective way to process with noisy images, due to existence of significant difference between derivative of noisy-free image and derivative of noisy image.

Motivated by this challenge to deal with super-resolution for noisy image, we introduce in the following section two novel example-learning-based methods for single-image super-resolution with the help of given standard images. The first method is a geometric one published in [TLR+12a] where we use the projection of a point onto the convex hull of a finite set of points in high-dimension vector space to perform super-resolution on image patches. In the second method [TLR+12b], super-resolution on image patches is performed by using a nonnegative sparse representation model. Both of two methods may be applied effectively for noise-free

images as well as for noisy images.

3.3 The Proposed Super-Resolution Methods

In this section, we will present our two example-learning based SR methods: the Super-resolution by Projection onto Convex Hull (SPOCH) and the Super-Resolution by Optimal Weight model (SROW). Let us begin by recalling the image degradation model (3.4). Assume that we are given a LR image $\mathbf{Y}$, that is generated from a HR image $\mathbf{X}$ by the model

$$\mathbf{Y} = D_s H \mathbf{X} + V,$$

where H is the blur operator, D_s is the decimation operator with factor s, and V is the additive noise component. Our aim is to estimate the unknown HR image $\mathbf{X}$ from $\mathbf{Y}$ with the help of a given set standard image images $\{\mathbf{A}_h\}$.

Similar to the other example-learning-based SR methods, here, each image can be represented as an arranged set of overlapping patches, and then super-resolution is performed on each patch. The LR image $\mathbf{Y}$ will be represented as a set of N overlapping image patches, that is

$$\mathbf{Y} = \{y_i^l,\, i = 1, 2, \ldots, N\}. \tag{3.25}$$

Here, y_i^l is a $\sqrt{m} \times \sqrt{m}$ image patch, N is the number of patches generated from the image $\mathbf{Y}$. N depends on the patch size and the sliding distance between adjacent patches. Similarly, the high-resolution image $\mathbf{X}$ can be also represented as a set of the same number N of paired HR patches $\{x_i^h,\, i = 1, 2, \ldots, N\}$. The size of x_i^h is set to be $\sqrt{n} \times \sqrt{n}$ where $\sqrt{n} = s\sqrt{m}$. The LR patch and the HR patch is related by

$$y_i^l = D_s H x_i^h + \eta_i, \tag{3.26}$$

where η_i is the noise in the i^{th} patch. For the sake of simplicity, we assume that

$$\eta_i \sim \mathcal{N}(0, \sigma_i^2) \tag{3.27}$$

is Gaussian, white, zero-mean, and i.i.d., with variance σ_i^2. Thus, we can consider y_i^l as a LR version of the HR patches x_i^l. In the remaining of this chapter, image patches are rearranged as vectors, for example, $x_i^h \in \mathbb{R}^n$ and $y_i^l \in \mathbb{R}^m$.

In order to estimate $\mathbf{X}$, the proposed algorithms are also performed into two phases: *database construction phase* and *super-resolution reconstruction phase.*

- In the first phase, a database of HR and LR patch pairs $(\mathbf{P}_m, \mathbf{P}_h) = \{(u_i^m, u_i^h), i \in \mathcal{I}\}$ is constructed from a given set of the standard images which are taken at nearly the same locations as the LR image $\mathbf{Y}$.

- The super-resolution reconstruction phase consists of the HR patch reconstruction and the reconstruction of the entire HR image.

In order to obtain a good database, the selection of these example images should be such that they would contain a variety of intensities as well as shapes and very little noise. Since the standard images and the LR image are often taken from nearby locations and thanks to the repetition of local structures of images, small image patches tend to recur many times inside these images. Thereby, we can believe that for a given LR image patch in $\mathbf{Y}$, a rich amount of similar patches can be extracted from the database. This property is very important in the success of the proposed methods

3.3.1 The first proposed method: Super-resolution by Projection Onto Convex Hull (SPOCH)

The first proposed method is a geometric method, where the projection of a point onto the convex hull of a finite set of points is for the first time applied to the problem of image super-resolution [TLR+12a]. The method can be considered as a K-nearest neighbor-based approach. Similar to the other K-nearest neighbor-based methods ([FP00, CYX04]), SR perform for a given input LR patch consist of extracting a subset of K best HR candidates in the database for the estimation of the output HR patch. However, unlike the other methods, in our proposed method, the image patches are regarded as points in a high dimensional vector space. To estimate the output for a given input LR point, we assume that the HR point lies on the convex hull of the K candidate points and is closest to the input point. Consequently, the projection of the input point onto the convex hull of the candidate points is the HR output point.

Before going into more details of the proposed method, we recall in Section 3.3.1.1 the concept of convex hull of a finite set of points and a geometric algorithm for finding the projection of a point onto convex hull. The application of the Projection Onto Convex Hull (POCH) to image SR will be presented in Section 3.3.1.2

3.3.1.1 Convex Hull and Projection Onto Convex Hull (POCH)

Let $\Omega = \{x_i, i \in I\}$ be a finite set of points in $\mathbb{R}^n$, where I is a set of indices. The convex hull of Ω, denoted by $\text{conv}(\Omega)$, is the smallest convex set containing Ω. It consists exactly of all convex combinations of elements of Ω, that is:

$$\text{conv}(\Omega) = \{\sum_{i\in I} \alpha_i x_i \mid x_i \in \Omega, \alpha_i \geq 0, \sum_{i\in I} \alpha_i = 1\}. \tag{3.28}$$

In $\mathbb{R}^n$, the projection of a point y onto the convex hull of a finite set of points Ω, denoted by $\mathbf{proj}_{\text{conv}(\Omega)}(y)$, is a point on $\text{conv}(\Omega)$ and the closest to y,

$$\mathbf{proj}_{\text{conv}(\Omega)}(y) = \underset{x \in \text{conv}(\Omega)}{\text{argmin}} \|y - x\|_2. \tag{3.29}$$

To find the projection $\mathbf{p} = \mathbf{proj}_{\text{conv}(\Omega)}(y)$, we propose to adapt in this work a fast iterative geometric algorithm, namely the Schlesinger-Kozinec algorithm, introduced in [FH03] by Franc and Hlaváč. This algorithm was introduced by the authors for finding optimal separating hyperplane between two linearly separable sets. The algorithm is defined as follows.

Schlesinger-Kozinec algorithm for POCH[FH03]
Input: y, $\Omega = \{x_i, i \in I\}$, and the precision ε.

1. *Initialization:* Set $\mathbf{p}$ to any point $x_i \in \Omega$.

2. *Stopping condition:* Find $x_t \in \Omega$ such that $x_t = \underset{x_i \in \Omega}{\text{argmin}}\ m(x_i)$, where

$$m(x_i) = \frac{\langle x_i - y, \mathbf{p} - y \rangle}{\|\mathbf{p} - y\|_2}. \tag{3.30}$$

   If the ε-optimality condition

$$\|y - \mathbf{p}\|_2 - m(x_t) < \varepsilon \tag{3.31}$$

   holds then $\mathbf{p}$ defines the ε-solution, $\mathbf{proj}_{\text{conv}(\Omega)}(y) = \mathbf{p}$.
   Otherwise, go to *Step 3.*

3. *Adaptation:* Compute the new value of vector $\mathbf{p}$ as

$$\mathbf{p}^{new} = (1 - \beta)\mathbf{p} + \beta x_t, \tag{3.32}$$

   where

$$\beta = \underset{\beta \in (0,1]}{\arg\min} \|y - ((1 - \beta)\mathbf{p} + \beta x_t)\|_2. \tag{3.33}$$

   Continue with *Step 2.*

The Schlesinger-Kozinec algorithm begins iterations from an arbitrary point $x \in \text{conv}(\Omega)$. For instance it can start from any point $x_i \in \Omega$. In Step 2, $m(x_i)$ is the the algebraic distance from x_i to hyperplane $(H) : \langle x - y, \mathbf{p} - y \rangle = 0$ ((H) goes through point y, and $\mathbf{p} - y$ is its normal vector). x_t is a point in Ω, that has the shortest algebraic distance to (H). If point $\mathbf{p}$ satisfies condition (3.31) then $\mathbf{p}$ defines the ε-solution, and $\mathbf{proj}_{\text{conv}(\Omega)}(y) = \mathbf{p}$. Otherwise $\mathbf{p}$ is adapted so that its new value

$\mathbf{p}^{new}$ is closest point to y and lies on the abscissa between the old $\mathbf{p}$ and the point x_t. The number β determines how much we must move from $\mathbf{p}$ towards the point x_t to define the new value the $\mathbf{p}^{new}$. As proposed in [FH03], the solution of the task (3.33), β, is explicitly computed as

$$\beta = \min\left\{1, \frac{\langle x_t - \mathbf{p}, y - \mathbf{p}\rangle}{\|x_t - \mathbf{p}\|_2^2}\right\}. \tag{3.34}$$

The adaptation rule (3.32) ensures that the vector $\mathbf{p}$ remains in the convex hull $\mathrm{conv}(\Omega)$ and the distance $\|y - \mathbf{p}\|_2$ monotonically decreases. Iteration is performed until the ε-optimality criterion is satisfied, and we obtain $\mathbf{proj}_{\mathrm{conv}(\Omega)}(y) = \mathbf{p}$.

3.3.1.2 The proposed SPOCH method

Similar to the other example-learning-based SR methods, here, each image can be represented as a set of overlapping patches, and then super-resolution is performed on each patch. In order to estimate $\mathbf{X}$, the proposed algorithm is also performed into two phases: database construction phase and super-resolution reconstruction phase.

In the first phase, a database of HR and LR patch pairs $(\mathbf{P}_l, \mathbf{P}_h) = \{(u_i^l, u_i^h), i \in \mathcal{I}\}$ is constructed from the example standard images. In the super-resolution phase, the estimation of the HR image $\mathbf{X}$ is performed via three steps as follows:

1. *Search for HR candidate patches:* For each LR input patch y_i^l (at position i of image $\mathbf{Y}$), find in the database $\mathbf{P}_l$ a subset of K LR patches $\Omega_i^l = \{u_k^l\}_{k=1}^K$ which are the most similar to the input y_i^l. Then, the set of the corresponding HR patches $\Omega_i^h = \{u_k^h\}_{k=1}^K$ are used to estimate the HR output x_i^h.

   In this step, to determine the nearest set Ω_i^l, we propose to use a new criteria to measure the similarity between image patches by combining the Euclidian distance with a statistical criterion. Using this criterion the reliability of the set of HR candidate patches Ω_i^h is improved.

2. *Super-resolution by POCH:* Using the POCH algorithm (see Section 3.3.1.1), the desired HR patch is estimated from the LR patch y_i^l and the HR candidate patches u_i^h.

3. *Reconstruction of the entire HR image:* The desired HR image $\mathbf{X}$ is reconstructed using the estimated HR patches.

These phases are detailed in the next subsections.

3.3.1.2.1 Database Construction

In this work the database $(\mathbf{P}_l, \mathbf{P}_h)$ is constructed in the steps as follows:

1. The example HR images $\mathbf{A}_h$ are sub-sampled to generate the corresponding LR images $\mathbf{A}_l$,
$$\mathbf{A}_l = D_s H \mathbf{A}_h. \tag{3.35}$$

2. For each training image pair $(\mathbf{A}_l, \mathbf{A}_h)$, a set of high- and low-resolution patch pairs is generated, that is $\{(u_i^l, u_i^h), i \in \mathcal{I}_k\}$, where
$$u_i^l = D_s H u_i^h. \tag{3.36}$$
The training database $(\mathbf{P}_l, \mathbf{P}_h) = \{(u_i^l, u_i^h), i \in \mathcal{I}\}$ is the union of such sets. Note that $u_i^l \in \mathbb{R}^m$ and $u_i^h \in \mathbb{R}^n$ are the vectorized noise-free patches, u_i^h is regarded as a HR patch and a_i^l is the corresponding LR version.

Unlike the other methods, where the vector pairs (u_i^l, u_i^h) present the features of the image patches, we use directly the luminance values of the pixels in the image patch to define the database.

3.3.1.2.2 Super-resolution phase

This phase is realized as follows:

1. **Search for candidate HR patches:**
For a given LR input patch y_i^l, we need to search in the database of HR patches $\mathbf{P}_h$ a subset Ω_i^h of K best candidates for x_i^h. In order to do that, we first find in $\mathbf{P}_l$ a subset of K_0 LR patches u_k^l which are the nearest to y_i^l in terms of Euclidean distance (that means $\|u_k^l - y_i^l\|_2$ closest to 0), using the k-NN tree algorithm proposed by Friedman *et al.* [FBF77]. Then, we refine this search by using another criterion established from the statistical properties on residual image, to keep only K $(K < K_0)$ candidates among the K_0 candidates.

   Specifically, since $y_i^l = D_s H x_i^h + \eta_i$, we have $y_i^l - D_s H x_i^h = \eta_i$. Therefore, a HR patch u_k^h in the database $\mathbf{P}_h$ is considered as the best candidate for x_i^h (i.e. $u_k^h \simeq x_i^h$) if
$$y_i^l - D_s H u_k^h \simeq y_i^l - D_s H x_i^h = \eta_i. \tag{3.37}$$
   Since the noise component η_i is supposed to be Gaussian noise, $\eta_i \sim \mathcal{N}(0, \sigma_i^2)$ (see (3.27)). Thus,
$$y_i^l - u_k^l \simeq \mathcal{N}(0, \sigma_i^2). \tag{3.38}$$
$$\Rightarrow \mathrm{E}(y_i^l - u_k^l) \simeq 0, \text{and } \mathrm{Var}(y_i^l - u_k^l) \simeq \sigma_i^2 \tag{3.39}$$
$$\Rightarrow d_k = |\mathrm{E}(y_i^l - u_k^l)| + |\mathrm{Var}(y_i^l - u_k^l) - \sigma_i^2| \simeq 0. \tag{3.40}$$

Criteria (3.40) represents in some extent the statistical similarity criteria using the difference image $y_i^l - u_k^l$.

Therefore, from the set of K_0 the nearest neighbors of y_i^l, we choose K ($K < K_0$) elements which have d_k closest to 0. Denote this set by $\Omega_i^l = \{u_k^l\}_{k=1}^K$, the set of K HR candidates for the output x_i^h is defined as

$$\Omega_i^h = \{u_k^h | u_k^l \in \Omega_i^l\}. \tag{3.41}$$

We will present the estimation of x_i^h from Ω_i^h and y_i^l in what follows.

2. **Image patch super-resolution by POCH:**
Suppose that for each given input LR patch y_i^l, a set Ω_i^h of K candidate HR patches has been found from the database. In order to obtain a good estimation $\hat{x}_i^h$ of x_i^h, a fundamental constraint is that $\hat{x}_i^h$ should be consistent with the input y_i^l. This can be expressed by minimizing the distance

$$\|D_s H \hat{x}_i^h - y_i^l\|. \tag{3.42}$$

Moreover, $\hat{x}_i^h$ should be similar to the HR candidates u_i^h in Ω_i^h. Accordingly, we can find $\hat{x}_i^h$ as a weighted mean of the HR candidates u_k^h:

$$\hat{x}_i^h = \sum_{k=1}^{K} \alpha_k u_k^h, \tag{3.43}$$

where $u_k^h \in \Omega_i^h$, the coefficients α_k satisfy the condition: $\alpha_k \geq 0$, $\forall k$ and $\sum_{k=1}^K \alpha_k = 1$.

