

Image Denoising via Correlation Based Sparse Representation and Dictionary Learning

Gulsher Lund Baloch

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical and Electronic Engineering

Eastern Mediterranean University
January 2018
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Assoc. Prof. Dr. Ali Hakan Ulusoy
Acting Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Doctor of Philosophy in Electrical and Electronic Engineering.

Prof. Dr. Hasan Demirel
Chair, Department of Electrical and
Electronic Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis of the degree of Doctor of Philosophy in Electrical and Electronic Engineering.

Prof. Dr. Hüseyin Özkaramanlı
Supervisor

Examining Committee

1. Prof. Dr. Bilge Günsel

2. Prof. Dr. Erhan A. İnce

3. Prof. Dr. Osman Kükrer

4. Prof. Dr. Hüseyin Özkaramanlı

5. Prof. Dr. Bülent Sankur

ABSTRACT

Error-based Orthogonal Matching Pursuit (OMP_e) employed in many image denoising algorithms (e.g., K-means Singular Value Decomposition (K-SVD) algorithm) tries to reconstruct the clean image patch by projecting the observed noisy patch onto a dictionary and picking the atom with maximum orthogonal projection. This approach does indeed minimize the power in the residual. However minimizing the power in the residual does not guarantee that selected atoms will match the clean image patch. This leaves behind a residual that contains structures from the clean image patch. This problem becomes more pronounced at high noise levels. Firstly, we develop a simple method to prove that autocorrelation of residual does not match that of the contaminating noise. Then we propose a correlation-based sparse coding algorithm that is better able to pick the atom that matches the clean patch. This is achieved by picking atoms that force the residual patch to have autocorrelation similar to the autocorrelation of contaminating noise. Autocorrelation-based sparse coding and dictionary update stages are iterated and dictionaries are learned from noisy image patches. Also, a new residual correlation based regularization for image denoising is developed. The regularization can effectively render residual patches as uncorrelated as possible. It allows us to derive analytical solution for sparse coding (atom selection and coefficient calculation). It also leads to a new online dictionary learning update. The clean image is obtained by alternating between the two stages of sparse coding and dictionary updating. Experimental results of peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) show that the proposed algorithm can significantly outperform the K-SVD denoising algorithm, especially at

high noise levels. The proposed algorithm is compared with the K-SVD denoising algorithm, BM3D, NCSR and EPLL algorithms. Our results indicate that the proposed algorithm is better than K-SVD and EPLL denoising. The proposed algorithm gives visual results that are comparable or better than BM3D and NCSR algorithms.

Keywords: Correlation regularization, dictionary learning, image denoising, residual correlation, sparse representation.

ÖZ

Pek çok görüntü temizleme algoritmasında (örnek: K-means tekil değer ayrıştırma (K-SVD) algoritması) kullanılan hata-tabanlı ortogonal eşleştirme kovalama (Error-based Orthogonal Matching Pursuit (OMP_e)), temiz görüntü yamasını tanımlayabilmek için gözlenen gürültülü yamanın izdüşümü bir sözlük üzerine yansıtılır ve maksimum ortogonal izdüşümlü atom seçilir. Bu yaklaşım, artık işaretin gücünü minimize eder. Ancak artık işaretin gücünü minimize etmek, seçilen atomların temiz görüntü yamasıyla eşleşeceğini garanti etmez. Bu durumda artık işaret, temiz görüntü yamasından yapılar içerir. Bu sorun, yüksek gürültü seviyelerinde daha belirgindir. İlk olarak, artık işaretin oto-korelasyonunun gürültüyle eşleşmediğini gösterecek basit bir yöntem geliştirilmiştir. Daha sonra, temiz yama ile eşleşen atomu seçebilen, korelasyona dayanan bir seyrek kodlama algoritması sunulmuştur. Bu amaçla, artık işaret, gürültünün otokorelasyonuna benzer bir otokorelasyona sahip olmaya zorlanmıştır. Otokorelasyona dayanan seyrek kodlama ve sözlük güncelleme aşamaları yinelenmiş ve sözlükler gürültülü görüntü yamaları ile eğitilmiştir. Bunun yanında, görüntü temizleme için, artık işaretin korelasyonuna dayanan yeni bir düzenleme geliştirilmiştir. Bu düzenleme, artık yamaların mümkün olduğu kadar korelasyonsuz olmasını sağlar. Bu durumda, seyrek kodlama için analitik çözümler (atom seçme ve katsayı hesaplama) elde edilebilir. Bu da yeni bir çevrimiçi sözlük öğrenme güncellemesi geliştirilmesine olanak sağlar. Temiz görüntü, seyrek kodlama ve sözlük güncelleme beraber uygulanarak elde edilir. Denemeler sonucu elde edilen işaret gürültü oranı ve yapısal benzerlik indisi değerleri, önerilen algoritmanın, özellikle yüksek gürültü değerlerinde, K-SVD algoritmasından daha iyi yanıtlar verdiğini göstermektedir.

algoritma, K-SVD gürültü temizleme algoritması, BM3D, NCSR ve EPLL algoritmalarıyla karşılaştırılmıştır. Sonuçlar, önerilen algoritmanın K-SVD ve EPLL gürültü temizleme algoritmalarından çok daha iyi çalıştığını göstermektedir. Önerilen algoritma, BM3D ve NCSR algoritmalarıyla karşılaştırılabilecek düzeyde veya daha iyi görsel sonuçlar vermektedir.

Anahtar Kelimeler: Korelasyon düzenlemesi, sözlük öğrenmesi, görüntü temizleme, seyrek temsiliyet.

Dedicated to

My parents, family, friends and teachers who have always been supportive during
my Doctoral studies

ACKNOWLEDGMENT

Special thanks to my worthy supervisor Prof. Dr. Hüseyin Özkaramanlı, who has guided, supported and encouraged me throughout my PhD program. His hard work, wisdom and understanding towards this research work were key factors for its success.

I am very grateful to Prof. Dr. Runyi Yu for his instrumental guidance and support throughout my PhD studies and especially for his untiring efforts for this research work.

TABLE OF CONTENTS

ABSTRACT.....	iii
ÖZ.....	v
ACKNOWLEDGMENT.....	viii
LIST OF TABLES.....	xiv
LIST OF FIGURES.....	xv
LIST OF SYMBOLS AND ABBREVIATIONS.....	xix
1 INTRODUCTION.....	1
1.1 Introduction.....	1
1.3 Thesis Objectives.....	4
1.4 Thesis Contribution.....	4
1.5 Thesis Overview.....	5
2 STATE-OF-THE ART METHODS IN IMAGE DENOISING.....	6
2.1 Introduction.....	6
2.2 Types of Image Noises.....	6
2.2.1 Gaussian Noise.....	6
2.2.2 White Noise.....	7
2.2.3 Impulse Valued Noise.....	7
2.2.4 Quantization Noise.....	8
2.2.5 Speckle Noise.....	8
2.2.6 Photon Noise.....	9

2.3 Inverse Problems.....	9
2.4 Image Denoising	10
2.5 Regularization	10
2.6 Sparse Representation	11
2.7 Types Of Sparse Representation Algorithms.....	12
2.7.1 Sparse Coding Based On Greedy Algorithms.....	13
2.7.1.1 Use of Matching Pursuit (MP) For Sparse Representation	13
2.7.1.2 Difference Between Matching Pursuit And Orthogonal Matching Pursuit.....	13
2.7.2 Sparse Representation Algorithms Based On L1 Norm.....	14
2.7.2.1 LASSO And Basis pursuit Sparse Representation Algorithms..	14
2.7.2.2 Sparse Representation Based On L-p Norm.....	15
2.8 Training Of Dictionary Atoms.....	15
2.8.1 Use Of The Method Of Optimized Directions (MOD)	16
2.8.2 Dictionary Learning Algorithm Based On K-SVD	17
2.8.3 Online Dictionary Learning (ODL).....	18
2.9 Image Denoising Via Sparse-Land Model.....	19
3 IMAGE DENOISING VIA CORRELATION BASED SPARSE REPRESENTATION AND DICTIONARY LEARNING.....	22
3.1 Introduction.....	22
3.2 Background	23
3.3 Motivation And Problem Definition.....	26

3.4 Proposed Correlation Based Sparse Coding Stage	29
3.4.1 Complexity Analysis	31
3.4.2 Limitations And Future Work	32
3.5 Types Metrics Used To Compare the Performance	33
3.6 Simulation And Results	33
3.6.1 Convergence Of Proposed Sparse Coding Algorithm OMPc	34
3.6.2 PSNR Results Comparison.....	35
3.6.3 Qualitative Comparison.....	40
3.7 Conclusion	44
4 RESIDUAL CORRELATION REGULARIZATION BASED IMAGE DENOISING.....	46
4.1 Introduction.....	46
4.2 Background.....	47
4.3 Motivation And Problem Statement	49
4.4 Residual Correlation Regularization.....	51
4.4.1 Sparse Coding.....	51
4.4.2 Dictionary Updating	54
4.5 Computational Complexity.....	59
4.6 Limitations And Future Work.....	59
4.7 Experimental Results And Comparison.....	60
4.7.1 Quantitative Performance Evaluation.....	60
4.7.1.1 Comparison Based On PSNR Results	60

4.7.1.2 Comparison Based On SSIM Results.....	66
4.7.1.3 Comparison Based On FSIM Results.....	68
4.7.2 Qualitative Experiments	69
4.8 Comprehensive Performance Evaluation At High Noise Levels.....	72
4.8.1 Performance Evaluation With Other Type Of Noises	73
4.8.2 Testing Variety Of Images At High Noise Levels	82
4.8.3 Testing With High Frequency Synthetic Images.....	84
4.9 Conclusion	86
5 COUPLED KSVD DICTIONARY LEARNING ALGORITHM IN WAVELET DOMAIN FOR SINGLE IMAGE SUPER-RESOLUTION	89
5.1 Introduction.....	89
5.2 Background	90
5.3 Image Super-Resolution.....	92
5.4 The Proposed Super-Resolution Approach.....	93
5.4.1 Dictionary Learning Based On Proposed Method.....	93
5.4.2 Image Reconstruction Based On Proposed Method.....	95
5.5 Simulation And Results	96
5.6 Conclusion	106
6 CONCLUSION AND FUTURE WORK	107
6.1 Conclusion	107
6.2 Future Work.....	108
6.2.1 Deep Learning	108

6.2.2 Application in Fingerprinting Algorithm	109
6.2.3 Simplifying Complexity of Algorithm by Proximal Calculus	109
6.2.4 Updating Multiple Dictionary Atoms.....	109
6.2.5 Group Sparsity.....	110
6.2.6 Analyzing the Performance by Varying Patch Sizes.....	110
6.2.6 Residual Correlation Based Single Image Super Resolution (SISR) ...	110
REFERENCES	111
APPENDICES	123
Appendix A: Derivation of expanded equation (4.6).....	124
Appendix B: Simplifying equation (4.10)	125
Appendix C: Derivation of equation (4.17) to obtain equation (4.21).....	127

LIST OF TABLES

Table 3.1: PSNR results in decibels. Top left K-SVD [1]. Top right: BM3D [6]. Bottom left EPLL [14]. Bottom right: Proposed algorithm	40
Table 3.2: PSNR results in decibels at various patch size and dictionary atoms.....	41
Table 4.1: PSNR results in decibels. Top left: KSVD [1]. Top right: NCSR [39]. Middle left: BM3D [6]. Middle right: EPLL [14]. Bottom left: Proposed algorithm.	62
Table 4.2: SSIM Results. Top left: KSVD [1]. Top right: NCSR [39]. Middle left: BM3D [6]. Middle right: EPLL [14]. Bottom left: Proposed algorithm.	67
Table 4.3: PSNR results for ACGN in decibels. Top left: KSVD [1]. Top right: NCSR [39]. Middle left: BM3D [6]. Middle right: EPLL [14]. Bottom left: Proposed algorithm.	74
Table 4.4: PSNR results for Laplacian noise in decibels. Top left: KSVD [1]. Top right: NCSR [39]. Middle left: BM3D [6]. Middle right: EPLL [14]. Bottom left: Proposed algorithm.	77
Table 4.5: PSNR results in decibels. Top left: KSVD [1]. Top right: NCSR [39]. Middle left: BM3D [6]. Middle right: EPLL [14]. Bottom left: Proposed algorithm.	82

LIST OF FIGURES

Figure 2.1: Image With Salt and Pepper Noise.....	7
Figure 2.2: Uniform Noise	8
Figure 3.1: Sum of nonzero lag autocorrelation versus number of iterations for Barbara image (a) $\sigma = 15$ (b) $\sigma = 50$ (c) $\sigma = 75$	35
Figure 3.2: Comparison of denoising results for Fingerprint image with noise level varying from 20 to 100	36
Figure 3.3: Comparison of denoising results for Barbara image with noise level varying from 20 to 100	38
Figure 3.4: Difference in PSNR comparison	39
Figure 3.5: Difference in PSNR comparison	39
Figure 3.6: Visual comparison of Fingerprint image with $\sigma = 100$, (a) original image (b) denoise by K-SVD [1], (c) denoised by EPLL [14], (d) denoised by BM3D [6], (e) denoised by KSVD _c	41
Figure 3.7: Barbara image reconstruction comparison with $\sigma = 50$, (a) original image (b) denoise by K-SVD [1], (c) denoised by EPLL [14], (d) denoised by BM3D [6], (e) denoised by KSVD _c	42
Figure 3.8: Visual comparison of Building image with $\sigma = 60$, (a) original image (b) denoise by K-SVD [1], (c) denoised by EPLL [14], (d) denoised by BM3D [6], (e) denoised by K-SVD _c	43
Figure 3.9: The trained dictionary for Fingerprint image with $\sigma = 100$ after 20 iterations (a) K-SVD [1], (b) K-SVD _c	44
Figure 3.10: The trained dictionary for Barbara image with $\sigma = 75$ after 20 iterations (a) K-SVD [1], (b) K-SVD _c	44

Figure 4.1: The flow chart of proposed algorithm.....	55
Figure 4.2: Neighborhood of current patch.....	58
Figure 4.3: PSNR results comparison.....	62
Figure 4.4: Difference in PSNR comparison for Barbara image.....	63
Figure 4.5: Difference in PSNR comparison for Fingerprint image.....	64
Figure 4.6: Difference in PSNR comparison for Straw image.....	65
Figure 4.7: PSNR comparison when $M = 1$ and $M = 2$	66
Figure 4.8: The SSIM results comparison.....	68
Figure 4.9: The FSIM results comparison.....	69
Figure 4.10: Visual comparison of Barbara image with $\sigma = 60$, (a) original image (b) denoise by K-SVD [1], (c) denoised by BM3D [6] (d) denoised by EPLL [14], (e) denoised by NCSR [39] (f) denoised by K-SVD _c	70
Figure 4.11: Visual comparison of Fingerprint image with $\sigma = 100$, (a) original image (b) denoise by K-SVD [1], (c) denoised by BM3D [6] (d) denoised by EPLL [14], (e) enoised by NCSR [39] (f) denoised by K-SVD _c	71
Figure 4.12: Visual comparison for Barbara image ($\sigma = 75$) and Fingerprint image ($\sigma = 100$).....	73
Figure 4.13: Frequency response of digital filter used to generate ACGN.....	74
Figure 4.14: Visual comparison for Barbara ($\sigma = 50$) corrupted with ACGN.....	76
Figure 4.15: Visual comparison for Fingerprint ($\sigma = 50$) corrupted with ACGN...	77
Figure 4.16: a Visual comparison for Barbara ($\sigma = 50$) corrupted with Laplacian noise.....	79
Figure 4.17: Visual comparison for Fingerprint ($\sigma = 50$) corrupted with Laplacian noise.....	80
Figure 4.18: PSNR results for ACGN and Laplacian noise.....	80

Figure 4.19: PSNR results for ACGN and Laplacian noise.....	81
Figure 4.20: PSNR results for ACGN and Laplacian noise.....	81
Figure 4.21: SSIM results for ACGN and Laplacian noise	82
Figure 4.22: SSIM results for ACGN and Laplacian noise	82
Figure 4.23: Visual comparison of Text image ($\sigma = 50$)	84
Figure 4.24: PSNR heat map of synthetic DCT images (with 5 coefficients).....	86
Figure 4.25: PSNR heat map of synthetic DCT images (with 15 coefficients).....	86
Figure 4.26: PSNR heat map of synthetic DCT images (with 25 coefficients).....	87
Figure 5.1: Example sets of HR subbands dictionary atoms (a) vertical details (b) horizontal details (c) diagonal details	95
Figure 5.2: PSNR results comparison.....	97
Figure 5.3: The proposed dictionary learning approach	98
Figure 5.4: The proposed super resolution approach.....	99
Figure 5.5: SSIM results comparison.....	100
Figure 5.6: Visual comparison for the image image number 1 in the kodak set	101
Figure 5.7: Visual comparison for the Boat image	102
Figure 5.8: Visual comparison of the zoomed Lena image. (a) Original Image, (b) Bicubic technique, (c) Algorithm of [23], (d) Algorithm of [29], (e) Algorithm of [35], (f) Proposed Algorithm	103
Figure 5.9: Visual comparison of the zoomed Peppers image. (a) Original Image, (b) Bicubic technique, (c) Algorithm of [23], (d) Algorithm of [29], (e) Algorithm of [35], (f) Proposed Algorithm	104
Figure 5.10: Visual comparison of the zoomed Barbara image. (a) Original Image, (b) Bicubic technique, (c) Algorithm of [23], (d) Algorithm of [29], (e) Algorithm of [35], (f) Proposed Algorithm	105

Figure 5.11: Samples of images used for dictionary learning 105

LIST OF SYMBOLS AND ABBREVIATIONS

α	Sparse coefficient vector
a_0^r	Power in the residual
σ	Standard deviation of noise
δ	Delta function
ψ	Blurring and down sampling operator
λ	Regularization weighting parameter
ε	Vector sparse approximation error tolerance
Sgn	Sign function
$\ \cdot \ _2$	Euclidean vector norm
$\ \cdot \ _0$	Number of nonzero elements in a vector
$\ \cdot \ _F$	Frobenius vector norm
a	Autocorrelation
D	A dictionary
K	Size of Dictionary
M	Number of neighboring residuals
N	signal space dimension
R	Residual patch
S	Maximum Number of nonzero elements
T	Transpose operator
Tr	Trace operator
x	A vector signal
AWGN	Additive White Gaussian Noise

BM3D	Image Denoising by Sparse 3-D Transform Domain Collaborative Filtering
BMP	Basis matching pursuit
BP	Basis Pursuit
BPD	Basis Pursuit Algorithm
DWT	Discrete wavelet transform
EPLL	Expected Patch Log Likelihood
FOCUSS	Focal underdetermined system solver
HR	High Resolution
K-SVD	k-means Singular value Decomposition
LASSO	Least Absolute Shrinkage and Selection Operator
LR	Low Resolution
LSE	Least-Squared Error
MOD	Method of optimized directions
MP	Matching Pursuit
MR	Middle Resolution
MSE	Mean-Squared Error
NCSR	Nonlocally Centralized Sparse Representation
NP	Non-deterministic polynomial-time
ODL	Online dictionary learning
OMP	Orthogonal Matching Pursuit
OMP _e	Error based Orthogonal Matching Pursuit
PSNR	Peak Signal-to-Noise Ratio
SISR	Single Image Super Resolution
SR	Super-Resolution
SSIM	Structural Similarity Index

Chapter 1

INTRODUCTION

1.1 Introduction

Noise is one of the major barriers in the way of quality data communication or data processing. Data can be in the form of text, voice, image or video. In this research work, we shall focus on data in the form of images. Images often contain noise due to various reasons like deficiency in sensors, reduced illumination, or communication errors. This noise should be removed in order to process or communicate image properly. The process of recovering or reconstructing an image from noise is called image denoising. This is considered as very important and simple inverse problem. It has many applications especially in computer vision and in medical imaging. Numerous attempts have been made to address this issue in last 50 years. However, in last two decades, sparse representation and redundant dictionary learning based approach has been one of the prominent methods for image denoising. Sparse representation and dictionary learning based model is commonly known as sparse-land model [1]. In this research work, we have focused on sparse-land model based image denoising algorithms. Sparse representation of any signal is a linear combinations of a few number of bases. These bases can be fixed such as wavelets, contourlets, Fourier basis functions, and the discrete cosine transform. However, due to their limitations, off-the-shelf bases are replaced by redundant set of trained bases called dictionary. Dictionary is trained by available training data so that it becomes

adequate to its function. This process of bases training is also known as dictionary learning.

K-means Singular Value Decomposition (K-SVD) is one of the benchmark methods [1, 2] based on sparse-land model. In this algorithm, noisy image is divided small overlapping square portions called patches. Then, current patch is assumed as residual (removed noise) and error based Orthogonal Matching Pursuit (OMP_e) algorithm is applied to approximate the clean patch. Second step is to update the dictionary based on known sparse representations. These two steps are iterated for few times. Finally, recovered patches are combined to reconstruct the original image. In this research work, the sparse coding and dictionary update stages are modified to improve the performance of sparse-land model based image denoising algorithms.

1.2 Problem Definition

The performance of sparse representation and dictionary learning (sparse-land model) based image denoising algorithms have been highly remarkable in the last two decades. In these patch based image denoising algorithms, the objective is to approximate the clean patch buried in noise. It is achieved by calculating the maximum orthogonal projection (inner product) between noisy patch and dictionary atom. An atom that gives maximum orthogonal projection and its corresponding sparse coefficient are used to approximate the noisy patch. Hence, an atom that matched clean image patch buried in noise is picked and noise (residual) is removed. Initially it is assumed that patch itself is a residual. Then, each time an atom is selected then new residual is calculated. This process continues till power of the residual goes below the noise power of the contaminating noise. This process works well at low noise levels but it fails at high noise levels [1]. It is due to fact that noise

process dominates the projection at high noise levels. In other words, when noise power is greater than signal power then an atom that matches the contaminating noise is selected and residual contains remnants from clean signal. This observation calls for studying the statistical properties of residual. An atom that produces the noise-like residual must be selected instead of an atom that produces maximum orthogonal projection. It is to note that if an atom that matches the noise is selected then contents of clean signal are lost in the form of residual. Hence, if atom that matches the image patch is selected, the residual becomes similar to the contaminating noise process.

In sparse coding stage, the information about statistics of the contaminating noise must be included for the better approximation of clean signal. In standard noise model, the additive white Gaussian noise (AWGN) with zero mean and known variance is used. Therefore, residual must possess statistical properties similar to the AWGN. In this research work, we develop a sparse coding stage where residual correlation is considered for picking the correct atom. In other words, we study correlation between the pixels in the residual during sparse coding stage. If pixels of residual patch are highly correlated then the selected atoms did not match the clean image patches. However, if pixels in the residual are highly uncorrelated then atom that matches the clean image patch is picked. This is achieved by forcing the autocorrelation of the residual patch to match the autocorrelation of contaminating noise. To achieve this objective, correlation based regularization is developed in this research work.

Our problem can be summarized as follows. Given a patch from noisy images, we aim to find a sparse code such that it gives a good approximation of the clean image,

and the resultant residual is uncorrelated to the residuals of the neighboring patches of the noisy image and also its internal patches are uncorrelated to each other.

1.3 Thesis Objectives

This thesis work is about understanding and analyzing the performance of sparse representation and dictionary update stages in image denoising algorithms. The main objectives of this research work are listed below:

1. Analyzing the usage of sparse representation and dictionary update stages for solving inverse problems in image processing.
2. Showing the reason behind limitation (given in literature) of sparse representation based image denoising algorithms.
3. Based on acquired knowledge, proposing a suitable solution to eliminate or at least reduce the magnitude of limitation of sparse representation based image denoising algorithms.
4. Implementing a dictionary learning algorithm in wavelet domain and analyzing its performance in image super-resolution.

1.4 Thesis Contribution

This research work is mainly focused on two major applications of image processing namely image denoising and image super-resolution. Its major contributions to each application are listed below:

1. Demonstrating the impact of picking an atom that gives maximum orthogonal projection on performance of image denoising.
2. Establishing the contribution of considering residual patch correlations for sparse coding in improving the performance of image denoising.
3. Introducing a new sparse coding strategy that picks an atom based on residual patch correlation to improve the performance of image denoising.

4. Developing a residual correlation regularization for sparse representation and dictionary update stages.
5. Introducing a new sparse coding algorithm and dictionary update stage based on residual correlation regularization for image denoising.
6. Presenting the performance of coupled K-SVD algorithm in wavelet domain for image super-resolution.

1.5 Thesis Overview

In Chapter 2 sparse-land model and its two major steps are discussed in details. Also brief literature review on types of sparse representation and dictionary learning algorithms is conducted. Chapter 3 presents the proposed image denoising via correlation-based sparse representation algorithm. It contains motivation, mathematical formulation and complexity analysis of the proposed algorithm. Finally its results are compared with state-of-the-art image denoising algorithms. In Chapter 4, we introduce a novel residual correlation regularization. A new sparse coding and dictionary update stages based on developed regularization are presented. Simulation results are compared with benchmark algorithms. Chapter 5 describes the coupled K-SVD dictionary learning algorithm in wavelet domain for single image super-resolution. Coupled K-SVD algorithm is implemented in wavelet domain. Dictionary learning and super-resolution approaches are proposed. Chapter 6 presents thesis conclusions. Also contribution of this thesis work is summarized. Future work based on this thesis work is also discussed.

Chapter 2

STATE-OF-THE ART METHODS IN IMAGE DENOISING

2.1 Introduction

Sparse-land model is one of the well-known models used for various applications of image processing. Due to its simplicity and effectiveness, it has become the standard model in the last two decades.

In this chapter, we shall discuss the methods used for sparse representation and also we shall summarize the famous dictionary learning algorithms. Finally, image denoising via sparse representation is summarized. However, since this research work is mainly based on image denoising, hence firstly the major type image noises are summarized in next section.

2.2 Types of Image Noises

Noise is defined as random unwanted signal that adds to desired signal and changes its originality. Data in any form can be corrupted by noise during acquisition, coding, transmission, and processing steps. Following are some well known types of noise.

2.2.1 Gaussian Noise

This model of noise is generated based on normal distribution with given mean and variance. This noise highly affects the gray values of image. Mostly Gaussian noise is generated by thermal vibration present inside atoms. The probability distribution function (PDF) is given by:

$$P(g) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(g-\mu)^2}{2\sigma^2}} . \quad (2.1)$$

Here g is the value of pixel, σ is standard deviation and μ is mean. Every Gaussian noise is not always white noise. Gaussian colored noise can be generated by passing white Gaussian noise through low pass or high pass filter [66].

2.2.2 White Noise

The term “white” is taken from white color where there are uniform emissions at all frequencies. Hence, here white noise has uniform power spectrum. Each pixel is uncorrelated from its neighboring pixel. Ideally, noise power in white noise is infinite (ranges from negative infinity to positive infinity in frequency domain) [66].

2.2.3 Impulse Valued Noise

Impulse valued noise also known as Salt and Pepper Noise. All pixel values are not affected by this kind of noise. Some of the pixels are changed due to Salt and Pepper noise. The affected values are changed to highest values or lowest value present in image. If pixel value is changed to lowest value due to the pepper noise than a dark spot or dead pixel is created in an image [66].

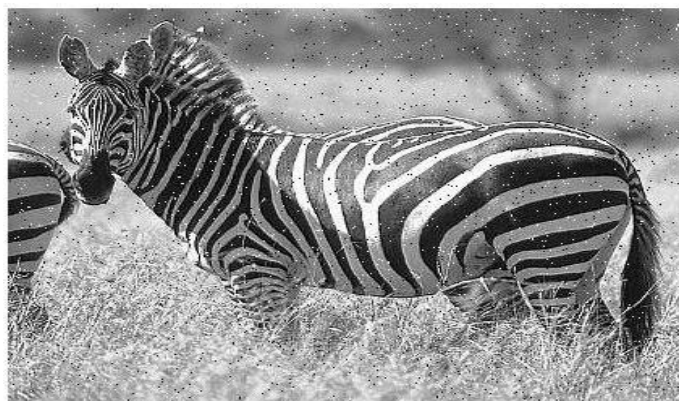


Figure 2.1: Image with Salt and Pepper Noise

2.2.4 Quantization Noise

When amplitude of the data is quantized then this change in amplitude is known as quantization error or quantization noise. It generally appears when analog information is converted to digital information. This type of noise follows the uniform distribution hence it is also known as uniform noise [66]. The PDF of quantization noise is given as:

$$P(g) = \frac{1}{b-a} \begin{cases} \text{if } a \leq g \leq b \\ 0 \text{ otherwise} \end{cases} \quad (2.2)$$

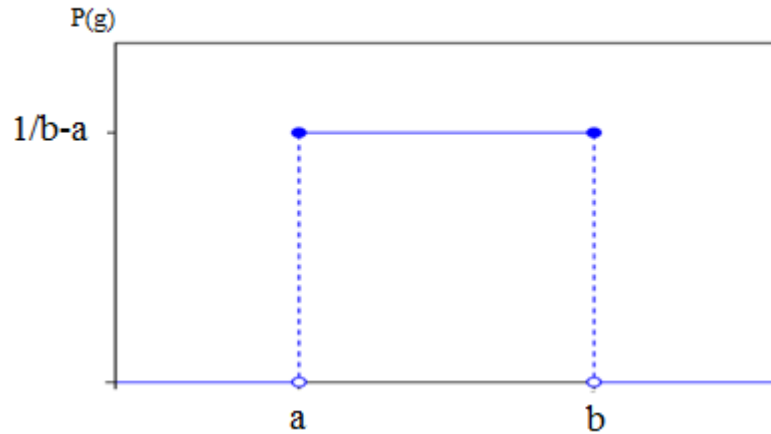


Figure 2.2: Uniform Noise.

Mean is given by $\mu = \frac{a+b}{2}$ and variance is $\sigma^2 = \frac{(b-a)^2}{12}$.

2.2.5 Speckle Noise

The PDF of a speckle noise is defined as a gamma distribution. Due to its multiplicative nature, it affects the radar, medical ultrasound and other such devices [66].

2.2.6 Photon Noise

This noise is modeled by Poisson distribution. Hence, it is also known as Poisson noise or Shot noise. Generally, this is produced due to electromagnetic waves such as gamma rays, x-rays e-t-c. Due to random movement of photons in sources of such rays the images obtained contains the spatial randomness [66].

2.3 Inverse Problems

Inverse problems are one of the very essential topics in the field of science. It is defined as the mathematical model used to extract unknown information from available observations [67]. In other words, we reverse the process in a sense that we develop a model based on observed measurement to extract unknown information. Therefore, given some previous knowledge about the lost data and some available information, the objective is to obtain missing data. Generally, inverse problems are ill posed and non linear. However, some additional information (regularization or prior information) about the unknown data plays key role to develop a model. They are very important in the field of signal processing, computer vision, medical imaging, astronomy, remote sensing, machine learning and many other fields [67].

Mathematically, if system of linear equations has more unknowns than the number of equations then either it has no solution or infinitely many solutions. Such system is known as a system of underdetermined linear equations. This system is often used to formulate a number of problems in image processing. For example image scale-up, image denoising, image super-resolution and many more. These problems are known as inverse problems. One can find infinite many solutions to these inverse problems. Image fusion is one of the inverse problems where information from two or more input images is combined to form a single input which contains more information

than any of the input images [68]. The source separation is another useful inverse process where original signal is recovered from combined signal that is formed by the number of signals mixed together [69]. Image super resolution is a process of recovering high resolution image from number of low resolution images available [70].

2.4 Image Denoising

Image denoising is one of the well known inverse problems. Noise should be removed from any form of data in order to improve the quality of data or prevent it from being lost. In literature, there are many methods to remove noise from data. Since, useful data to be extracted from noisy one is unknown, therefore, one of the well known method is to develop a model to best fit the noise in the data. Therefore, noise is modeled such that it becomes prominent and then it becomes easy task to remove it. Sparse representation and dictionary learning method is one of the very successful methods to denoise data. It projects the noisy data on a low dimensional subspace formed by linear combination of few atoms. This low dimensional projection makes sure that noise does not fit in this space and hence denoising is achieved.

In order to model a noise, it is very important to know the properties of noise. Some types of image noises are summarized in next section.

2.5 Regularization

One of the major hindrances in solving inverse problems is to find a suitable single solution out of infinite many solutions. A well known method to do this is a regularization function. This function examines the desirability of solutions and helps to find an appropriate solution. In literature, many regularization functions are used

like smoothness, adaptive smoothness, total variation and energy. However, sparse representation is one of the widely used regularization functions.

2.6 Sparse Representation

A solution to underdetermined system of linear equations having fewest nonzero entries is known as sparse representation or sparse approximation. Recently, finding the sparse solutions to underdetermined linear systems have become much more practical. Especially, data like image and video can also be sparsely represented using transform-domain methods.

Bases used for representation can be fixed like wavelets, contourlets, Fourier basis functions, and the discrete cosine transform. However, we shall focus on online basis training called dictionary learning.

A signal can be sparsely represented by searching a suitable basis from a dictionary. This sparse approximation process can be formulated as:

$$\underset{\alpha}{\operatorname{argmin}} \|\alpha\|_0 \text{ s.t. } \|\mathbf{x} - \mathbf{D}\alpha\|_2 \leq \varepsilon. \quad (2.3)$$

Note that *s. t.* refers to “subject to”.

The $\|\cdot\|_0$ and $\|\cdot\|_2$ operators denote ℓ_0 and ℓ_2 norm respectively. Whereas, \mathbf{x} is signal for approximation, $\alpha \in \mathbb{R}^n$ is sparse coefficient vector, $\mathbf{D} \in \mathbb{R}^{n \times K}$ is a dictionary (n is length of atoms (columns) in dictionary, K is number of atoms) and ε is maximum acceptable representation error. This is sparsest approximation for a signal $\mathbf{x} \in \mathbb{R}^n$ since it uses the ℓ_0 norm (number of nonzero entries in α).

In the last decade, dictionaries trained over example signals have become the topic of interest. Especially, redundant (over-complete) dictionaries ($K > n$) have great significance in image processing.

In terms of sparsity level, (2.1) can also be formulated as follows:

$$\underset{\alpha}{\operatorname{argmin}} \|\mathbf{x} - \mathbf{D}\alpha\|_2 \text{ s.t. } \|\alpha\|_0 < S \quad (2.4)$$

where S is sparsity limit.

This vector selection problem is computationally expensive and a non-deterministic polynomial-time (NP)-hard problem. The pursuit methods are used to solve this problem. Brief description of these sparse approximation methods is presented in next section.

2.7 Types Of Sparse Representation Algorithms

Sparse representation algorithms are divided into two major categories namely greedy algorithms and convex relaxation algorithms. This categorization is based mainly on the type of norm used to solve this NP hard problem. In greedy algorithms, the signal approximation process is carried out by minimizing the ℓ_0 norm iteratively. The algorithms such as matching pursuit (MP) [47, 48], orthogonal matching pursuit (OMP) [50, 44] and order recursive matching pursuit (ORMP) algorithms [52] are greedy algorithms. Whereas, convex relaxation approaches uses ℓ_1 minimization to further minimize the computational cost of the process.

2.7.1 Sparse Coding Based On Greedy Algorithms

In this section, well known greedy sparse approximation algorithms [55] are summarized with perspective of image processing. These methods work iteratively. Since this is a vector search process, the signal is represented iteratively with one atom at a time drawn from the dictionary till representation error goes below certain level. If bases are orthogonal then an atom that gives maximum inner product with a signal is picked. Mallat and Zhang in [43] gave basic greedy algorithms that led to other such algorithms.

2.7.1.1 Use of Matching Pursuit (MP) For Sparse Representation

Let signal \mathbf{x} be represented by $\mathbf{Q}_i = [\mathbf{d}_1 \dots \mathbf{d}_i]$ number of atoms chosen from a dictionary $\mathbf{D} = [\mathbf{d}_1 \dots \mathbf{d}_k]$ during iteration i . Then, MP iteratively solves the the following to sparsely represent signal :

$$\underset{\mathbf{d}_i, \alpha_i}{\operatorname{argmin}} \|\mathbf{x} - \mathbf{d}_i \alpha_i\|_2^2 \quad (2.5)$$

Here $\alpha_i = [\alpha_1 \dots \alpha_i]$ are the coefficients for selected atoms. Hence the approximation of \mathbf{x} is given by $\hat{\mathbf{x}} = \mathbf{d}_i \alpha_i$. Firstly, residual is initialized as $\mathbf{r}_i = \mathbf{x}$, and then an atom \mathbf{d} that gives maximum orthogonal projection with residual \mathbf{r}_i is picked for approximation. Hence, this inner product is given by $\mathbf{d}_i^T \mathbf{r}_i$ (Note that \mathbf{d}_i and \mathbf{r}_i are in vector form). Finally, residual is updated as $\mathbf{r}_i = \mathbf{x} - \hat{\mathbf{x}}$.

This process is repeated at each iteration until representation error goes below a certain level or maximum sparsity limit is reached [43].

2.7.1.2 Difference Between Matching Pursuit And Orthogonal Matching Pursuit

OMP algorithm is a modified version of the previously defined MP algorithm [43].

As discussed in the last section about MP, OMP also finds the best atom similarly and updates the residual for fixed number of iterations until stopping criteria is met.

However, the way of selecting an atom is different. In OMP, the atom that is selected for signal representation is eliminated from \mathbf{D} . Hence, an atom that is selected once cannot be selected again. An atom \mathbf{d}_i that gives maximum orthogonal projection with residual \mathbf{r}_{i-1} is selected as follows.

$$\operatorname{argmax}_{\mathbf{d}_i} |\mathbf{d}_i^T \mathbf{r}_{i-1}| \quad (2.6)$$

If \mathbf{Q}_i is the matrix of all the atoms selected then the representation coefficients are updated as $\boldsymbol{\alpha}_i = \mathbf{Q}_i^+ \mathbf{x}$. Here $\mathbf{Q}_i^+ = \mathbf{Q}_i(\mathbf{Q}_i^T \mathbf{Q}_i)^{-1}$ is the Moore-Penrose pseudo-inverse of matrix \mathbf{Q}_i . Finally, residual is updated before going to the next iteration. The advantage of OMP is that it does not consider the same atoms for selection again. Hence, computational complexity is reduced because number of atoms to be considered is reduced after each iteration.

2.7.2 Sparse Representation Algorithms Based On L1 Norm

The computational complexity of minimizing the ℓ_0 -norm is considered as major drawback of matching pursuit algorithms. Hence, in convex relaxation algorithms ℓ_0 -norm is relaxed with the ℓ_1 norm. The main advantage of using the ℓ_1 norm is the reduced computational complexity of sparse representation. Also, this reduction leads to standard optimization approaches [50] for sparse representation.

2.7.2.1 LASSO And Basis pursuit Sparse Representation Algorithms

Sparse representation of any given signal can be obtained by the basis pursuit (BP) algorithm which uses the ℓ_1 norm [45],

$$\operatorname{argmin}_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1 \text{ s. t } \mathbf{x} = \mathbf{D}\boldsymbol{\alpha} \quad (2.7)$$

It is to note that the ℓ_1 norm considers the value of entries only and not the number of entries.

The least absolute shrinkage and selection operator (LASSO) algorithm [57] is a type of BP algorithm. It is commonly known as basis pursuit denoising (BPD). In this algorithm, some restrictions are introduced in the ℓ_1 norm. This is given as follows:

$$\underset{\alpha}{\operatorname{argmin}} \|\mathbf{x} - \mathbf{D}\alpha\|_2 \text{ s.t. } \|\alpha\|_1 < S \quad (2.8)$$

where S is the sparsity limit. LASSO is the commonly used sparse approximation algorithm because sparsest solution can be obtained under the right conditions.

2.7.2.2 Sparse Representation Based On L-p Norm

The Focal Underdetermined System Solver (FOCUSS) approximation algorithm uses the ℓ_p ($p < 1$) norm for sparse representation. This is achieved by solving,

$$\underset{\alpha}{\operatorname{argmin}} \|\alpha\|_p \text{ s.t. } \mathbf{x} = \mathbf{D}\alpha \quad (2.9)$$

It is also used in many different applications since it has advantages of both classical optimization and learning-based algorithms.

2.8 Training Of Dictionary Atoms

Dictionary \mathbf{D} is the collection of bases used to sparsely represent any given signal. These bases are arranged in each column of a matrix. A dictionary may contain fixed bases like Fourier basis functions, wavelet frames, Gabor, etc. However, dictionary can also be trained from randomly chosen signals. In the last decade, over complete trained dictionaries are proved to be the best fit to a variety of signals [12], whereas, fixed dictionaries are unable to represent a wide variety of signals.

The dictionary is trained using random signals based on controlled parameters such that it adapts to the best signal approximation. This dictionary training is also known as dictionary learning (DL) [12]. This learning makes sure that trained dictionary

bases are optimal in representing a given signal and also representation is as sparse as possible.

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M] \in \mathbb{R}^{n \times M}$ be the random training signals. The representation coefficients $\mathbf{A} \in \mathbb{R}^{k \times M}$ are updated based on given signal and trained dictionary. Hence, DL is formulated as following optimization problem,

$$f(\mathbf{D}, \mathbf{A}) = \underset{\mathbf{D}, \mathbf{A}}{\operatorname{argmin}} \|\mathbf{X} - \mathbf{DA}\|_F^2. \quad (2.10)$$

$\|\cdot\|_F$ denotes the Frobenius norm.

Here \mathbf{D} is a matrix of trained atoms $[\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_K] \in \mathbb{R}^{n \times K}$. It is to note that initially during DL the sparse coefficient and dictionary atoms are unknowns. Therefore, this process is divided into two stages. During the first stage, the dictionary is assumed to be known and initialized with any random signals and sparse representation coefficients are obtained. Then, sparse approximation coefficients are fixed and dictionary is updated in the second stage. In the next sections, the most relevant of the state-of-the-art DL algorithms are summarized.

2.8.1 Use Of The Method Of Optimized Directions (MOD)

The MOD [49, 51] is a technique to design a frame and it is used with vector selection methods such as matching pursuit algorithms. In this method, dictionary update is considered as least square (LS) problem. In other words, under-determined set of equations are solved by LS solution using pseudo-inverse $\mathbf{D} = \mathbf{XA}^+$.

Dictionary is obtained by alternating between sparse approximation and dictionary update stages. MOD is proved to give local optimal solution.

2.8.2 Dictionary Learning Algorithm Based On K-SVD

The K-SVD dictionary learning algorithm is a well known method to train a dictionary for a number of signal processing applications [54]. In this algorithm the dictionary is trained based on singular value decomposition and it also uses the k -means clustering algorithm. The K-SVD algorithm tries to solve the following objective function for updating any atom \mathbf{d}_k :

$$\begin{aligned} f(\mathbf{D}, \mathbf{A}) &= \left\| \mathbf{X} - \sum_{j=1}^K \mathbf{d}_j \boldsymbol{\alpha}_j^T \right\|_F^2 = \left\| \left(\mathbf{X} - \sum_{j \neq k} \mathbf{d}_j \boldsymbol{\alpha}_j^T \right) - \mathbf{d}_k \boldsymbol{\alpha}_k^T \right\|_F^2 \\ &= \left\| \mathbf{E}_k - \mathbf{d}_k \boldsymbol{\alpha}_k^T \right\|_F^2 \end{aligned} \quad (2.11)$$

In the KSVD algorithm, a partial residual matrix \mathbf{E}_k is instrumental in updating sparse approximation and the dictionary atom jointly. The above defined function f is minimized by determining the best rank-one approximation to partial residual matrix \mathbf{E}_k . It is to note that each atom is updated independently. The main steps of this algorithm are listed below (for updating an atom \mathbf{d}_k).

- I. The locations of training signals that have used the atom \mathbf{d}_k are defined in label matrix ($\boldsymbol{\Lambda}_k$).
- II. Put those training signals in columns of matrix \mathbf{E}_k .
- III. Now find the solution of best rank-one approximation of matrix (\mathbf{E}_k) and update the dictionary atom \mathbf{d}_k and coefficients $\boldsymbol{\alpha}_k$. Generally, SVD is used to find this solution.

It is to note that sparse approximation coefficients are not modified during dictionary update stage. Furthermore, a matrix (\mathbf{E}_k) with its rank-one approximation is confined

to a particular set of signals that use the k th atom in the sparse coding stage.

Algorithm 1 summarizes the main steps of the K-SVD algorithm.

2.8.3 Online Dictionary Learning (ODL)

The computational complexity is one of the constraints in developing dictionary learning algorithms. In the literature, most of the dictionary learning algorithms are based on accessing all given training signals at each iteration. Therefore, when the set of training signals is very large then the efficiency of these algorithms decreases. ODL [13, 56] is designed to overcome this problem or at least reduce the magnitude of it. This algorithm considers stochastic approximations and it uses a small subset of the training for processing. The authors [13, 56] also proved that this algorithm converges to the optimum solution. It is to note that training samples are assumed to be i.i.d (independent, identically distributed), hence all the training vectors are independent of each other. ODL tries to minimize the objective function given

$$\begin{aligned} \mathbf{D}_t &\cong \underset{\mathbf{D} \in \mathcal{C}}{\operatorname{argmin}} \frac{1}{T} \sum_{i=1}^t \frac{1}{2} \|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1 \\ &= \underset{\mathbf{D} \in \mathcal{C}}{\operatorname{argmin}} \frac{1}{T} \left(\frac{1}{2} T_r(\mathbf{D}^T \mathbf{D} \mathbf{A}_t) - T_r(\mathbf{D}^T \mathbf{B}_t) \right) \end{aligned} \quad (2.12)$$

Here T is the number of iterations. \mathcal{C} is a space where all dictionary atoms are normalized. \mathbf{B}_t and \mathbf{A}_t are the matrices containing the information about previous iterations. They are formed as shown in algorithm 2 [9,40].

Algorithm 1:K-SVD dictionary learning algorithm

1: Input
2: $\mathbf{X} \in \mathbb{R}^{n \times M}$: noisy patches, $\mathbf{D} \in \mathbb{R}^{n \times K}$: dictionary, σ^2 : noise power
3: S : sparsity level
4: N : number of iterations
5: OUTPUT: \mathbf{D}, \mathbf{A}
6: procedure
7: Initially let: $\mathbf{D} \leftarrow \mathbf{D}_0, i \leftarrow 1$
8: **while** $i \leq N$ **do**
9: **for** $k = 1: K$ **do**
10: set $\Lambda_k \leftarrow i \subseteq 1, 2, \dots, m$ s. t. $\mathbf{A}_{k,i} \neq 0$
11: set $\mathbf{E}_k \leftarrow [\mathbf{X} - \sum_{j \neq k} \mathbf{d}_j \boldsymbol{\alpha}_j^T] \Lambda_k$
12: $[\mathbf{U}, \mathbf{W}, \mathbf{V}] \leftarrow \text{SVD}(\mathbf{E}_k)$
13: $\mathbf{d}_k \leftarrow \mathbf{u}_1$
14: $\mathbf{A}_{\Lambda_k} \leftarrow \sigma_{1,1} \mathbf{v}_1^T$
15: **endfor**
16: $i = i + 1$
17: endwhile

Algorithm 2:Dictionary update stage of ODL

1: Input
2: $\mathbf{D} = [\mathbf{d}_1, \dots, \mathbf{d}_K] \in \mathbb{R}^{n \times K}$ (initialized dictionary)
3: $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_K] \in \mathbb{R}^{n \times K} = \sum_{i=1}^t \boldsymbol{\alpha}_i \boldsymbol{\alpha}_i^T$
4: $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_K] \in \mathbb{R}^{n \times K} = \sum_{i=1}^t \mathbf{x}_i \boldsymbol{\alpha}_i^T$
5: procedure
6: Initialization: $\mathbf{D} \leftarrow \mathbf{D}_0, i \leftarrow 1$
7: **for** $j = 1: K$ **do**
8: Update the j th column to optimize (2.10)
9: $\mathbf{u}_j \leftarrow \frac{1}{A_{jj}} (\mathbf{b}_j - \mathbf{D} \mathbf{a}_j) + \mathbf{d}_j$
10: $\mathbf{d}_j = \frac{1}{\arg\max_j (\|\mathbf{u}_j\|_{2,1})} \mathbf{u}_j$
11: endfor
12: endprocedure

2.9 Image Denoising Via Sparse-Land Model

Elad et al., [1] developed a sparse-land model based image denoising algorithm. It is considered as one of the benchmark methods for image denoising. In this section we shall define how this approach works. Let the image \mathbf{X} be divided into overlapping patches (vectorized form of $n \times n$ portions of the image). Assume that a clean patch \mathbf{x}^c is corrupted with AWGN \mathbf{w} with zero mean and variance σ^2 such that the

observed noisy patch is given by $\mathbf{x} = \mathbf{x}^c + \mathbf{w}$. For convenience assume that patches are arranged as column vectors.

Given a noisy patch \mathbf{x} , we initialize a dictionary \mathbf{D} . In the sparse representation framework, the task of approximating \mathbf{x}^c involves the selection of atoms from a given dictionary $\mathbf{D} \in \mathbb{R}^{n \times K}$ where n and K are length and number of atoms respectively. When the k^{th} atom \mathbf{d}_k is selected, the approximation of \mathbf{x}^c can be expressed as $\mathbf{x}^c = \mathbf{d}_k \alpha_k$, where α_k is the representation coefficient. Once the sparse coding coefficients of all the patches in the training set are computed, the dictionary update stage is performed. The sparse coding and dictionary update stages are iterated a few times and the dictionary that will be used to approximate the clean image is obtained.

The process of iterating between sparse coding and dictionary update continues until representation error goes below certain threshold level or sparsity limit is reached.

This can be formulated as

$$\boldsymbol{\alpha} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\boldsymbol{\alpha}\|_0 \text{ s.t. } \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 \leq \varepsilon \quad (2.13)$$

Where ε is bounded representation error, $\|\cdot\|_2$ operator represents the ℓ_2 norm and $\|\cdot\|_0$ is the ℓ_0 norm. However, the solution to (2.11) is non-deterministic polynomial-time (NP)-hard and hence it is computationally expensive. This optimization task can be rewritten as:

$$\boldsymbol{\alpha} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \mu \|\boldsymbol{\alpha}\|_0 \quad (2.14)$$

Now the constraint has turned to a penalty. In this image denoising method, orthogonal matching pursuit (OMP) is used for sparse coding stage due to its simplicity [1]. After the sparse coding stage, each column is updated independently using KSVD dictionary learning as mentioned in the previous section [2]. Sparse coding and dictionary update stages are alternated for few iterations. Finally, image is reconstructed as:

$$\widehat{\mathbf{X}}^c = \underset{\mathbf{X}}{\operatorname{argmin}} \lambda \|\mathbf{X}^c - \mathbf{X}\|_2^2 + \sum_{ij} \|\mathbf{D}\boldsymbol{\alpha}_{ij} - \mathbf{R}_{ij}\mathbf{X}^c\|_2^2 \quad (2.15)$$

Note that \mathbf{X} represents noisy image, \mathbf{X}^c is clean image and \mathbf{R}_{ij} is binary matrix to extract patch from specified locations [1].

Chapter 3

IMAGE DENOISING VIA CORRELATION BASED SPARSE REPRESENTATION AND DICTIONARY LEARNING

3.1 Introduction

Algorithms based on the sparse and redundant representation model have been successfully applied to the image denoising problem. K-means Singular Value Decomposition (K-SVD) based dictionary learning is one of the most important works along this line of research [1, 2]. K-SVD denoising is a patch based algorithm and it learns a dictionary from the noisy image to approximate the clean image. In the dictionary learning stage, the algorithm initializes the current patch as the residual and employs error based Orthogonal Matching Pursuit (OMP_e) algorithm to approximate the clean patch. The OMP_e algorithm picks the atom that gives maximum orthogonal projection and calculates the new residual based on the selected atom. When the residual power goes below the noise power, the next patch in the training set is processed. Once the sparse coding coefficients of all patches in the training set are evaluated, the dictionary update stage is performed. The sparse coding and dictionary update stages are iterated a few times and the dictionary that will be used to approximate the clean image is obtained.

Note that the residual formed during the sparse coding stage is supposed to be similar to the contaminating noise. In this research work, a sparse coding method based on analysis of properties of the residual is proposed. In the proposed algorithm, we pick an atom such that residual formed is as similar to the noise as possible. In order to achieve this, we considered the autocorrelation property of the AWGN. Additive means it is added to already present intrinsic noise. White represents the uniformity in power distribution. Finally, it is Gaussian because it is generated by normal distribution. Hence, we obtain autocorrelation of the residual and pick an atom that produces the autocorrelation of residual similar to that of the contaminating noise.

The proposed algorithm is compared with the K-SVD [1] denoising algorithm, BM3D [6] and EPLL [14] algorithms. Our results indicate that the proposed algorithm is significantly better than K-SVD and EPLL denoising. At the noise level 100, the improvement over the K-SVD denoising algorithm for Barbara and Fingerprint images is 1.14 dB and 2.64 dB respectively. The proposed algorithm gives results that are visually comparable with the BM3D algorithm.

3.2 Background

Objective of the error based Orthogonal Matching Pursuit (OMP_e) algorithm is to minimize the power in the residual. To minimize the residual power, OMP_e picks the atom that gives maximum orthogonal projection. In order for the maximum projection based OMP_e algorithm to work properly, the atom that is picked must match the clean image patch. However, when the noise is additive and its power is high relative to the clean image patch, the projection of noisy patch onto the dictionary atoms is dominated by noise. Thus, the atom that maximizes absolute

projection is very likely to match the noise instead of the clean image patch. This is the main drawback of the OMP_e atom selection algorithm.

We note once again that, when the atom that matches the image patch is picked, the residual gets closer to the contaminating noise process. If one knows or can estimate the statistics of the contaminating noise, then this information can be incorporated into the sparse coding algorithm. K-SVD denoising algorithm assumes that the contaminating noise is additive, white and Gaussian (AWGN) with zero mean and known variance and it uses maximum projection based OMP_e algorithm to select atoms that match the clean image patch. However, OMP_e exploits only the variance information. OMP_e terminates the atom selection process when the residual power goes just below the contaminating noise power. At high noise levels the variance information alone is insufficient in making sure that the correct atom is selected. This leaves the pixels in the residual patch highly correlated. Highly correlated residual patch pixels is a manifestation of the fact that the selected atoms did not match the clean image patches. In order to make sure that the atom that matches the clean image patch is selected one needs to force the autocorrelation of the residual patch to match the autocorrelation of contaminating noise.

There exist image denoising algorithms that exploit correlations [3, 4, 5, 6, 7]. However, none of these algorithms embed correlation reduction in the framework of sparse representation via learned dictionaries. Also other image denoising algorithms in the literature like [8, 9, 10, 11] have different approach than our proposed correlation based approach.

In this research work, we first show that the atom which gives maximum projection does not necessarily minimize residual correlations. We then develop a simple strategy that takes into account the residual patch correlations. We achieve this by making a simple modification to the OMP_e algorithm. We consider slightly bigger size patches and for each atom in the dictionary we first form the residual and then estimate its autocorrelation. Since the residual must have autocorrelation similar to the autocorrelation of the noise process, the atom selection should ideally continue till the autocorrelation of the residual acceptably matches the autocorrelation of the noise. If, for example, the noise is known to be AWGN with zero mean and variance σ^2 as in [1], then the atom selection continues till the power in the residual (zero lag autocorrelation) goes down to noise power σ^2 and nonzero lag autocorrelations approach zero. We refer to this sparse coding algorithm as OMP_c . We then use the two stage dictionary learning approach employed in [1, 2] where the sparse coding stage is replaced with the proposed OMP_c algorithm to learn the dictionary that will be used to approximate the clean image. The proposed denoising algorithm that employs OMP_c is referred to as K-SVD_c denoising.

Simulations indicate that proposed K-SVD_c algorithm produces better results both visually and in terms of Peak Signal to Noise Ratio (PSNR) when compared to K-SVD for images that are rich in high frequency content and strong pixel correlations like Barbara and Fingerprint images. The improvement over K-SVD denoising is 1.14 dB and 2.64 dB for Barbara and Fingerprint images respectively at $\sigma = 100$. Also it outperforms EPLL [14] denoising significantly at all noise levels in terms of PSNR as well as visual results obtained. Visual results obtained by K-SVD_c algorithm are as good if not better than state of the art BM3D algorithm. Whereas,

BM3D denoising algorithm recover images with high PSNR as compared to proposed K-SVD_c algorithm. However, margin of difference in PSNR obtained by BM3D and K-SVD_c algorithm decreases with increase in noise level and also for images that are rich in high frequency content.

3.3 Motivation And Problem Definition

Assume that a clean patch \mathbf{x}^c is corrupted with AWGN \mathbf{w} with zero mean and variance σ^2 such that the observed noisy patch is given by $\mathbf{x} = \mathbf{x}^c + \mathbf{w}$. For convenience assume that patches are arranged as column vectors.

In the sparse representation framework the task of approximating $\mathbf{x}^c \in \mathbb{R}^n$ involves the selection of atoms from a given dictionary $\mathbf{D} \in \mathbb{R}^{n \times K}$. When the k th atom \mathbf{d}_k is selected, the approximation of \mathbf{x}^c can be expressed as $\hat{\mathbf{x}}^c = \mathbf{d}_k \alpha_k$, where α_k is the representation coefficient. The residual is then given by:

$$\mathbf{r} = \mathbf{x} - \hat{\mathbf{x}}^c = \mathbf{x}^c - \hat{\mathbf{x}}^c + \mathbf{w} = \mathbf{e} + \mathbf{w}, \quad (3.1)$$

where \mathbf{e} is the error in the representation.

We note that as one continues to select more atoms that match the clean image patch, then power in the error \mathbf{e} is expected to decrease and thus the residual is expected to behave like the noise \mathbf{w} . More specifically the residual \mathbf{r} is expected to have the statistical properties of the noise process \mathbf{w} .

Let us consider the projection based approach employed in the OMP_c algorithm for selecting atoms that approximates the clean image patch. The projection of the noisy patch onto the dictionary atoms \mathbf{d}_i ($i = 1, 2, \dots, k$) can be expressed as,

$$\mathbf{d}_i^T \mathbf{x} = \mathbf{d}_i^T (\mathbf{x}^c + \mathbf{w}) = \|\mathbf{d}_i^T\| \|\mathbf{x}^c\| \cos(\theta_{\mathbf{x}^c, \mathbf{d}_i^T}) + \|\mathbf{d}_i^T\| \|\mathbf{w}\| \cos(\theta_{\mathbf{w}, \mathbf{d}_i^T}) \quad (3.2)$$

Here $\theta_{\mathbf{x}^c, \mathbf{d}_i^T}$ and $\theta_{\mathbf{w}, \mathbf{d}_i^T}$ are the angles between the dictionary atom \mathbf{d}_i and clean patch vector \mathbf{x}^c and the noise \mathbf{w} respectively. $\|\mathbf{x}^c\|$ and $\|\mathbf{w}\|$ are square roots of the powers in the clean image patch and the \mathbf{w} noise respectively. Also note that the dictionary atoms are normalized to unit norm i.e., $\|\mathbf{d}_i^T\| = 1$. Given the noisy patch the aim is to select the atom that gives maximum projection. When $\theta_{\mathbf{x}^c, \mathbf{d}_i^T}$ is small and \mathbf{w} is comparable to or greater than \mathbf{x}^c , then the projection is dominated by the noise term $\|\mathbf{w}\| \cos(\theta_{\mathbf{w}, \mathbf{d}_i^T})$. Thus, the atom \mathbf{d}_i that matches the noise term is likely to be picked. When this happens the maximum projection based algorithm picks the atom that matches the contaminating noise. This happens even if $\theta_{\mathbf{x}^c, \mathbf{d}_i^T}$ is small i.e., the similarity of the clean patch and the atom \mathbf{d}_i is high. This contradicts with the premise of the OMP_e algorithm which requires that the selected atom should match the clean image patch. Therefore, at high noise levels the atom picked does not match the clean image patch and thus the residual does not behave like the contaminating noise.

We propose a remedy for this problem by incorporating additional constraints that force the selected atom to match the clean image patch. This is achieved by forcing the residual to behave like the contaminating noise. Thus, when atoms are selected instead of maximizing absolute projections, we pick atoms that force residual autocorrelation sequence to be similar to the autocorrelation sequence of the noise. For an AWGN process, if one assumes that \mathbf{e} and \mathbf{w} are uncorrelated, then the autocorrelation of the residual \mathbf{r} is $\mathbf{a}_k^r = \mathbf{a}_k^w + \mathbf{a}_k^e = \sigma^2 \delta_k + \mathbf{a}_k^e$, where \mathbf{a}_k^w and \mathbf{a}_k^e are the autocorrelation sequences of the noisy patch and the error patch respectively. We note that when atoms that match the clean image patch are selected, the norm of the error decreases such that $\sigma^2 \gg \|\mathbf{e}\|^2$ and the autocorrelation of the

residual then can be approximated as $\mathbf{a}_k^r \approx \sigma^2 \delta_k$ where δ_k is the Dirac delta sequence. The 2D autocorrelation sequence \mathbf{A} of a 2D residual patch \mathbf{R} can be estimated by,

$$\mathbf{A}_{k_1, k_2} = \frac{1}{N} \sum_i \sum_j \mathbf{R}_{i,j} \mathbf{R}_{i+k_1, j+k_2} \quad (3.3)$$

Here i, j denotes the location of residual patch and k_1 and k_2 are horizontal and vertical shifts (lags) from residual patch. Note that for simplicity in (3.3) border effects are not explicitly shown.

Since the patch is of finite size, in order to make sure that the autocorrelation estimates are statistically meaningful, we only consider small lags $|k_1|, |k_2| \leq 2$. Furthermore, for simplicity we reorder this two dimensional autocorrelation sequence and rewrite it as a one dimensional sequence \mathbf{a}_k^r such that \mathbf{a}_0^r represents the residual power (autocorrelation at zero lag) and $\mathbf{a}_k^r (k \neq 0)$ are the nonzero lag autocorrelations.

Now let us consider the sparse coding stage OMP_e of the K-SVD denoising algorithm. Given the dictionary \mathbf{D} and the training patch \mathbf{x} , OMP_e solves,

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\text{argmin}} \|\boldsymbol{\alpha}\|_0 \text{ s.t. } \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 \leq \varepsilon \quad (3.4)$$

where $\|\cdot\|_2$ and $\|\cdot\|_0$ are ℓ_2 and ℓ_0 norms respectively and ε is the representation error. The first term in equation (3.4) forces the representation to be as sparse as possible. The term $\mathbf{D}\boldsymbol{\alpha}$ is the approximation of \mathbf{x}^c and thus $\mathbf{x} - \mathbf{D}\boldsymbol{\alpha} = \mathbf{x} - \hat{\mathbf{x}}^c$ is the residual patch \mathbf{r} .

Therefore, the second term in (3.4) represents the power in the residual patch. To solve (3.4) OMP_e algorithm is used. OMP_e in K-SVD denoising represents each clean patch by picking atoms one at a time till either the power in the residual (zero lag autocorrelation) goes just below $1.15 \sigma^2$ (as given in [1]) or the sparsity limit S_{max} (maximum number of atoms allowed in the representation) is reached. In a way, the OMP_e algorithm assumes that the residual \mathbf{r} should have the properties of the contaminating noise process \mathbf{w} , however it does not go beyond to take advantage of the nonzero lag autocorrelations $\mathbf{a}_k^r (k \neq 0)$.

3.4 Proposed Correlation Based Sparse Coding Stage

We now turn to the formulation of the proposed strategy. Given the dictionary \mathbf{D} , the selection of sparse coding coefficients must ensure that the autocorrelation of the residual patch at all lags must conform to the statistics of the contaminating noise. Thus the sparse coding problem is formulated as,

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\boldsymbol{\alpha}\|_0 \text{ s.t. } \sum_k (|\mathbf{a}_k^r - \sigma^2 \delta_k|) \leq \varepsilon. \quad (3.5)$$

As in (3.3), the first term is the sparsity constraint. The second term constrains the autocorrelation of the residual and it forces it to behave like the autocorrelation of the contaminating AWGN. It contains not only the power in the residual \mathbf{a}_0^r but also the residual correlations at all nonzero lags $\mathbf{a}_k^r (k \neq 0)$.

The solution of (3.5) is very similar to the OMP_e used in K-SVD denoising. Instead of picking the atom that minimize residual power, we adopt a strategy that reduce the residual power (autocorrelation at zero lag) and at the same time minimize sum of nonzero lag correlations in absolute sense. To achieve this, we consider not only the atom that gives maximum projection but a subset of atoms with large projections.

For each atom in this subset, we form the new candidate residuals and estimate their autocorrelation sequences using (3.3). We then pick the atom that reduces the residual power and at the same time minimize the sum of nonzero lag autocorrelations. With selected atom the new residual is formed and atom selection is repeated for the new residual. Similar to the OMP_e algorithm, the proposed algorithm is terminated when the power in the residual goes just below the noise power σ^2 or the sparsity level S_{max} is reached. We formulate stopping criteria in terms of residual power (zero lag correlations) as in [2].

The above description for OMP_c assumes that the dictionary is known. However, if one is to learn the dictionary from the noisy input patches, then the optimization problem that one needs to solve is given by,

$$\{\hat{\boldsymbol{\alpha}}, \hat{\mathbf{D}}\} = \underset{\boldsymbol{\alpha}, \mathbf{D}}{\operatorname{argmin}} \sum_i \|\boldsymbol{\alpha}_i\|_0 \quad s.t. \quad \sum_i \sum_k (|\boldsymbol{\alpha}_i^r - \sigma^2 \delta_k|) \leq \varepsilon \quad (3.6)$$

As in many dictionary learning algorithms [2, 12, 13], the solution of (3.6) is approximated by a two stage process. In the first stage \mathbf{D} is fixed and sparse representation coefficient vectors $\boldsymbol{\alpha}_i$ are calculated. This is the same as the optimization problem formulated in (3.4).