Denote by y_i^* a patch obtained by interpolating y_i^l with the factor s (here, we use the nearest interpolation), resulting in y_i^* which has the same size as HR patches u_k^h. Intuitively, from (3.43) we can see that the estimated HR patch $\hat{x}_i^h \in \text{conv}(\Omega_i^h)$ and $\hat{x}_i^h$ is closest to y_i^*. Therefore, we propose to use the projection of y_i^* onto the convex hull of the candidate set Ω_i^h as the estimation for x_i^h:

$$\hat{x}_i^h = \mathbf{proj}_{\text{conv}(\Omega_i^h)}(y_i^*). \tag{3.44}$$

This is done by applying the POCH algorithm in Section 3.3.1.1

3. **Reconstruction of the entire HR image:**
For each LR patch y_i^l as input, our aim is to find the closest estimation of x_i^h. Suppose that $\hat{x}_i^h$ is an estimation of x_i^h. The final HR image $\hat{\mathbf{X}}$ can be reconstructed from $\{\hat{x}_i^h\}_{i=1}^N$. Then, we put the estimated HR patches $\hat{x}_i^h$ in their proper locations and perform averaging in overlap regions to get the

image $\hat{\mathbf{X}}$. This allows enforcing then the consistency between neighboring patches.

If the LR image Y_l is not corrupted by noise, i.e. $\mathbf{Y} = D_s H\mathbf{X}$, we can get better results of reconstruction using the back-projection technique. For that, the final HR image is determined by

$$\mathbf{X}^{final} = \underset{\mathbf{X}}{\operatorname{argmin}} \|\mathbf{X} - \hat{\mathbf{X}}_h\|_2^2 \quad s.t. \quad D_s H\mathbf{X} = \mathbf{Y}. \tag{3.45}$$

The back-projection method [IS93] is used to solve this problem. The update equation for this iterative algorithm is

$$\mathbf{X}_{t+1} = \mathbf{X}_t + ((\mathbf{Y} - D_s H\mathbf{X}_t) \uparrow_s) * p, \tag{3.46}$$

where $\mathbf{X}_t$ is the estimate of the HR image at the t-th iteration, $\uparrow_s$ denotes up-scaling by factor s, p is a back-projection filter. Here we use the symmetric Gaussian filter of size 5 with standard deviation 1.

3.3.1.3 Performance evaluation

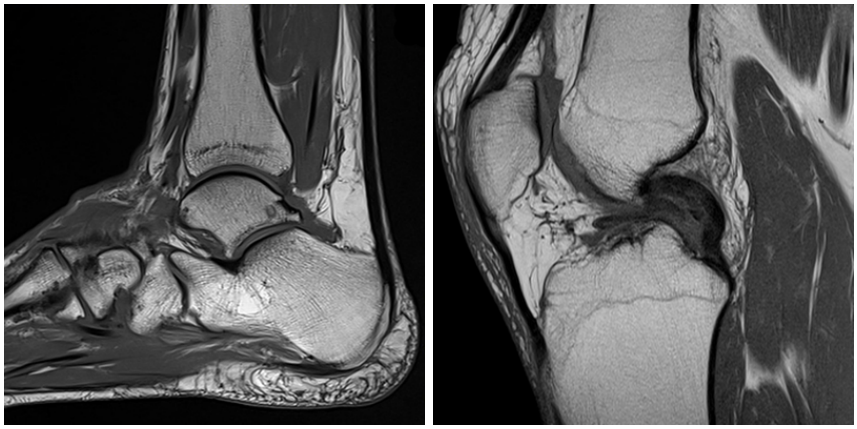
To demonstrate the robustness of the proposed SPOCH method, we compare it with the bicubic interpolation and two other nearest neighbors-based methods: the Markov Network-based method (MN) of Freeman *et al.* [FJP02][1] and the Neighbor Embedding-based method (NE) of Chang *et al.* [CYX04][2]. In order to obtain an objective comparison, we use two quality metrics: the PSNR and the SSIM. The experiments are performed on both noise-free images and noisy images. The example databases of the methods MN and NE are established with the same standard images used for our SPOCH method.

For the SPOCH method, in all experiments, super-resolution is performed with magnification factor of $s = 2$. For larger magnification factors, it can be performed multiple times. Default size of the HR and LR patches are 7×7 ($n = 49$) and 4×4 ($m = 16$), respectively. The HR patches are randomly extracted from the standard images to build the database. The parameter K_0 is set to 100. We set the number of nearest neighbors K to 10 for all of our experiments.

Here, we report some experimental results on five test images including three CT images of size 540×360 and two MRI images of size 400×400 (as illustrated in Figure 3.8). For each of the test images, we use another image taken at nearly the same location as the LR image to establish the database of 50000 HR and LR patch pairs. The standard images are presented in Figure 3.9. The test images are

[1] Matlab code: http://research.microsoft.com/en-us/um/people/celiu/
[2] Matlab code: http://www.jdl.ac.cn/user/hchang/publication.htm

(a) (b) (c)

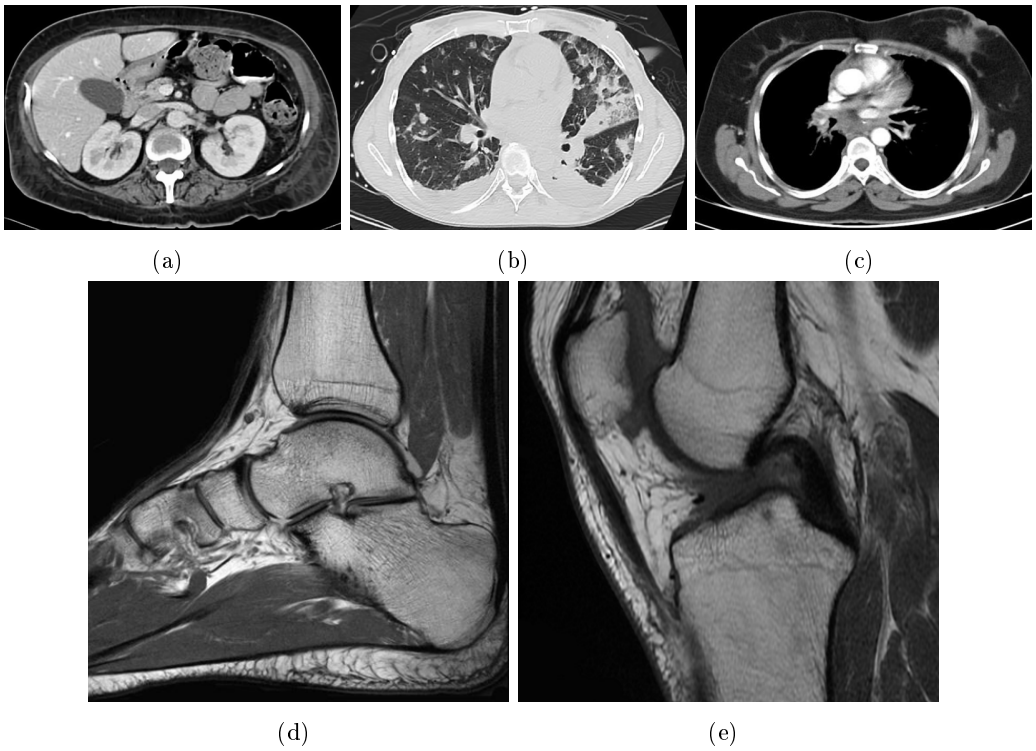


(d) (e)

Figure 3.8: Test HR images: (a) CT image of abdomen (source: http://radiopaedia.org/), (b) CT image of lung (source: http://radiopaedia.org/), (c) CT image of thorax (source: http://radiopaedia.org/), (d) MRI of ankle (source: http://www.cedars-sinai.edu/), (e) MRI of knee (source: http://healthcare.siemens.com/).

blurred by a 7×7 Gaussian filter with standard deviation 1 and downsampled by a decimation factor of s, and then added with Gaussian white noise of standard deviation σ to produce the corresponding LR images. The experiments are performed with magnification factor $s = 4$ and with three levels of noise $\sigma = 0$, 5 and 10. The objective results of experiments are reported in Table 3.1. As can be seen, our method yields significant PSNR and SSIM gains over the other methods, especially in the cases of $\sigma = 5$ and $\sigma = 10$ (LR images are corrupted by noise).

For subjective comparison, we show in Figure 3.10 the super-resolution results obtained by the bicubic interpolation method, the MN method [FJP02], the NE method [CYX04] and the proposed method on the knee MRI image (Figure 3.8(e)). In this figure, the input LR image is a noisy image ($\sigma = 10$) of size 100×100. We can observe that some methods are seriously affected when the image to enhance is noisy, such as the interpolation method (see Figure 3.10(b)) and the NE-based method of Chang *et al* [CYX04] (see Figure 3.10(c)). As to the MN-based method



(a) (b) (c)

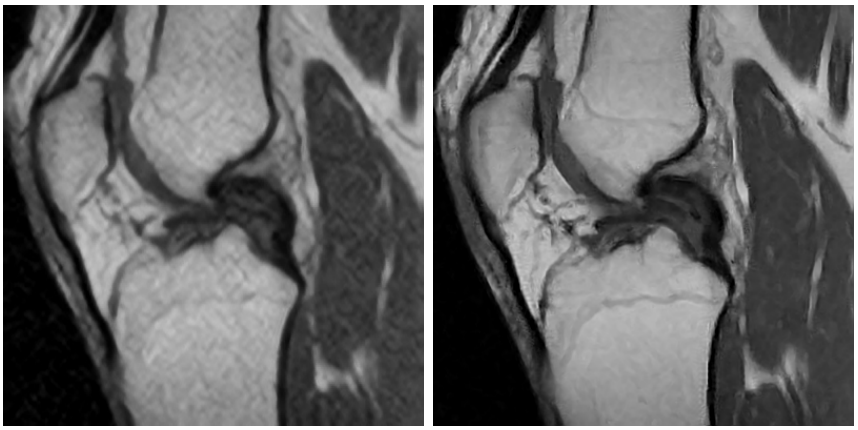
(d) (e)

Figure 3.9: Standard images used to construct the database: (a) CT image of abdomen (source: http://radiopaedia.org/), (b) CT image of lung (source: http://radiopaedia.org/), (c) CT image of thorax (source: http://radiopaedia.org/), (d) MRI of ankle (source: http://www.westbrookopenmri.com/), (e) MRI of knee (source: http://healthcare.siemens.com/).

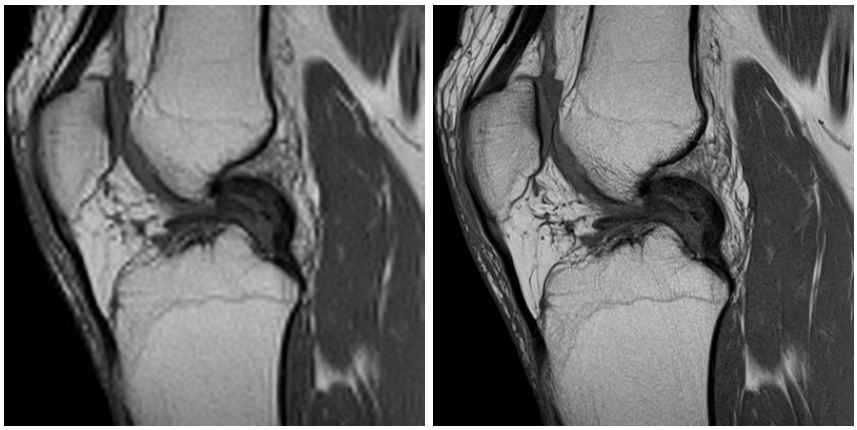
of Freeman *et al.* [FJP02] and the proposed SPOCH method, both of them give the best results (see Figure 3.10(d) and Figure 3.10(e)), with sharp edges while noise is effectively removed. However, our method's result is closer to the original test image with textures and details are more preserved than the MN-based method.

We also present in Figure 3.11 the results of a CT image of lung (size 200×200) with magnification factor $s = 4$. This experiment has a potential application of pulmonary micro-nodules (small white nodules). In fact, their presence indicates a potential risk of cancer and an early detection is crucial for patients. In this experience, we directly use this input image as example standard image to construct the database for all methods MN, NE and SPOCH. We only show here the results of a desired region of interest in the LR image (see Figure 3.11(b)-3.11(f)). As it can be seen, the bicubic interpolation results in a blurred image (see Figure 3.11(c)). The MN-based method of Freeman *et al.* [FJP02] produces sharper image than the other methods. However, this makes the result image somewhat "non-photorealistic" near

(a) Low-resolution image (b) Bicubic



(c) NE [CYX04] (d) MN [FJP02]



(e) SPOCH (f) Original image

Figure 3.10: Super-resolution results on a knee MRI image with magnification factor of 4. (a) input 100×100 LR image with standard deviation of noise $\sigma = 10$ (shown with nearest neighbor interpolation), (b) bicubic interpolation (PSNR = 31.19, SSIM = 0.560), (c) NE [CYX04] (PSNR = 31.27, SSIM = 0.572), (d) MN [FJP02] (PSNR = 32.41, SSIM = 0.661), (e) SPOCH (PSNR = **32.49**, SSIM = **0.706**), and (f) ground truth image.

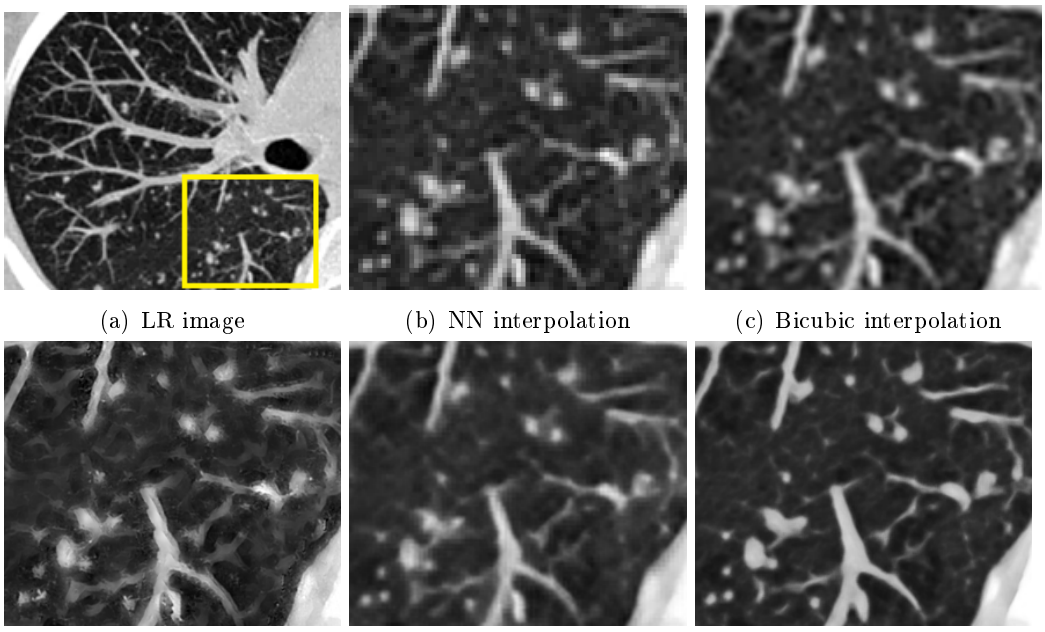
Table 3.1: PSNR and SSIM Comparison

Image	σ	PSNR				SSIM			
		Bicubic	MN	NE	SPOCH	Bicubic	MN	NE	SPOCH
(a)	0	33.88	33.70	**34.42**	34.32	0.844	0.829	0.880	**0.882**
	5	33.24	33.69	32.53	**34.10**	0.780	0.824	0.780	**0.862**
	10	32.00	33.63	31.68	**34.04**	0.682	0.814	0.728	**0.860**
(b)	0	32.45	32.59	33.07	**33.13**	0.715	0.732	0.736	**0.738**
	5	31.87	32.27	31.91	**32.35**	0.672	0.729	0.683	**0.735**
	10	30.77	32.18	31.00	**32.31**	0.581	0.723	0.621	**0.734**
(c)	0	34.20	34.40	**35.02**	34.98	0.850	0.850	**0.876**	0.873
	5	33.22	34.28	32.38	**34.32**	0.752	0.838	0.773	**0.859**
	10	32.50	34.14	32.52	**34.30**	0.649	0.820	0.709	**0.850**
(d)	0	32.21	32.18	32.29	**32.50**	0.712	0.695	0.721	**0.747**
	5	31.64	32.13	31.77	**32.42**	0.644	0.686	0.647	**0.741**
	10	31.35	32.08	31.44	**32.40**	0.563	0.675	0.578	**0.737**
(e)	0	32.62	32.52	32.73	**32.91**	0.690	0.671	0.699	**0.718**
	5	32.23	32.43	32.21	**32.79**	0.641	0.665	0.646	**0.708**
	10	31.19	32.41	31.27	**32.49**	0.560	0.661	0.572	**0.706**

the edges, and it seems to introduce many unwanted details (see Figure 3.11(d)). The result generated by the NE [CYX04] in Figure 3.11(e) is slightly blurred and details are less enhanced. In contrast, it is interesting to see that our method provides enhanced contrast while noise is well removed (see it is obvious that Figure 3.11(f)). Visually, SPOCH well preserves subtle details while reducing noise. Moreover, we can clearly see the presence of many micro-nodules with sharper form when compared to the bicubic interpolation method and the NE method.