In the second stage the sparse representation coefficient vectors $\boldsymbol{\alpha}_i$ are fixed and the dictionary is updated. For the dictionary update stage we ignore the nonzero lag correlations and adopt the K-SVD dictionary update method [2]. After the sparse coding (OMP_c) and dictionary update stages are iterated several times, a local minimum is reached. Once the dictionary is learned the sparse representation coefficients can be calculated and the clean image can be reconstructed as in [1]. The

proposed correlation reduction strategy for the sparse coding stage OMP_c is given in Algorithm 3.

Algorithm 3: Proposed Correlation-Based Sparse Coding Algorithm: OMP_c

1: Input
2: \mathbf{x}_i : noisy patches ($i = 1, 2, 3, \dots, P$), \mathbf{D} : dictionary, σ^2 : noise power
3: S_{max} : Maximum number of atoms in the representation of \mathbf{x}_i
4: K_{max} : Subset of atoms with large projection
5: procedure
6: **for** $i = 1, 2, \dots, P$ **do**
7: $s = 0; \mathbf{r} = \mathbf{x}_i$
8: Calculate residual correlation \mathbf{a}_k^r
9: **while** $\mathbf{a}_0^r < \sigma^2$ and $s < S_{max}$
10: project \mathbf{r} onto \mathbf{D}
11: Select K_{max} atoms with large projections
12: **for** $l = 1, 2, \dots, K_{max}$
13: Calculate residual $\mathbf{r}^l = \mathbf{r} - \mathbf{d}^l \alpha^l$,
14: Calculate residual correlations $\mathbf{a}_k^{r^l}$,
15: **endfor**
16: Pick atom \mathbf{d}^{l_0} that reduces $\text{sum}(\text{abs}(\mathbf{a}_k^{r^l}))$ the most
17: $s = s + 1$
18: $\mathbf{r} = \mathbf{r} - \mathbf{d}^{l_0} \alpha^{l_0}$
19: **endwhile**
20: **endfor**
21: endprocedure

3.4.1 Complexity Analysis

In this section, the computational complexity of proposed algorithm is compared with that of K-SVD [1] because proposed algorithm is also based on sparse-land model as K-SVD [1]. Similar to the K-SVD algorithm, the computational complexity of K-SVD_c algorithm is evaluated by considering the sparse coding and dictionary updated stages. These stages perform $O(NKLJ)$ operations per pixel [1], where N is patch size, K is number of dictionary atoms, J is number of iterations and L is number of nonzero entries in each sparse coefficient vector. The proposed K-SVD_c algorithm differs from K-SVD algorithm only in the sparse coding stage. K-SVD calculates only one residual for each patch whereas K-SVD_c considers a subset (i.e.,

20 atoms) of atoms (Z_b) and calculates Z_b residuals. Then it further calculates the autocorrelation sequences. It then compares the calculated autocorrelation sequences with that of contaminating noise and determines the atom to be picked. Hence, the proposed algorithm performs $O(NZ_bKLJ)$ operations per pixel. For Barbara image at noise level $\sigma = 10$, the average L is 2.96 for K-SVD whereas L is 5.13 for K-SVD_c.

3.4.2 Limitations And Future Work

In this section, we would point out that the proposed algorithm is less effective for images that do not possess significant high frequency content. Also at low noise levels it does not perform significantly better. It is due to the fact that if there are no sufficient nonzero lag correlations i.e., autocorrelation of residual is similar to that of AWGN, our proposed algorithm will run the same as K-SVD [1] denoising algorithm.

The other issue is that the proposed OMP_c algorithm is computationally expensive since it considers each atom in the dictionary and for each atom calculates the resulting residual autocorrelation sequence. However, if one considers small subset of atoms with large projections and autocorrelations with small lags, the computational complexity can be significantly reduced with little loss in performance. As a future work, it is possible to formulate different atom selection strategies and stopping criteria that can more effectively balance the reduction in residual power and nonzero lag autocorrelations; however our aim in this research work is not to obtain the optimal strategy but rather motivate the concept of sparse representation based correlation reduction and show its utility and effectiveness in the denoising problem.

3.5 Types Metrics Used To Compare the Performance

The peak signal-to- noise ratio (PSNR), structure similarity index measure and feature similarity index measure (FSIM) are used to compare performance of the proposed algorithm with state-of-the-art algorithms.

PSNR is measured as $10(\log_{10} 255^2 / MSE)$ where $MSE = \frac{1}{n} ||X - \hat{X}||^2$. Where X and \hat{X} are the original and denoised images respectively.

SSIM is measured based on (3.6) as given in [64],

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (3.6)$$

Here x and y are original and recovered images respectively. μ and σ are mean and standard deviation respectively. C_1 and C_2 are constants.

FSIM is measure based on (3.7). It is combination of phase congruency and gradient magnitude measure as:

$$FSIM = \frac{S_L(x).PC_m(x)}{PC_m(x)} \quad (3.7)$$

Here PC is phase congruency measure and S_L is similarity measure in terms of PC and GMM as given in [63].

3.6 Simulation And Results

In this section we first briefly study the convergence of the proposed algorithm in terms of nonzero lag autocorrelation reduction and compare it with the maximum projection based OMPe algorithm in terms of speed of convergence and degree of reduction achieved. Then PSNR results comparing the performance of K-SVD [1],

EPLL [14] and BM3D [6] denoising with the proposed K-SVD_c denoising algorithm are presented. Finally, qualitative results of proposed algorithm are compared with state of the art image denoising algorithms.

3.6.1 Convergence Of Proposed Sparse Coding Algorithm OMP_c

In order to study and compare the convergence behavior of the proposed correlation based OMP_c and maximum projection based OMP_e in terms of non zero lag autocorrelation reduction, we start with the noisy Barbara image, extract patches of size 16×16 and learn two dictionaries using K-SVD[1] and K-SVD_c algorithms. Both algorithms are iterated 20 times. The experiment is repeated for $\sigma = 15, 50$ and 75. Note that after adding noise to the image there is possibility of pixel saturation which means pixel value can exceed 255 (overflow) or at the same time it can go below zero (underflow) considering gray scale image uint8 data type. In order to avoid such effect the image is converted to larger data type (using Matlab command “im2double”) before adding noise. For the K-SVD_c algorithm autocorrelations are calculated using maximum lag of 2 ($|k|_1, |k|_2 \leq 2$). In this simulation we considered a subset of 20 dictionary atoms with largest projections. In both algorithms dictionaries are initialized by randomly selected patches from the training set. Dictionaries in both algorithms have $K = 512$ atoms.

At the end of each iteration, we form the residual, calculate the sum of absolute value of nonzero lag autocorrelations and normalize it with the initial sum of absolute value of nonzero lag autocorrelations. The results of this experiment are presented in Figure 3.1. For all noise levels K-SVD_c achieves a lower total of nonzero lag autocorrelation at every iteration. Figure 3.1 clearly indicates that for all noise levels considered and at every iteration K-SVD_c algorithm achieves a lower nonzero lag

autocorrelation level. For high noise levels K-SVD_c achieves significant reduction in nonzero lag autocorrelations. For $\sigma = 50$ and $\sigma = 75$ the reduction at 20th iteration is 19% and 34% more when compared to K-SVD [1] algorithm. We note that even though K-SVD_c algorithm achieves significant reduction in nonzero lag autocorrelations, it converges slower than the K-SVD algorithm especially at high noise levels. The proposed OMP_c algorithm does an excellent job in decorrelating the residual patches and rendering their autocorrelation function much closer to the autocorrelation of contaminating AWGN.

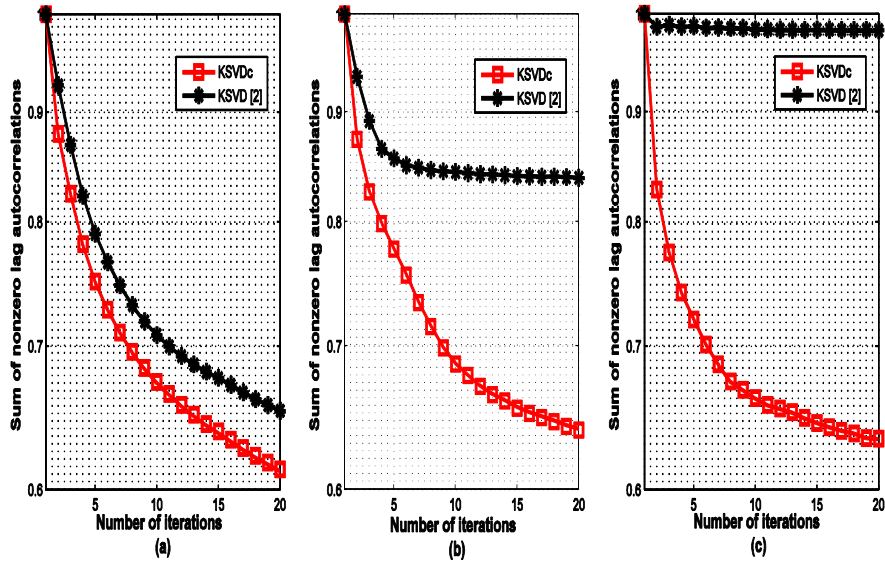


Figure 3.1: Sum of nonzero lag autocorrelations versus number of iterations for Barbara image (a) $\sigma = 15$ (b) $\sigma = 50$ (c) $\sigma = 75$.

3.6.2 PSNR Results Comparison

We now present simulations comparing the performance of K-SVD_c algorithm with the K-SVD algorithm [1], EPLL [14] and BM3D algorithm [6] in terms of PSNR.

In the light of the results presented in Figure 3.1, dictionaries for both algorithms are obtained after 20 iterations. The clean image is then reconstructed as in [1]. Table

3.1 and Table 3.2 gives the Peak Signal to Noise Ratio (PSNR) results for several benchmark images. This simulation is carried out for noise levels varying from 20 to 100.

In order to have fair comparison, patch sizes of 8, 12 and 16 corresponding respectively to dictionary sizes of $K = 256$, $K = 400$ and $K = 512$ are considered as shown in Table 3.2. As we analyze results in Table 3.2, K-SVDc achieves better denoising results as patch size is increased and it outperforms K-SVD by a significant margin. For patch size 8, K-SVD is slightly better than the K-SVDc algorithm except for the fingerprint image ($\sigma \geq 50$). When the patch size is 12 the autocorrelation estimates become more accurate and K-SVDc performs better denoising especially for images with high frequency content. K-SVDc outperforms K-SVD at patch size 16 ($K = 512$) by significant margin.

Now best PSNR results obtained by both algorithms are compared. We thus compare K-SVD at patch size 8 with K-SVDc at patch size 16. K-SVDc outperforms K-SVD by a significant margin at high noise levels and for images with repeating structures like Barbara and Fingerprint images. For the Fingerprint image at noise $\sigma = 100$, the improvement is 2.64 dB. For Barbara image at $\sigma = 100$, K-SVDc outperforms K-SVD by 1.14 dB. Also as shown in Table 3.1, proposed K-SVD_c algorithm performs significantly better than state-of-the-art EPLL [14] denoising algorithm for all images at almost all noise levels. It is to note that the proposed algorithm does not perform significantly better at low noise levels and/or for images that contain large amounts of low-frequency contents. It is due to the fact that if autocorrelation of residual is closely related to that of AWGN, then the proposed algorithm will turn

into maximum projection-based image denoising scheme as same K-SVD [1] denoising algorithm.

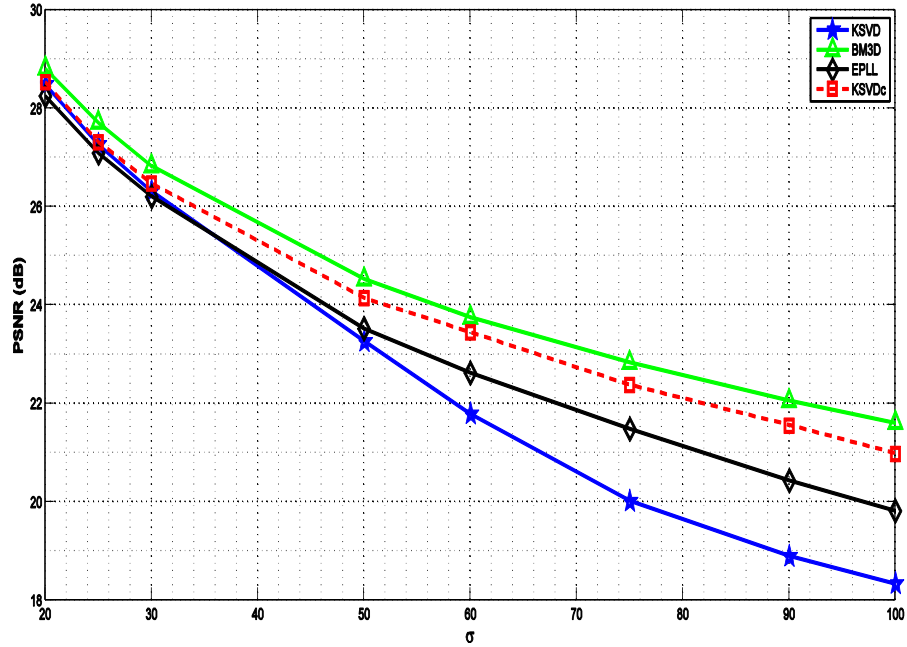


Figure 3.2: Comparison of denoising results for Fingerprint image with noise level σ varying from 20 to 100.

Figure 3.2 and Figure 3.3 show that for images that are rich in high frequency contents the performance of proposed algorithm keeps improving as noise level is increased with respect to state of the art algorithms. It demonstrates that the proposed algorithm is highly effective to recover repeated structures at high noise levels. It also reveals that performance of K-SVD [1] and EPLL [14] keeps decreasing as noise level is increased when compared to proposed K-SVD_c and BM3D [6] algorithms. However, K-SVD_c produces highly consistent results when compared to the BM3D denoising algorithm. Whereas, visual results obtained by proposed algorithm are as good as BM3D algorithm.

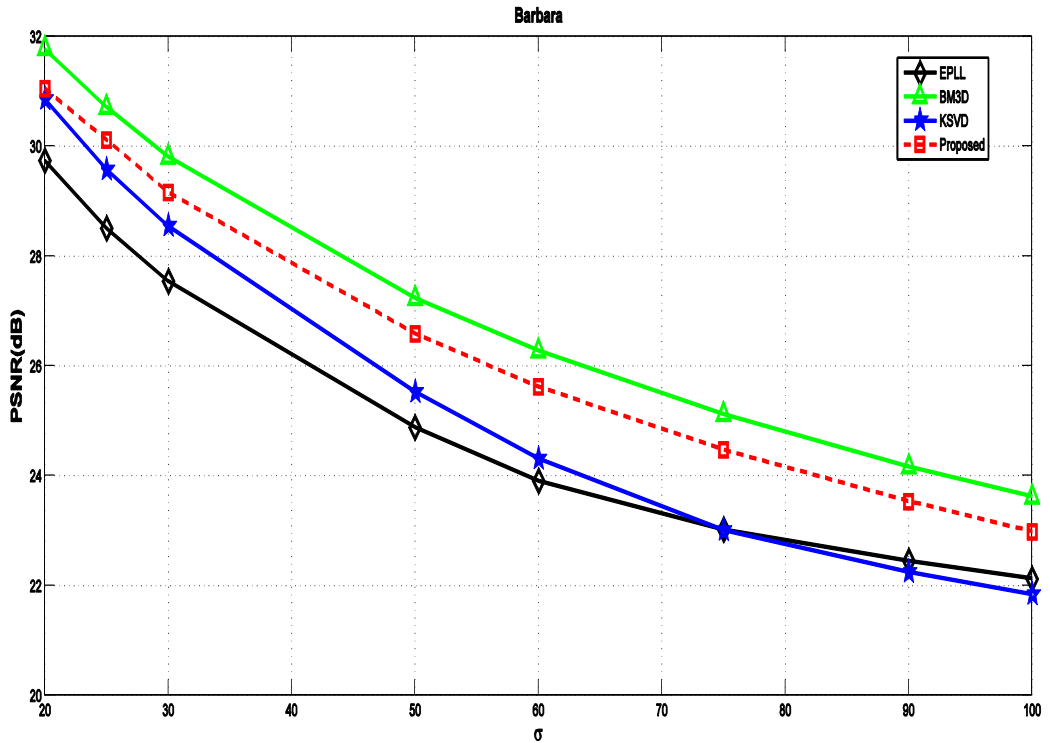


Figure 3.3: Comparison of denoising results for Barbara image with noise level σ varying from 20 to 100.

Now, in order to further investigate the comparison, we plot difference in PSNR results of all the algorithms with respect to KSVD image denoising algorithm. Hence, we consider KSVD as a zero line reference (shown as straight line in Figure 3.4). From Figure 3.4 and Figure 3.5, we conclude that the performance of residual correlation based algorithm keeps improving with respect to base line KSVD algorithm. Also it produces highly consistent results with respect to BM3D algorithm. It further verifies that the residual correlation information is highly essential especially for the images that possess large quantity of high frequency content and/or at high noise levels. It also proves that at high noise levels maximum projection based sparse coding algorithms fail to pick correct atom at high noise levels. As a result its performance keeps decreasing with increasing noise levels.

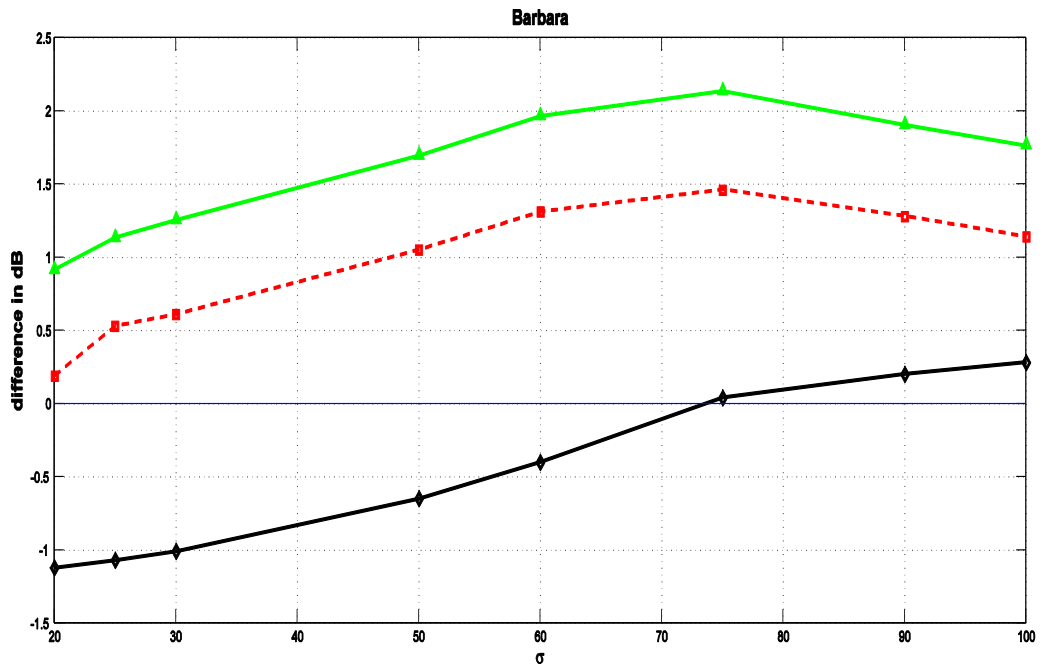


Figure 3.4: Difference in PSNR comparison .

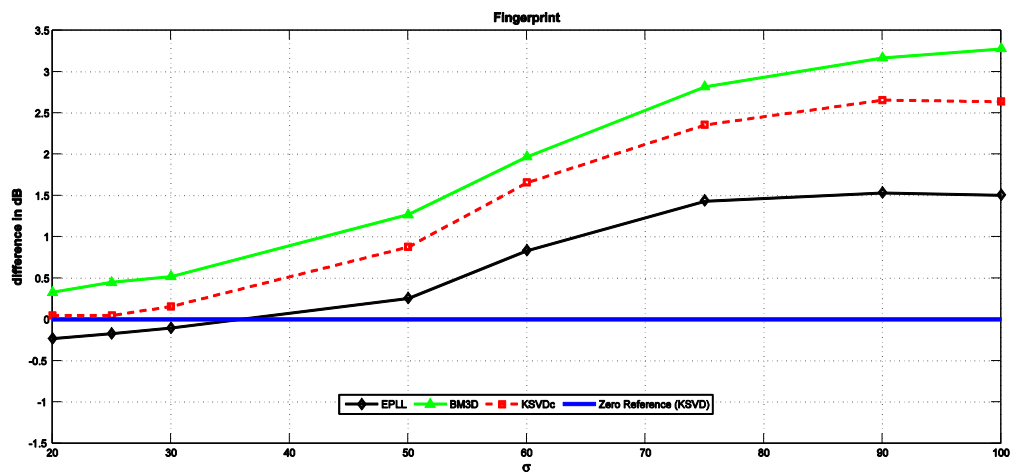


Figure 3.5: Difference in PSNR comparison .

Table 3.1: PSNR results in decibels. Top left: Results of K-SVD [1]. Top right: BM3D[6]. Bottom left EPLL [Bottom right: Proposed Algorithm.

Sigma	Barbara		Boat		Fingerprint		Lena		Building		MRI		Average		Avg $\Delta PSNR$ w.r.t K-SVD
20	30.86	31.78	30.33	30.88	28.48	28.81	32.39	33.05	28.43	28.93	28.37	28.54	30.29	30.82	-0.07
	29.74	31.05	30.61	30.16	28.25	28.53	32.53	32.41	27.68	27.99	28.35	28.27	30.02	30.22	
25	29.58	30.72	29.27	29.91	27.26	27.71	31.32	32.08	27.09	27.62	27.63	27.96	29.18	29.88	0.06
	28.51	30.11	29.61	29.31	27.09	27.31	31.54	31.49	26.39	26.79	27.61	27.52	28.96	29.24	
30	28.55	29.81	28.42	29.12	26.31	26.83	30.45	31.26	25.94	26.59	27.11	27.51	28.28	29.03	0.13
	27.54	29.16	28.77	28.58	26.21	26.47	30.79	30.71	25.43	25.83	27.19	26.84	28.16	28.41	
50	25.53	27.23	25.94	26.78	23.26	24.53	27.87	29.05	22.79	23.48	24.94	26.09	25.47	26.69	0.62
	24.88	26.58	26.58	26.32	23.52	24.14	28.32	28.36	22.79	23.36	25.53	25.33	25.79	26.09	
60	24.31	26.28	25.06	26.02	21.78	23.75	26.89	28.27	21.48	22.65	24.16	25.48	24.35	25.88	0.92
	23.91	25.62	25.82	25.58	22.62	23.44	27.42	27.44	22.09	22.56	24.95	24.49	24.98	25.27	
75	23.01	25.12	23.95	25.12	20.01	22.83	25.65	27.26	20.51	21.74	23.15	24.69	23.09	24.89	1.12
	23.02	24.47	24.87	24.61	21.48	22.37	26.47	26.34	21.13	21.58	24.19	23.72	23.99	24.21	
90	22.25	24.16	23.24	24.39	18.89	22.06	24.88	26.45	19.81	21.05	22.31	24.01	22.23	24.08	1.15
	22.45	23.53	24.08	23.83	20.43	21.55	25.64	25.35	20.49	20.97	23.41	22.99	23.16	23.38	
100	21.84	23.62	22.81	23.97	18.33	21.61	24.49	25.95	19.57	20.67	22.06	23.59	21.65	23.61	1.13
	22.13	22.98	23.66	23.27	19.82	20.97	25.33	24.87	20.02	20.45	23.08	22.46	22.76	22.78	

Table 3.2: PSNR results in decibels at various patch sizes and dictionary sizes

Sigma	Patch Size	Barbara			Boat			Fingerprint			House			Lena			
Value	and	KSVD	BM3D	KSVDc	KSVD	BM3D	KSVDc	KSVD	BM3D	KSVDc	KSVD	BM3D	KSVDc	KSVD	BM3D	KSVDc	
	Dictionary atoms																
20	$8 \times 8 (K = 256)$	30.86	31.78	30.05	30.33	30.88	29.81	28.48	28.81	28.22	33.17	33.77	31.46	32.39	33.05	31.18	
	$12 \times 12 (K = 400)$	30.71		30.79	29.86		30.16	28.11		28.53	33.16		32.46	32.06		32.07	
	$16 \times 16 (K = 512)$	30.36		31.06	29.51		30.16	27.79		28.59	32.75		32.66	31.66		32.22	
30	$8 \times 8 (K = 256)$	28.55	29.81	27.68	28.42	29.12	27.55	26.31	26.83	26.02	31.18	32.09	28.85	30.45	31.26	28.78	
	$12 \times 12 (K = 400)$	28.35		28.59	27.83		29.12	28.21		25.75	26.33		31.46	30.64		30.09	30.11
	$16 \times 16 (K = 512)$	27.83		29.16	27.29		28.29	25.38		26.41	30.92		30.76	29.59		30.25	
50	$8 \times 8 (K = 256)$	25.53	27.23	24.56	25.94	26.78	24.66	23.26	24.53	23.35	27.97	29.69	25.41	27.87	29.05	25.56	
	$12 \times 12 (K = 400)$	25.36		25.97	25.41		25.77	22.88		23.79	28.26		27.36	27.41		27.32	
	$16 \times 16 (K = 512)$	24.91		26.59	24.88		26.07	22.46		23.88	28.15		28.09	26.96		27.89	
60	$8 \times 8 (K = 256)$	24.31	26.28	23.32	25.06	26.02	23.48	21.78	23.75	22.29	26.81	28.74	24.09	26.89	28.27	24.03	
	$12 \times 12 (K = 400)$	24.06		25.02	24.65		24.88	21.71		22.91	26.81		26.46	26.52		26.32	
	$16 \times 16 (K = 512)$	23.61		25.62	24.11		25.38	21.24		23.11	26.44		27.14	25.91		27.01	
75	$8 \times 8 (K = 256)$	23.01	25.12	21.81	23.95	25.12	20.99	20.01	22.83	20.87	25.23	27.51	22.27	25.78	27.26	22.66	
	$12 \times 12 (K = 400)$	22.65		23.68	23.52		23.76	19.65		21.92	25.06		24.99	25.37		24.99	
	$16 \times 16 (K = 512)$	21.97		24.47	22.93		24.32	19.41		22.15	24.72		26.11	24.59		25.99	
90	$8 \times 8 (K = 256)$	22.25	24.16	20.51	23.24	24.39	20.83	18.89	22.06	19.68	24.26	26.48	20.98	24.88	26.45	21.42	
	$12 \times 12 (K = 400)$	21.69		22.74	22.69		22.87	18.16		21.07	23.95		23.83	24.44		24.01	
	$16 \times 16 (K = 512)$	21.03		23.53	21.99		23.55	17.85		21.41	23.08		24.89	23.52		25.15	
100	$8 \times 8 (K = 256)$	21.84	23.62	19.81	22.81	23.97	20.08	18.33	21.61	18.95	23.64	25.87	20.17	24.49	25.95	20.66	
	$12 \times 12 (K = 400)$	21.36		22.12	22.32		22.33	17.38		20.58	23.27		23.03	24.02		23.35	
	$16 \times 16 (K = 512)$	20.64		23.11	21.63		22.93	17.12		21.12	22.46		24.19	23.14		24.48	

3.6.3 Qualitative Comparison

In this section, we compare the proposed algorithm with state of the art image denoising algorithms in terms of visual results obtained.

Figure 3.6 shows that K-SVD [1] and EPLL [14] fail to recover repeating structures like ridges in Fingerprint image at high noise levels. However, proposed K-SVD_c algorithm does excellent job to denoise these images and it produces highly competitive visual results when compared to BM3D algorithm.

Furthermore, Figure 3.7 shows a portion of the Barbara image reconstructed using K-SVD, BM3D and K-SVD_c for $\sigma = 50$. Visually it is clear that the textures in the upper right corner, the stripes on the scarf near the hand are reconstructed fairly correctly for the K-SVD_c method. K-SVD [1] on other hand does a poor job in recovering such fine structures. Similarly, visual results show that K-SVD_c is as good if not better than BM3D [6]. A closer investigation reveals that, the stripes on scarf and on background are recovered much sharply by K-SVD_c as compared to the state of art BM3D denoising algorithm [6].

Similarly, Figure 3.8 also shows that fine structures of windows in building image are better restored by the proposed K-SVD_c denoising algorithm as compared to K-SVD and EPLL algorithms. Visual results obtained by proposed K-SVD_c image denoising algorithm are as good as state of the art BM3D algorithm [6].

Figure 3.9 and Figure 3.10 show the dictionaries learned via K-SVD [1] and K-SVD_c algorithms. The best denoising results are obtained at $K = 256$ and $K = 512$ dictionary atoms for K-SVD and K-SVD_c respectively. Therefore, number of atoms

are based on best results produced by each algorithm. Dictionary obtained by K-SVD_c are highly structured. It is noted that first atom in both algorithms is reserved for DC.

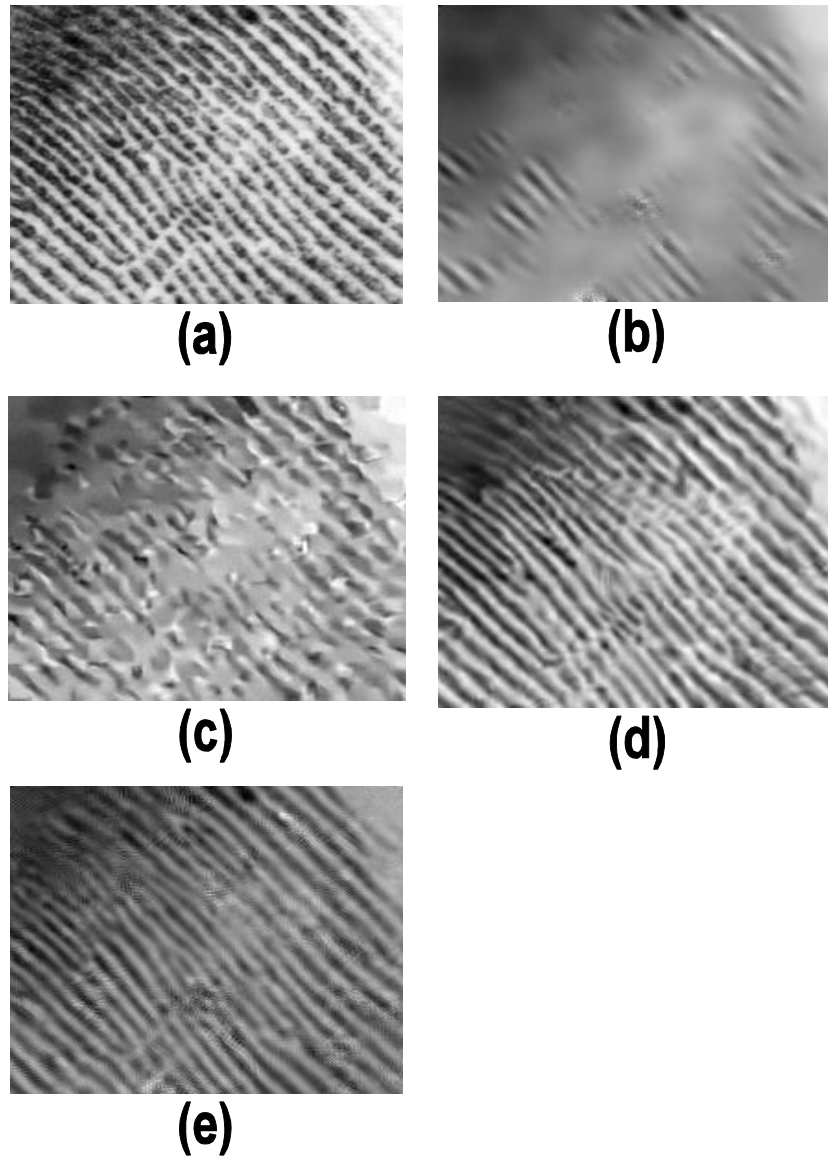


Figure 3.6: Visual comparison of Fingerprint image with $\sigma = 100$ (a) original image (b) denoised by K-SVD [1] (c) denoised by EPLL [14] (d) denoised by BM3D [6] (e) denoised by K-SVD_c



(a)



(b)



(c)



(d)

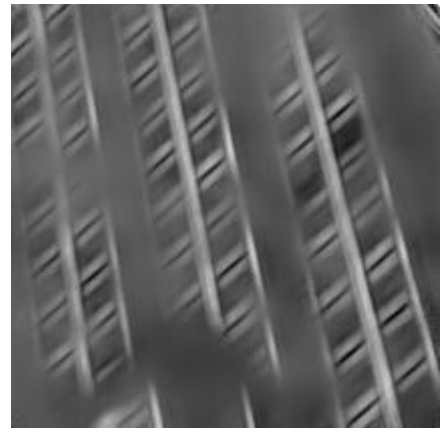


(e)

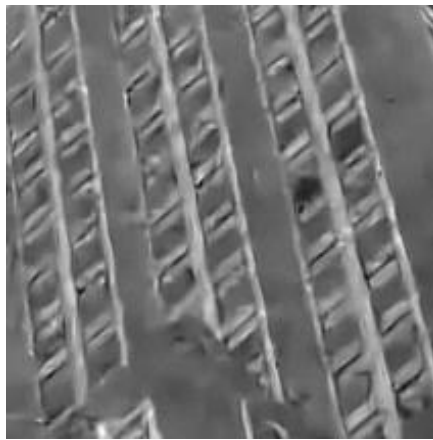
Figure 3.7: Barbara image reconstruction comparison with $\sigma = 50$ (a) original image (b) denoised by K-SVD [1] (c) denoised by EPLL [14] (d) denoised by BM3D [6] (e) denoised by K-SVD_c



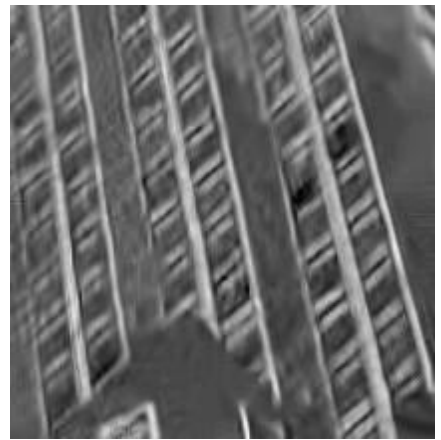
(a)



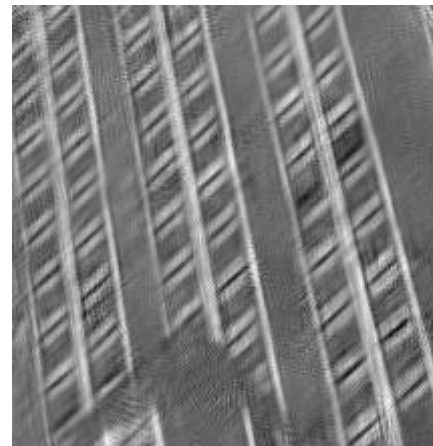
(b)



(c)



(d)



(e)

Figure 3.8: Visual comparison of Building image with $\sigma = 60$ (a) original image (b) denoised by K-SVD [1] (c) denoised by EPLL [14] (d) denoised by BM3D [6] (e) denoised by K-SVD_c

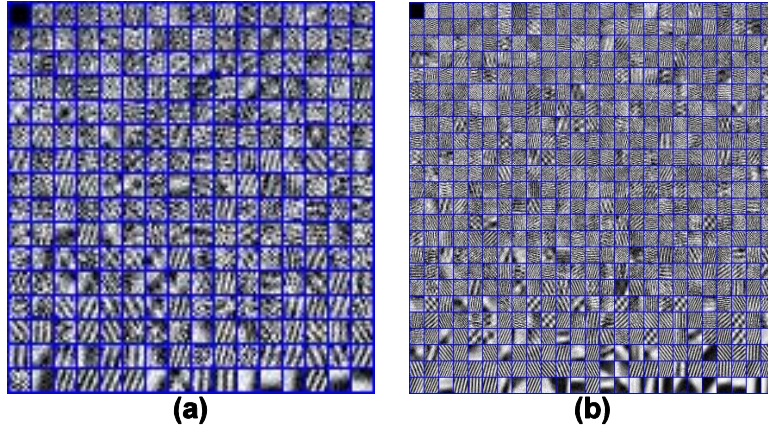


Figure 3.9: The trained dictionary for Fingerprint image with $\sigma = 100$ after 20 iterations. (a) K-SVD (b) K-SVD_c

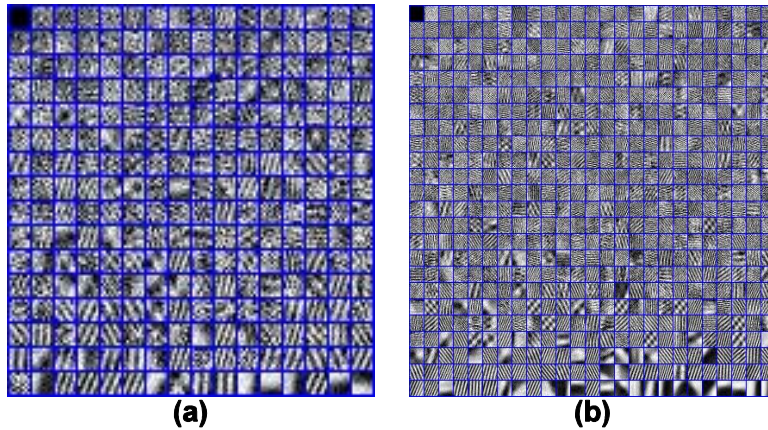


Figure 3.10: The trained dictionary for Barbara image with $\sigma = 75$ after 20 iterations. (a) K-SVD (b) K-SVD_c

3.7 Conclusion

In this chapter, firstly it was shown through simple experiment that maximum orthogonal projection based image denoising algorithms fail to pick correct atom especially at high noise levels. Then, later in this chapter a new correlation reduction strategy in the framework of sparse representation is proposed. Simulation results obtained through proposed strategy show that incorporating residual correlations in sparse representation does indeed improve the performance of image denoising

problem. Hence, the purpose of this chapter was to show only the effectiveness of residual correlation reduction based sparse coding in image denoising. However, in next chapter, a comprehensive sparse coding algorithm is derived that is based on new residual correlation regularization. Also, a new dictionary update stage is derived that uses the proposed residual patch regularization to update the dictionary atoms.

Chapter 4

RESIDUAL CORRELATION REGULARIZATION BASED IMAGE DENOISING

4.1 Introduction

Patch based denoising algorithms aim to reconstruct the clean image patch leaving behind the residual as contaminating noise. It is very likely that the residual patch contains remnants from the clean image patch. However, residual should possess statistical properties of contaminating noise. In this chapter, we propose a new residual correlation based regularization for image denoising. The regularization can effectively render residual patches as uncorrelated as possible. It allows us to derive analytical solution for sparse coding (atom selection and coefficient calculation). It also leads to a new online dictionary learning update. The clean image is obtained through alternating between the two stages of sparse coding and dictionary updating. The performance of proposed algorithm is compared with state-of-the-art denoising algorithms in terms of peak signal-to-noise ratio, structural similarity index and feature similarity index, as well as through visual comparison. Experimental results show that the proposed algorithm is highly competitive and often better than leading denoising algorithms. The proposed algorithm is also shown to offer an efficient complement to the benchmark algorithm of BM3D especially.

4.2 Background

The main objective in patch based image denoising algorithms that employ learned dictionaries is to make sure that atoms that best match clean image patch are picked. K-SVD [1] denoising for example does this by projecting the noisy patch onto the dictionary atoms and picking the atom that gives maximum orthogonal projection. As a result, at high noise levels, the residue usually contains structures from clean image patch, thus it does not match the contaminating noise [36]. On the other hand, after the sparse coding stage is completed, the residual is expected to possess properties similar to those of contaminating noise. One such property is that the residues of different patches should be uncorrelated. We adopt a strategy that will render the residual patches uncorrelated for AWGN. This observation calls for processing patches in groups by considering local neighborhoods and making sure that the neighboring residuals are uncorrelated. Thus in selecting atoms for a given patch, we determine the sparse coefficient that leaves behind a residual which is as uncorrelated with the neighboring residuals as possible.

This approach was adapted in [36]. However, the sparse coefficients were not estimated based on residual correlation and also dictionary update stage was similar to the one proposed in [1].

In [36] large patches were used and the atom that rendered the correlation between smaller sub-patches within the bigger patch was selected. Similarly, Riot et al. [38] also proposed a variation in fidelity term to control the residual distribution. This is achieved by considering statistical moments of residual and the correlation on

patches. It differs from our proposed algorithm since it does not embed the framework of sparse representation via learned dictionaries.

There also exist other image denoising algorithms based on residual correlations such as [4, 5, 6, 7]. However, except [36] and [5], none of these algorithms are based on sparse-land model. In [6], similar 2D image blocks are arranged in 3D groups. Then, collaborative filtering is developed to denoise these 3D image blocks. In [4], web images are used to match the noisy image patch. The accuracy of matching is increased by graph based optimization and then image cubes (group of similar noisy image patches) are filtered in the transform domain. He et al. [5] introduced a correlation coefficient criterion. Meaningful structures are extracted from noisy image using correlation based coefficient criterion. Also multi-scale sparse coding is proposed to improve the performance. In [7], the importance of exploiting residual image to improve performance of image denoising is discussed. The authors proposed a algorithm based on mean-squared-error (MSE) and structural similarity index measure (SSIM) estimation of residual image without any reference image.

In this chapter, we introduce a new residual correlation regularization based image denoising algorithm. This regularization minimizes the correlation between neighboring residual patches. We derive analytical solution for sparse coding (the atom selection and coefficient estimation). We also propose a new dictionary update that is based on the correlation regularization. The final clean image reconstruction is obtained via alternating between the sparse coding and dictionary update stages. Our experimental results show that the proposed algorithm is highly comparable and often superior to the state-of-the-art denoising algorithms. The performance of the proposed algorithm is compared with state-of-the-art algorithms at various noise

levels ranging from 25 to 100. Experimental results show that the proposed algorithm significantly outperforms K-SVD [1] and EPLL [14] in terms of the peak signal-to-noise ratio (PSNR), especially at high noise levels and for images that are rich in high frequency content (like Barbara and Fingerprint image). Also it outperforms K-SVD [1] in terms of structural similarity index (SSIM) and produces competitive SSIM results when compared to benchmarks for image denoising algorithms such as KSVD [1] and NCSR [39]. The improvement over K-SVD denoising is 1.22 dB and 2.93 dB for Barbara and Fingerprint images respectively at $\sigma = 100$. A visual comparison also suggests that the proposed algorithm allows denoising results that are as good if not better than BM3D [6] and NCSR [39] algorithms.

4.3 Motivation And Problem Statement

We consider the standard model for the image denoising problem: A clean image is corrupted by an additive white Gaussian (AWGN) uncorrelated noise. Let the image be partitioned into overlapping patches and each patch is arranged as column vector $\mathbf{x} \in \mathbb{R}^n$, which is modeled as

$$\mathbf{x} = \mathbf{x}^c + \mathbf{w} \quad (4.1)$$

Where \mathbf{x}^c is the clean patch and \mathbf{w} is the noise patch. A dictionary \mathbf{D} is given with atoms $k = 1, 2, \dots, K$. If \mathbf{x}^c is approximately represented by its code coefficients $\boldsymbol{\alpha}$, i.e., $\hat{\mathbf{x}}^c = \mathbf{D}\boldsymbol{\alpha}$, then the approximation error is $\mathbf{e} = \mathbf{x}^c - \hat{\mathbf{x}}^c$ and the residue is:

$$\mathbf{r} = \mathbf{x} - \hat{\mathbf{x}}^c = \mathbf{x}^c + \mathbf{w} - \hat{\mathbf{x}}^c = \mathbf{e} + \mathbf{w} \quad (4.2)$$

This implies that if the code is correctly determined so that $\mathbf{e} \cong 0$, then residue $\mathbf{r} \approx \mathbf{w}$. As a result, the residue should possess the same statistical properties of the contaminating noise. In maximum projection based algorithms, approximation of a

noisy patch \mathbf{x} is achieved by projecting it onto dictionary atoms and picking the atom that gives the maximum projection. Note that the performance of maximum projection based algorithms deteriorates as the noise level increases [40]. According to [36], the projection coefficient on atom \mathbf{d}_k is

$$\begin{aligned} \mathbf{d}_k^T \mathbf{x} &= \mathbf{d}_k^T (\mathbf{x}^c + \mathbf{w}) = \|\mathbf{d}_k^T\| \|\mathbf{x}^c\| \cos(\theta_{\mathbf{d}_k, \mathbf{x}^c}) + \|\mathbf{d}_k^T\| \|\mathbf{w}\| \cos(\theta_{\mathbf{d}_k, \mathbf{w}}) \\ &= \|\mathbf{x}^c\| \cos(\theta_{\mathbf{d}_k, \mathbf{x}^c}) + \|\mathbf{w}\| \cos(\theta_{\mathbf{d}_k, \mathbf{w}}) \end{aligned} \quad (4.3)$$

where $\|\mathbf{d}_k^T\| = 1$ and $\theta_{\mathbf{a}, \mathbf{b}}$ denotes the angle between vectors \mathbf{a} and \mathbf{b} . At high noise levels where the magnitude of noise \mathbf{w} is greater than that of the clean patch \mathbf{x}^c , the noise \mathbf{w} dominates the maximum projection and thus would dictate the atom selection process. The atom that matches the contaminating noise is then picked. Consequently, the residual \mathbf{r} contains remnants from clean signal and it would not possess properties of the noise [36].

In this research work, we develop a new correlation based regularization to ensure that the residuals of different patches are minimally correlated, hence they behave like contaminating noise. Our problem can be summarized as follows.

Given a patch \mathbf{x} formed from a noisy images, we aim to find a sparse code $\boldsymbol{\alpha}$, such that the representation $\hat{\mathbf{x}}^c = \mathbf{D}\boldsymbol{\alpha}$ gives a good approximation of the clean image, i.e., $\hat{\mathbf{x}}^c \approx \mathbf{x}^c$ and the resultant residue $\mathbf{r} = \mathbf{x} - \hat{\mathbf{x}}^c$ is uncorrelated with the residues of the neighboring patches of the noisy image.

4.4 Residual Correlation Regularization

4.4.1 Sparse Coding

Let the current patch being processed be denoted by \mathbf{x} . Assume that M of its immediate neighbors have been processed, and the corresponding residuals are $\mathbf{r}^m, m = 1, 2, \dots, M$. We initialize its residue as $\mathbf{r}_0 = \mathbf{x}$. We shall then determine the first atom by a regularization based on residual patch correlation regulation. We then proceed to next patch by the same approach. And then similarly we pick second atom for each patch. The process is repeated either the maximum number of atoms to be used is reached or the residual power is reduced below the noise power. Assume that we are going to pick the s th atom for \mathbf{x} . Denote by \mathbf{r}_{s-1} the residual formed after selection of $s - 1$ atoms. If the atom picked is \mathbf{d}_{k_s} , and the corresponding coefficient is α_s , then the new residual is:

$$\mathbf{r}_s = \mathbf{r}_{s-1} - \mathbf{d}_{k_s} \alpha_s \quad (4.4)$$

Our atom selection is performed by minimizing the following objective function:

$$J_c(k_s, \alpha_s) = \frac{1}{2} \|\mathbf{r}_s\|_2^2 + \sum_{m=1}^M \lambda_m |\mathbf{r}_s^T \mathbf{r}^m| \quad (4.5)$$

where $\lambda_m > 0, m = 1, 2, \dots, M$, are regularization weighting parameters, which can be selected according to the level of the noise. Note that the first term in the right-hand side of (4.5) represents a fidelity term, whereas the second term realizes the regularization on residual correlation between the current residual \mathbf{r}_s and neighboring residuals \mathbf{r}^m . Therefore, minimizing J_c in (4.5) enforces the residual of the current patch to be uncorrelated with those of neighboring patches as possible.

Thus, the atom and the corresponding coefficient can be determined as

$$(k_s^*, \alpha_s^*) = \underset{k_s, \alpha_s}{\operatorname{argmin}} J_c(k_s, \alpha_s) \quad (4.6)$$

Expanding $J_c(k_s, \alpha_s)$ in terms of k_s and α_s , we have:

$$\begin{aligned} & J_c(k_s, \alpha_s) \\ &= \frac{1}{2} (\mathbf{r}_{s-1} - \mathbf{d}_{k_s} \alpha_s)^T (\mathbf{r}_{s-1} - \mathbf{d}_{k_s} \alpha_s) + \sum_{m=1}^M \lambda_m |(\mathbf{r}_{s-1} - \mathbf{d}_{k_s} \alpha_s)^T \mathbf{r}^m| \\ &= \frac{1}{2} (\alpha_s^2 \mathbf{d}_{k_s}^T \mathbf{d}_{k_s} - 2\alpha_s (\mathbf{d}_{k_s})^T \mathbf{r}_{s-1} + \mathbf{r}_{s-1}^T \mathbf{r}_{s-1}) + \sum_{m=1}^M \lambda_m |(\mathbf{r}_{s-1} - \mathbf{d}_{k_s} \alpha_s)^T \mathbf{r}^m|. \end{aligned}$$

To solve (4.6) we take the derivative of $J_c(k_s, \alpha_s)$ with respect to (w.r.t.) α_s :
(detailed steps are presented in Appendix A)

$$\frac{\partial f}{\partial \alpha_s} = -\mathbf{d}_{k_s}^T (\mathbf{r}_{s-1} - \mathbf{d}_{k_s} \alpha_s) - \sum_{m=1}^M |s_m \lambda_m \mathbf{d}_{k_s}^T \mathbf{r}^m| \quad (4.7)$$

Where

$$s_m = \operatorname{sgn} \left((\mathbf{r}_{s-1} - \mathbf{d}_{k_s} \alpha_s)^T \mathbf{r}^m \right) \quad (4.8)$$

with sgn being the sign function. Letting the derivative be equal to zero and noting:

$$\alpha_s = \mathbf{d}_{k_s}^T \mathbf{r}_{s-1} + \sum_{m=1}^M s_m \lambda_m \mathbf{d}_{k_s}^T \mathbf{r}^m \quad (4.9)$$

Note that if $\lambda_m = 0$, function in J_c (4.5) gives the standard least square error (LSE) expression. And the solution in equation (4.9) coincides with the very solution of the corresponding LSE problem when no patch correlation regularization is used [1]. Furthermore, equation (4.9) shows explicitly how the LSE solution is modified when any neighboring patch is used for regularization. There are two issues in calculating

α_s from (4.9):1) α_s is dependent on atom \mathbf{d}_{k_s} , which is yet to be determined, and more importantly 2) α_s itself also appears in the signs s_m of (4.9). Fortunately, Issue 1) can be resolved by examining all the atoms in the dictionary that have not been selected for patch \mathbf{x} . And for each atom considered, Issue 2) can also be circumvented by testing all 3^M (because sign can be +1,-1 or 0) combinations of s_m and selecting the combination that yields the least $J_c(k_s, \alpha_s)$. As a result, (4.6) is then solved. That is, we complete the s th atom selection and its coefficient calculation. Proposed sparse coding stage is summarized in Algorithm 4. Flow chart of proposed algorithm is presented in Figure 4.1.

Algorithm 4: Sparse Coefficient Estimation Based On Inter-patch Correlation regularization

Input

- 1: \mathbf{x} : noisy input patch, $\mathbf{D} = [\mathbf{d}_k]$, $k = 1, 2, \dots, K$: dictionary, σ : noise level
- 2: S : Maximum number of atoms to be selected
- 3: M : number of neighbors
- 4: \mathbf{r}^m , $m = 1, 2, \dots, M$: the number of neighboring residuals of current patch \mathbf{x}

EndInput

5: **procedure**

6: $\mathbf{r}_0 = \mathbf{x}$

7: **for** $s = 1, 2, \dots, S$ **do**

8: Obtain m neighboring residuals of current patch \mathbf{x}

9: **if** $\|\mathbf{r}_{s-1}\| > \sigma^2$ **then**

10: **for** $k = 1, 2, \dots, K$ **do**

11: **for** $z = 1, 2, \dots, Z$ **do**

12: For each combination of sign z , compute α_s using equation (4.9) and save it with corresponding atom

13: Calculate (4.6) for current k_s, α_s and sign z

14: **endfor**

15: **endfor**

16: Select sign k_s and α_s that minimizes (4.6)

17: $\mathbf{r}_s = \mathbf{r}_{s-1} - \mathbf{d}_{k_s} \alpha_s$

18: **endif**

19: **endfor**

20: **endprocedure**

4.4.2 Dictionary Updating

The above described sparse coding stage assumed that the dictionary \mathbf{D} is known and fixed. If we wish to learn the dictionary from noisy image patches, we can resort to the two stage dictionary learning algorithm employed in [2, 7, 13, 37], that is, we alternate between the sparse coding stage and a new dictionary update stage. We now consider the dictionary update stage; the new sparse coding stage was described in the previous section.

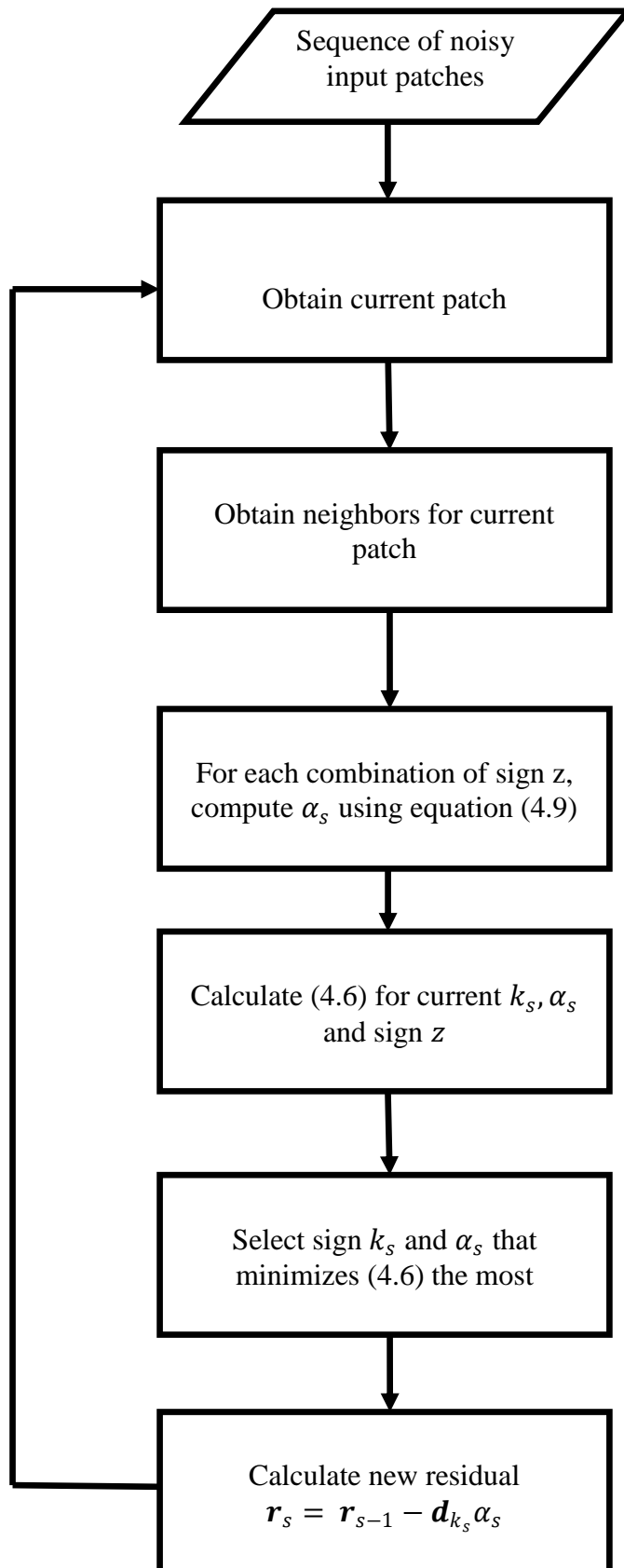


Figure 4.1: The flow chart of proposed algorithm.

Consider the t th iteration, with dictionary \mathbf{D}_{t-1} . Let the current patch be \mathbf{x}_t with sparse code α_t , and then the corresponding residue is $\mathbf{r}_t = \mathbf{x}_t - \mathbf{D}_{t-1}\alpha_t$. Similarly, let \mathbf{r}_i be the residuals of the neighboring patches that have been processed under \mathbf{D}_{t-1} with the corresponding code α_i , $i = 1, \dots, t-1$. The new dictionary can be updated by minimizing the objective function (Detailed steps are presented in Appendix B)

$$\begin{aligned} J_d(D) &= \frac{1}{t} \left[\sum_{i=1}^t \|\mathbf{r}_i\|_2^2 + \sum_{i=1}^{t-1} \lambda_i |\mathbf{r}_t^T \mathbf{r}_i| \right] \\ &= \frac{1}{t} \left[\sum_{i=1}^t \|\mathbf{x}_i - \mathbf{D}\alpha_i\|_2^2 + \sum_{i=1}^{t-1} \lambda_i |(\mathbf{x}_t - \mathbf{D}\alpha_t)^T (\mathbf{x}_i - \mathbf{D}\alpha_i)| \right] \end{aligned} \quad (4.10)$$

$$\begin{aligned} J_d(D) &= \frac{1}{t} \left[\left(\frac{1}{2} (\text{Tr}(\mathbf{D}^T \mathbf{D} \mathbf{A}_t - 2\mathbf{D}^T \mathbf{B}_t + \mathbf{x}_t \mathbf{x}_t^T)) \right) \right. \\ &\quad \left. + \text{Tr}(\mathbf{D}^T \mathbf{D} \mathbf{f}_{t-1} \alpha_{t-1}^T - \mathbf{D}^T \mathbf{g}_{t-1} \alpha_{t-1}^T - \mathbf{D} \mathbf{f}_{t-1} \mathbf{x}_t^T \right. \\ &\quad \left. + \mathbf{g}_{t-1} \mathbf{x}_t^T) \right] \end{aligned} \quad (4.11)$$

where

$$\mathbf{A}_t = \sum_{i=1}^t \alpha_i \alpha_i^T = \mathbf{A}_{t-1} + \alpha_t \alpha_t^T \quad (4.12)$$

$$\mathbf{B}_t = \sum_{i=1}^t \mathbf{x}_i \alpha_i^T = \mathbf{B}_{t-1} + \mathbf{x}_t \alpha_t^T \quad (4.13)$$

$$\mathbf{f}_t = \sum_{i=1}^{t-1} \lambda_i s_i \alpha_i = \mathbf{f}_{t-1} + \lambda_{t-1} s_{t-1} \alpha_{t-1} \quad (4.14)$$

$$\mathbf{g}_t = \sum_{i=1}^{t-1} \lambda_i s_i \mathbf{x}_i = \mathbf{g}_{t-1} + \lambda_{t-1} s_{t-1} \mathbf{x}_{t-1} \quad (4.15)$$

With:

$$s_i = \text{sgn}((\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}_t)^T(\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i)) \quad (4.16)$$

Function J_d in (4.10) explicitly shows that the neighboring residual patches contribute to the dictionary update. Note that if no neighborhood patch \mathbf{x}_i is used in the constraints in equation 4.10, then the proposed dictionary update reduces to the method of optimal directions for frame design [41]. We now solve for the new dictionary \mathbf{D} :

$$\mathbf{D}_t = \underset{\mathbf{D}}{\text{argmin}} J_d(\mathbf{D}) \quad (4.17)$$

Taking the derivative of the objective function in (4.11) w.r.t. \mathbf{D} and setting it to be zero (Detailed steps are presented in Appendix C), we have:

$$\mathbf{D}\mathbf{A}_c - \mathbf{B}_c = 0 \quad (4.18)$$

where

$$\mathbf{A}_c = \mathbf{A}_t + 2\sigma_\lambda \mathbf{f}_{t-1} \boldsymbol{\alpha}_{t-1}^T \quad (4.19)$$

$$\mathbf{B}_c = \mathbf{B}_t + \sigma_\lambda (\mathbf{g}_{t-1} \boldsymbol{\alpha}_{t-1}^T + \mathbf{f}_{t-1} \mathbf{x}_t^T) \quad (4.20)$$

The linear equation (4.18) can be approximately solved by:

$$\mathbf{D} = \mathbf{D}_{t-1} + (\mathbf{B}_c - \mathbf{D}_{t-1}\mathbf{A}_c)\mathbf{A}_c^{-1} \quad (4.21)$$

Since the coefficient vectors $\boldsymbol{\alpha}$ are sparse, when λ_i are selected to be small, the coefficients of \mathbf{A}_c are generally diagonal. As a result, the k th column of \mathbf{D} can be approximately updated as

$$\mathbf{D} = \mathbf{D}_{t-1} + (\mathbf{B}_c - \mathbf{D}_{t-1}\mathbf{A}_c)[\text{diag}(\mathbf{A}_c)]^{-1} \quad (4.22)$$

And the k th column of dictionary can be calculate by an approximation followed by a normalization:

$$\mathbf{u}_k \leftarrow \frac{1}{\mathbf{A}_c(k, k)} (\mathbf{B}_c(:, k) - \mathbf{D}_{t-1} \mathbf{A}_c(:, k)) + \mathbf{D}_{t-1}(:, k)$$

$$\mathbf{D}_t(:, k) \leftarrow \frac{1}{\max(\|\mu_k\|_2, 1)} \mathbf{u}_k \quad (4.23)$$

Note that \mathbf{A}_c and \mathbf{B}_c are related to the signs $s_i, i = 1, \dots, t - 1$, which are dependent on \mathbf{D} . To circumvent this dependency, we can in principle consider all combinations of the possible signs. However, it would quickly become infeasible as t increases. Nevertheless, we can practically overcome this problem in two ways:

- 1) We include only a small number of correlation terms in the objective function (4.10) (e.g., let $M = 2$); and 2) we use \mathbf{D}_{t-1} instead of \mathbf{D}_t in determining the sign, that is,

$$s_i = \text{sgn}((\mathbf{x}_t - \mathbf{D}_{t-1} \boldsymbol{\alpha}_t)^T (\mathbf{x}_i - \mathbf{D}_{t-1} \boldsymbol{\alpha}_i)), i = 1, \dots, t - 1 \quad (4.24)$$

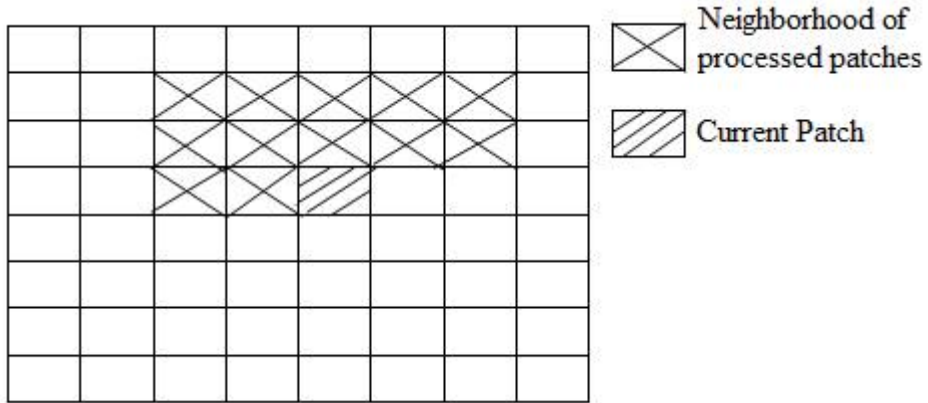


Figure 4.2 Neighborhood of processed patches

4.5 Computational Complexity

In this section, we discuss the computational complexity of the proposed algorithm. Recall that the K-SVD performs $O(nKLP)$ operations per pixel [1], where n, K, L, P are the size of the patch, the size of the dictionary atoms, the maximum number of atoms selected, and the number of iterations, respectively.

Similar approach is adapted in the proposed algorithm, except that for each patch being processed, we need to test for different combinations of signs in order to select the atom and to update the dictionary. As a result, the computational cost of the proposed algorithm becomes $O(cnK^2LP)$, where c is the number of sign combinations considered at each atom selection (sparse coding) and updating. Therefore, the proposed algorithm is computationally more expensive than K-SVD [1]. On the other hand, we shall show in Section 4.6 that the correlation based regularization can help produce much better results than the K-SVD, both qualitatively and quantitatively. Note that the computational burden can be alleviated by searching for suboptimal solutions of (4.6) and (4.17); for example, by selecting atoms that reduce the objective function below certain threshold value. This shall significantly decrease the computational complexity at the cost of minor loss in performance.

4.6 Limitations And Future Work

In this section, the limitations of the proposed algorithm are discussed and also future work is suggested.

Computation cost can further be improved as discussed in Section 4.5. Hence, further investigation in this regard is suggested as a future work. Also performance of the proposed algorithm is not as good at low noise levels as it is at high noise levels.

Thus, this limitation also creates opportunity to further investigate and improve the performance of the residual correlation based proposed image denoising algorithm.

4.7 Experimental Results And Comparison

This section compares the performance of the proposed algorithm with four state-of-the-art algorithms of image denoising, namely, the K-SVD [1], BM3D [6], EPLL[14] and NCSR [39] algorithm. Firstly, quantitative performance evaluation is performed. Later, qualitative experiments are compared.

4.7.1 Quantitative Performance Evaluation

In this subsection, the quantitative performance of proposed algorithm is compared with state-of-the-art algorithms. This quantitative comparison is conducted based on PSNR (Peak Signal to Noise Ratio), SSIM (Structure Similarity Index Measure) and FSIM (Feature Similarity Index Measure).