3.3.1.4 Conclusion

In this section, we have proposed a geometric solution for single image super-resolution by using the projection onto convex hull of a finite set of points in high-dimension vector space. The proposed method is a K-nearest neighbor-based approach. The main difference between the proposed SPOCH method and the NE-based method is that while the proposed method estimate HR patch by finding a convex combination of the K nearest neighbors (HR candidate patches), the NE-based method [CYX04] finds an affine combination of these candidates. Moreover, we used pixel values as features to represent each patch (both LR and HR), while the NE-based method uses a gradient-based representation for LR patches.



(a) LR image (b) NN interpolation (c) Bicubic interpolation

(d) MN [FJP02] (e) NE [CYX04] (f) SPOCH

Figure 3.11: Super-resolution results on a CT image of lung with magnification factor of 4. (a) LR image of size 200×200 with a windowed region of interest. (b)-(f) Super-resolution results of the region of interest obtained by the Nearest Neighbor (NN) interpolation, bicubic interpolation, Freeman *et al.*'s method [FJP02], Chang *et al.*'s method [CYX04], and SPOCH, respectively. Image courtesy of Dr. Jean-Marie Rocchisani (Avicenne University Hospital, Bobigny, France)

Although the experimental results have demonstrated the performance of the proposed method over some other state-of-the-art methods, the proposed method has some limitations. First, its effectiveness is highly dependent on the similarity between the input LR image and the example standard images. Second, it often requires a large database of HR and LR patch pairs. So, the computational time of the proposed method is still expensive. It takes around 4 minutes per 255×255 image, with magnification factor of 2 and a database of 70000 patch pairs (the program is implemented using Matlab and runs on PC of 3.06 GHz, 4GB RAM). As perspectives, these issues will be considered in the future.

3.3.2 The second proposed method: Super-Resolution by Optimal Weight model (SROW)

As presented above, the K-nearest neighbor-based methods are often limited by the choice of K and the quality of the nearest neighbors. Due to blurring, downsampling, and noisy data, it is very difficult to propose an ideal metric for measuring the

similarity between image patches. The sparse coding (SC)-based SR methods (see subsection 3.2.3) proposed to perform SR via sparse representation. By that way, these methods can avoid the problem of finding the nearest neighbors. Unlike the K-nearest neighbor-based methods, in the conventional sparse representation model such as in (3.20) each patch is coded individually without considering other patches, e.g. the similarity between image patches is not considered. Although these methods offer promising performances, learning the dictionary from the database is often time-consuming. Another difficulty is the debatable performance of these methods when dealing with noisy images.

In this section, we introduce an optimal weight model for single image super-resolution. This method is an extension of the sparse weight model for Gaussian denoising (SWMGD) presented in Section 2.4. Hence, the advantage of this model is an integrated framework of super-resolution and denoising, providing us both super-resolved and denoised solutions. This model is very suitable for medical images. Indeed, medical images are affected not only by limited spatial resolution but also by noise, making the structures or objects of interest indistinguishable. This method can improve the detection by enhancing the spatial resolution while removing noise. The basic idea is to find a non-negative sparse representation of the input y_i^l over the training database $\mathbf{P}_l = \{u_j^l,\, j \in \mathcal{I}\}$, in which the non-zero representation coefficients can be assigned to the example patches u_j^l which are congruent to y_i^l. By that way, we can take the advantages of both SC-based methods and K-nearest neighbors-based methods. Our method is referred to as SROW (Super-Resolution by Optimal Weight model).

Without loss of generality, the images in this work are assumed to be positive. Before going into the details of the proposed method, we start our presentation with the database construction phase.

3.3.2.1 Database construction

In this method, the example database of high- and low-resolution patch pairs, $(\mathbf{P}_l, \mathbf{P}_h)$, is simply constructed as follows:

- Step 1: A set of high- and low-resolution patch pairs $\{(u_i^l, u_i^h) \in \mathbb{R}^m \times \mathbb{R}^n,\, i \in \mathcal{I}\}$ is established from the example standard images $\mathbf{A}_h$ by the same way as in the SPOCH method above (see section 3.3.1.2.1). We can directly use this set as the example database. However, the database are then highly dependent on luminous intensity of the example standard images. To overcome this disadvantage we propose to use a database of scaled patches as in the next step.

- Step 2: The patch pairs (u_i^l, u_i^h) above are scaled before being saved to database,

$$(\mathbf{P}_l, \mathbf{P}_h) = \left\{ \left(\frac{u_i^l}{\|u_i^l\|_2}, \frac{u_i^h}{\|u_i^l\|_2} \right), i \in \mathcal{I} \right\}, \tag{3.47}$$

  Simply, we denote below the training set as

$$(\mathbf{P}_l, \mathbf{P}_h) = \left\{ (u_i^l, u_i^h) \in \mathbb{R}^m \times \mathbb{R}^n, i \in \mathcal{I} \right\}, \tag{3.48}$$

  in which $\|u_i^l\|_2 = 1$ and $D_s H u_i^h = u_i^l$ for all $i \in \mathcal{I}$. We can consider the set $\mathbf{P}_l$ and $\mathbf{P}_h$ as the matrices in $\mathbb{R}^{m \times card(\mathcal{I})}$ and $\mathbb{R}^{n \times card(\mathcal{I})}$, respectively.

Note that by this construction way, we can write $\mathbf{P}_l = D_s H \mathbf{P}_h$.

Compared with the SPOCH method, by scaling the vectors in the training database we can establish the database $(\mathbf{P}_l, \mathbf{P}_h)$ from the example standard images $\mathbf{A}_h$ with luminance intensity different from the LR input image $\mathbf{Y}$. Hence, this database is more robust to luminance intensity.

3.3.2.2 Super-Resolution by Optimal Weight model

In this subsection, we will present in details the proposed model for denoising and super-resolution.

Let us consider a LR patch $y_i^l = D_s H x_i^h + \eta_i$ as in (3.26), with the noise component $\eta_i \sim \mathcal{N}(0, \sigma_i^2)$. The problem is to find an estimate of the HR patch x_i^h, denoted by $\hat{x}_i^h$, from y_i^l with the help of the database $(\mathbf{P}_l, \mathbf{P}_h)$. Thanks to the repetition of local structures of images, we can believe that there exists a subset of patches $u_k^h \in \mathbf{P}_h$ which have similar structures as these of x_i^h. Such patches will play an important role in determining the estimate $\hat{x}_i^h$.

In this work, it is assumed that $x_i^h \in \mathbb{R}^n$ can be represented as a sparse non-negative linear combination of the HR patches in $\mathbf{P}_h$,

$$x_i^h = \sum_{k \in \mathcal{I}} \alpha_k u_k^h, \tag{3.49}$$

where vector of representation coefficients $\alpha^i = (\alpha_1, \alpha_2, \ldots, \alpha_k, \ldots)^T \geq 0$. Note that unlike the previous SC-based methods ([YWHM08, YWHM10, YWL+12, WZG10, ZEP10]), in (3.49) we use non-negative constraint on the coefficients α_k. This can be explained by considering two sides of equation (3.49). Due to the fact that the vectors x_i^h and u_k^h are defined from the pixel values of image patches, they are then positive vectors. Thus, in the equation $x_i^h = \mathbf{P}_h \alpha^i$, both x_i^h and $\mathbf{P}_h$ are positive. This is why we can require the non-negative constraint on α^i.

Now, consider the corresponding LR patch y_i^h of x_i^h. Since it is assumed that $x_i^h = \mathbf{P}_h \alpha^i$, multiplication of this equation by $D_s H$ gives

$$D_s H x_i^h = D_s H \mathbf{P}_h \alpha^i = \mathbf{P}_l \alpha^i. \tag{3.50}$$

Exploiting the relation between the LR and the HR patches $y_i^l = D_s H x_i^h + \eta_i$, we obtain

$$\mathbf{P}_l \alpha^i = y_i^l - \eta_i, \tag{3.51}$$

implying that

$$y_i^l - \mathbf{P}_l \alpha^i = \eta_i. \tag{3.52}$$

Thus,

$$\|y_i^l - \mathbf{P}_l \alpha^i\|_2 \leq \xi_i, \tag{3.53}$$

where ξ_i is related to the noise power σ_i of η_i.

As it can be seen, the LR patch y_i^l can be represented by the same sparse vector α^i over the database of HR patches $\mathbf{P}_l$, with a controlled error ξ_i. This implies that for a given LR patch y_i^l, the estimate of the corresponding HR patch x_i^h is performed as follows. First, we determine sparse representation vector, α^i, of y_i^l with respect to the database of LR patches $\mathbf{P}_l$. Then, x_i^h can be recovered by simply multiplying this representation by the database $\mathbf{P}_h$, $\hat{x}_i^h = \mathbf{P}_h \alpha^i$. This is the core idea behind the proposed method.

Our aim is to find the patches u_k^h which are similar to x_i^h and then to use them for the estimation of x_i^h. Therefore, we try to force coefficient vector $\alpha^i = (\alpha_1, \alpha_2, \ldots, \alpha_k, \ldots)^T$ such that most of its zero components, α_k, correspond to the elements u_k^h which are dissimilar to x_i^h. To this end, we formulate the problem of finding the vector α^i as a following sparse decomposition problem:

$$\begin{aligned} \alpha^i \quad = \quad & \underset{\alpha \geq 0}{\operatorname{argmin}} \ \|\alpha\|_0 + \sum_{k \in \mathcal{I}} w_{ik} \alpha_k \\ & \text{subject to} \quad \Big\| y_i^l - \sum_{k \in \mathcal{I}} \alpha_k u_k^l \Big\|_2^2 \leq \varepsilon \sigma_i^2, \end{aligned} \tag{3.54}$$

where ε is a given positive number, the constants $w_{ik} > 0$ depend on the dissimilarity between x_i^h and u_k^h. We want to force small α_k for high dissimilarity w_{ik}, i.e. for weak similarity between x_i^h and u_k^h. However, due to the fact that the HR patch x_i^h is not available, the dissimilarity between x_i^h and u_k^h is evaluated through their LR versions y_i^l and u_k^l.

In (3.54), the ℓ_0-norm assures that the solution α^i is a sparse one. In the second term, w_{ik} may be viewed as a penalty coefficient on the variable α_k in the sense that if the value of w_{ik} is suitably large, the term $w_{ik}\alpha_k$ will be penalized a heavy

cost if α_k is large. Thus, in the cases where u_k^l and y_i^l are very dissimilar (i.e. w_{ik} is large), objective function in (3.54) can be minimized with α_k often very small or null. Therefore, in the sparse solution α^i of (3.54), the non-zero components often correspond to the small penalty coefficients. The constraint in (3.54) implies that the output HR patch $\hat{x}_i^h = \sum_{k\in\mathcal{I}} \alpha_k u_k^h$ has to be consistent with the input y_i^l. Therefore, $\hat{x}_i^h$ can be determined optimally from several example HR patches $u_k^h \in \mathbf{P}_h$ which have similar in structure with the desired patch x_i^h. By that way, it can avoid the influence of the inconsistent patches in the determining the estimate $\hat{x}_i^h$.

In this work, the penalty coefficients w_{ik} is defined as the form

$$w_{ik} = \Phi_i(d(y_i^l, u_k^l)), \tag{3.55}$$

where $d: \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^+$ is a criteria evaluating the dissimilarity between y_i^l and u_k^l, $\Phi: \mathbb{R} \to \mathbb{R}^+$ is a non-negative increasing function. The value of the function $d(y_i^l, u_k^l)$ express the dissimilarity in structure between y_i^l and u_k^l. So, a smaller dissimilarity value implies higher similarity. Normally, to measure the extent of dissimilarity among the image patches, one of the most popular dissimilarity criteria is the Euclidian distance. However, in this case y_i^l is a vector defined from the pixel values of the i^th patch of the input LR image $\mathbf{Y}$ while u_k^l is a normalized example vector in the database. Moreover, y_i^l is also corrupted by noise η_i. Thus, using the Euclidian distance may not be effective enough. To obtain a better dissimilarity criteria, let us consider the relationship of y_i^l and u_k^l.

We start with definition of congruence of image patches.

Definition 3.3.1 *Two image patches* $\mathbf{x}_1$ *and* $\mathbf{x}_2$ *are congruent if there exists a non-zero constant* $\mu \in \mathbb{R}$*, with* $\mathbf{x}_1 = \mu\mathbf{x}_2$*.*

As mentioned above, y_i^l is assumed to be corrupted by Gaussian white noise $\eta_i \sim \mathcal{N}(0, \sigma_i^2)$, $y_i^l = D_s H x_i^h + \eta_i$ (see (3.26)). Thus, the patch u_k^l is ideally similar to y_i^l if u_k^l is congruent to $D_s H x_i^h$. That means there exists a constant $\mu_{ik} > 0$ such that

$$y_i^l = \mu_{ik} u_k^l + \eta_i. \tag{3.56}$$

Since the assumption that the noise component $\eta_i \sim \mathcal{N}(0, \sigma_i^2)$, we have the mean of η_i, $\mathrm{E}(\eta_i) \approx 0$. Therefore, the constant μ_{ik} can be approximately computed as:

$$\mathrm{E}(y_i^l) = \mu_{ik}\mathrm{E}(u_k^l) + \underbrace{\mathrm{E}(\eta_i)}_{\approx 0} \Longrightarrow \mu_{ik} = \frac{\mathrm{E}(y_i^l)}{\mathrm{E}(u_k^l)}. \tag{3.57}$$

From equation (3.56), we have $(y_i^l - \mu_{ik} u_k^l) \sim \mathcal{N}(0, \sigma_i^2)$. It can be inferred that,

$$\begin{cases} \mathrm{E}(y_i^l - \mu_{ik} u_k^l) & \simeq 0 \\ \mathrm{Var}(y_i^l - \mu_{ik} u_k^l) - \sigma_i^2 & \simeq 0. \end{cases}$$

Therefore,

$$a_{i,k} = \left|\mathrm{E}(y_i^l - \mu_{ik}u_k^l)\right| + \left|\mathrm{Var}(y_i^l - \mu_{ik}u_k^l) - \sigma_i^2\right| \simeq 0. \tag{3.58}$$

The parameter $a_{i,k}$ allows us to evaluate the statistical property of noise in the residual patch. So, in this work, the dissimilarity criteria is defined by

$$d(y_i^l, u_k^l) = \|y_i^l - \mu_{ik}u_k^l\|_2^2 + a_{i,k}, \tag{3.59}$$

where μ_{ik} is defined by (3.57).