4.7.1.1 Comparison Based On PSNR Results

The PSNR is calculated as $10(\log_{10} 255^2 / MSE)$ where $MSE = \frac{1}{n} \|\mathbf{X} - \hat{\mathbf{X}}\|^2$. Where \mathbf{X} and $\hat{\mathbf{X}}$ are the original and denoised images respectively. The experiment is performed through MATLAB program. We select some standard test images, all of size 512×512 . Then image segments of size 80×80 are extracted from each image. These segments are further divided into 15×15 fully overlapping patches. The AWGN noise \mathbf{w} of power between 25 to 100 is generated. The number of immediate neighborhood patches is set to be $M = 2$. We find one atom for the patch and form its residual. Then, we move to next patch to pick its first atom and so on. When we pick first atom for all the patches and formed their residuals then we pick second atom for all the patches. This process continues until residual power goes below noise power and/or maximum number of atoms to be picked is reached. Figure 4.2 shows that how sequentially patches are processed and neighborhood of

processed patches is obtained. Finally, when sparse coefficients for all the patches are estimated then dictionary is updated. Note that we process (denoise) 80×80 segments of image separately. Thus, we initialize dictionary of size 225×35 ($n = 225, K = 35$) with pre-chosen and fixed dictionary as given in [1].

The PSNR results are presented in Table 4.1. We evaluate the performance of proposed algorithm for each image separately. In order to get clear comparison the PSNR results for each image are shown in Figure 4.3.

As proposed algorithm is based on sparse-land model thus its performance is compared with K-SVD [1]. Also, for fair comparison, different types of the standard images are selected. For example, the images with abundance of high frequency content like Barbara, Fingerprint and Straw images are tested. Furthermore, in order to evaluate the performance at low noise levels and also at high noise levels the AWGN noise with noise level with sigma from 25 to 100 is generated.

Let us evaluate the results of each image given in Figure 4.3. For the Barbara image, the performance of proposed algorithm is better than state of the art KSVD [1] and EPLL [14]. Contrary to K-SVD [1], performance of the proposed algorithm keeps improving with increasing noise levels. On the other hand, for Barbara image, BM3D [6] and NCSR [39] produce high PSNR results when compared to KSVD [1], EPLL [14] and proposed algorithm.

Table 4.1: PSNR results in decibels. Top left: Results of K-SVD[1]. Top right: NCSR [39]. Middle left: BM3D [6]. Middle right EPLL [14]. Bottom Left: Proposed Algorithm.

Sigma	Barbara		Boat		Fingerprint		Lena		Straw		Avg $\Delta PSNR$ w.r.t KSVd
	25	30	29.27	29.67	27.26	27.77	31.32	31.91	25.45	25.51	
25	29.58	30.61	29.27	29.67	27.26	27.77	31.32	31.91	25.45	25.51	-0.02
	30.72	28.51	29.91	29.61	27.71	27.09	32.08	31.54	25.04	25.23	
	29.56	-0.02	29.13	-0.14	27.56	0.3	31.25	-0.07	25.65	0.2	
30	28.55	29.68	28.42	28.79	26.31	26.97	30.45	31.07	24.39	24.71	1.08
	29.81	27.54	29.12	28.77	26.83	26.21	31.26	30.79	24.22	24.39	
	28.68	0.13	28.44	0.02	26.68	0.37	30.52	0.07	24.88	0.49	
50	25.53	27.03	25.94	26.52	23.26	24.48	27.87	29.01	22.08	22.56	1.7
	27.23	24.88	26.78	26.58	24.53	23.52	29.05	28.32	22.41	22.24	
	25.79	0.26	26.01	0.07	23.86	0.6	28.07	0.2	22.65	0.57	
75	22.98	24.76	23.95	24.67	20.04	22.65	25.65	27.02	21.04	21.33	1.15
	25.12	23.02	25.12	24.87	22.83	21.48	27.26	26.47	21.52	21.29	
	24.43	1.45	24.41	0.46	22.61	2.57	26.35	0.7	21.63	0.59	
100	21.85	23.24	22.83	23.42	18.31	21.34	24.42	25.66	20.45	20.71	1.08
	23.62	22.13	23.97	23.66	21.61	19.82	25.95	25.33	21.05	20.81	
	23.07	1.22	23.25	0.42	21.24	2.93	24.72	0.3	20.99	0.54	

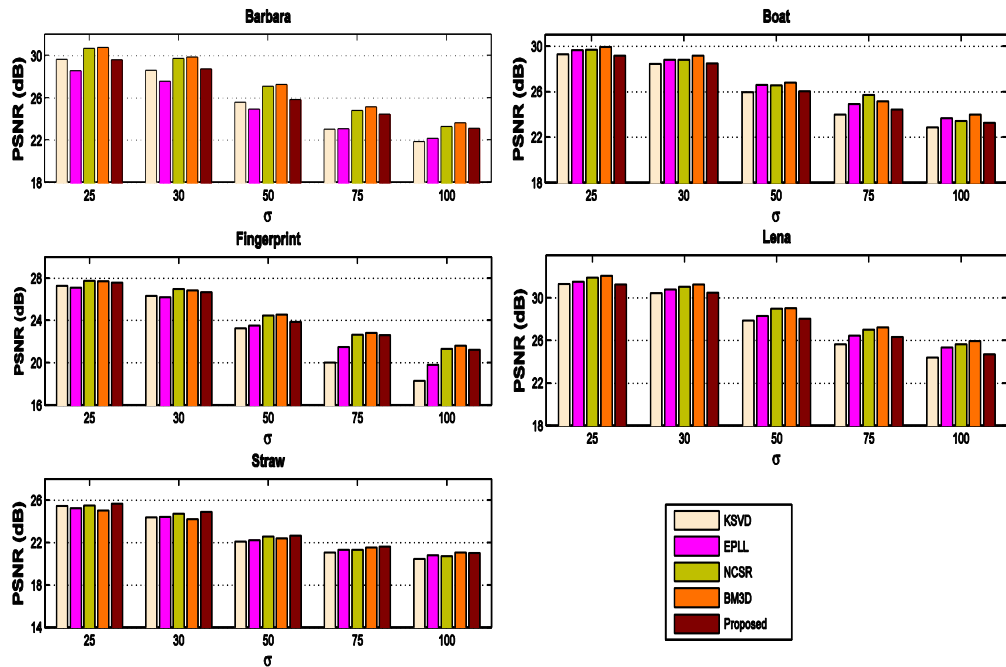


Figure 4.3: The PSNR results comparison.

Now let us analyze performance of the proposed algorithm with increasing noise levels when compared to other image denoising results. In order to understand this, we keep K-SVD[1] results as a zero reference. This reference line is shown in Figure 4.4 as a straight line at zero. Figure 4.4 shows that the difference in PSNR results of the proposed algorithm keeps decreasing with increasing noise levels when compared to BM3D [6] and NCSR [39] algorithms. Hence, we conclude that residual correlation regularization based image denoising is highly effective at high noise levels.

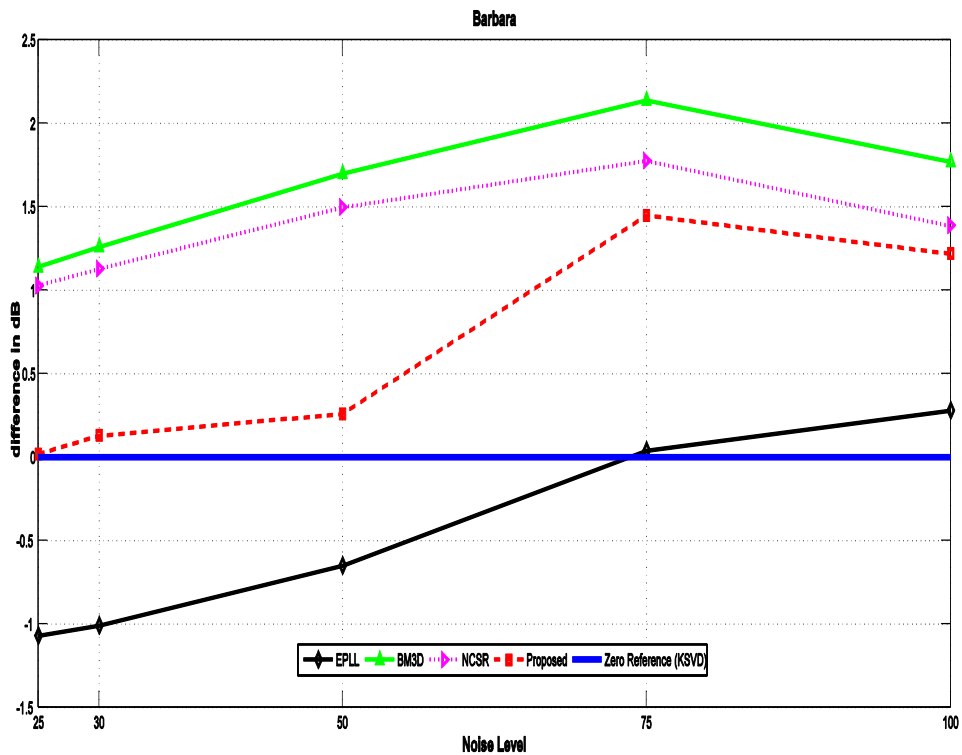


Figure 4.4: Difference in PSNR comparison for Barbara image.

Similarly, when we compare the results of Boat and Lena images given in Figure 4.3, it is evident that proposed algorithm performs reasonably better than KSVD [1] especially at high noise levels. However, for these images, the performance of proposed algorithm is not as good as it is for Barbara image. Hence, we conclude that

residual correlation regularization is more effective for images with plenty of high frequency content. Let us verify this claim, by comparing the PSNR results of Fingerprint and Straw images. As both of these images contain large quantities of high frequency contents, performance of the proposed algorithm is better as compared to its performance for Lena and Boat images. We can see from Figure 4.3 that the proposed algorithm outperforms KSVD and EPLL algorithms for the Fingerprint and Straw images. Especially, for the Straw image, the performance of the proposed algorithm exceeds BM3D at some noise levels. Also for the Fingerprint image performance of the proposed algorithm is a good complement to the BM3D and NCSR algorithms. Hence, it verifies our claim that the information of residual correlation becomes effective when dealing with images that possess large quantity of high frequency content. Figure 4.5 presents the difference in PSNR results of all algorithms with zero reference KSVD for the Fingerprint image.

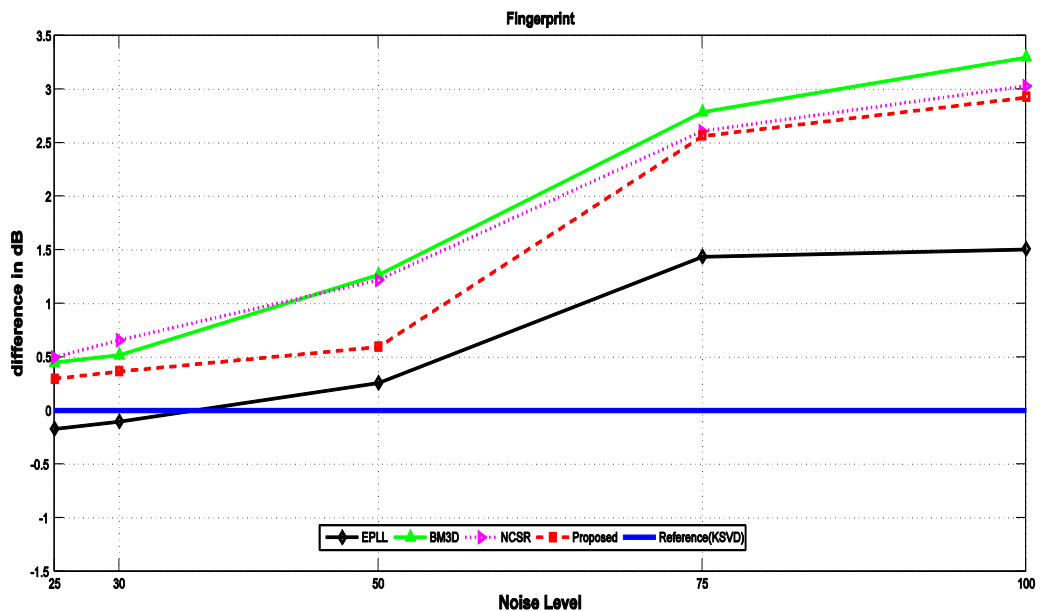


Figure 4.5: Difference in PSNR comparison for Fingerprint image.

It is evident from Figure 4.5 that the proposed residual correlation regularization based image denoising algorithm keeps improving with increasing noise levels when compared to state-of-the-art image denoising algorithms. Similarly, Figure 4.6 shows the difference in PSNR results for the Straw image. It shows that for the Straw image the proposed algorithm produces best PSNR results for some noise levels when compared to all other algorithms.

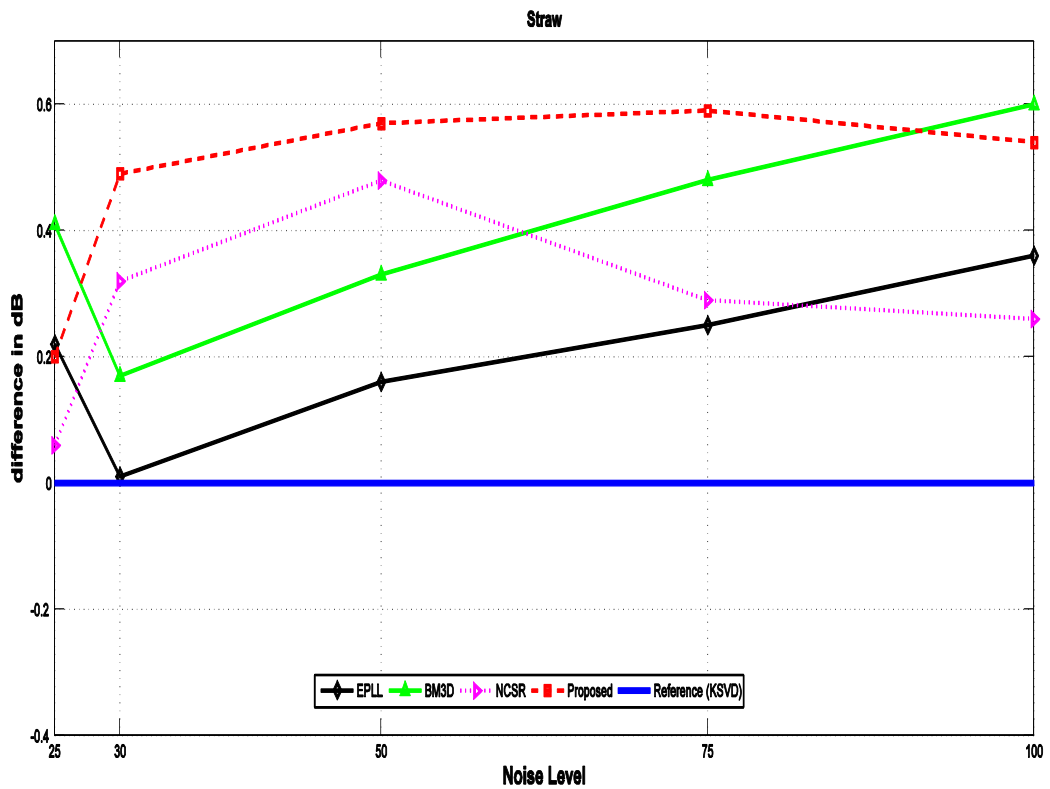


Figure 4.6: Difference in PSNR comparison for Straw image.

Now, let us analyze the effect of number of neighboring residuals considered for sparse coding and dictionary update stages. In order to evaluate this comparison, we set $M = 1$ and $M = 2$. Figure 4.7 presents PSNR results at noise levels 50, 75 and 100 when $M = 1$ and $M = 2$. Figure 4.7 shows that performance of the proposed algorithm improves when we increase the number of neighboring residuals. It shows

that often performance of the proposed algorithm improves if more information about neighboring residuals is incorporated during sparse coding and dictionary update stages. In order to limit the computational cost we set $M = 2$ for our experiments.

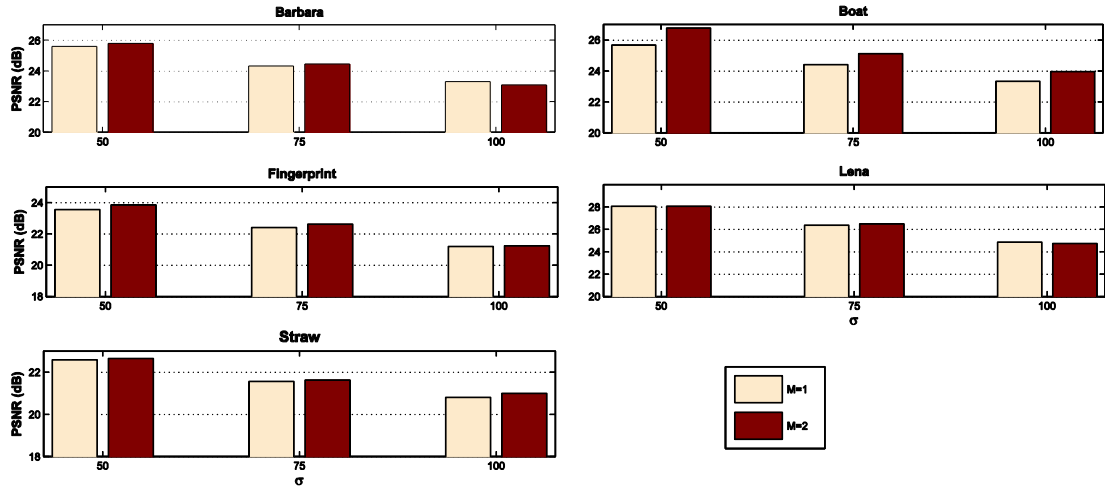


Figure 4.7: PSNR comparison when $M = 1$ and $M = 2$.

4.7.1.2 Comparison Based On SSIM Results

In this section, we compare the proposed algorithm with state of the art algorithms with respect to SSIM. Table 4.2 presents the SSIM results obtained for few standard images at various noise levels. However, we plot these values to better understand the behavior of proposed algorithm in terms of SSIM.

Figure 4.8 presents the comparison of proposed algorithm with state of the art algorithms in terms of SSIM results. Similar to PSNR comparison, it is evident that performance of the proposed algorithm is better for the Barbara, Fingerprint and Straw images as compared to its performance for Lena and Boat images.

If we evaluate the SSIM results, we come to know that proposed algorithm performs better than KSVD and NCSR algorithms especially at high noise levels. However, EPLL and BM3D perform better than KSVD, NCSR and proposed algorithm in terms of SSIM.

Table 4.2: SSIM results. Top left: Results of K-SVD[1]. Top right: NCSR[39]. Middle left: BM3D [6]. Middle right EPLL [14]. Bottom Left: Proposed Algorithm.

Sigma	Barbara		Boat		Fingerprint		Lena		Straw	
25	0.651	0.796	0.676	0.681	0.908	0.923	0.721	0.735	0.095	0.081
	0.851	0.797	0.811	0.811	0.922	0.922	0.833	0.841	0.711	0.458
	0.782		0.711		0.915		0.743		0.741	
30	0.574	0.703	0.638	0.649	0.891	0.905	0.668	0.684	0.172	0.141
	0.823	0.743	0.781	0.789	0.906	0.899	0.801	0.801	0.638	0.312
	0.734		0.698		0.903		0.714		0.685	
50	0.356	0.499	0.462	0.486	0.775	0.844	0.442	0.535	0.314	0.342
	0.629	0.495	0.669	0.673	0.861	0.828	0.696	0.686	0.414	0.311
	0.583		0.523		0.819		0.563		0.414	
75	0.213	0.284	0.345	0.395	0.569	0.781	0.312	0.401	0.095	0.079
	0.481	0.401	0.579	0.581	0.813	0.727	0.591	0.571	0.291	0.249
	0.411		0.445		0.764		0.445		0.241	
100	0.178	0.199	0.274	0.315	0.383	0.727	0.273	0.262	0.172	0.141
	0.401	0.374	0.514	0.511	0.768	0.577	0.507	0.452	0.241	0.248
	0.366		0.351		0.736		0.361		0.291	

For the straw image the proposed algorithm outperforms all the algorithms at some noise levels. Moreover, SSIM results for Barbara and Fingerprint presented in Figure 4.8 also verify the competitiveness of proposed algorithm with state of the art denoising algorithms.

We conclude that structures of the recovered images are well preserved by the proposed algorithms. Later, in this chapter, the visual comparison of the proposed algorithm is also verifying the SSIM results.

4.7.1.3 Comparison Based On FSIM Results

In this section, the proposed algorithm is compared with the state-of-the-art algorithms. Similar to the SSIM comparison, the proposed algorithm outperforms KSVD and NCSR in terms of FSIM as shown in Figure 4.9. However, EPLL and BM3D produce better FSIM results when compared to KSVD, NCSR and proposed algorithm.

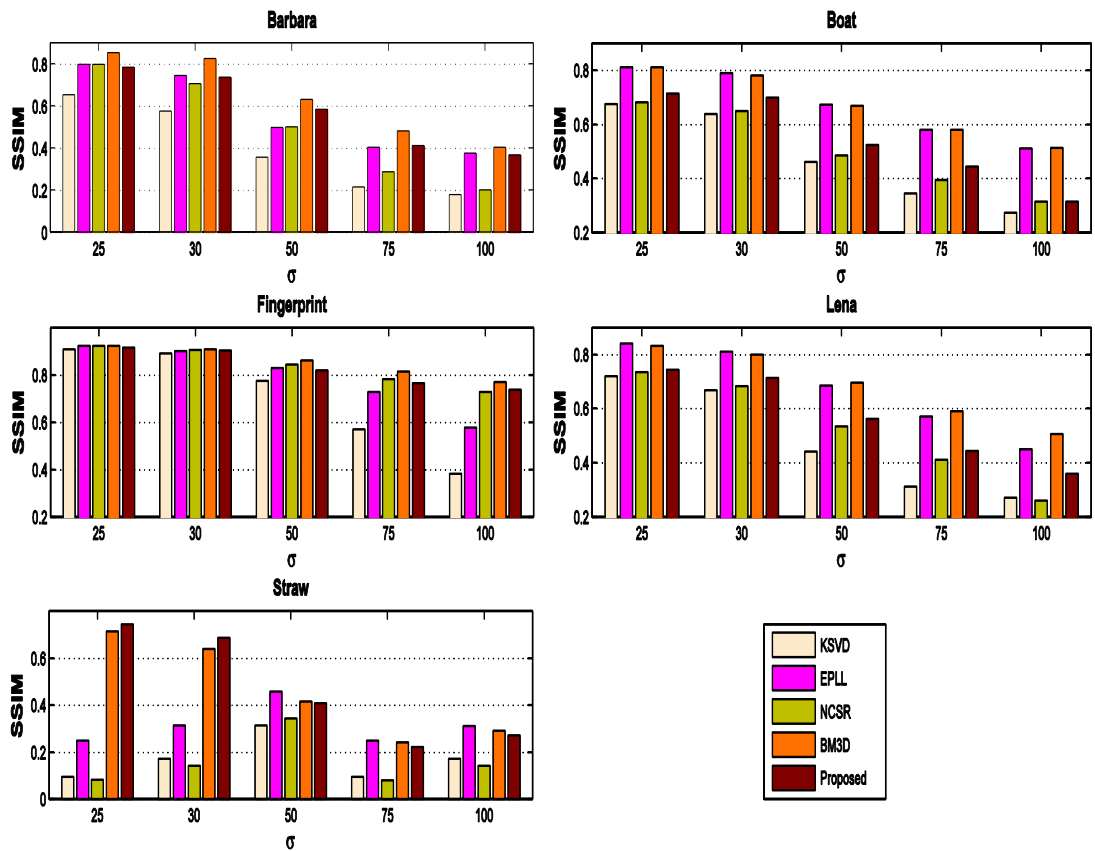


Figure 4.8: The SSIM results comparison.

We conclude that for the images that are rich in high frequency content like Barbara and Fingerprint images, the performance of proposed algorithm is relatively better.

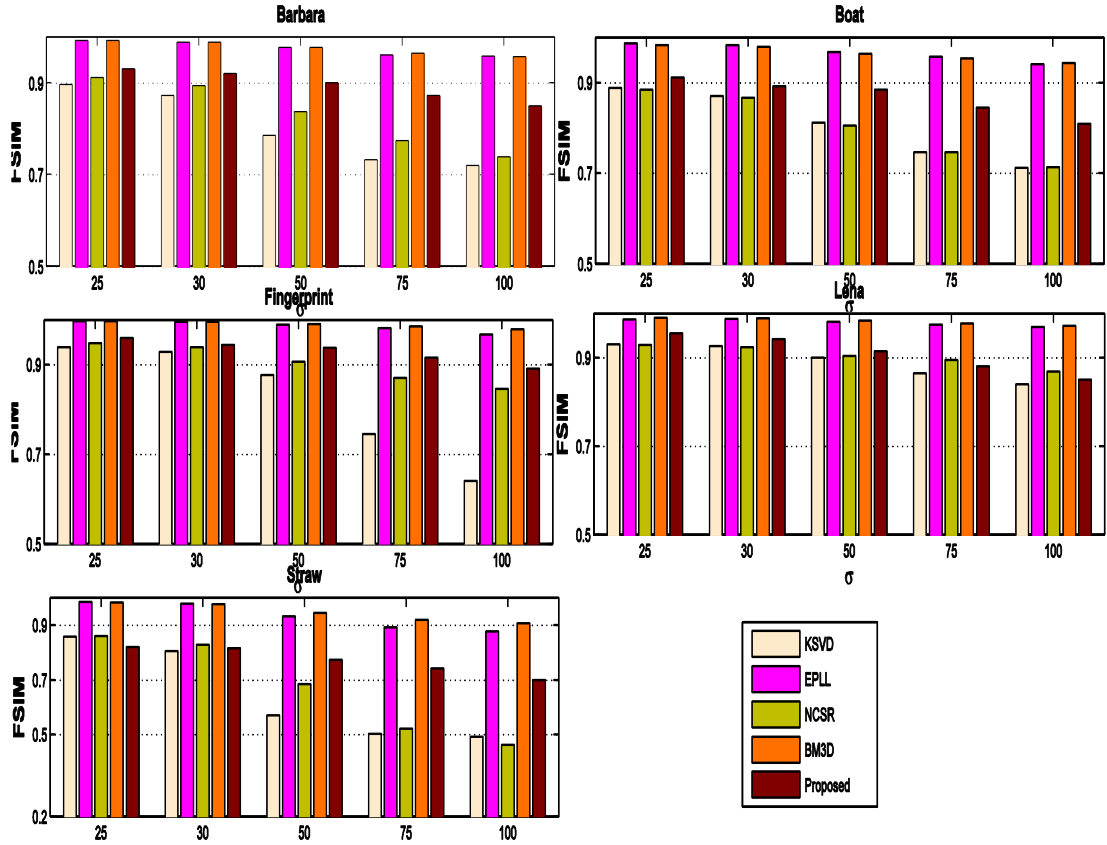


Figure 4.9: The FSIM results comparison.

4.7.2 Qualitative Experiments

In this section, comparison of the visual results is presented. As in previous section, we conclude that information about residual correlation is very essential for recovering highly repeating structures at high noise levels. Thus, portion of Barbara and Fingerprint images at high noise levels is denoised by all comparing algorithms.

As the noise level increases the regularization parameter λ_m is also increased from 0.5 to 1. With increased noise levels, the performance of the proposed algorithm improves more in terms of both indices.

Firstly, we present results for high frequency portion of Barbara image Figure 4.10. This image segment is corrupted with $\sigma = 60$ and then it is recovered by KSVD,

EPLL, NCSR, BM3D and the proposed algorithm. It is clear from Figure 4.10 that the repeating structures of the scarf and portion near the hand are well preserved by proposed algorithm. It is to note that KSVD and EPLL do a poor job in recovering such fine structures. Also, it shows that proposed algorithm performs as good if not better than NCSR and BM3D algorithms.

Similarly, the portion of recovered Fingerprint image is presented in Figure 4.11. We corrupt the segment of Fingerprint image with $\sigma = 100$. Then, we recover it with all the algorithms to compare their qualitative performance.

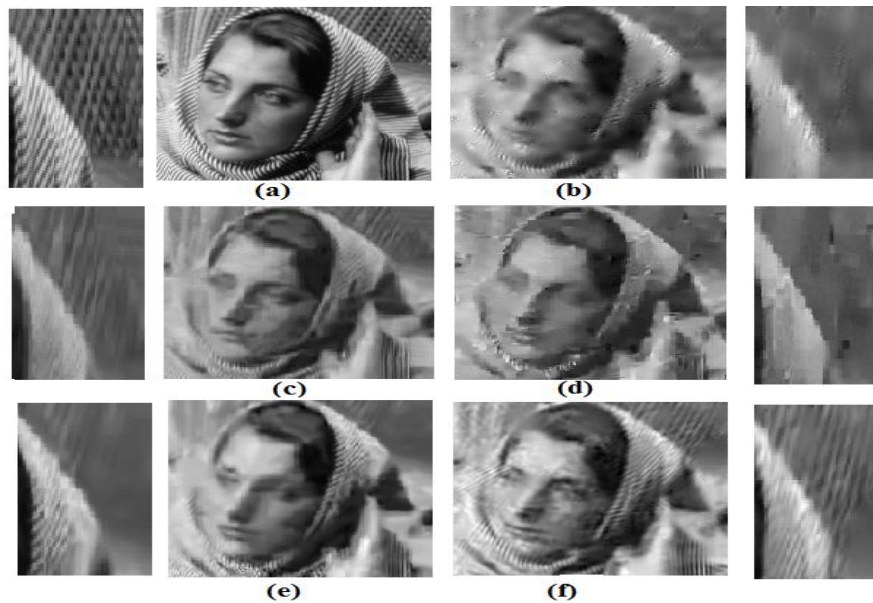


Figure 4.10: Visual Comparison of Barbara image with $\sigma = 60$ (a) Original image (b) denoised by KSVD [1] (c) denoised by BM3D [6] (d) denoised by EPLL [14] (e) denoised by NCSR [39] (f) denoised by proposed algorithm

If we closely investigate the Figure 4.11, we find that KSVD and EPLL fail to recover the ridges of Fingerprint image at $\sigma = 100$. Whereas, proposed algorithm does excellent job to recover these fine structures at high noise level. Usually, these structures are considered as “hard-to-recover”. As a result most of the image denoising algorithms fail to preserve the structures of these portions after denoising.

Especially, as result of selecting incorrect atom, maximum projection based algorithms fail to recover these structures. However, information about the correlation between neighboring residuals has proved to be useful in recovering such fine structures.

Furthermore, visual results obtained in Figure 4.10 shows that the proposed algorithm is as good if not better than BM3D and NCSR algorithms.

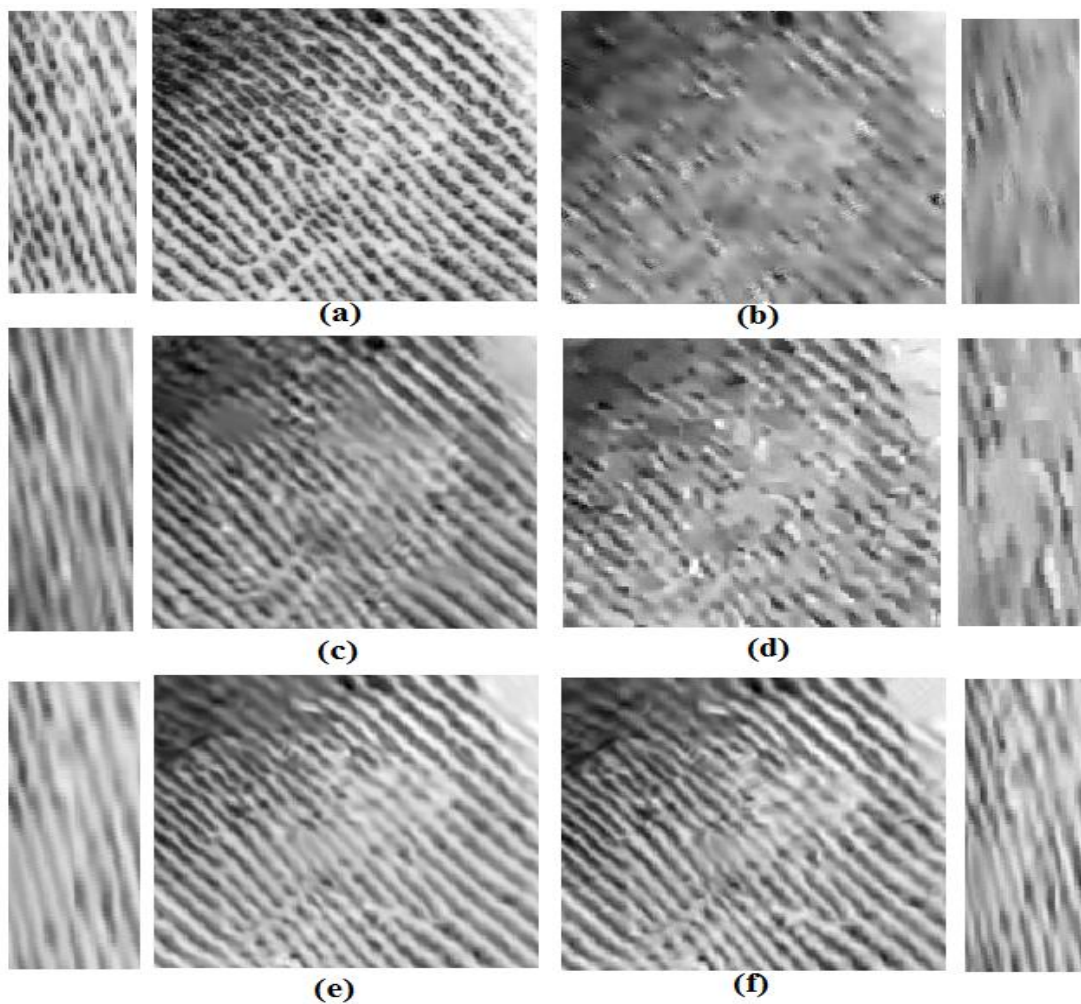


Figure 4.11: Visual Comparison of Fingerprint image with $\sigma = 100$ (a) Original image (b) denoised by KSVD [1] (c) denoised by BM3D [6] (d) denoised by EPLL [14] (e) denoised by NCSR [39] (f) denoised by proposed algorithm

Finally, we combine Barbara and Fingerprint image as shown in Figure 4.12. The portion of Barbara image recovered for $\sigma = 75$ and the portion of Fingerprint image denoised for $\sigma = 100$ is presented in Figure 4.12. Visual results in Figure 4.12 verify that proposed algorithm is highly effective to recover repeated structures at high noise levels. For the $\sigma = 75$, the stripes on the scarf are very well recovered by proposed algorithm.