In this work, the function $\Phi_i(\cdot)$ in (3.55) is defined by

$$\Phi_i(t) = \begin{cases} e^t & \text{if } t > \rho_i \\ t & \text{if } t \leq \rho_i, \end{cases} \tag{3.60}$$

where ρ_i is a positive threshold corresponding to y_i^l. As can be seen, the function $\Phi_i(t)$ strongly increases when $t > \rho_i$. This makes the penalty coefficients corresponding to the example patches to have $d(y_i^l, u_k^l) > \rho_i$, and so these penalty coefficients will be very high. Note that in the ideal case, $\mu_{ik}u_k^l = D_sHx_i^h$, we have,

$$d(y_i^l, u_k^l) \approx m\sigma_i^2, \tag{3.61}$$

where m is the number of elements in vector y_i^l ($y_i^l \in \mathbb{R}^m$). Therefore, for $d(y_i^l, u_k^l) \leq m\sigma_i^2$ we can believe that u_k^l is congruent with $D_sHx_i^h$, and thus u_k^h is congruent with the desired HR patch x_i^h. Then, the threshold ρ_i in (3.60) is set to $m\sigma_i^2$.

It is easy to see that the objective in (3.54) is not a convex function, since ℓ_0-norm is not a true norm. This problem is too complex to solve in general. To avoid the above problem we replace ℓ_0-norm by ℓ_1-norm, and problem (3.54) is then convex and can be rewritten as:

$$\begin{aligned} \alpha^i &= \underset{\alpha \geq 0}{\operatorname{argmin}} \|\alpha\|_1 + \sum_{k \in \mathcal{I}} w_{ik}\alpha_k \\ &\quad \text{subject to} \quad \Big\|y_i^l - \sum_{k \in \mathcal{I}} \alpha_k u_k^l\Big\|_2^2 \leq \varepsilon\sigma_i^2 \end{aligned} \tag{3.62}$$

$$\begin{aligned} \Leftrightarrow \alpha^i &= \underset{\alpha \geq 0}{\operatorname{argmin}} \sum_{k \in \mathcal{I}} (1 + w_{ik})\alpha_k \\ &\quad \text{subject to} \quad \Big\|y_i^l - \sum_{k \in \mathcal{I}} \alpha_k u_k^l\Big\|_2^2 \leq \varepsilon\sigma_i^2. \end{aligned} \tag{3.63}$$

Lagrange multipliers allow an equivalent formulation

$$\alpha^i = \underset{\alpha \geq 0}{\operatorname{argmin}} \frac{1}{2}\Big\|y_i^l - \sum_{k \in \mathcal{I}} \alpha_k u_k^l\Big\|_2^2 + \lambda \sum_{k \in \mathcal{I}} (1 + w_{ik})\alpha_k, \tag{3.64}$$

where the parameter λ balances sparsity of the solution and fidelity of the approximation to y_i^l.

Let us denote $S(y_i^l) = \{k \in \mathcal{I} : \alpha_k^i > 0\}$ as the support set of $\mathbf{y}_i^l$. As analyzed above, $S(y_i^l)$ involves u_k^l where $d(y_i^l, u_k^l)$ is not very large. Thus, with a suitable value of the threshold r_i, there exists a subset $\mathcal{I}_i = \{j \in \mathcal{I} : d(y_i^l, u_j^l) \leq r_i\}$ of $\mathcal{I}$ such that

$$S(\mathbf{y}_i^l) \subseteq \mathcal{I}_i. \tag{3.65}$$

Thus, to save computing time, problem (3.64) should be considered on the subset $\mathcal{I}_i$,

$$\Rightarrow \alpha^i = \arg\min_{\alpha \geq 0} \frac{1}{2}\|y_i^l - \sum_{k \in \mathcal{I}_i} \alpha_k u_k^l\|_2^2 + \lambda \sum_{k \in \mathcal{I}_i} (1 + w_{ik})\alpha_k. \tag{3.66}$$

It is easily to see that problem (3.66) can be rewritten as:

$$\begin{aligned} \alpha^i &= \arg\min_{\alpha \geq 0} \frac{1}{2}\|\mathbf{y}_i^l - \mathbf{U}_i\alpha\|_2^2 + \lambda \mathbf{w}_i^T \alpha \\ &= \arg\min_{\alpha \geq 0} \frac{1}{2}\alpha^T ({\mathbf{U}_i}^T \mathbf{U}_i)\alpha + \left(\lambda \mathbf{w}_i - {\mathbf{U}_i}^T \mathbf{y}_i^l\right)^T \alpha \end{aligned} \tag{3.67}$$

where $\mathbf{U}_i$ is the matrix whose columns are the vectors u_k^l, $\mathbf{w}_i$ is the vector formed by concatenating all the penalty coefficients w_{ik}, here $k \in \mathcal{I}_i$. We can see that (3.67) is a nonnegative quadratic programming, which can be solved, specifically by the multiplicative updates algorithm proposed by Sha *et al.* in [SSL02] as presented in subsection 2.3.2.2.

When α^i is obtained, the desired HR patch $\hat{x}_i^h$ can be estimated as

$$\hat{x}_i^h = \sum_{k \in \mathcal{I}_i} \alpha_k^i u_k^h. \tag{3.68}$$

Likely, we can obtain a denoised version, denoted by $\hat{y}_i^l$, of y_i^l

$$\hat{y}_i^l = \mathbf{U}_i \alpha^i = \sum_{k \in \mathcal{I}_i} \alpha_k^k u_k^l. \tag{3.69}$$

In the next section, we will present how to construct the entire HR image using the estimated HR patches $\hat{x}_i^h$, and the denoised patches $\hat{y}_i^l$.

3.3.2.3 Reconstruction of the entire HR image

We first put all the estimated HR patches $\hat{x}_i^h$ in the proper locations in the HR grid. A coarse estimate of $\mathbf{X}$, $\hat{\mathbf{X}}^{coarse}$, is then computed by an average in overlapping regions. In the same way, we obtain a denoised image, denoted by $\mathbf{Y}^{denoise}$, of $\mathbf{Y}$ by replacing the noisy patches y_i^l by the denoised $\hat{y}_i^l$, and then take the average in overlapping regions.

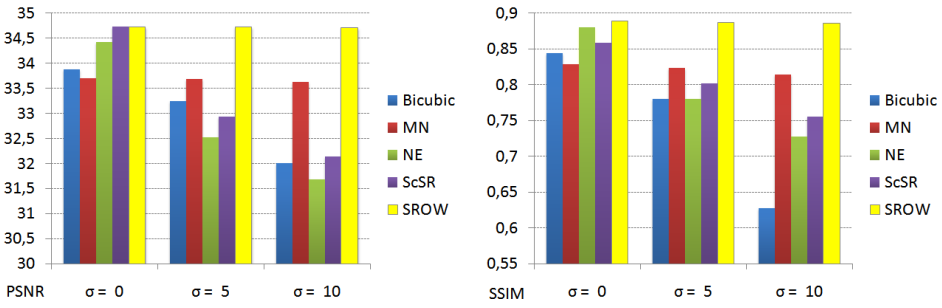
Similarly to [YWHM08], we determine the final HR image $\hat{X}^{final}$ as the minimizer of the following problem:

$$\min_{\mathbf{X}} \|\mathbf{X} - \hat{\mathbf{X}}^{coarse}\|_2^2 \text{ subject to } \|D_s H\mathbf{X} - \mathbf{Y}^{denoise}\|. \tag{3.70}$$

The back-projection algorithm [IS93] is used to solve this problem:

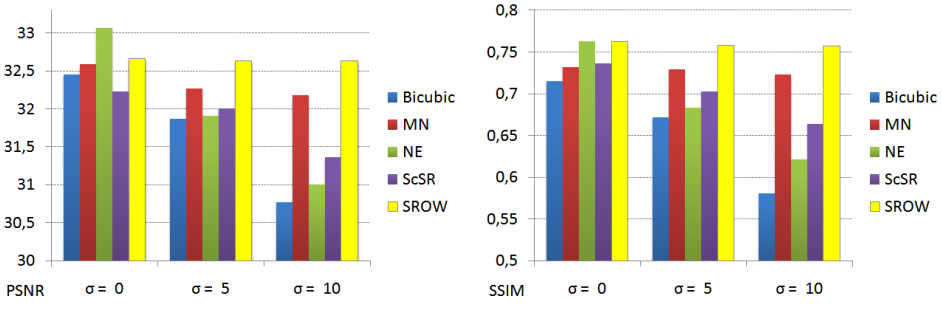
$$\mathbf{X}_{t+1} = \mathbf{X}_t + ((\mathbf{Y}^{denoise} - D_s H\mathbf{X}_t) \uparrow_s) * p, \tag{3.71}$$

where $\mathbf{X}_t$ is the estimate of the HR image at the t-th iteration, $\uparrow_s$ denotes up-scaling by factor s, p is a symmetric Gaussian filter. Here, we use the symmetric Gaussian filter of size 5 with standard deviation 1. Note that in the case where the input LR image $\mathbf{Y}$ can be considered as a noise-free image, $\mathbf{Y}^{denoise}$ in (3.70) and (3.71) is replaced by $\mathbf{Y}$.



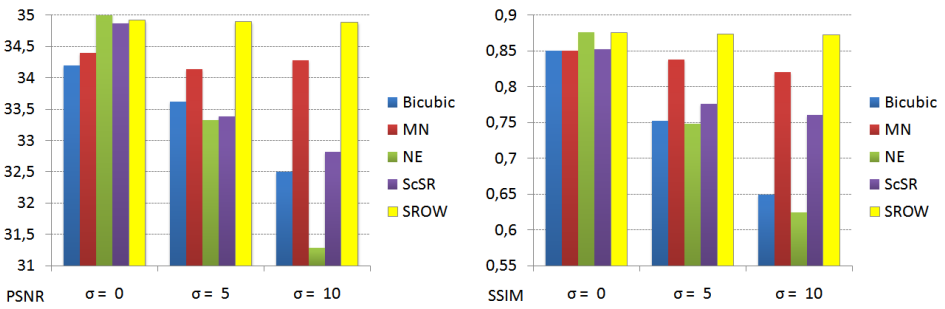
(a) (b)

Figure 3.12: Objective comparison on the CT image of abdomen with the noise levels: $\sigma = 0$, 5 and 10.



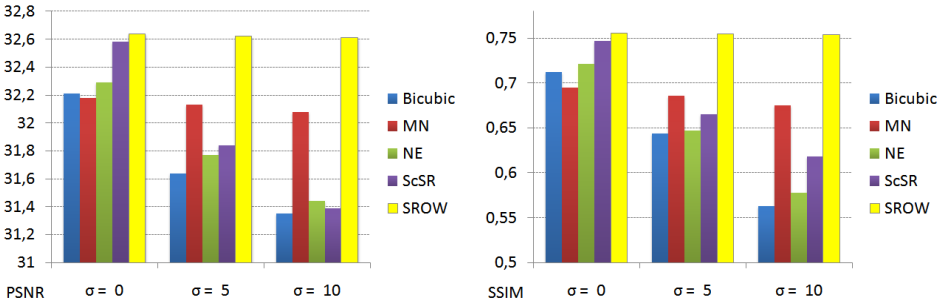
(a) (b)

Figure 3.13: Objective comparison on the CT image of lung with the noise levels: $\sigma = 0$, 5 and 10.



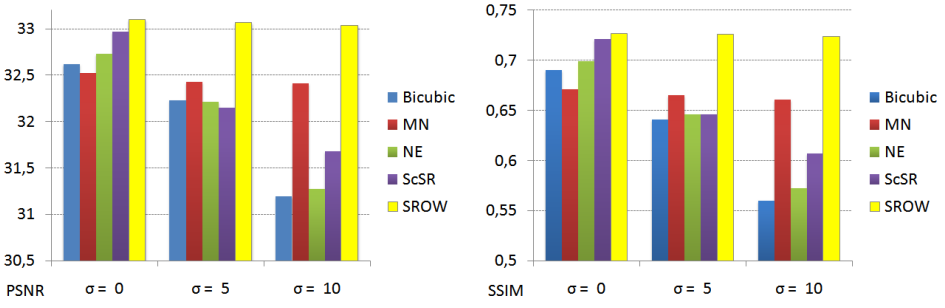
(a) (b)

Figure 3.14: Objective comparison on the CT image of trorax with the noise levels: $\sigma = 0, 5$ and 10.



(a) (b)

Figure 3.15: Objective comparison on the CT image of ankle with the noise levels: $\sigma = 0, 5$ and 10.



(a) (b)

Figure 3.16: Objective comparison on the CT image of knee with the noise levels: $\sigma = 0, 5$ and 10.

3.3.2.4 Performance evalution

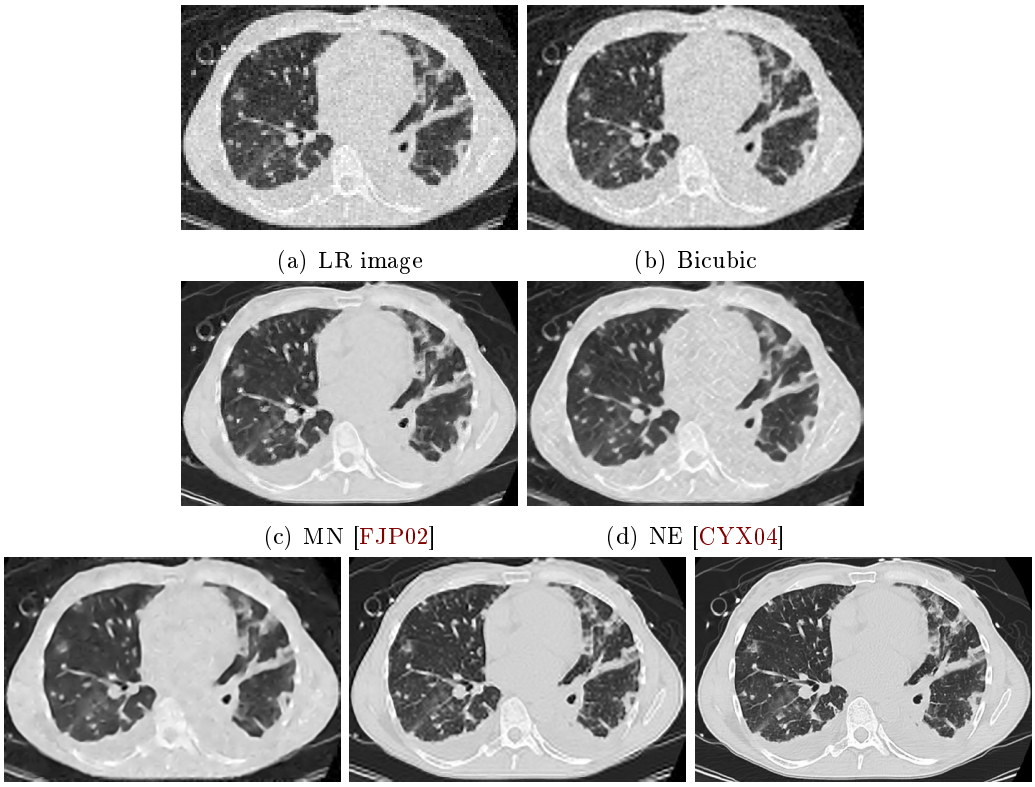
For medical images, the ability to detect anomalies or object of interest may be limited not only by limitations in spatial resolution but also by the presence of noise. Thus, we present here some experiments performed on both images with noise-free and noisy images. The proposed method (called SROW) is compared with the bicubic interpolation, the Markov-network-based SR method (MN) of Freeman *et al.* [FJP02], the neighbor embedding-based SR method (NE) of Chang *et al.*[CYX04], and the Sparse codding-based SR method (ScSR) [YWHM10] [3].

The experimental tests are performed on five 8-bit images shown in Figure 3.8: CT of abdomen, CT of lung, CT of thorax, MRI of ankle, and MRI of knee with magnification factor $s = 4$. The training databases of all the methods are established with the same set of five standard images as illustrated in Figure 3.9. In all experiments, the LR image is created from the corresponding test image in three steps: first, the test image is blurred by a 7×7 Gaussian filter with standard deviation 1, then downsampling by a decimation factor of s is performed, and finally the Gaussian white noise with standard deviation σ is added into the decimated image. For the proposed SROW method, super-resolution is realized two times with magnification factor 2. The database includes 150000 patch pairs extracted randomly form the standard images. Default size of the HR patches is 7×7. The parameter λ in equation (3.67) is set to 1.

For objective comparison, we use the PSNR and SSIM to evaluate the performance of the methods. The best results of the methods are reported through Figure 3.12 to 3.16. As it can be seen, for $\sigma = 0$, the NE and ScSR methods obtain high quality indices and sometimes higher values than those of the proposed method. However, the quality measures of the SROW method are significantly higher than those of the other methods, in the cases of noise for $\sigma = 5$ and $\sigma = 10$. This confirms the outperformance of the proposed method in the cases of noisy images.

For a subjective evaluation, Figure 3.19 and Figure 3.17 show the SR results with magnification $s = 4$ of the different methods for the CT image of thorax (Figure 3.8(b)) and the MRI image of ankle (Figure 3.8(d)), respectively. In these experiments, the LR images are corrupted by white Gaussian white noise with $\sigma = 10$. Visually, the bicubic interpolation's result (Figure 3.17(b)) is blurred while the results of the NE method of Chang *et al.* [CYX04] (Figure 3.19(c) and Figure 3.17(d)) and the ScSR method of Yang *et al.* [YWHM10] (Figure 3.19(d) and Figure 3.17(e)) are smoothed, many small details are lost. The results of the MN method of Freeman *et al.* [FJP02] looks better compared to the NE and the ScSR methods. As

[3] Matlab code available at http://www.ifp.illinois.edu/~jyang29/index.html



(a) LR image (b) Bicubic

(c) MN [FJP02] (d) NE [CYX04]

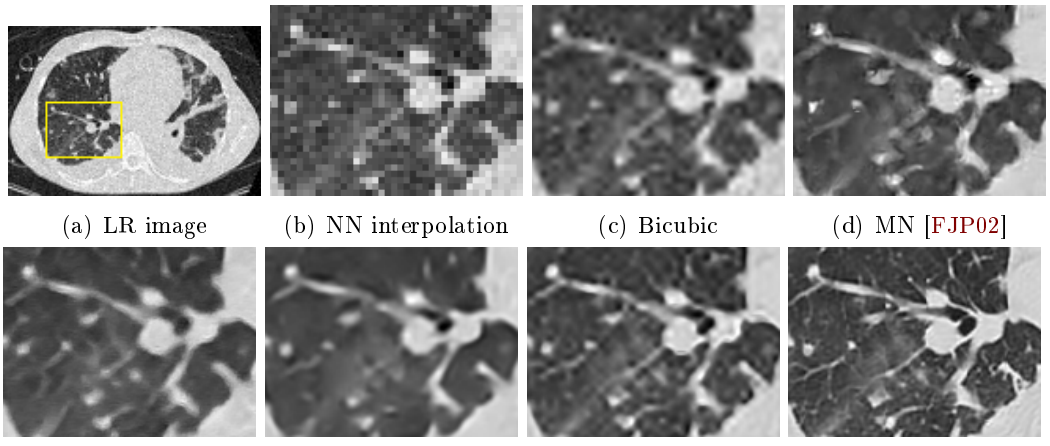
(e) ScSR [YWHM10] (f) SROW (g) Test image

Figure 3.17: Results on the CT image of thorax (Figure 3.8(b)) with magnification factor of 4. (a) LR image (size 135×90) corrupted by Gaussian noise with $\sigma = 10$ (shown with nearest neighbor interpolation); (b) bicubic interpolation (PSNR = 30.77, SSIM = 0.581), (c) Freeman *et al.*'s method (PSNR = 32.18, SSIM = 0.723); (d) Chang et al.'s method (PSNR = 31.00, SSIM = 0.621); (e) Yang *et al.*'s method (PSNR = 31.36, SSIM = 0.664); (f) the proposed SROW method (PSNR = **32.63**, SSIM = **0.757**); (g) Original test image.

it can be seen in Figure 3.17(f), the proposed SROW method sometimes introduces jaggy artifacts along the major edges. However, the proposed SROW method gives the best results in cases of noise by effectively reducing noise and well preserving small details (see Figure 3.19(e) and Figure 3.17(f)). To see more clearly, we can observe in Figure 3.18, the super-resolution results in a Desired Region Of Interest (DROI) of the LR image (the yellow rectangle in Figure 3.18(a)).

3.3.2.5 Conclusion

In this section, we have proposed a very competitive example-based SR model capable of enhancing resolution and denoising in the same framework. The method



(a) LR image (b) NN interpolation (c) Bicubic (d) MN [FJP02]

(e) NE [CYX04] (f) ScSR [YWHM10] (g) SROW (h) Original

Figure 3.18: The images of DROI in the super-resolution results in Figure 3.17. (a) is the LR image in which the yellow rectangle illustrates a desired region of interest. (b)-(g) present the DROI up-scaled with magnification of 4 by the Nearest Neighbor (NN) interpolation, the bicubic interpolation, the MN-based method of Freeman *et al.* [FJP02], the NE-based method of Chang *et al.* [CYX04], the SC-based method of Yang *et al.* [YWHM10], and the proposed SROW method, respectively. (h) shows the DROI in the original test image.

is based on finding a sparse positive linear representation of the HR patches in the database constructed from given standard images. The experimental results have demonstrated the effectiveness of our method over some leading state-of-the-art SR methods. It is very useful in the case when the low-resolution image is corrupted by noise. The results of the SROW method for medical images are very promising, demonstrating the ability of the method for the potential benefit of diagnosis. In the future works, we will study the construction of the training database of example patch pairs optimally from the given standard images.

3.4 Comparison of two proposed SR methods

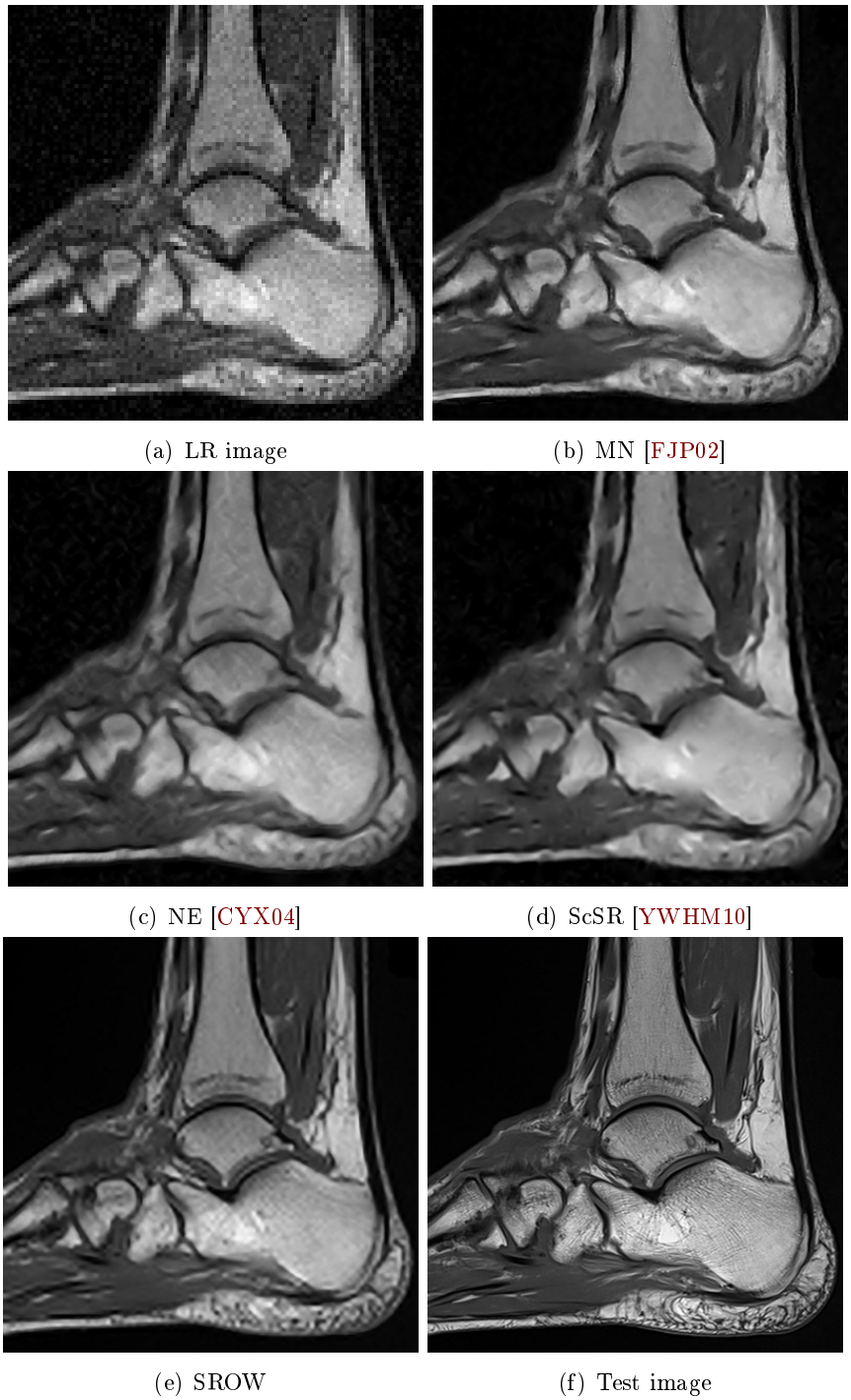
In previous sections the proposed methods have been compared with the other methods. In this section, we propose a comparison of the SPOCH method and the SROW method.

To this end, we report the experimental results of the two methods performed on five test images of Figure 3.8. The experiments are performed, on both noise-free image and noisy image, with magnification factor $s = 4$. The LR images are tested with Gaussian noise with standard deviations $\sigma = 0, 5$ and 10. The objective comparison between SPOCH and SROW using the PSNR and SSIM metrics are

illustrated through Figures 3.20, 3.21 and 3.22. We can see that the SROW method achieves noticeable quality gains over the SPOCH method in most cases. This confirms the outperformance of the SROW method.

3.5 Summary

In this chapter, we have presented an overview of the example-based super-resolution approach and introduced two novel example-based learning SR methods which perform SR and denoising effectively. The first proposed method, namely SPOCH, is a geometric solution for single image super-resolution. In this method, super-resolution is realized based on using the projection onto convex hull of a finite set of points in high-dimension vector space. The second method, namely SROW, we propose an optimal weight model for single image super-resolution, which integrates SR and denoising in the same framework. Through extensive experimental tests, we have shown the outperformance of the proposed methods compared with some state-of-the-art methods. The proposed SR approaches gives promising potential to be integrated in imaging technologies.

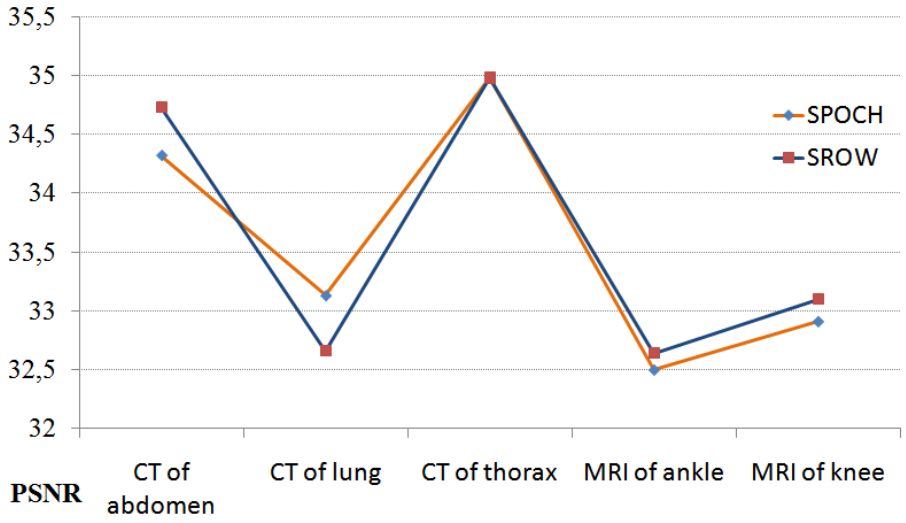


(a) LR image (b) MN [FJP02]

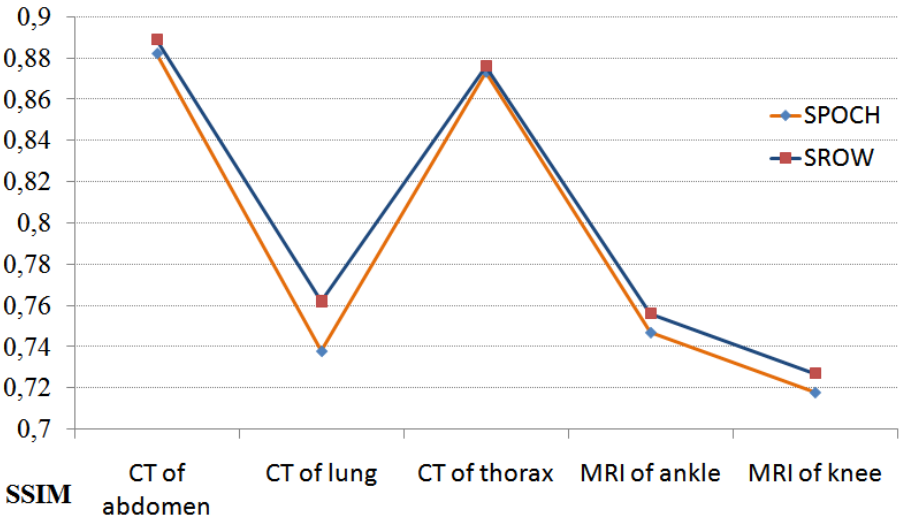
(c) NE [CYX04] (d) ScSR [YWHM10]

(e) SROW (f) Test image

Figure 3.19: Results on the MRI image of ankle (Figure 3.8(d)) with magnification factor of 4. (a) LR image (size 100×100) corrupted by Gaussian noise with $\sigma = 10$ (shown with nearest neighbor interpolation); (b) Freeman *et al.*'s method (PSNR = 32.08, SSIM = 0.675); (c) Chang et al.'s method (PSNR = 31.44, SSIM = 0.578); (d) Yang *et al.*'s method (PSNR = 31.39, SSIM = 0.618); (e) the proposed SROW method (PSNR = **32.61**, SSIM = **0.754**); (f) Original test image.