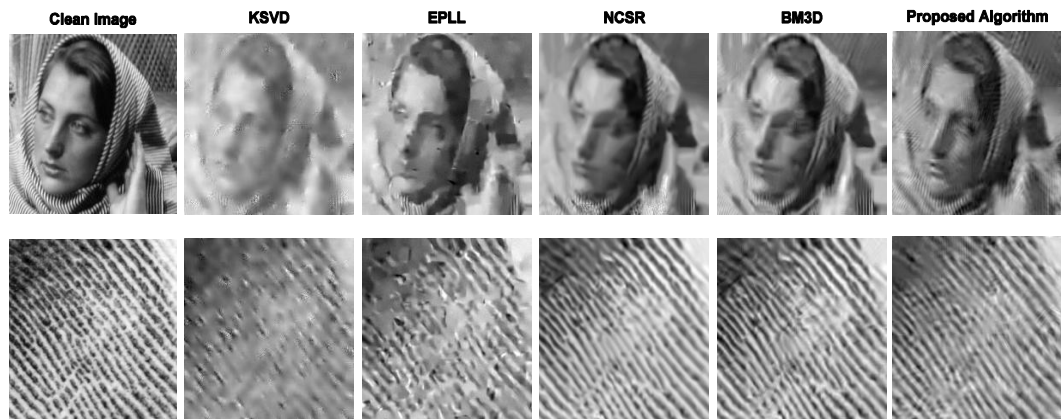


Figure 4.12: Visual Comparison for Barbara ($\sigma = 75$) and Fingerprint ($\sigma = 100$).

4.8 Comprehensive Performance Evaluation At High Noise Levels

According to motivation presented in Section 4.3, the performance of the proposed residual correlation reduction based image denoising algorithm should be improving with increasing noise levels. This phenomenon was observed in the experimental results presented in Section 4.7. It is observed that producing better results at high noise levels is the specialty of the proposed algorithm. In order to further investigate this characteristic of the proposed algorithm the experiments are performed on variety of images in next Sections. Also various types of noises are used in denoising process to verify the persistence of the proposed algorithm. Finally, synthetic images with high frequency content are created using fixed DCT (Discrete Cosine

Transform) bases. The results of these synthetic images are evaluated using heat map.

4.8.1 Performance Evaluation With Other Type Of Noises

The proposed algorithm is used for denoising AWGN in previous Section. However, Gaussian noise can also be colored in nature which is commonly known as Addictive White Colored Noise (ACGN). Also noise can be long tailed Laplacian noise. Now, performance of the proposed algorithm is evaluated for denoising images containing ACGN and Laplacian noise.

4.8.1.1 Denoising Images Corrupted With ACGN

ACGN can be generated by passing AWGN through a low pass or high pass digital filter. Firstly, ACGN is produced by using FIR digital filter with frequency response and impulse response as shown in Figure 4.13.

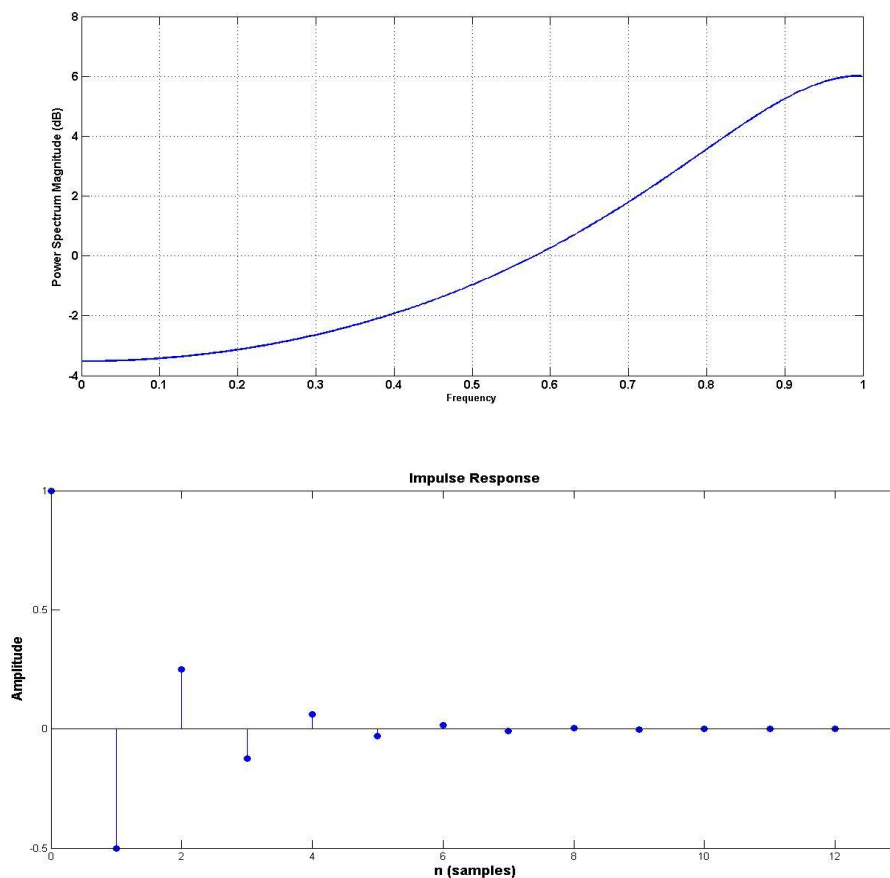


Figure 4.13: Frequency Response and Impulse Response of Digital Filter Used To Generate ACGN.

Table 4.3: PSNR results for ACGN in decibels. Top left: Results of K-SVD[1]. Top right: NCSR [39]. Middle left: BM3D [6]. Middle right EPLL [14]. Bottom Left: Proposed Algorithm, Bottom Right: $\Delta PSNR$ w.r.t KSVD.

Sigma	Barbara		Boat		Fingerprint		Lena		Straw		Avg $\Delta PSNR$ w.r.t KSVD
50	21.63	21.14	21.50	20.31	20.43	19.34	22.15	20.78	19.37	19.66	0.9
	22.05	24.99	21.77	26.81	19.23	23.79	20.08	27.75	20.48	22.18	
	22.69	1.06	23.12	1.62	20.63	0.20	23.01	0.86	20.26	0.89	
75	18.43	17.63	18.76	17.08	17.57	16.49	19.30	22.24	17.47	16.99	3.06
	19.01	23.26	18.82	25.35	18.49	22.01	18.49	25.34	17.97	21.34	
	21.73	3.3	22.48	3.72	19.71	2.14	21.83	2.53	21.09	3.62	
100	16.54	14.08	17.01	14.28	15.33	13.86	16.94	14.45	15.83	14.10	4.26
	16.92	22.64	16.85	24.22	16.57	20.23	17.05	23.34	16.15	21.01	
	21.11	4.57	21.73	4.72	18.95	3.62	20.52	3.53	20.73	4.9	

When ACGN (with zero mean and known variance) is added to clean images and the proposed sparse coding and dictionary update stages are iterated few times then image denoising results are obtained as presented in Table 4.3.

Note that proposed algorithm performs better in terms of PSNR when compared to all algorithms except EPLL [14]. At noise level 100, it outperforms KSVD by 4.26 dB on average. Visual results for Barbara and Fingerprint images presented in Figure 4.14 and Figure 4.15 respectively shows that the proposed algorithm along with EPLL [14] successfully recovers images as compared to other algorithms.

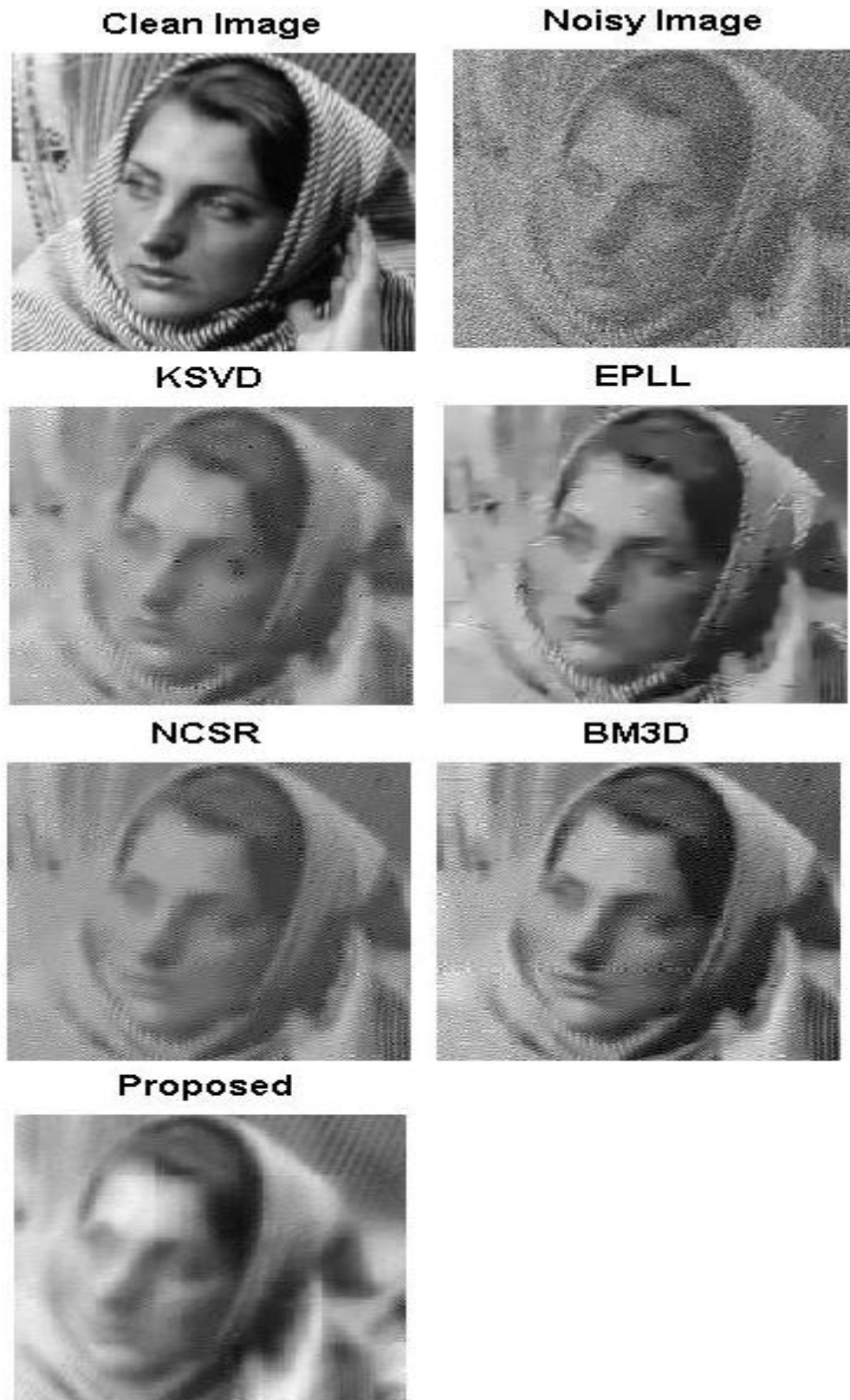


Figure 4.14: Visual Comparison for Barbara ($\sigma = 50$) Corrupted With ACGN.

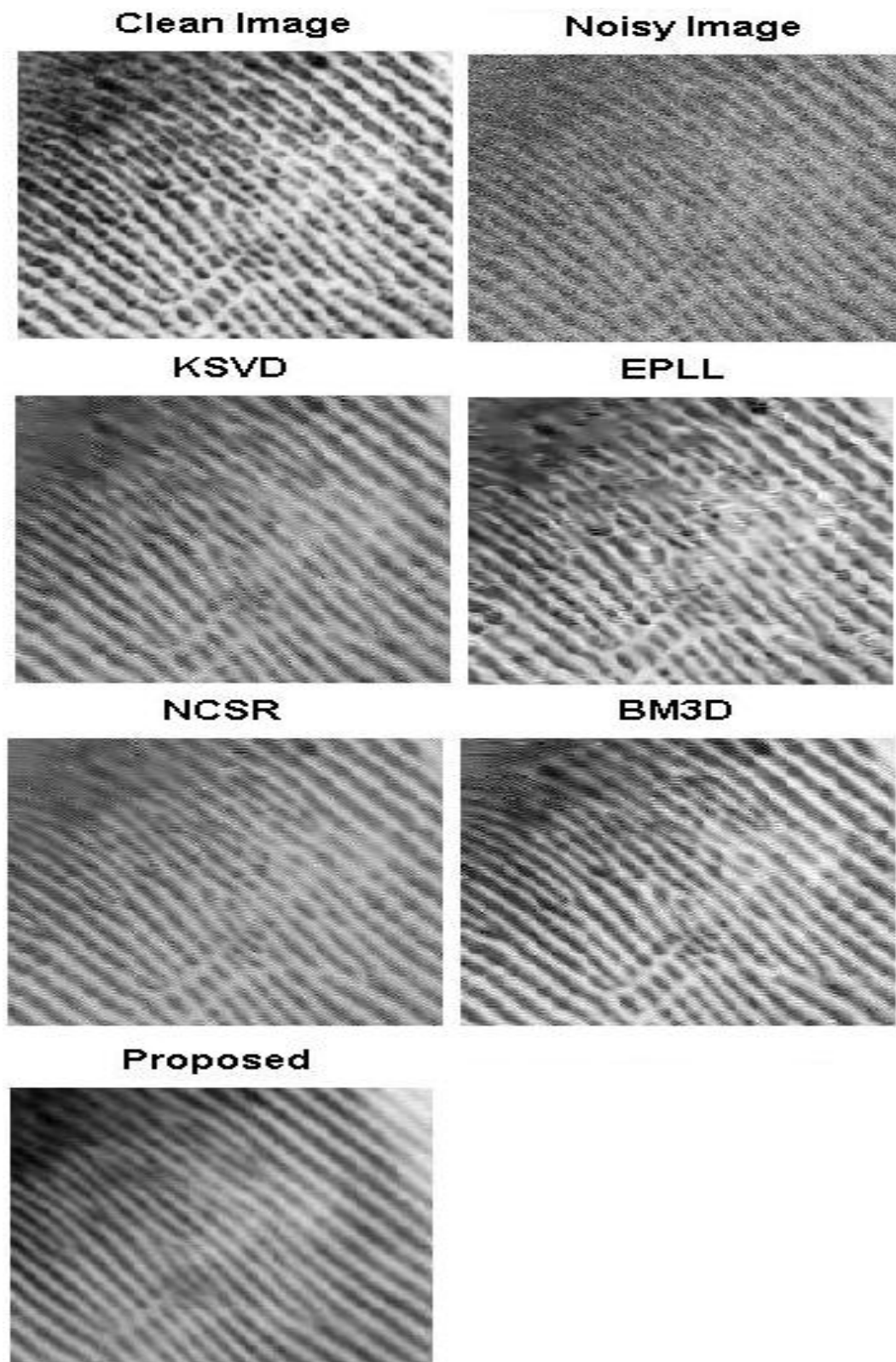


Figure 4.15: Visual Comparison for Fingerprint ($\sigma = 50$) Corrupted With ACGN.

4.8.1.2 Denoising Images Corrupted With Laplacian Noise

Now, performance of the proposed algorithm is evaluated for the images corrupted with Laplacian noise (also called biexponential). The PSNR results obtained are summarized in Table 4.4.

Table 4.4: PSNR results for Laplacian noise in decibels. Top left: Results of K-SVD [1]. Top right: NCSR [39]. Middle left: BM3D [6]. Middle right EPLL [14]. Bottom Left: Proposed Algorithm, Bottom Right: $\Delta PSNR$ w.r.t KSVD.

Sigma	Barbara		Boat		Fingerprint		Lena		Straw		Avg $\Delta PSNR$ w.r.t KSVD
50	23.87	26.88	24.21	26.41	22.01	24.48	25.71	28.90	21.17	22.37	-0.91
	27.21	24.51	26.71	25.96	24.58	23.31	28.93	27.58	22.26	22.05	
	22.72	-1.15	23.17	-1.04	20.52	-1.49	23.88	-1.83	22.13	0.96	
75	21.21	24.52	22.08	24.61	18.82	22.61	23.25	26.73	19.79	21.12	0.73
	25.04	22.06	25.01	24.34	22.78	21.22	27.36	25.84	21.39	21.09	
	21.96	0.75	22.63	0.55	19.67	0.85	23.11	-0.14	21.47	1.68	
100	19.73	22.91	20.77	23.17	17.02	21.30	21.81	25.26	18.68	20.42	1.65
	23.65	21.88	23.87	23.30	21.64	19.67	25.90	24.73	20.94	20.49	
	21.50	1.77	22.04	1.27	18.97	1.95	22.78	0.97	21.01	2.33	

The PSNR results obtained for Laplacian noisy images show that the proposed algorithm does not perform as well as it has performed for AWGN and ACGN. It is due to the fact that there are significant nonzero lag correlations in AWGN and ACGN. Hence, residual correlation reduction method is much more effective when there are good numbers of nonzero lag correlations which are supposed to be minimized. The visual results for Laplacian noise are presented in Figure 4.15 and 4.16. It is observed that proposed algorithm produces better visual results when compared to baseline method KSVD. Although, PSNR results of the proposed algorithms are not better than EPLL, NCSR and BM3D algorithms, however, it outperforms KSVD by 1.65 dB on average at noise level 100.

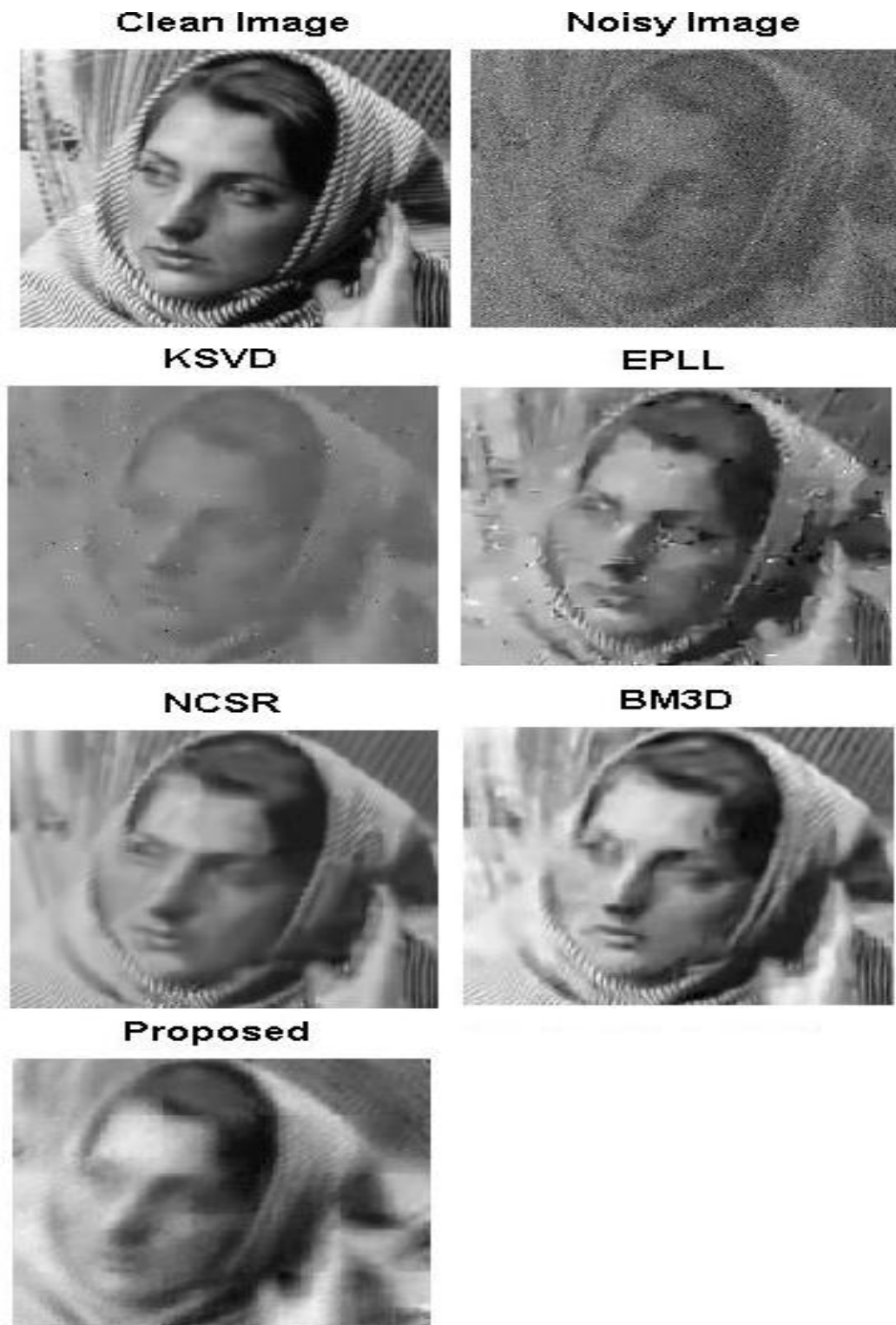


Figure 4.16: Visual Comparison ($\sigma = 50$) Corrupted With Laplacian Noise.

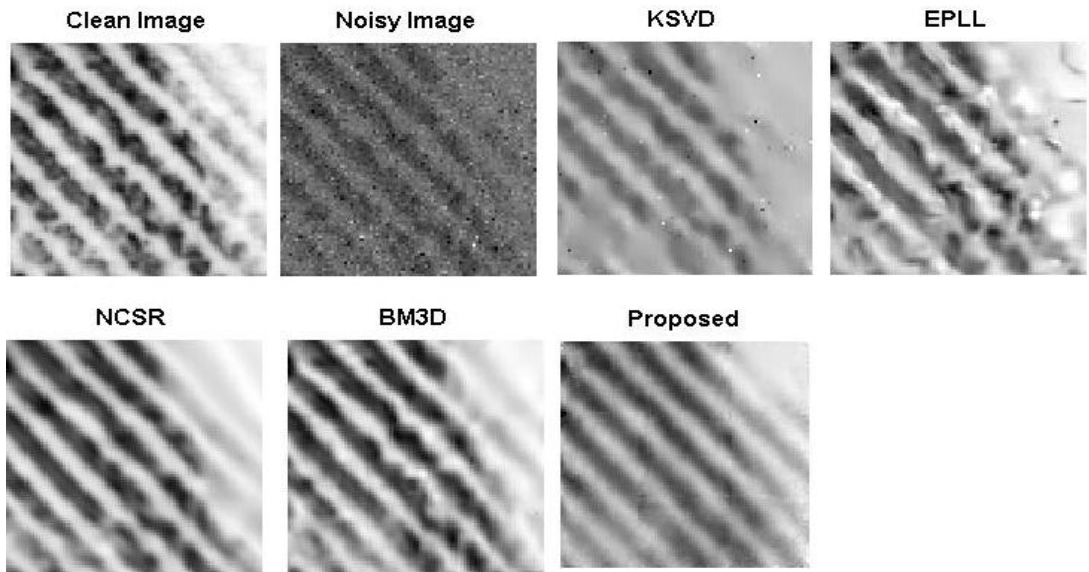


Figure 4.17: Visual Comparison ($\sigma = 50$) Corrupted With Laplacian Noise.

Now, PSNR results obtained for the Laplacian and ACGN are compared in the Figure 4.17, 4.18 and 4.20.

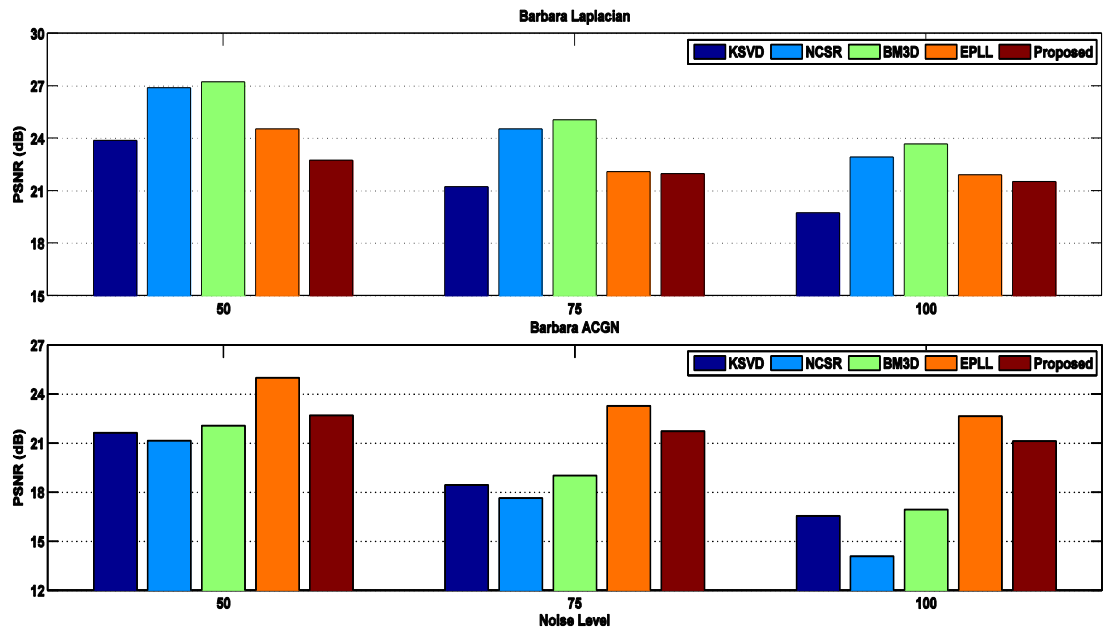


Figure 4.18: PSNR results for ACGN and Laplacian Noise.

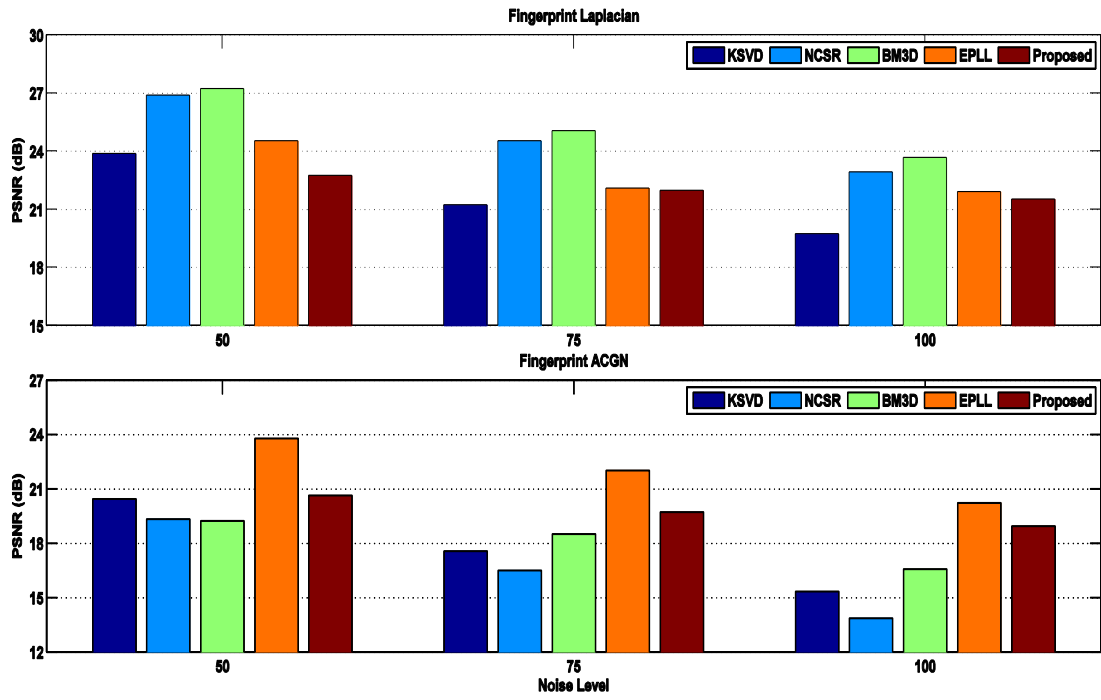


Figure 4.19: PSNR results for ACGN and Laplacian Noise.

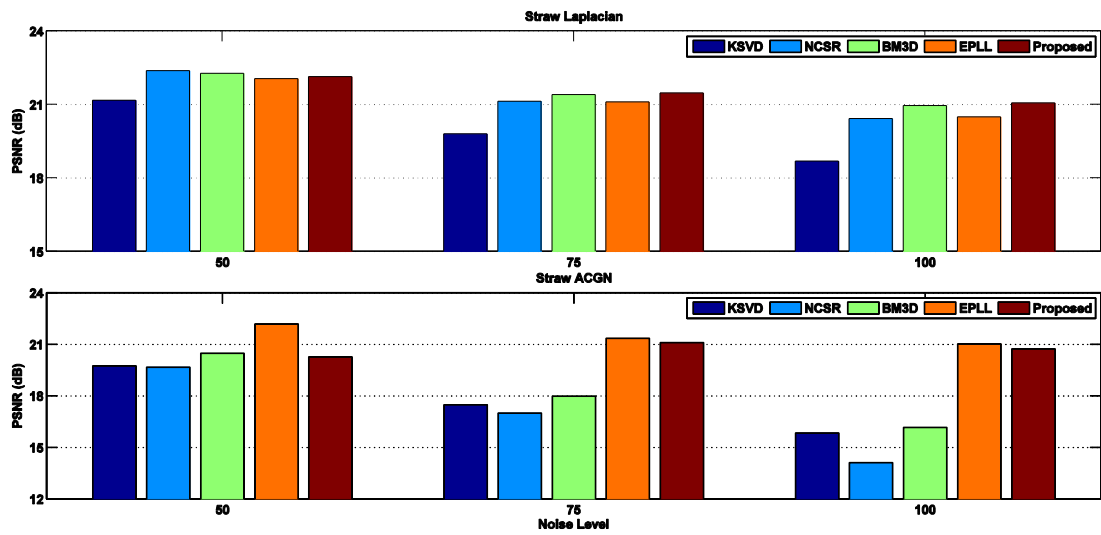


Figure 4.20: PSNR results for ACGN and Laplacian Noise.

Also, since the visual results of Barbara and Fingerprint images are presented.

Therefore, the SSIM results of these images are shown in Figure 4.21 and 4.22.

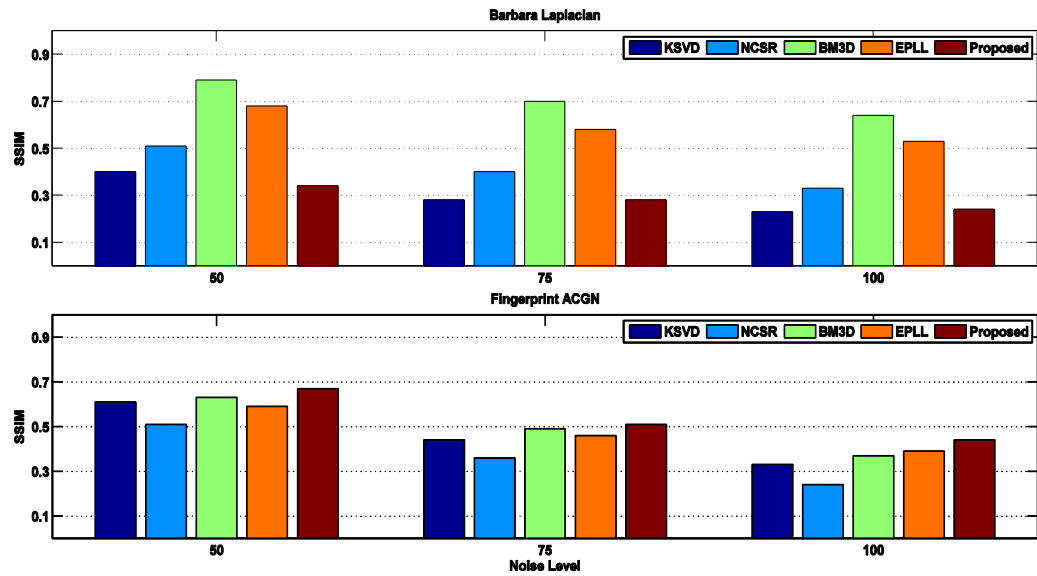


Figure 4.21: SSIM results of Barbara Image for ACGN and Laplacian Noise.