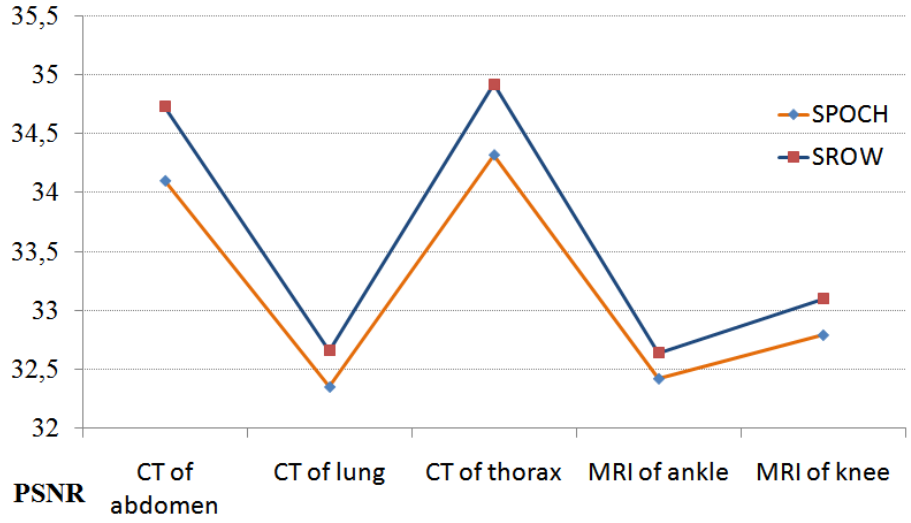


(a) PSNR

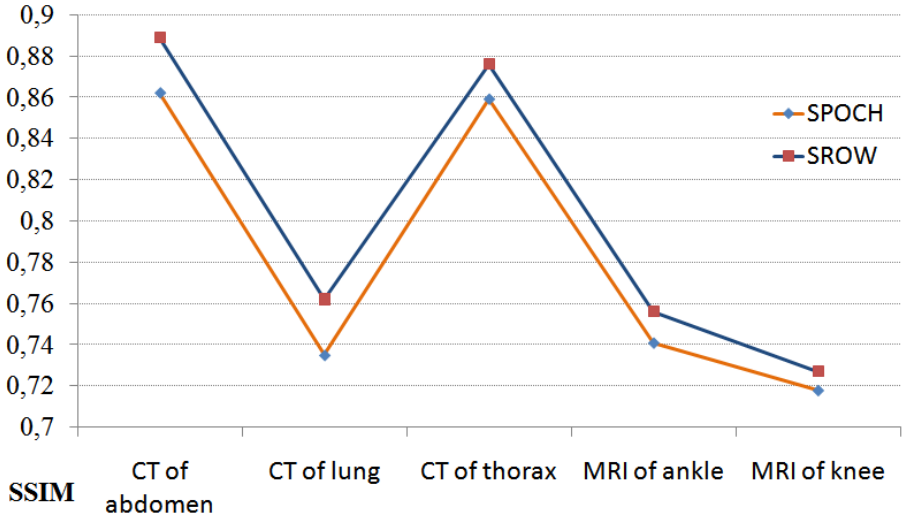


(b) SSIM

Figure 3.20: Objective comparison of the SPOCH method and the SROW method in the case of noise levels $\sigma = 0$: (a) PSNR comparison, (b) SSIM comparison.

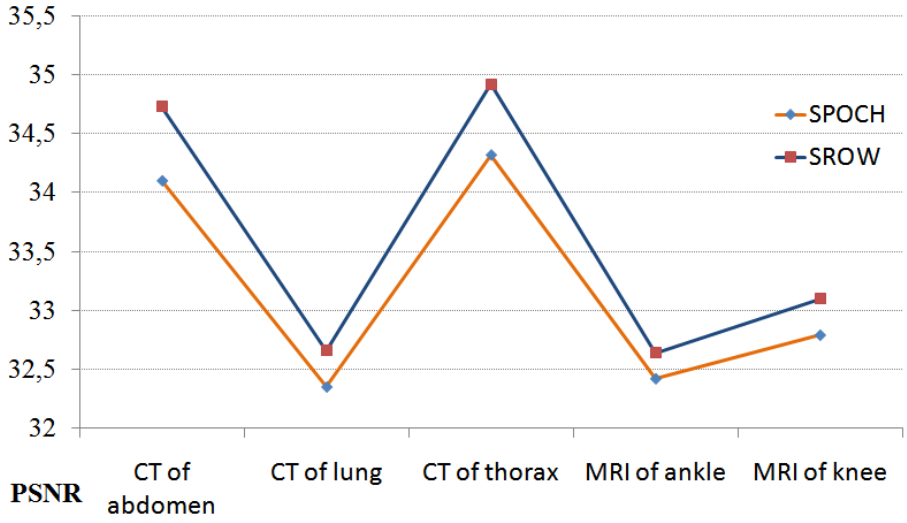


(a) PSNR

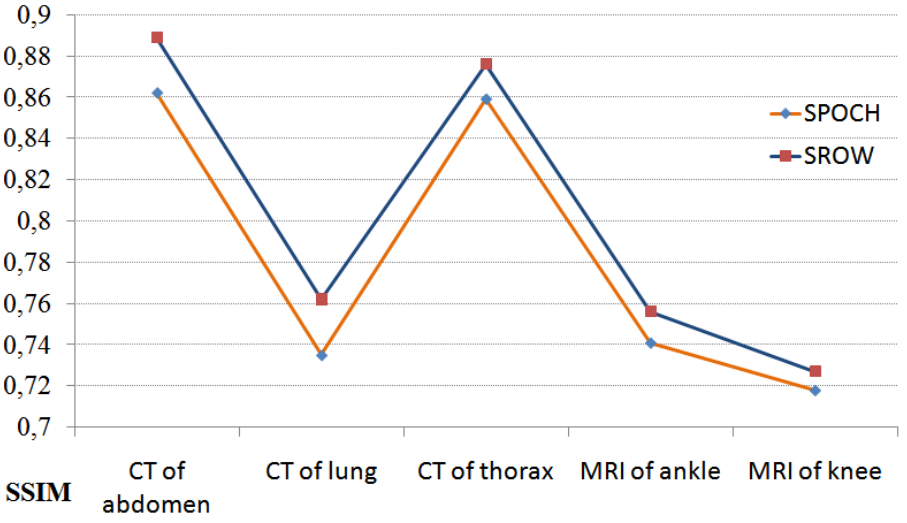


(b) SSIM

Figure 3.21: Objective comparison of the SPOCH method and the SROW method in the case of noise levels $\sigma = 5$: (a) PSNR comparison, (b) SSIM comparison.



(a) PSNR



(b) SSIM

Figure 3.22: Objective comparison of the SPOCH method and the SROW method in the case of noise levels $\sigma = 10$: (a) PSNR comparison, (b) SSIM comparison.

Chapter 4

Conclusions and Future Work

Contents

This chapter provides a summary of the work presented and the conclusions drawn from this work. It lists the contributions to knowledge already achieved in this research and provides directions for future work.

4.1 Summary and conclusions

In this thesis, we have investigated some issues related to quality improvement in medical imaging. Specifically, we have focused on two important factors affecting the image quality including noise and resolution. Several example-learning-based methods have been developed for denoising and resolution enhancement of medical images. The proposed methods are based on the observation that a lot of medical images are taken from the same type of subjects at similar locations, and many of them can be considered as standard images (high quality or acceptably and proven as noiseless images or passed a preprocessing step) by experts.

Chapter 1 gives a short overview of some common noise models found in medical images and some recent state-of-the-art approaches in the image denoising domain as well as their applications for medical image denoising. The approaches that have been addressed in this chapter include the Total Variation [ROF92, BKP10, KBPS11], the Non-local means [BCM05, BCM06], the Data-adaptive Kernel Regression [TFM07], the Sparse Representation [EA06, AEB06], and the Block matching with 3D (BM3D) [DFKE07].

Chapter 2 provides three novel example-learning-based denoising solutions. In the proposed methods, denoising is performed with the help of a given set of standard images (example images) taken at nearly the same location with the image that needs to be denoised.

- The first method is constructed based on the machine learning theory. It consists of two independent phases: training phase performs training the regression functions from a training set established from the standard images, and denoising phase performs denoising using the regression functions trained from the training phase. This method is proposed to reduce Gaussian noise and Rician noise. Thus, it can be used to reduce noise on some types of medical imaging such as CT, MRI.

- The second method is a k-nearest neighbor-based one designed for Gaussian noise and Poisson noise. In this method, a non-negative non-linear regression model has been proposed and used for denoising. This method can be consistent with imaging systems involving counting procedures like Fluorescence Confocal Microscopy, X-ray Projection Radiography, Fluoroscopy, Mammography, CT, PET and SPECT.

- The third method is designed for Gaussian denoising in low-dose CT images. In this method, the standard images are used to construct a database of example patches, and denoising is performed patch-wise with the help of such database. In order to denoise on image patches, we introduced a positive sparse weighted model. This model may be applied to images where the noise is supposed to be additive white Gaussian on each patch.

The obtained results are promising and show the effectiveness of the proposed methods.

In Chapter 3, the problem of resolution enhancement has been addressed. The approach of single-image super-resolution methods has been considered in our work. First, we have presented in this chapter the main content of some typical methods in this approach including the Markov-Network-based method of Freeman *et al.* [FJP02], the Neighbor Embedding-based method of Chang *et al.* [CYX04], and the super-resolution method via sparse representation of Yang *et al.* [YWHM08, YWL+12]. Our contributions were presented through two super-resolution solutions. The first one is a geometric solution where we proposed to apply projection onto convex hull of a finite set of points in high-dimension vector space for single image super-resolution. In the second one, super-resolution is performed via a novel model based on sparse positive linear representation. In particular, this model incorporates denoising and super-resolution in the same framework offering the possibility to improve spatial resolution while improving the quality of the image. The experimental tests on both synthetic and medical images have shown that the proposed solutions can effectively enhance resolution for noiseless as well as noisy images.

4.2 Future directions

Apart from some promising results, the proposed methods have still some limitations which need to be improved. They are listed below:

1. The proposed methods in this thesis highly depend on the set of standard images used to construct the training set, since the training set could be indeed dominated by standard images. In the general case, a larger data set of standard images is necessary to construct a good training set. Thus, it is often required a large amount of memory.

2. In the case of denoising problem, the first method using kernel ridge regression has many parameters and thus it is not easy to determine the optimal values for these parameters.

3. In the case of super-resolution, the proposed methods sometime introduce jaggy artifacts along the major edges. This may be due to the algorithms which have no any condition for suppressing such artifacts .

As perspectives, some issues will be considered in the future. We can list some important tasks as follows:

1. The first one is to construct a standard data set for medical images in such a way to reduce the dependence of the proposed methods on the example standard images. It is desirable for example that the test image could be of any characteristic and taken at any location.

2. The second one is to reduce the computational time of the methods. To this end, one of feasible solutions is to use Graphic Processing Unit (GPU).

3. The next one is how to extend the denoising solutions for different types of noise such as speckle noise and mixed noise.

4. In the SPOCH method, by considering each image patch as a point in high-dimension space and using projection onto convex hull, we have proposed a geometric solution for image super-resolution. Thus, some techniques of the computational geometry may be very useful to solve certain problems of image processing. This is a promising approach and deserves attention.

5. Finally, in the proposed super-resolution methods we have used the assumption that low-resolution image is affected by additive white Gaussian noise. Thus, an interesting perspective to extend these methods to other types of noise.

List of Notations

$\langle \cdot, \cdot \rangle_{\mathcal{F}}$	Inner product on a space $\mathcal{F}$
$\mathbb{N}$	Set of natural numbers
$\mathbb{R}^d$	d-dimensional real space
$\mathcal{H}$	Hessian operator
$\mathcal{K}(\cdot, \cdot)$	Kernel function
$\mathcal{N}(\mu, \sigma^2)$	Gaussian distribution with mean μ and variance σ^2
∇	Gradient operator
σ	Standard deviation
$\det(\mathbf{H})$	the determinant of matrix $\mathbf{H}$
div	Divergence operator
$\mathrm{E}(x)$	The mean of vector x
$\mathrm{Var}(x)$	The variance of vector x
BV	Space of functions of bounded variation
$L^1(\Omega)$	Space of integrable functions on Ω
p_X	The probability density function of random variable X
k-NN	k-nearest neighbor algorithm
ADF	Anisotropic diffusion filter
BM	Block matching
BM3D	Block matching 3D-based filter
CT	Computerized tomography
FC	Fluoroscopy
HR	High resolution
ISKR	Iterative steering kernel regression

K-SVD	A dictionary learning algorithm for sparse representation which is a generalizationg of the K-means algorithm and is based on the singular value decomposition
kNND	k-nearest neighbors-based denoising method
KRR	Kernel ridge regression
KRRD	Kernel ridge regression-based denoising method
LASSO	Least-Absolute-Shrinkage-and-Selection-Operator
LR	Low resolution
MAP	Maximum a posteriori
MG	Mammography
MN	Markov Network-based super-resolution method
MR	Magnetic resonance
MRI	Magnetic resonance imaging
MSE	Mean square error
NE	Neighbor Embedding-based super-resolution method
NLM	Non-local means
NN	Nearest neighbor
NQP	Nonnegative quadratic programming
PDF	Probability density function
PET	Positron emission tomography
POCH	Projection onto convex hull
PSNR	Peak signal to noise ratio
PURE	Poisson unbiased risk estimator
QP	Quadratic programming
RBF	Radial basis function
RNLM	Reference-based non-local means

SC	Sparse codding
ScSR	Sparse codding-based super-resolution
SNR	Signal to noise ratio
SPECT	Single photon emission computed tomography
SPOCH	Super-resolution using projection onto convex hull
SR	Super-resolution
SROW	Super-Resolution by Optimal Weights
SSIM	Structural similarity
SURE	Stein's unbiased risk estimate
SVD	Singular value decomposition
SWMGD	Sparse Weight Model for Gaussian Noise
TGV	Total generalized variation
TV	Total variation
UNLM	Unbiased nonlocal means
US	Ultrasound
X-PR	X-ray projection radiography

List of Figures

List of Tables

Bibliography

[ABHY00] M. S. Alam, J. G. Bognar, R.C. Hardie, and B.J Yasuda, *Infrared image registration and high-resolution reconstructionusing multiple translationally shifted aliased video frames*, Instrumentation and Measurement **49** (2000), no. 5, 915–923. (Cited on page 148.)

[AEB06] M. Aharon, M. Elad, and A. M. Bruckstein, *The k-svd: An algorithm for designing of overcomplete dictionaries for sparse representation*, IEEE Trans. On Signal Processing **54** (2006), 4311–4322. (Cited on pages 28, 65 and 195.)

[AMN+94] Sunil Arya, David M. Mount, Nathan S. Netanyahu, Ruth Silverman, and Angela Y. Wu, *An optimal algorithm for approximate nearest neighbor searching*, SODA, ACM/SIAM, 1994, pp. 573–582. (Cited on page 107.)

[Ans48] F. J. Anscombe, *The transformation of poisson, binomial and negativebinomial data*, Biometrika **35** (1948), no. 3–4, 246–254. (Cited on pages 31, 32, 50 and 108.)

[AT79] J. G. Abbott and F. L. Thurstone, *Acoustic speckle: Theorie and experimental analysis*, Ultrason. Imag. **1** (1979), 303–324. (Cited on page 19.)

[BCM05] A. Buades, B. Coll, and J.M. Morel, *A non local algorithm for image denoising*, IEEE Computer Vision and Pattern Recognition 2005 (CVPR 2005) (2005), 60–65. (Cited on pages 28, 47, 112, 115, 116, 117, 136, 137, 195, 209 and 210.)

[BCM06] A. Buades, B. Coll, and J. M. Morel, *A review of image denoising algorithms, with a new one*, SIAM J. Multiscale Modeling and Simulation **4** (2006), 490–530. (Cited on pages 28, 47, 48, 49, 50, 51, 52, 54, 56, 62, 63, 65, 66, 67, 70, 72, 94, 98, 104, 112, 113, 114, 134, 135, 138, 195, 205, 206, 207 and 210.)

[Bez81] James C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, Kluwer Academic Publishers, Norwell, MA, USA, 1981. (Cited on page 89.)

[BKP10] K. Bredies, K. Kunisch, and T. Pock, *Total generalized variation*, SIAM J. on Imaging Sciences **3** (2010), no. 3, 492–526. (Cited on pages 28, 37, 44, 45, 94, 98, 112, 113, 114, 115, 116, 117, 134, 135, 136, 137, 138, 195, 207, 209 and 210.)

[Bur98] Christopher J. C. Burges, *A tutorial on support vector machines for pattern recognition*, Data Min. Knowl. Discov. **2** (1998), no. 2, 121–167. (Cited on page 82.)

[BV11] K. Bredies and T. Valkonen, *Inverse problems with second-order total generalized variation constraints*, Proc. of SampTA 2011 - 9th Int. Conf. on Sampling Theory and Applications, 2011. (Cited on pages 44, 45, 46, 47 and 204.)

[BZ03] Paul Bao and Lei Zhang, *Noise reduction for magnetic resonance images via adaptative multiscale products thresholding*, IEEE Trans. on Me **22** (2003), no. 9, 1089–1099. (Cited on page 144.)

[CDS01] S. Chen, D. Donoho, and M. Saunders, *Atomic decompositions by basis pursuit*, Commun. Pure Appl. Math. **43** (2001), 129–159. (Cited on page 64.)

[CDV10] P. L. Combettes, D. Dung, and B. C. Vu, *Dualization of signal recovery problems*, Set-Valued and Variational Analysis **18** (2010), 373–404. (Cited on pages 38, 48, 51, 52, 204 and 205.)

[Cha04] A. Chambolle, *An algorithm for total variation minimization and applications*, Journal of Mathematical Imaging and Vision **20** (2004), 89–97. (Cited on page 38.)

[CHKB09] P. Coupe, P. Hellier, C. Kervrann, and C. Barillot, *Nonlocal means-based speckle filtering for ultrasound images*, IEEE Trans. on Image Process. **18** (2009), no. 10, 2221–2229. (Cited on pages 50, 55, 57, 58, 104 and 205.)

[CL97] A. Chambolle and P.L. Lions, *Image recovery via total variational minimization and related problems*, Numer. Math. **76** (1997), no. 2, 167–188. (Cited on page 44.)

[CMG+10] Pierrick Coupe, Jose V. Manjon, Elias Gedamu, Douglas L. Arnold, Montserrat Robles, and D. Louis Collins, *Robust rician noise estimation for mr images*, Medical Image Analysis **14** (2010), no. 4, 483–493. (Cited on page 86.)

[CMM00] T. Chan, A. Marquina, and P. Mulet, *High-order total variation-based image restoration*, SIAM J. Sci. Comput. **22** (2000), no. 2, 503–516. (Cited on page 44.)

[CP04] Patrick L. Combettes and Jean-Christophe Pesquet, *Image resolution subject to a total variation constraint*, IEEE Trans. o **13** (2004), no. 9, 1213–1222. (Cited on page 44.)

[CYX04] H. Chang, D. Yeung, and Y. Xiong, *Super-resolution through neighbor embedding*, in Proc. IEEE CVPR (2004), 275–282. (Cited on pages 150, 151, 158, 160, 165, 166, 168, 173, 174, 176, 177, 178, 187, 188, 189, 191, 196, 211, 212 and 213.)

[CZ06] Tak Ming Chan and Junping Zhang, *An improved super-resolution with manifold learning and histogram matching*, Advances in Biometrics, Lecture Notes in Computer Science, vol. 3832, Springer Berlin Heidelberg, 2006, pp. 756–762. (Cited on page 161.)

[CZPH09] T. M. Chan, J. Zhang, J. Pu, and H. Huang, *Neighbor embedding based super-resolution algorithm through edge detection and feature selection*, Pattern Recognition Letters **30** (2009), no. 5, 494–502. (Cited on page 161.)

[DD12] Julie Delon and Agnes Desolneux, *A patch-based approach for random-valued impulse noise removal*, ICASSP, IEEE, 2012, pp. 1093–1096. (Cited on page 28.)

[DDD04] I. Daubechies, M. Defriese, and C. DeMol, *An iterative thresholding algorithm for linear inverse problems with a sparsity constraint*, Commun. Pure Appl. Math. **57** (2004), 1413–1457. (Cited on page 64.)

[DFKE07] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, *Image denoising by sparse 3d trans-form-domain collaborative filtering*, IEEE Trans. on Image Process. **16** (2007), no. 8, 2080–2095. (Cited on pages 28, 67, 70, 72, 94, 98, 112, 113, 114, 115, 116, 117, 134, 135, 136, 137, 138, 195, 206, 207, 209 and 210.)

[DK00] F. Dibos and G. Koepfler, *Global total variation minimization*, SIAM J. Numer. Anal. **37** (2000), no. 2, 646–664. (Cited on pages 38, 39, 40, 46, 47, 203 and 204.)

[DlZS11] W. Dong, X. li, L. Zhang, and G. Shi, *Sparsity-based image denoising via dictionary learning and structural clustering*, CVPR, IEEE, 2011, pp. 457–464. (Cited on pages 28 and 66.)

[Dou09] G. Dougherty, *Digital image processing for medical applications*, Cambridge University Press, 2009. (Cited on pages 5, 6, 7, 12, 14, 15, 18, 19 and 203.)

[DTD10] C. A. Deledalle, F. Tupin, and L. Denis, *Poisson nl means: unsupervised non local means for poisson noise*, In Proc. of the 17th IEEE Int. Conf. on Image Processing (ICIP), IEEE, 2010, pp. 801–804. (Cited on pages 50, 51, 52, 54, 73, 75, 76, 120, 124, 125, 127, 128, 205, 206 and 209.)

[EA06] M. Elad and M. Aharon, *Image denoising via sparse and redundant representations over learned dictionaries*, IEEE Trans. on Image Process. **15** (2006), no. 2, 3736–3745. (Cited on pages 28, 63, 66, 67, 70, 72, 94, 98, 112, 113, 114, 115, 116, 117, 134, 135, 136, 137, 138, 195, 206, 207, 209 and 210.)

[EAH00] K. Engan, S.O. Aase, and J.H. Husoy, *Multi-frame compression: Theory and design*, EURASIP Signal Processing **80** (2000), no. 10, 2121–2140. (Cited on page 65.)

[ED07] M. Elad and D. Datsenko, *Example-based regularization deployed to super-resolution reconstruction of a single image*, The Computer Journal **52** (2007), no. 1, 15–30. (Cited on page 155.)

[EF99] M. Elad and A. Feuer, *Super-resolution restoration of an image sequence: Adaptive filtering approach*, IEEE Transactions on Image Processing **8** (1999), 387–395. (Cited on page 148.)

[EHJT04] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, *Least angle regression*, Ann. Statist. **32** (2004), no. 2, 407–499. (Cited on page 64.)

[Ela10] M. Elad, *Sparse and redundant representations: From theory to applications in signal and image processing*, Springer, 2010. (Cited on pages 64, 65 and 150.)

[FBF77] J. H. Friedman, J. L. Bentley, and R. A. Finkel, *An algorithm for finding best matches in logarithmic expected time*, ACM Transactions on Mathematical Software **3** (1977), 209–226. (Cited on pages 107 and 171.)

[FH03] V. Franc and V. Hlavac, *An iterative algorithm learning the maximal margin classifier*, Pattern Recognition **39** (2003), no. 9, 1985–1996. (Cited on pages 169 and 170.)

[FJP02] W. T. Freeman, T. R. Jones, and E. C. Pasztor, *Example-based super-resolution*, IEEE Comp. Graph. and Appl. **22** (2002), no. 2, 56–65. (Cited on pages 150, 151, 153, 155, 166, 173, 174, 175, 176, 178, 187, 188, 189, 191, 196, 212 and 213.)

[FM02] Xiaoguang Feng and Peyman Milanfar, *Multiscale principal components analysis for image local orientation estimation*, Proceedings of the 36th Asilomar Conference on Signals, Systems and Computers, 2002, pp. 478–482. (Cited on pages 60 and 87.)

[Foi11] A. Foi, *Noise estimation and removal in mr imaging: the variance-stabilization approach*, Proc. 2011 IEEE Int. Symposium on Biomedical Imaging, ISBI 2011 (Chicago (IL), USA), IEEE, April 2011. (Cited on page 73.)

[FP00] W. Freeman and E. Pasztor, *Markov networks for super-resolution*, Proc. of 34th Annual Conference on Information Sciences and Systems, Princeton University, 2000. (Cited on pages 150, 151, 153, 155, 156, 157, 158, 163, 165, 166, 168 and 211.)

[FREM04a] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, *Advances and challenges in super-resolution*, International Journal of Imaging Systems and Technology **14** (2004), 47–57. (Cited on page 148.)

[FREM04b] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, *Fast and robust multi-frame super resolution*, IEEE Trans. Image Process. **13** (2004), no. 10, 1327–1344. (Cited on page 148.)

[FY07] Wei Fan and Dit-Yan Yeung, *Image hallucination using neighbor embedding over visual primitive manifolds*, IEEE conference on Computer Vision and Pattern Recognition (CVPR), 2007, pp. 1–7. (Cited on page 161.)

[GBG04] P. Gravel, G. Beaudoin, and J.A. De Guise, *A method for modeling noise in medical images*, IEEE Trans. Med. Imag. **10** (2004), no. 23, 1221–1232. (Cited on page 31.)

[GBI09] D. Glasner, S. Bagon, and M. Irani, *Super-resolution from a single image*, ICCV, 2009. (Cited on page 150.)

[GEW10] Ali Gholipour, Judy A. Estroff, and Simon K. Warfield, *Robust super-resolution volume reconstruction from slice acquisitions: Application to fetal brain mri*, IEEE Trans. Med. Imaging **29** (2010), no. 10, 1739–1758. (Cited on page 148.)

[Giu84] E. Giusti, *Minimal surfaces and functions of bounded variation*, Birkhauser Verlag, Basel, 1984. (Cited on page 37.)

[GP95] H. Gudbjartsson and S. Patz, *The rician distribution of noisy mri data*, Magn. Reson. Med. **34** (1995), 910–914. (Cited on pages 19, 33, 34 and 144.)

[GTV11] P. Getreuer, M. Tong, and L. Vese, *A variational model for the restoration of mr images corrupted by blur and rician noise*, ISVC (1), 2011, pp. 686–698. (Cited on pages 40, 42, 97, 100, 101, 102, 204, 207 and 208.)

[HBBW98] R.C. Hardie, J.G. Bognar, K.J. Barnard, and D. A. Watson, *High-resolution image reconstruction from a sequence of rotated and translated frames and it's application to an infrared imaging system*, Optical Engineering **37** (1998), 247–260. (Cited on page 148.)

[IP91] M. Irani and S. Peleg, *Improving resolution by image registration*, CVGIP: Graphic Models and Image Processing **53** (1991), 231–239. (Cited on page 148.)

[IS93] M. Irani and S.Peleg, *Motion analysis for image enhancement: resolution, occlusion and transparency*, JVCI, 1993. (Cited on pages 173 and 185.)

[Jan06] Jiri Jan, *Medical image processing, reconstruction and restoration: Concepts and methods*, Taylor and Francis Group, 2006. (Cited on page 7.)

[KB06] C. Kervrann and J. Boulanger, *Optimal spatial adaptation for patch-based image denoising*, IEEE Trans. Image Process **15** (2006), 2866–2878. (Cited on page 49.)

[KBC07] C. Kervrann, J. Boulanger, and P. Coupe, *Bayesian non-local means filter, image redundancy and adaptive dictionaries for noise removal*, in Proc. Conf. Scale-Space and Variational Methods, Ischia, Italy (2007), 520–532. (Cited on pages 49 and 55.)

[KBPS11] F. Knoll, K. Bredies, T. Pock, and R. Stollberger, *Second order total generalized variation (tgv) for mri*, Magn. Reson. Med. **65** (2011), no. 2, 480–491. (Cited on pages 28, 46, 47, 48, 51, 52, 62, 63, 70, 72, 195, 204, 205 and 206.)

[KK08] K. I. Kim and Y. Kwon, *Example-based learning for single-image super-resolution*, Proc. DAGM (2008), 456–465. (Cited on page 150.)

[KKWV05] K. Krissian, R. Kikinis, C. F. Westin, and K. Vosburgh, *Speckle-constrained filtering of ultrasound images*, Computer Vision and Pattern Recognition (CVPR), IEEE, 2005, pp. 547–552. (Cited on page 43.)

[LCA07] T. Le, R. Chartrand, and T.J. Asaki, *A variational approach to reconstructing images corrupted by poisson noise*, Journal of Mathematical Imaging and Vision **27** (2007), no. 3, 257–263. (Cited on pages 40, 41, 73, 75, 76, 120, 124, 125, 127, 128, 204, 206 and 209.)

[LFB94] M. Lindenbaum, M. Fischer, and A.M. Bruckstein, *On gabor's contribution to image-enhancement*, Pattern Recognition **27** (1994), 1–8. (Cited on page 28.)

[LFSK06] C. Liu, W. T. Freeman, R. Szeliski, and S. B. Kang, *Noise estimation from a single image*, CVPR, vol. 1, 2006, pp. 901–908. (Cited on page 86.)

[LGS99] T. M. Lehmann, C. Gonner, and K. Spitzer, *Survey: Interpolation methods in medical image processing*, IEEE Trans. on Medical Imaging **18** (1999), no. 11, 1049–1075. (Cited on page 148.)

[LHLC12] Z. Liao, S. Hu, M. Li, and W. Chen, *Noise estimation for single-slice sinogram of low-dose x-ray computed tomography using homogenous patch*, Mathematical Problems in Engineering **2012** (2012), 16. (Cited on page 86.)

[LHLL01] Hongbing Lu, Ing-Tsung Hsiao, Xiang Li, and Zhengrong Liang, *Noise properties of low-dose ct projections and noise treatment by scale transformations*, Nuclear Science Symposium Conference Record, vol. 3, 2001, pp. 1662–1666. (Cited on page 78.)

[Li09] Dalong Li, *Support vector regression based image denoising*, Image Vision Comput. **27** (2009), no. 6, 623–627. (Cited on pages 79, 80 and 106.)