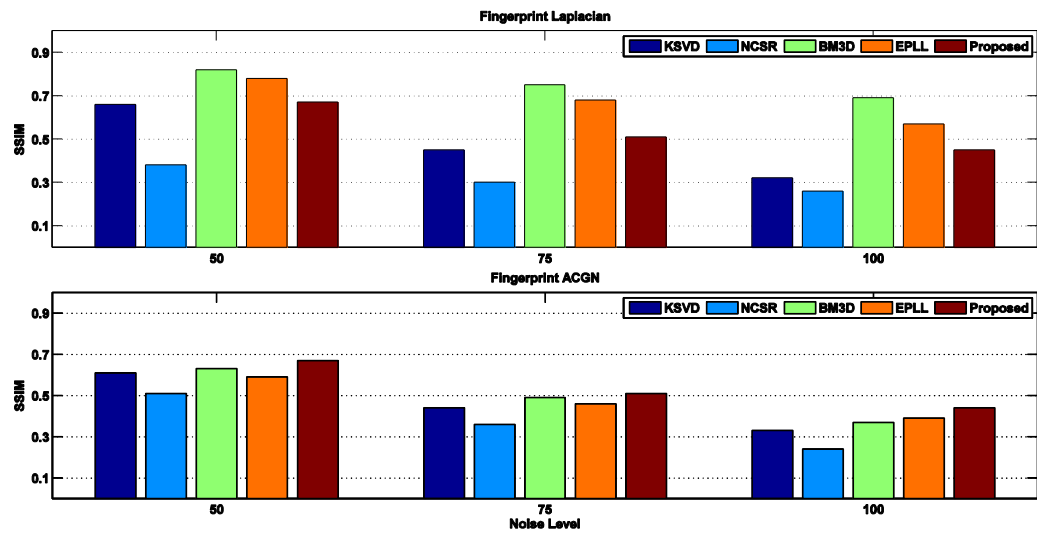


Figure 4.22: SSIM results of Fingerprint Image for ACGN and Laplacian Noise.

4.8.2 Testing Variety Of Images At High Noise Levels

In previous sections the standard test images are used to compare performance of the proposed algorithm with state-of-the-art algorithms. However, now the proposed algorithm is evaluated for images with different structures in order to further study the effectiveness of residual correlation reduction on image denoising. Well known images like House, Pepper, Pirate and Mandril are chosen. Furthermore, the images with distinctive features like text image and face image is also evaluated. The PSNR results obtained are summarized in Table 4.5.

Table 4.5 PSNR results in decibels. Top left: Results of K-SVD[1]. Top right: NCSR [39]. Middle left: BM3D [6]. Middle right: EPLL [14]. Bottom Left: Proposed Algorithm, Bottom Right: $\Delta PSNR$ w.r.t KSV D.

Sigma	House		Pepper		Pirate		Text Image		Mandril		Face		Avg $\Delta PSNR$ w.r.t KSV D
50	28.37	30.05	26.11	26.42	25.04	25.34	21.01	22.03	22.78	23.22	25.69	26.43	-0.56
	29.69	28.96	26.68	26.46	25.55	25.59	20.91	20.93	23.03	23.31	26.88	26.58	
	27.98	-0.39	25.31	-0.81	24.31	-0.73	20.57	-0.44	23.24	0.46	24.65	-1.04	
75	25.56	27.55	23.55	24.24	23.04	23.77	18.63	20.34	21.73	22.11	23.84	24.53	0.37
	29.02	26.95	25.96	24.56	24.85	23.95	20.16	19.29	22.36	22.16	26.13	24.83	
	26.25	0.69	24.38	0.83	23.38	0.34	19.03	0.67	22.34	0.61	22.97	-0.87	
100	23.54	25.86	21.72	22.77	21.99	22.7	17.79	19.71	21.13	21.21	22.70	23.02	0.66
	28.58	25.31	25.50	23.06	24.49	24.83	19.99	18.31	22.13	21.49	25.66	23.33	
	24.47	0.93	22.64	0.92	22.45	0.46	18.55	0.78	21.95	0.82	22.79	0.09	

From Table 4.5 it is obvious that the proposed algorithm keeps improving with increasing noise levels. Also, it is evident that the residual correlation reduction approach for image denoising is more effective for images with high frequency content rather than flat images like House, Pepper or Text-image. However, performance of the proposed algorithm is better for Mandril. Figure 4.23 shows that the proposed algorithm is less effective for flat images such as text image.

Clean Image

al solution of the optimization. The worst-case complexity with the problem sizes n and m problem instances encountered when the optimization does occur, it is not typical, can take a very long time.

blems with a small number of the value of finding the true global design is *worst-case analysis*.

Noisy Image

al solution of the optimization. The worst-case complexity with the problem sizes n and m problem instances encountered when the optimization does occur, it is not typical, can take a very long time.

blems with a small number of the value of finding the true global design is *worst-case analysis*.

KSVD

al solution of the optimization. The worst-case complexity with the problem sizes n and m problem instances encountered when the optimization does occur, it is not typical, can take a very long time.

blems with a small number of the value of finding the true global design is *worst-case analysis*.

EPLL

al solution of the optimization. The worst-case complexity with the problem sizes n and m problem instances encountered when the optimization does occur, it is not typical, can take a very long time.

blems with a small number of the value of finding the true global design is *worst-case analysis*.

NCSR

al solution of the optimization. The worst-case complexity with the problem sizes n and m problem instances encountered when the optimization does occur, it is not typical, can take a very long time.

blems with a small number of the value of finding the true global design is *worst-case analysis*.

BM3D

al solution of the optimization. The worst-case complexity with the problem sizes n and m problem instances encountered when the optimization does occur, it is not typical, can take a very long time.

blems with a small number of the value of finding the true global design is *worst-case analysis*.

Proposed

al solution of the optimization. The worst-case complexity with the problem sizes n and m problem instances encountered when the optimization does occur, it is not typical, can take a very long time.

blems with a small number of the value of finding the true global design is *worst-case analysis*.

Figure 4.23: Visual Comparison of Text image ($\sigma = 50$).

4.8.3 Testing With High Frequency Synthetic Images

Now the images with desired effective bandwidth are generated in order to further evaluate performance of the proposed algorithm. These images are generated by randomly selecting DCT coefficients. The images are generated by selecting 5, 15 and 25 DCT coefficients. The coefficients are selected in order to control the effective bandwidth of generated synthetic images. Six images are generated with effective bandwidth of 0.1π , 0.3π , 0.5π , 0.7π , 0.9π and π respectively both in horizontal and vertical directions. These images are varying from low effective bandwidth to high effective bandwidth. Note that it is made sure that the selected coefficients contain random weights. Finally, we test image denoising performance of the proposed algorithm on these images. Images are tested at noise levels ranging from 25 to 100. Performance of the proposed algorithm is compared with baseline method KSVD. The PSNR values obtained are used to generate the heat map as shown in Figure 4.24, 4.25, and 4.26. The horizontal axis corresponds to image effective bandwidth ranging from 0.1π to π in both directions. The vertical axis is the noise level ranging from $\sigma = 25$ to $\sigma = 100$.

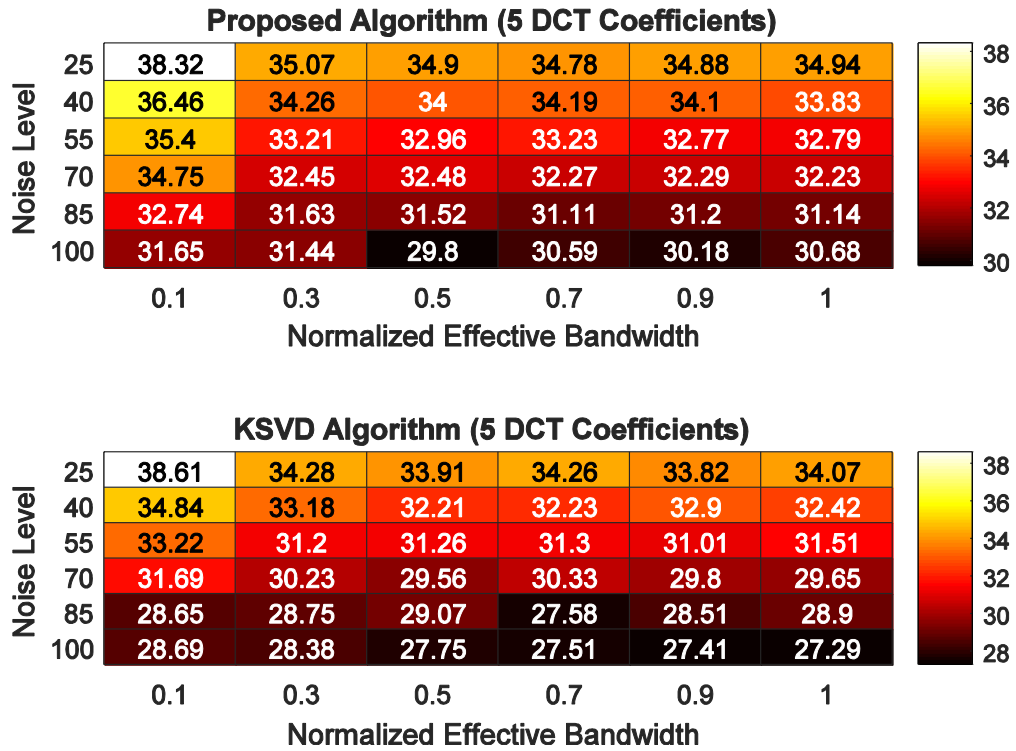


Figure 4.24: PSNR heat map of synthetic DCT images (with 5 coefficients)

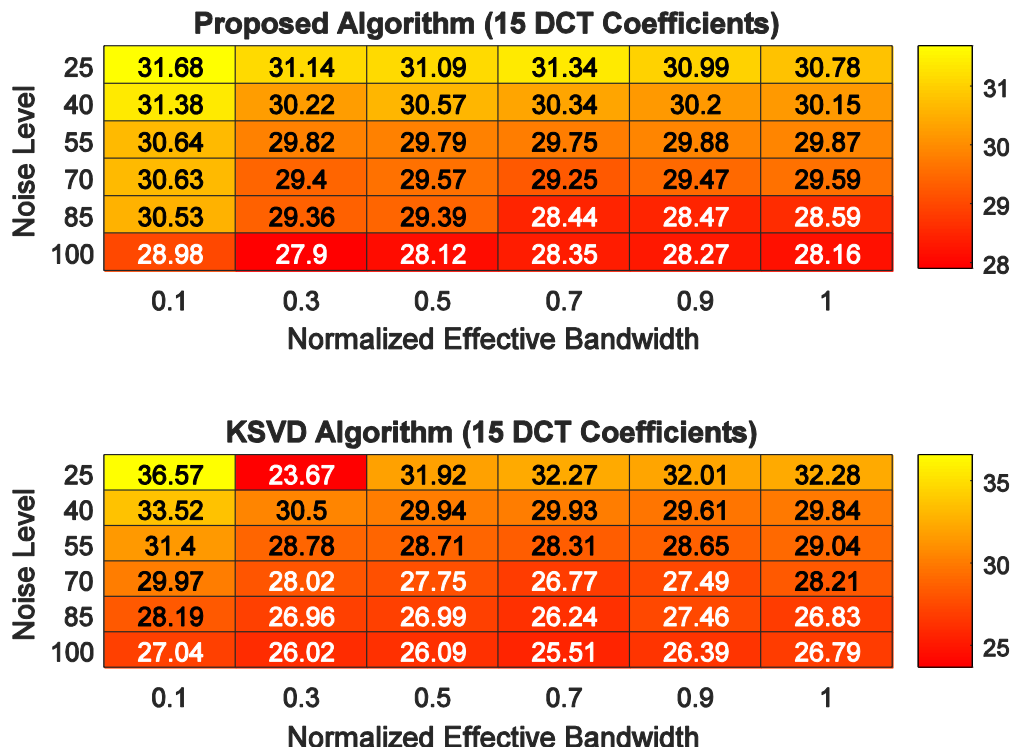


Figure 4.25: PSNR heat map of synthetic DCT images (with 15 coefficients)

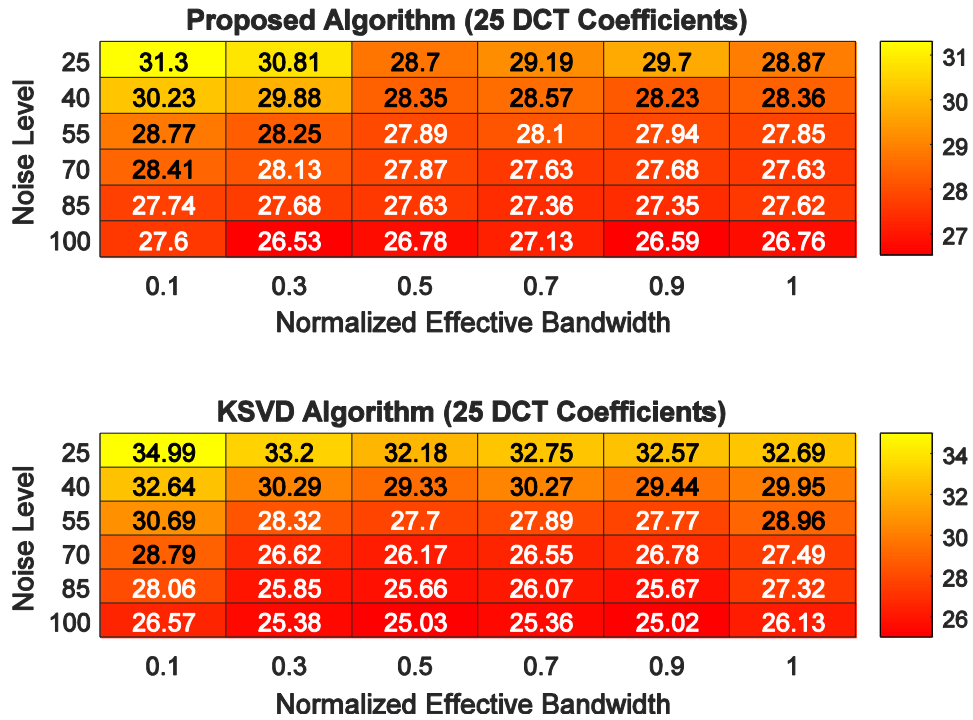


Figure 4.26: PSNR heat map of synthetic DCT images (with 25 coefficients)

The heat maps presented above show that as effective image bandwidth increases the performance of the proposed algorithm improves. Also it gives clear indication that at high noise levels the proposed algorithm performs better when compared to KSVD algorithm. Better performance at high noise levels justifies the motivation and problem definition of the proposed algorithm presented in section 4.3. Hence, we conclude that the proposed algorithm is highly effective for the images with abundance of high frequency content.

4.9 Conclusion

We presented a new residual correlation regularization that enhances the image denoising performance. This regularization helps to make statistical properties of residual to be similar to those of the contaminating noise. This is achieved by a new method of sparse coefficient estimation based on proposed regularization. A new dictionary update stage is developed based on residual correlation minimization. Our

experimental results demonstrate that the proposed regularization and sparse coefficient estimation is highly effective to recover fine structures at high noise levels.

It is indeed true that BM3D produces best PSNR results often. However, it introduces artifacts when it fails to obtain large amount of matched blocks [6]. Also, BM3D algorithm is unable to separate the edges from noise at high noise levels. As a result it can blur the edges. We have shown in Figure 4.11 that the proposed algorithm is as good if not better than BM3D algorithm in terms of visual perception.

Also as mentioned earlier that for Barbara image the proposed algorithm does a fairly good job in terms of PSNR. Furthermore, also for Fingerprint and Straw images the proposed algorithm performs reasonably better. Note that all of the above mentioned images are rich in high frequency content. Also note that contrary to BM3D, performance of the proposed algorithm improves with increasing noise levels as shown in Figure 4.4. Therefore, the reason behind comparing the proposed method with BM3D algorithm is to show that residual correlation regularization is highly effective when it is used to recover fine structures (like ridges in Fingerprint image) at high noise levels and especially images that possess abundance of high frequency content.

We conclude that the proposed algorithm have limitations in terms of PSNR when compared to BM3D algorithm, nevertheless, it is as good if not better than BM3D algorithm in terms of visual perception. Also, we can conclude that our residual correlation regularization based algorithm performs significantly better both visually and in terms of PSNR at high noise levels when compared to K-SVD [1] and EPLL

[14]. Also in terms of FSIM and SSIM results the proposed algorithm outperforms the KSVD and NCSR algorithms. It is thus a good complement to the state of the art image denoising algorithms.

Chapter 5

COUPLED KSVD DICTIONARY LEARNING ALGORITHM IN WAVELET DOMAIN FOR SINGLE IMAGE SUPER-RESOLUTION

5.1 Introduction

In the coupled K-Singular Value Decomposition (K-SVD) algorithm [35] coupled low resolution (LR) and high resolution (HR) dictionaries are trained using singular value decomposition (SVD). The best low-rank estimate is determined using SVD to make sure that the sparse coding of the two resolution levels is similar. In this chapter, we implement algorithm [35] in wavelet domain. Due to directionality and persistence properties of wavelet domain, performance of coupled KSVD in SISR (Single Image Super Resolution) is further improved. Also, in wavelet domain, dictionaries that are trained are small in size and also they are highly structured. Hence, instead of training one dictionary, multiple small dictionaries are trained to reduce the computational cost. Furthermore, since they are structured, so the performance of SISR is improved. For each low and high resolution scale, three pairs of dictionaries of wavelet subband are designed. Firstly, sparse coding of low resolution image is obtained by applying LR dictionary. Then using same sparse coding and HR dictionary a high resolution image is reconstructed. This method produces better high resolution images when compared to state of the art SISR

algorithms. Performance of algorithms is compared with respect to PSNR and SSIM results.

5.2 Background

SISR is a ill-posed inverse problem. One of the solution to this problem is by introducing the constraint priors [21, 26, 32]. Sparsity is one of the well known prior used to produce highly competitive results. Note that it has been proved that signal \mathbf{x} can be sparsely represented over dictionary \mathbf{D} and this representation is unique and reliable [16, 18, 25].

In order to better match the given signal, the signal-fitting characteristics of sparse coding is used. Hence, the representation error is reduced and better approximation is achieved but the computational cost is increased. In the literature, the idea of coupled dictionary learning is given by Yang et al [28] and they further modified it in [29]. Firstly, the LR and HR patches are concatenated to form a single feature space. Thus, the dictionaries trained contain features of both LR and HR patches. This problem is addressed by alternatively optimizing LR and HR dictionaries [29]. Similarly in [35] coupled K-SVD algorithm is presented that further improves the coupling by using the best low-rank approximation given by the SVD. In literature there are many algorithms [17, 31, 33] with improved coupling between LR and HR coefficients and dictionaries with better representation power. The idea of designing multiple dictionaries instead of single one has already been proven useful for sparse representation of signals. In [34] Elad et al. has proposed a method to learn the multi-scale dictionaries using wavelets which helps better capture the intrinsic image features. Furthermore in [23] the author has proposed the wavelet domain dictionary learning for single image super-resolution using K-SVD. Motivated by these ideas a

dictionary learning algorithm based on coupled K-SVD in wavelet domain is proposed and it is applied to the problem of Single Image Super-Resolution (SISR).

Now let us summarize the proposed SISR approach. Firstly, HR and LR patches are extracted from training data set. Then, HR and LR wavelet coefficient subbands of training images are obtained by two-level wavelet decomposition. Hence, HR and LR subbands are Level 1 and Level 2 subbands respectively. Then, pair of dictionaries is trained using the coupled K-SVD algorithm. Note that these pairs of dictionaries are trained from vertical, horizontal and diagonal wavelet subbands. Furthermore, one of basic points in coupled K-SVD algorithm is to enforce HR and LR patch pair to be at similar indices. Other important point is that for HR and LR patch pair similar sparse representation coefficient is used. Finally, it is to make sure that original HR and LR patch pair dictionary is obtained using similar sparse coding. This is achieved by alternatively calculating sparse coding for low resolution and high resolution patches and using them to update to HR and LR dictionary atoms with similar indices. We have divided image by patches of 6×6 size, each dictionary contains 256 atoms and process is iterated 20 times. Each patch is reconstructed using trained HR and LR dictionaries. At the reconstruction stage, wavelet subband of LR image is obtained by 1 level wavelet decomposition. Now, sparse coding of LR subband patch and HR dictionary is used to recover HR subband patch. Finally, HR image is obtained by one level wavelet reconstruction. Note that due to the persistence property of the wavelet coding the similarity between HR and LR subband sparse coefficients is further improved. The proposed algorithm is highly competitive if not better than state-of-the-art super-resolution algorithms. We have compared performance of the proposed algorithm in terms of visual results and also by quantitative results obtained. According to Peak-Signal to-Noise Ratio

(PSNR) results comparison, the proposed algorithm produces PSNR improvement of 1.19 dB over the algorithm of [23]. This improvement is achieved because of obtaining better estimate of high resolution subbands by implementing coupled K-SVD dictionary learning in wavelet domain.

5.3 Image Super-Resolution

One of the well known regularization for SISR is sparsity. In order to reduce computational cost the image is divided into patches [28]. In the patch based method, each patch from HR and LR image are sparsely coded using HR and LR dictionary respectively. This is given as

$$\mathbf{x}_H \approx \mathbf{D}_H \boldsymbol{\alpha}_H \quad (5.1)$$

$$\mathbf{x}_L \approx \mathbf{D}_L \boldsymbol{\alpha}_L \quad (5.2)$$

where $\boldsymbol{\alpha}_H$ and $\boldsymbol{\alpha}_L$ are the representation coefficient vectors of \mathbf{x}_H and \mathbf{x}_L respectively.

The HR and LR images are related by the blurring down-sampling as

$$\mathbf{x}_L = \psi \mathbf{x}_H \approx \psi \mathbf{D}_H \boldsymbol{\alpha}_H \approx \mathbf{D}_L \boldsymbol{\alpha}_H \quad (5.3)$$

Note that $\mathbf{D}_L = \psi \mathbf{D}_H$. Here ψ is blurring and down-sampling operator.

In [29] and [35] two dictionaries are obtained by coupled dictionary training of LR and HR patches. Finally, LR dictionary is used to find sparse representation of each patch of the LR image. It is assumed that $\boldsymbol{\alpha}_L \approx \boldsymbol{\alpha}_H$. Then, reconstruction of HR patch becomes simple by sparse code of corresponding LR patch and the HR dictionary, as

$$\mathbf{x}_H \approx \mathbf{D}_H \boldsymbol{\alpha}_H \approx \mathbf{D}_H \boldsymbol{\alpha}_L \quad (5.4)$$

In wavelet domain based SISR approach [24], dictionary is trained by using [2] for each LR subband. Then, high resolution dictionary is obtained by finding pseudo

inverse of high resolution patches and low resolution sparse codes. However, this approach does not implement any further coupling between HR and LR coefficients. Thus, in [29] and [35] concept of coupled dictionary learning is introduced.

5.4 The Proposed Super-Resolution Approach

Effectiveness of training multiple dictionaries as a replacement for of single dictionary is shown in [20]. Furthermore in [17,20] the training data is classified and divided into clusters which proved to be highly effective. Then, for each classified cluster a dictionary is trained. Based on such idea, structures and properties of wavelet subbands are transferred in dictionaries using directionality characteristics of Discrete Wavelet Transform (DWT). DWT can split image information into horizontal, vertical and diagonal features. In order to utilize these features, coupled K-SVD algorithm is implemented in the wavelet domain for SISR. When LR and HR patch pairs are kept at same indices in coupled K-SVD algorithm then better coupling between HR and LR sparse coding is achieved. Furthermore, in coupled K-SVD the HR and LR dictionary atoms are updated using single sparse code. This is achieved by alternative selection of HR and LR sparse codes to update HR and LR dictionary atoms. The coupled K-SVD proves to be effective due to better exploitation of the persistence property of wavelet codes at various resolution stages. Thus, better coupling in sparse codes is achieved by implementing coupled K-SVD on wavelet subband images.

5.4.1 Dictionary Learning Based On Proposed Method

For the training of the coupled dictionaries, we first extract a large number of HR/LR image patch pairs from some predefined database containing clean images. In order to form training detail subbands, the two-level decomposition is obtained for each HR image. Moreover, the level-one (level-two) subbands are considered as HR (LR)

training data. Also, wavelet interpolation is performed on each LR subband in order to keep its directionality same as HR [28, 30, 35]. We get the training data by developing pairs of HR and LR patches and then features from them are extracted. Note that $y = \{h, v, d\}$ represents horizontal, vertical and diagonal subbands. This process of feature extraction is same as in [29]. Then, dictionaries of HR and LR subband images are trained jointly as:

$$\begin{aligned} \min_{\mathbf{D}_L^y, \mathbf{D}_H^y, \alpha_i} &= \sum_{i=1}^N \|\mathbf{w}_L^y - \mathbf{D}_L^y \alpha_i\|_2^2 + \|\mathbf{w}_H^y - \mathbf{D}_H^y \alpha_i\|_2^2 \\ \text{s.t } &\|\alpha_i\|_0 \leq T_0, \|\mathbf{d}_L^y\|_2 \leq 1, \|\mathbf{d}_H^y\|_2 \leq 1, \\ &i = 1, 2, \dots, N \end{aligned} \quad (5.5)$$

Where \mathbf{D}_L^y and \mathbf{D}_H^y denotes LR and HR dictionaries and \mathbf{w}_L^y and \mathbf{w}_H^y represents LR and HR subband images.

In the proposed method, coupled K-SVD algorithm [35] is used for training dictionaries. Furthermore, we have chosen the Symlets wavelets [19] for DWT analysis and synthesis. Using DWT one can easily distinguish between the horizontal, vertical and diagonal details. The LR and HR subband dictionaries that is horizontal, vertical and diagonal dictionaries are trained. The proposed dictionary learning algorithm is given in Figure 5.3.

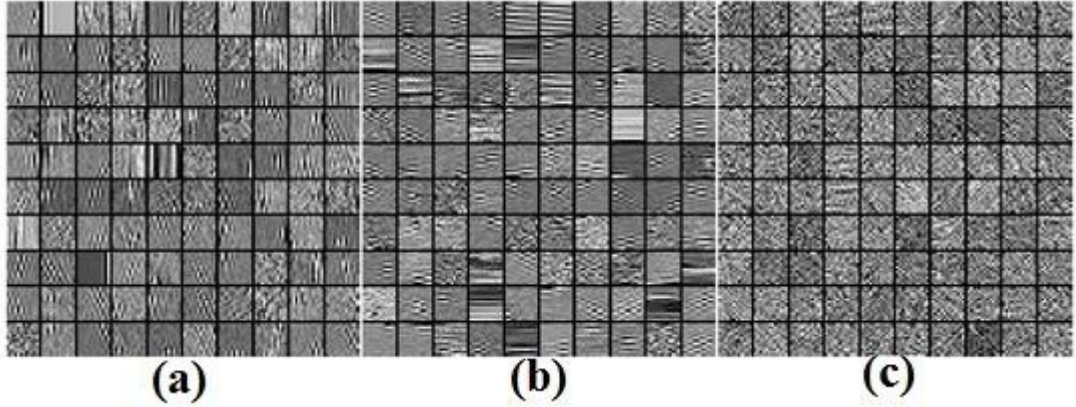


Figure 5.1: Example of HR subbands dictionary atoms. (a) Vertical Detail
(b) Horizontal Detail (c) Diagonal Detail

Figure 5.1 shows portion trained high resolution subband dictionaries. Since these dictionaries are trained on horizontal, vertical and diagonal wavelet subbands, therefore, these subband structures can clearly be observed from Figure 5.1.

5.4.2 Image Reconstruction Based On Proposed Method

In proposed SISR algorithm, wavelet subbands of HR image are determined in order to obtain original high resolution image. Note that LR image is also considered as approximation wavelet subband of high resolution image. The given LR image is first decomposed with a one-level DWT. Then LR image subband is up-converted to the size of High Resolution image by wavelet interpolation, and finally we perform overlapping of the patches on the interpolated Low Resolution image. For each LR image patch, we compute its sparse representation with respect to learned dictionary \mathbf{D}_y^L . This is performed by solving

$$\underset{\alpha_L^y}{\operatorname{argmin}} \|\mathbf{w}_L^y - \mathbf{D}_L^y \alpha_L^y\|_2 \text{ s.t. } \|\alpha_L^y\|_0 < S \quad (5.6)$$

Here S represents sparsity, and the $\|\cdot\|_0$ and $\|\cdot\|_2$ are ℓ_0 and ℓ_2 norm respectively. Note that ℓ_1 norm minimization is used to solve this vector selection problem as done in [29]. Furthermore, it is also assumed that the sparse representation of HR

wavelet subbands is same as that of the low resolution wavelet subbands $\alpha_L^y \approx \alpha_H^y$, the high resolution subbands are given by

$$\mathbf{w}_H^y \approx \mathbf{D}_H^y \alpha_H^y \approx \mathbf{D}_H^y \alpha_L^y \quad (5.7)$$

The reconstructed HR patches are used to construct the HR image by tiling operation, here for each pixel in overlapping region we take the average of many predictions and constitute a final image. In this scenario we use full overlapping mechanism. This Process is summarized in Figure 5.4.

5.5 Simulation And Results

In [23] DWT is used for image super-resolution; however, coupled dictionary training is not implemented. Thus, in order to know the effect of coupled dictionary training the proposed algorithm is compared with [23]. The proposed algorithm is also compared with [29, 35] and bicubic interpolation. [29] is considered as one of the state-of-the-art algorithm, thus, we compare its performance with the proposed algorithm. PSNR can be determined as :

$$PSNR(\mathbf{x}, \hat{\mathbf{x}}) = 10 \log_{10} \frac{\sum_{i=1}^M \sum_{j=1}^N 255^2}{\sum_{i=1}^M \sum_{j=1}^N ((\mathbf{x}(i, j) - \hat{\mathbf{x}}(i, j)))} \quad (5.8)$$

where \mathbf{x} is original image and $\hat{\mathbf{x}}$ is reconstructed image. Whereas, $M \times N$ represents the dimensions of image. We select 6×6 patch size, and number of dictionary atoms are set to be 256. Algorithm is run for 20 iterations. Note that the training set does not include image to be super-resolved.

The PSNR, SSIM and visual results are obtained for the Kodak set and other standard images. According to results the proposed algorithm outperforms Bicubic,

[29], [35], and [23] algorithms. Figure 5.2 shows the PSNR results comparison for standard test images.

It proves that the wavelet domain based coupled K-SVD algorithm is highly effective in terms of PSNR. This is due to fact that proposed algorithm makes sure that there is strong coupling between HR and LR sparse coefficients. Also, in terms of SSIM, the proposed algorithm produces as good if not better than state of the art SISR algorithms as shown in Figure 5.5.