[LLHL02] H. Lu, X. Li, I. T. Hsiao, and Z. Liang, *Analytical noise treatment for low-dose ct projection data by penalized weighted least squares smoothing in the k-l domain*, Proc. SPIE, Medical Imaging **4682** (2002), 146–152. (Cited on pages 19, 31, 126 and 144.)

[LMA89] T. Loupas, W. McDicken, and P. Allan, *An adaptive weighted median filter for speckle suppression in medical ultrasound image*, IEEE Trans. Circuits Syst. **36** (1989), 129–135. (Cited on page 34.)

[LS92] T. Lei and W. Sewchand, *Statistical approach to x-ray ct imaging and its applications in image analysis-part i: Statistical analysis of x-ray ct imaging*, IEEE Trans. Med. Imag. **11** (1992), 62–69. (Cited on pages 19, 31 and 144.)

[LSM07] Dalong Li, S. Simske, and R. M. Mersereau, *Image denoising through support vector regression*, ICIP, 2007. (Cited on pages 79, 80 and 106.)

[LWC+08] Y.-L. Liu, J. Wang, X. Chen, Y.-W. Guo, and Q.-S. Peng, *A robust and fast non-local means algorithm for image denoising*, Journal of Computer Science and Technology **23** (2008), 270–279. (Cited on page 49.)

[MA06] D.M. Mount and S. Arya, *Ann: A library for approximate nearest neighbor searching*, Ph.D. thesis, version 1.1.1, 2006. (Cited on page 107.)

[MBP+09] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, *Non-local sparse models for image restoration*, 12th International Conference on Computer Vision, IEEE, 2009, pp. 2272–2279. (Cited on page 66.)

[MCB+10] J V. Manjon, P. Coupe, A. Buades, L. Collins, and M. Robles, *Mri super-resolution using self-similarity and image priors*, Int. Journal of Biomedical Imaging. Article ID 425891 (2010). (Cited on page 148.)

[MCCL+08] J. V. Manjon, J. Carbonell-Caballero, J.J Lull, G. Garcia-Marti, L. Marti-Bonmati, and M. Robles, *Mri denoising using nonlocal means*, Medical Image Analysis **12** (2008), 514–523. (Cited on pages 50, 53, 54, 55, 56, 97, 100, 101, 102, 205, 207 and 208.)

[MCMB+10] J. V. Manjon, P. Coupe, L. Marti-Bonmati, D. L. Collins, , and M. Robles, *Adaptive non-local means denoising of mr images with spatially varying noise level*, Journal of Magnetic Resonance Imaging **31** (2010), no. 1, 192–203. (Cited on page 50.)

[MF11] M. Makitalo and A. Foi, *Optimal inversion of the anscombe transformation in low-count poisson image denoising*, IEEE Trans. on Image Process. **20** (2011), no. 1, 99–109. (Cited on pages 32, 73, 75, 76, 120, 124, 125, 127, 128, 206 and 209.)

[MGS11] A. Martin, J.F. Garamendi, and E. Schiavi, *Iterated rician denoising*, Proceedings of the IPCV'11, vol. 1, 2011, pp. 959–963. (Cited on page 40.)

[MHFZ11] Jianhua Ma, Jing Huang, Qianjin Feng, and Hua Zhang, *Low-dose computed tomography image restoration using previous normal-dose scan*, Medical Physics **38** (2011), no. 10, 5713–5731. (Cited on page 79.)

[Mod04] J. Modersitzki, *Numerical methods for image registration*, Numerical Mathematics and Scientific Computation, Oxford University Press, 2004. (Cited on page 29.)

[MP03] D. D. Muresan and T.W. Parks, *Adaptive principal components and image denoising*, Proc. ICIP, IEEE, 2003, pp. 101–104. (Cited on pages 86, 93 and 135.)

[MS03] G. N. Morano and J. P. Seibyl, *Technical overview of brain spect imaging: Improving acquisition and processing of data*, Journal of Nuclear Medicine Technology **31** (2003), no. 4, 191–195. (Cited on pages 13 and 203.)

[Nad64] E. A. Nadaraya, *On estimating regression*, Theory Prob. Appl. **9** (1964), 141–142. (Cited on page 104.)

[NN07] Karl S. Ni and Truong Q. Nguyen, *Image superresolution using support vector regression*, IEEE Transactions on Image Processing **16** (2007), no. 6, 1596–1610. (Cited on page 150.)

[Now99] R.D. Nowak, *Wavelet-based rician noise removal for magnetic resonance imaging*, IEEE Trans. on Image Process. **8** (1999), no. 10, 1408–1419. (Cited on pages 31 and 52.)

[OF97] J. M. Ollinger and J. A. Fessler, *Positron emission tomography*, IEEE Signal Processing Magazine **14** (1997), no. 1, 43–55. (Cited on pages 31 and 144.)

[Ouw06] J. Van Ouwerkerk, *Image super-resolution survey*, Image and Vision Comput. **26** (2006), no. 10, 1039–1052. (Cited on page 148.)

[PM90] P. Perona and J. Malik, *Scale-space and edge detection using anisotropic diffusion*, IEEE Trans. Pattern Anal. Mach. Intell. (1990), 629–639. (Cited on pages 28, 39, 40 and 203.)

[PPK03] S. C. Park, M. K. Park, and M. G. Kang, *Super-resolution image reconstruction: a technical overview*, IEEE Signal Processing Magazine **20** (2003), no. 3, 21–36. (Cited on pages 148 and 149.)

[PPLA03] A. Pizurica, W. Philips, I. Lemahieu, and M. Acheroy, *A versatile wavelet domain noise filtration technique for medical imaging*, IEEE Trans. on Medical Imaging **23** (2003), no. 3, 323–331. (Cited on pages 97, 100, 101, 102, 207 and 208.)

[PPM+09] Vamsi K. Potluru, Sergey M. Plis, Morten Mørup, Vincent D. Calhoun, and Terran Lane, *Efficient multiplicative updates for support vector machines*, SDM'09, 2009, pp. 1218–1229. (Cited on page 110.)

[PRK93] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad, *Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition*, Proc. Conf. Signals, Systems and Computers Record of The Twenty-Seventh Asilomar Conf., 1993. (Cited on page 64.)

[RCL+04] D. Robinson, S. J. Chiu, J. Y. Lo, C. Toth, J. A. Izatt, and S. Farsiu, *Novel applications of super-resolution in medical imaging, in super-resolution imaging*, IEEE Trans. Image Process. **13** (2004), no. 4, 600–612. (Cited on page 148.)

[ROF92] L.I. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Physica D **60** (1992), 259–268. (Cited on pages 28, 38 and 195.)

[RPJS10] J. Rajan, D. Poot, J. Juntu, and J. Sijbers, *Noise measurement from magnitude mri using local estimates of variance and skewness*, Physics in Medicine and Biology **55** (2010), 441–449. (Cited on page 86.)

[SBMB09] A. Sawatzky, C. Brune, J. Muller, and M Burger, *Total variation processing of images with poisson statistics*, Proceedings of the 13th International Conference on Computer Analysis of Images and Patterns, Springer, 2009, pp. 533–540. (Cited on page 40.)

[SdDA+98] J. Sijbers, A.J. den Dekker, J. Van Audekerke, M. Verhoye, and D. Van Dyck, *Estimation of the noise in magnitude mr images*, Magnetic Resonance Imaging **16** (1998), 87–90. (Cited on page 86.)

[SGV98] C. Saunders, A. Gammerman, and V. Vovk, *Ridge regression learning algorithm in dual variables*, Proceedings of the 15th International Conference on Machine Learning (ICML98) (Madison-Wisconsin), 1998, pp. 515–521. (Cited on page 83.)

[SLZZ04] H. Shen, P. Li, L. Zhang, and Y. Zhao, *A map algorithm to super-resolution image reconstruction*, Proceedings of Third International Conference on Image and Graphics, 2004, pp. 544–547. (Cited on page 148.)

[Spr95] Perry Sprawls, *Physical principles of medical imaging*, Medical Physics Publishing, 1995. (Cited on pages 6 and 7.)

[SS08] J. Seabra and J. Sanches, *Modeling log-compressed ultrasound images for radio frequency signal recovery*, IEEE Engineering in Medicine and Biology Society., 2008. (Cited on page 34.)

[SSL02] Fei Sha, Lawrence K. Saul, and Daniel D. Lee, *Multiplicative updates for nonnegative quadratic programming in support vector machines*, NIPS, 2002, pp. 1041–1048. (Cited on pages 110, 133 and 184.)

[SSXS08] J. Sun, J. Sun, Z. Xu, and H. Shum, *Image superresolution using gradient profile prior*, In CVPR, 2008. (Cited on page 150.)

[Ste81] C. M. Stein, *Estimation of the mean of a multivariate normal distribution*, The Annals of Statistics **9** (1981), no. 6, 1135–1151. (Cited on page 49.)

[TFM07] Hiroyuki Takeda, Sina Farsiu, and Peyman Milanfar, *Kernel regression for image processing and reconstruction*, IEEE Transactions on Image Processing **16** (2007), no. 2, 349–366. (Cited on pages 28, 58, 59, 60, 62, 63, 65, 66, 67, 72, 94, 98, 112, 113, 114, 115, 116, 117, 134, 135, 136, 137, 138, 195, 205, 206, 207, 209 and 210.)

[TH99] T. R. Tuinstra and R. C. Hardie, *High-resolution image reconstruction from digital video by exploitation of non-global motion*, Optical Engineering **38** (1999), 806–814. (Cited on page 148.)

[TK10] T. Thaipanich and C.-C. J Kou, *An adaptive nonlocal mean scheme for medical image denoising*, Proc. SPIE 7623, 76230M, 2010. (Cited on page 50.)

[TKG95] B. C. Tom, A. K. Katsaggelos, and N.P. Galatsanos, *Reconstruction of a high resolution image by simultaneous registration, restoration, and interpolation of low-resolution images*, IEEE International Conference on Image Processing, vol. 2, 1995, pp. 539–542. (Cited on page 148.)

[TLBL10] Y. W. Tai, S. Liu, M. S. Brown, and S. Lin, *Super resolution using edge prior and single image detail synthesis*, In CVPR, 2010. (Cited on page 150.)

[TLR+12a] Dinh Hoan Trinh, Marie Luong, Jean-Marie Rocchisani, Canh Duong Pham, Huy Dien Pham, and Francoise Dibos, *Image resolution enhancement by projection onto convex hull*, the 12th IEEE International Symposium on Signal Processing and Information Technology (ISSPIT) (Ho Chi Minh City, Vietnam), December 2012. (Cited on pages 147, 151, 166 and 168.)

[TLR+12b] ———, *An optimal weight model for single image super-resolution*, Int. Conf. on Digital Image Computing: Techniques and Applications (DICTA) (Fremantle, Australia), December 2012. (Cited on pages 147, 151 and 166.)

[TSP88] T.A. Tuthill, R.H. Sperry, and K.J. Parker, *Deviation from rayleigh statistics in ultrasonic speckle*, Ultrason. Imag. **10** (1988), 81–90. (Cited on page 34.)

[Vap95] V. N. Vapnik, *The nature of statistical learning theory*, Springer, 1995. (Cited on page 82.)

[vGM06] Bram van Ginneken and Adrienne Mendrik, *Image denoising with k-nearest neighbor and support vector regression*, ICPR (3), IEEE, 2006, pp. 603–606. (Cited on pages 79, 104, 105 and 106.)

[VK09] D. Van De Ville and M. Kocher, *Sure-based non-local means*, IEEE Signal Process. Lett. 16 (2009), 973–976. (Cited on page 49.)

[VSK85] Y. Vardi, L. A. Shepp, and L. Kaufman, *A statistical model for positron emission tomography*, J. Am. Stat. Assoc. **40** (1985), 8–20. (Cited on pages 31 and 144.)

[Wat64] G. S. Watson, *Smooth regression analysis*, Sankhya **26** (1964), 359–372. (Cited on page 104.)

[WBO08] Norman S. Williams, Christopher J. K. Bulstrode, and P. Ronan O'Connell, *Bailey and love's short practice of surgery 25th edition*, 25 ed., Hodder Arnold, 2008. (Cited on page 16.)

[WBSS04] Z. Wang, A. C. Bovik, H R. Sheikh, and E. P. Simoncelli, *Image quality assessment: from error visibility to structural similarity*, IEEE Trans. Image Process. **13** (2004), no. 4, 600–612. (Cited on pages 35, 36, 92 and 112.)

[WDPC+08] N. Wiest-Daessle, S. Prima, P. Coupe, S. Morrissey, and C. Barillot, *Rician noise removal by non-local means filtering for low signal-to-noise ratio mri: Applications to dt-mri*, In Proc. of the 11th Int. Conf. on Medical Image Computing and Computer-Assited Intervention (MICCAI'08), 2008, pp. 171–179. (Cited on page 50.)

[Wei98] J. Weickert, *Anisotropic diffusion in image processing*, Stuttgart, Germany: Teubner-Verlag, 1998. (Cited on page 44.)

[WZG10] Jinjun Wang, Shenghuo Zhu, and Yihong Gong, *Resolution enhancement based on learning the sparse association of image patches*, Pattern Recognition Letters **31** (2010), no. 1, 1–10. (Cited on pages 151, 157, 161, 162, 163, 166 and 180.)

[XM11] Wei Xu and Klaus Mueller, *A reference image database approach for nlm filter-regularized ct reconstruction*, 11th Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine, July 2011, pp. 116–119. (Cited on pages 79, 104, 112, 113, 114, 115, 116, 117, 134, 135, 136, 137, 138, 209 and 210.)

[YHY10] C. Y. Yang, J. B. Huang, and M. H. Yang, *Exploiting self-similarities for single frame super-resolution*, Asian Conference on Computer Vision (ACCV), 2010. (Cited on pages 150 and 151.)

[YLL+09] L. Yu, X. Liu, S. Leng, J.M. Kofler, J. C. Ramirez-Giraldo, M. Qu, J. Christner, J. G. Fletcher, and C. H. McCollough, *Radiation dose reduction in computed tomography: techniques and future perspective*, Imaging in Medicine **1** (2009), no. 1, 65–84. (Cited on page 78.)

[YWHM08] J.C. Yang, J. Wright, T. Huang, and Y. Ma, *Image super-resolution as sparse representation of raw image patches*, IEEE Computer Vision and Pattern Recognition (CVPR) (2008), 1–8. (Cited on pages 150, 151, 161, 162, 163, 166, 180, 185 and 196.)

[YWHM10] J.C. Yang, J. Wright, T.S. Huang, and Y. Ma, *Image super-resolution via sparse representation*, IEEE Trans. on Image Process. **19** (2010), no. 11, 2861–2873. (Cited on pages 150, 151, 161, 162, 163, 166, 180, 187, 188, 189, 191 and 213.)

[YWL+12] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, *Couple dictionary training for image super-resolution*, IEEE Trans. on Image Process. **21** (2012), no. 8, 3467–3478. (Cited on pages 150, 161, 162, 163, 164, 165, 166, 180, 196 and 211.)

[ZEP10] R. Zeyde, M. Elad, and M. Protter, *On single image scale-up using sparse-representations*, Curves and Surfaces, Avignon-France (2010), 711–730. (Cited on pages 150, 151, 161, 162, 163, 166 and 180.)

[ZY11] Z. Zin and X. Yang, *A variational model to remove the multiplicative noise in ultrasound images*, J. Math. Imaging Vis. **39** (2011), 62–74. (Cited on pages 43, 57, 58, 204 and 205.)