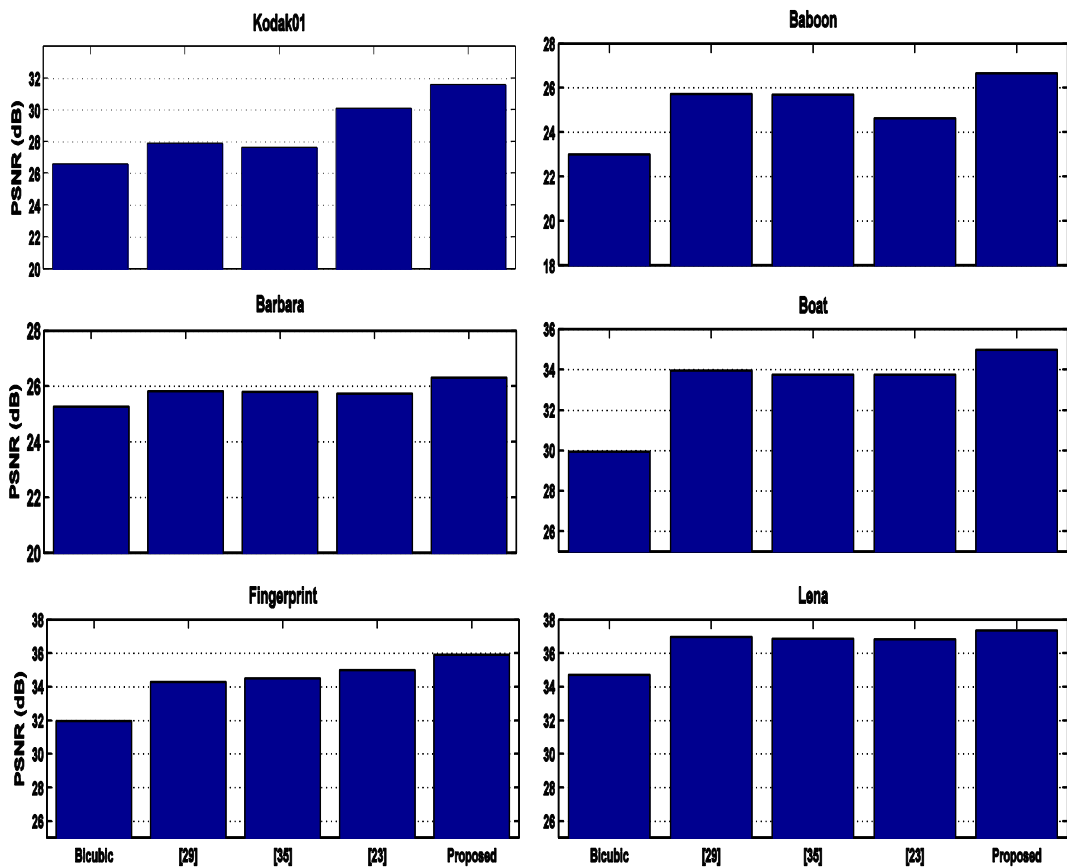


Figure 5.2: PSNR results comparison

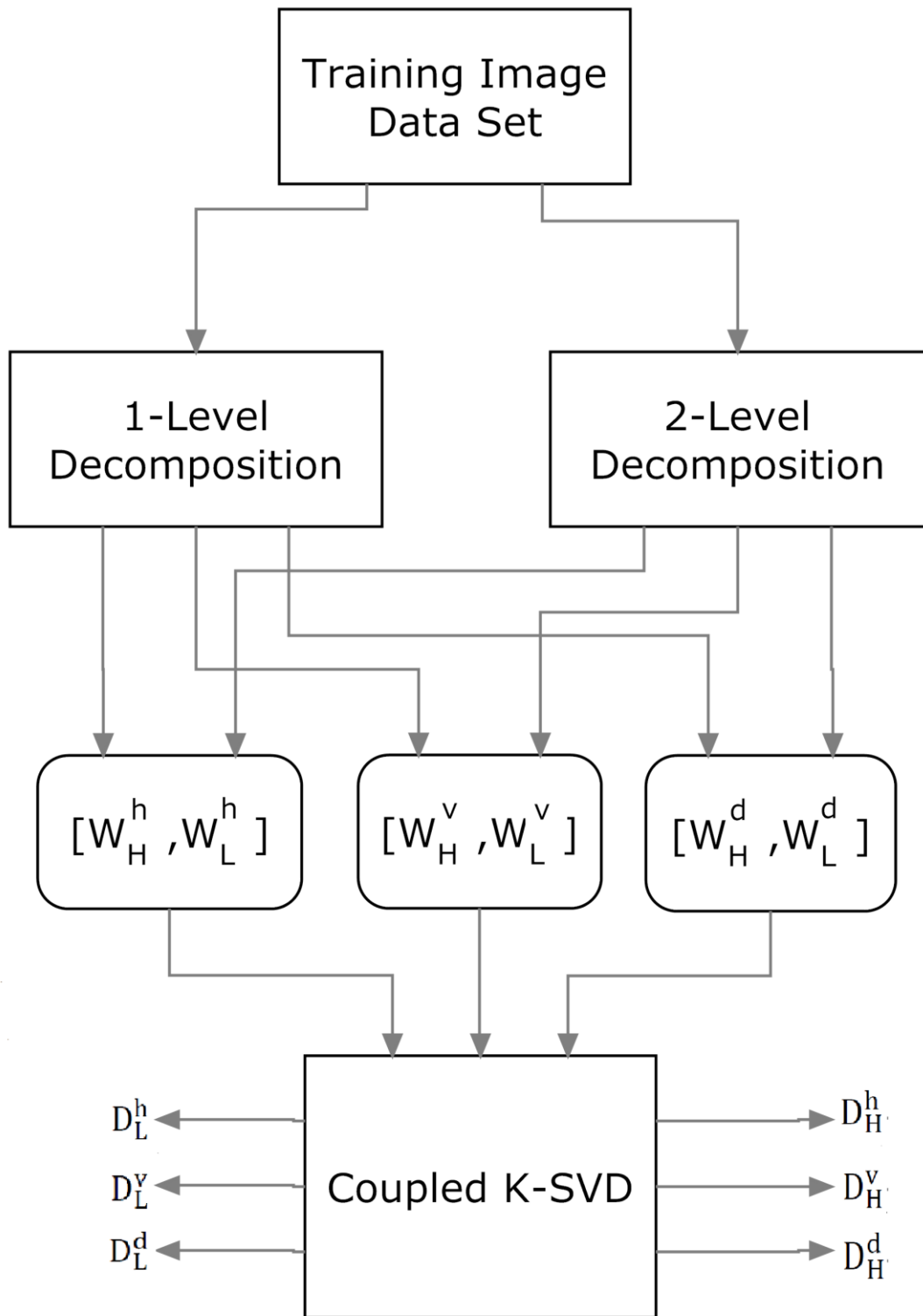


Figure 5.3: Proposed Dictionary Learning Approach

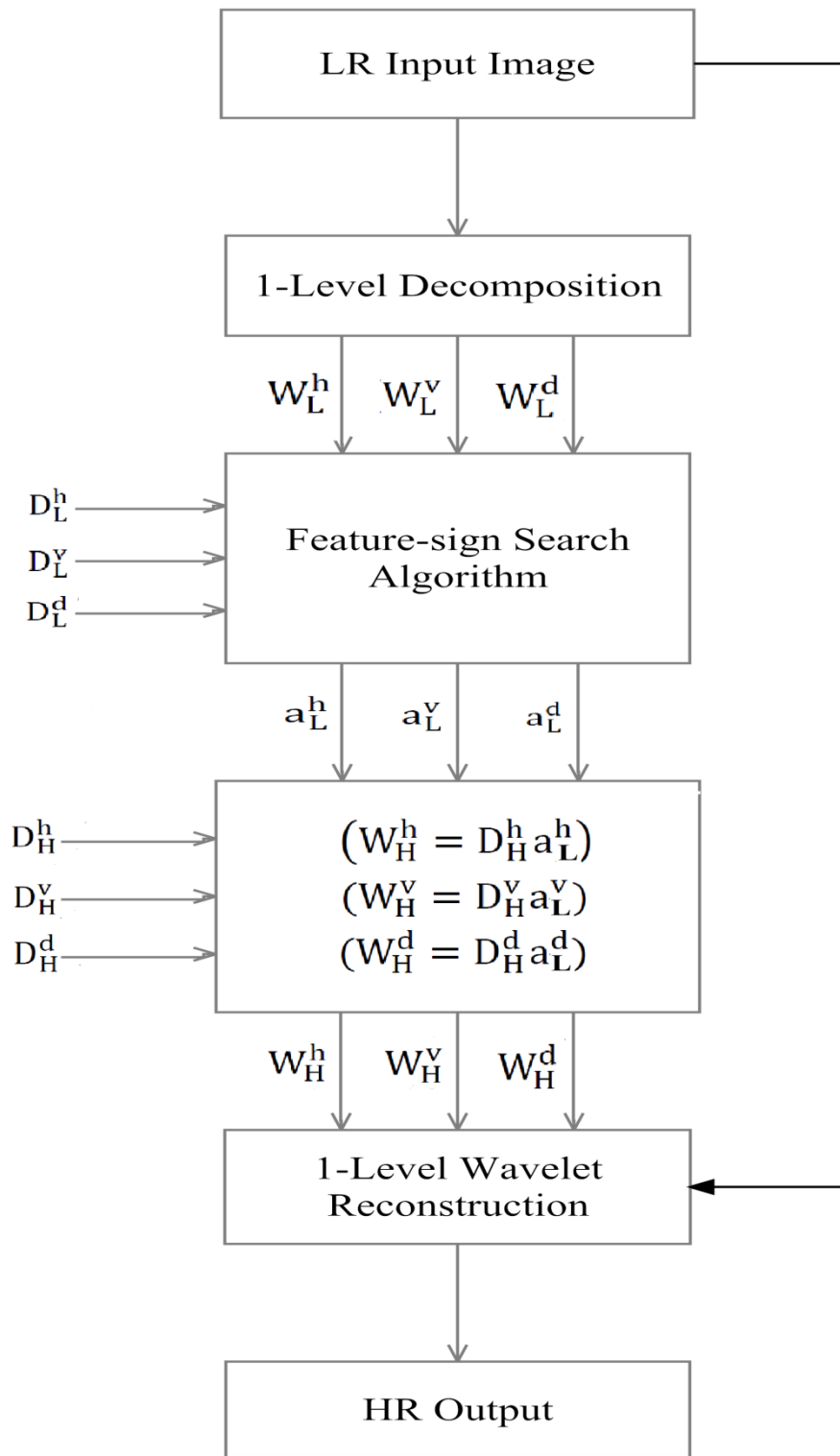


Figure 5.4: Proposed super-resolution approach

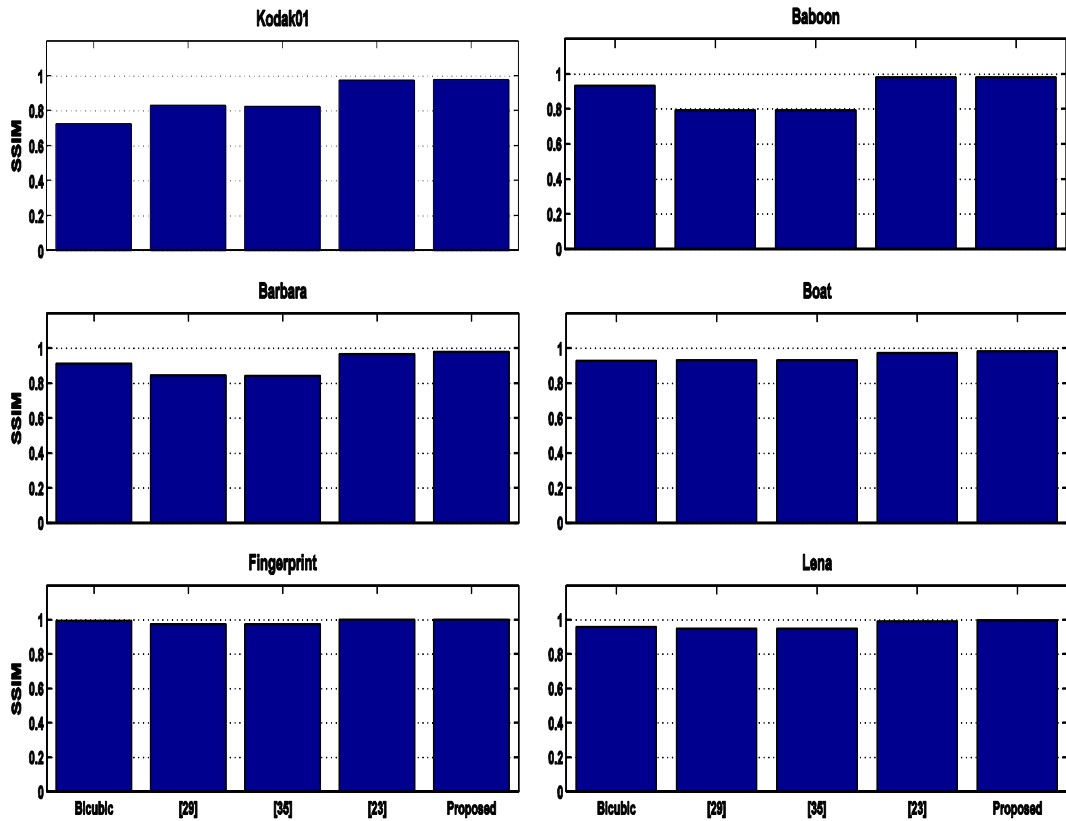


Figure 5.5: SSIM results comparison

Visual comparison is shown in Figure 5.6. Reconstructed images obtained with the proposed algorithm, [23], [29], [35] and Bicubic interpolation for the image number 1 in the Kodak set. The proposed algorithm shows improvement over [35] and [29], especially the continuity in diagonal patterns are well preserved.

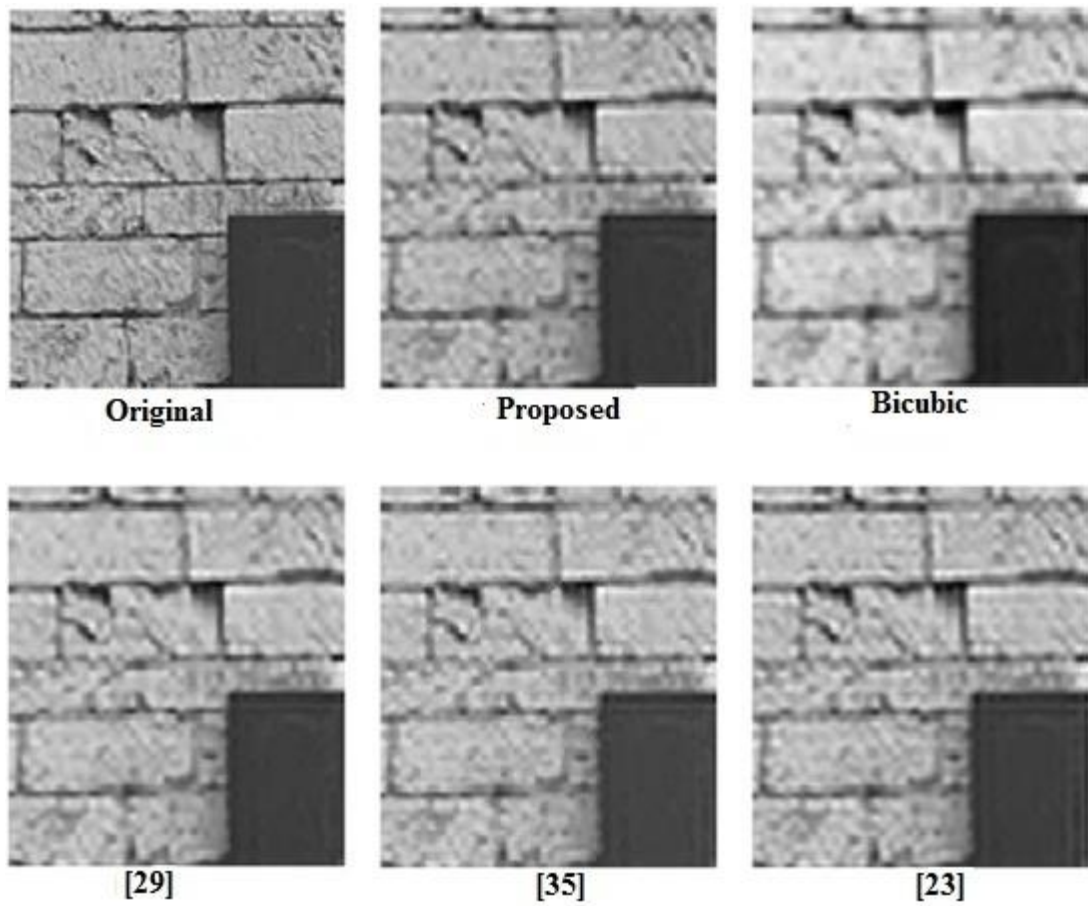


Figure 5.6: Visual comparison of the image number 1 in the Kodak set.

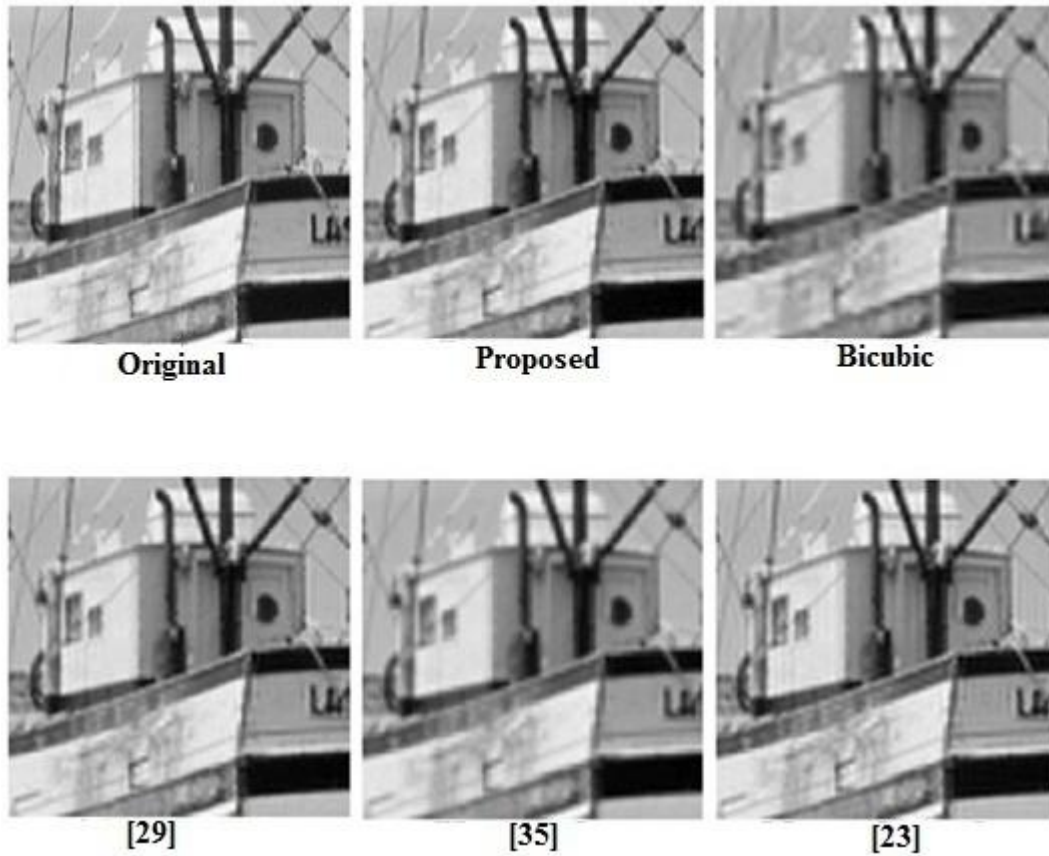


Figure 5.7: Visual comparison of the Boat image.

Figure 5.7, 5.8, 5.9 and 5.10 show the visual comparison of standard test images. It is evident that proposed algorithm perform significantly better than Bicubic, [29] and [35] in terms of visual perception. Also, in terms of visual comparison, the proposed algorithm is as good as [23].

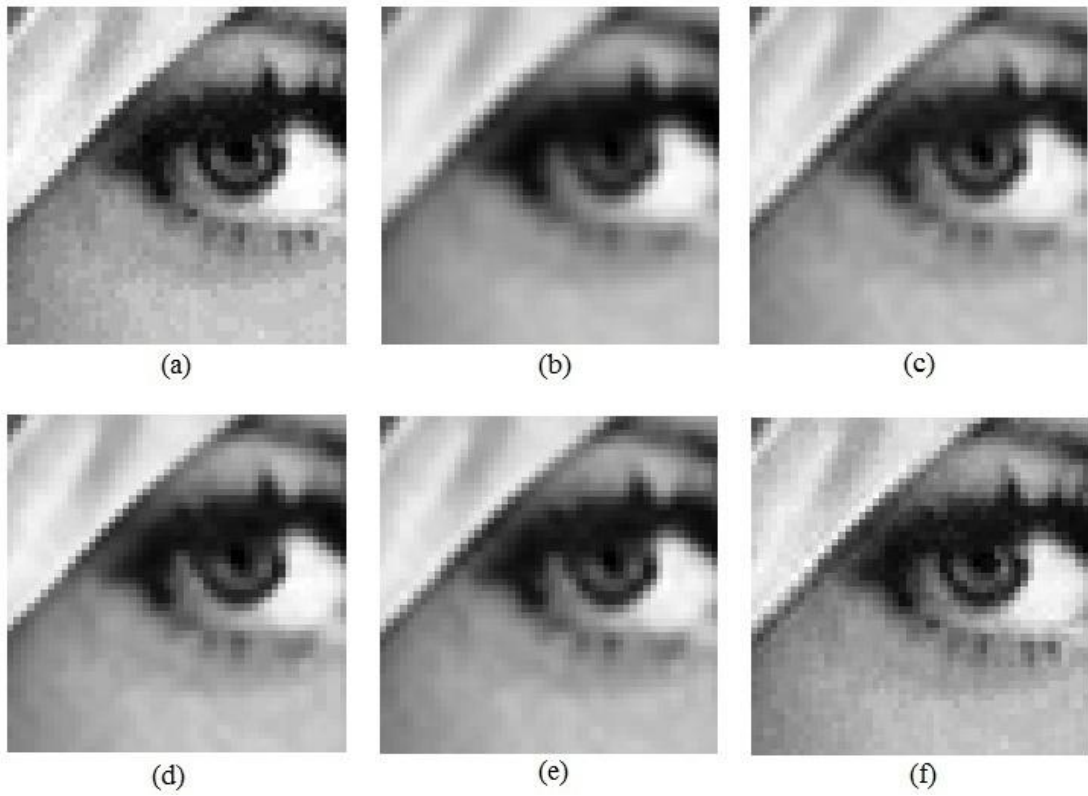


Figure 5.8: Visual comparison of the zoomed Lena image.
(a) Original Image, (b) Bicubic technique, (c) Algorithm of [23], (d) Algorithm of [29], (e) Algorithm of [35], (f) Proposed Algorithm.

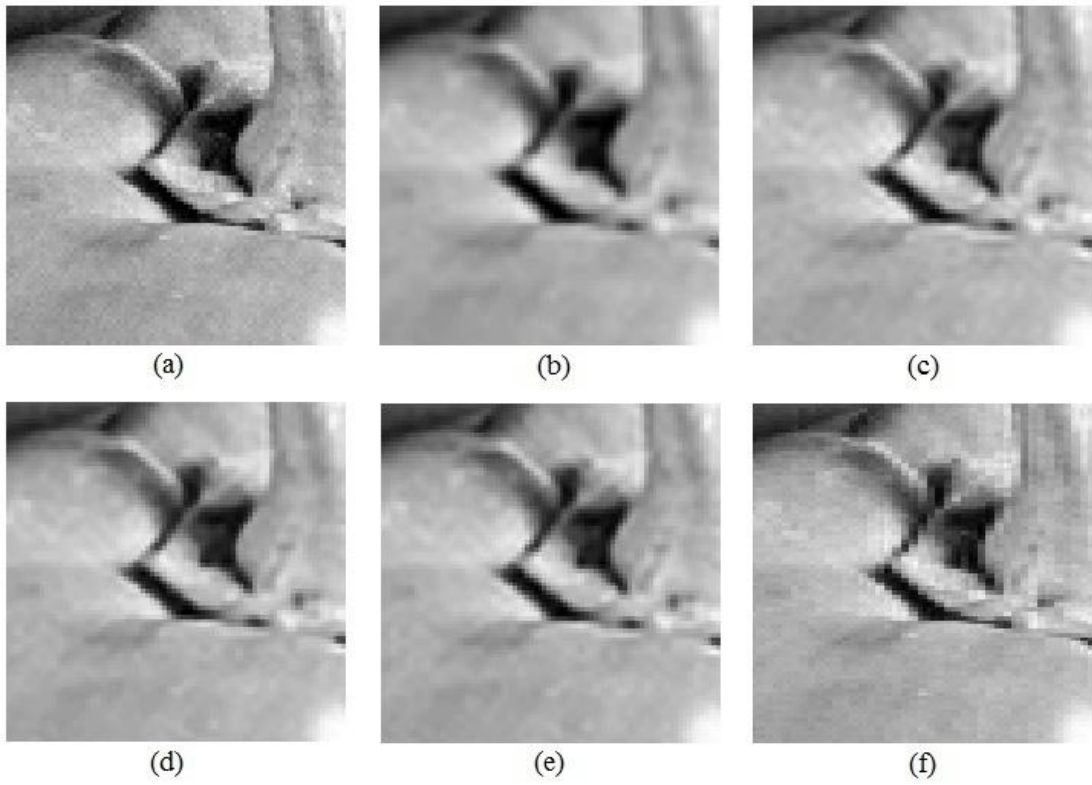


Figure 5.9: Visual comparison of the zoomed Peppers image.
(a) Original Image, (b) Bicubic technique, (c) Algorithm of [23], (d) Algorithm of [29], (e) Algorithm of [35], (f) Proposed Algorithm.

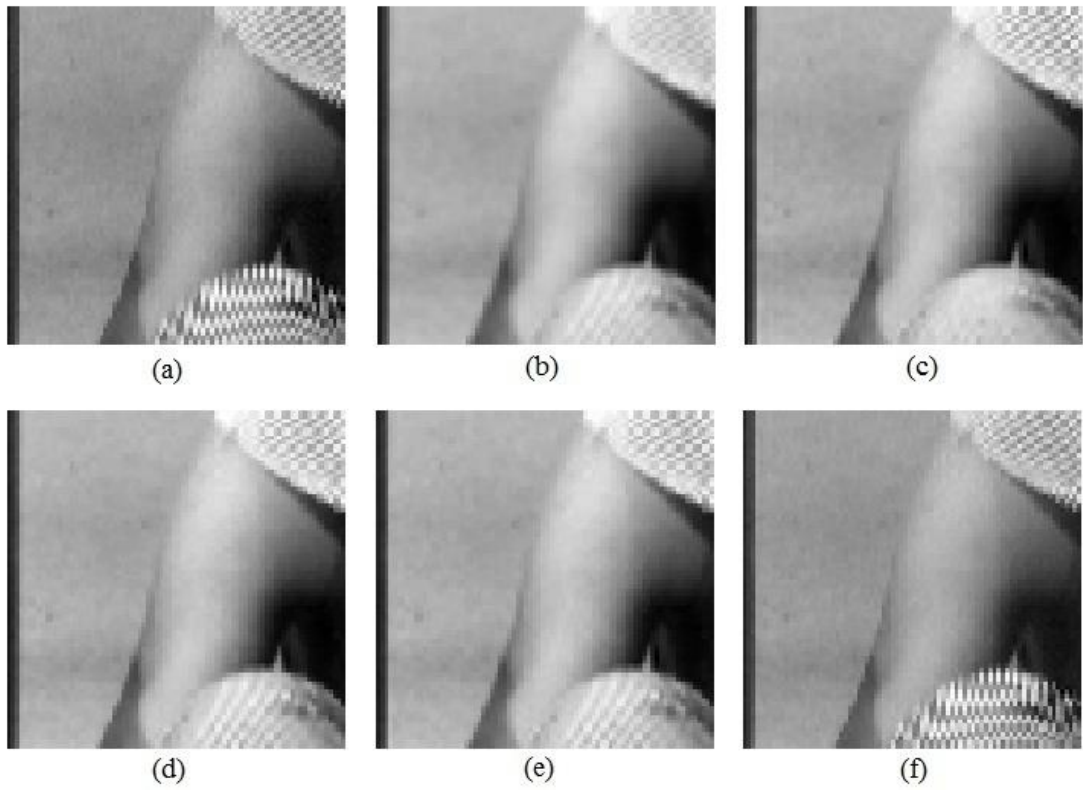


Figure 5.10: Visual comparison of the zoomed Barbara image.
 (a) Original Image, (b) Bicubic technique, (c) Algorithm of [23], (d) Algorithm of [29], (e) Algorithm of [35], (f) Proposed Algorithm.



Figure 5.11: Samples of images used in dictionary learning

5.6 Conclusion

In this chapter implementation of coupled dictionary learning in the wavelet domain for SISR is presented. Due to strong coupling between LR and HR sparse coefficients, the performance of SISR is improved. According to PSNR and SSIM results the proposed algorithm outperforms well known SISR algorithms.

Chapter 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

In summary, this study is conducted to present a novel contribution in the field of image processing. In this regard, a new approach to image denoising is developed. This method is developed based on findings about limitation of maximum orthogonal projection based sparse representation algorithm in image denoising. The proposed image denoising algorithm is based on sparse-land model and it provides new dimension to this line of research. In the proposed algorithm, information about residual correlation is incorporated in sparse coding stage. To best of my knowledge, this way of utilizing residual correlation regularization for sparse coding has not been used before. It is proved through mathematical modeling and simulation results that reducing residual correlation both internally and externally does indeed improve the performance of image denoising. In chapter 3, a new strategy for sparse representation verified our claim that minimizing residual correlation indeed improves image denoising performance especially when image is corrupted by high noise level. Another interesting finding is that the image with abundance of high frequency contents improves significantly by correlation based approach. In chapter 4, coupled K-SVD algorithm is implemented in wavelet domain and it is used for image super-resolution. A dictionary learning and super-resolution approach is developed based on wavelet domain. This study is carried out to understand the advantages of wavelet domain and also effectiveness of coupled K-SVD for image

super-resolution. Simulation results show significant improvement in terms of PSNR when compared to well known SISR algorithms. In Chapter 5, based on motivation presented for algorithm in chapter 3, residual correlation regularization is developed and employed in objective function of sparse representation and dictionary update stage. This regularization makes sure that residuals are as uncorrelated to each other as possible. Hence, residual possess statistical properties of contaminating noise. Results are compared with state-of-the-art image denoising algorithms. Comparison shows that it is highly competitive with state-of-the-art and often better especially at high noise levels or in terms of visual results obtained. In Chapter 6, a SISR algorithm is proposed in which super-resolution is attached by enforcing the invariance of the sparse representation at various scales. Data is divided into three clusters based on correlation between the patches and horizontal, vertical and non directional templates. Simulation results are compared with benchmark SISR algorithms.

6.2 Future Work

The performance of proposed algorithms presented in chapter 3 and chapter 4 can further be analyzed by implementing them for computer vision algorithms such as fingerprinting. Some of these future works are listed below.

6.2.1 Deep Learning

Learning through many layers of neural network is known as deep learning. Training of a model through deep learning has proved to be very successful in image denoising. The proposed algorithm can be pre-trained using stacked denoising auto-encoders (SDA). This training of image patches based on residual reduction information in deep network can lead to state-of-the-art image denoising results.

6.2.2 Application in Computer Vision Algorithms

The proposed algorithm proved to be successful in recovering fine structures even at very high noise levels especially for images like Fingerprint and Barbara images as shown in Chapter 3 and Chapter 4. This algorithm may prove to be helpful in one of the computer vision algorithms such as data fingerprinting. Such algorithms are used to identify the original data for practical use based on its fingerprint as human beings are identified through fingerprinting. The proposed algorithms can be tested for such applications. Also the proposed algorithm can also be effective if implemented in image classification where data to be classified is noisy. Also in case of pattern recognition the proposed algorithm can be tested.

6.2.3 Simplifying Complexity of Algorithm by Proximal Calculus

One of the complexities in the proposed objective function is to determine the correct sign by testing all of the possible signs which is a ad hoc solution of prospective function. This can be alleviated by applying one of the proximal calculus methods as given in [65]. Such methods are used to find the proximity operator of functions present optimization problem [65]. Also usage of subderivative or subdifferential can also prove to be useful in simplifying solution of the proposed optimization function. According to property of sudifferential; a global minimum of the convex function can be x_0 if subdifferential contains zero. Referring to subdifferetial for further simplifying the objective function and hence reducing computational complexity can be a good extension of this work.

6.2.4 Updating Multiple Dictionary Atoms

In dictionary update stage of the proposed algorithm, information of residual correlations played important role in achieving competitive performance. The process of updating one atom at one time is mostly used method. However, in

proposed algorithm, we not only consider current residual but also neighboring residuals, hence, it can be useful if dictionary atoms are also updated in groups with based on information about dictionary atoms used by neighboring residuals.

6.2.5 Group Sparsity

The proposed algorithm can be more effective if sparse coefficients are picked for group of neighboring patches or similar patches. Such sparse representation coefficient estimation method is commonly known group sparsity method. The performance of the proposed algorithm can further be if single sparse coefficient is picked for neighboring residuals. Hence, implementing the proposed algorithm based on group sparsity can be a good extension of this work.

6.2.6 Analyzing the Performance by Varying Patch Sizes

Performance of the proposed algorithm presented in Chapter 4 can also be tested for various patch sizes as shown in Table 3.2. According to results obtained, one can presume that the bigger patch sizes are useful at higher noise levels and smaller patch sizes can be effective for low noise levels for the proposed algorithm.

6.2.6 Residual Correlation Based Single Image Super Resolution (SISR)

The proposed algorithm can prove to be effective in improving performance of the SISR. This approach leads us to select an atom for current patch such that it becomes highly correlated with neighboring patches. Thus, the proposed algorithm can be tested for the inverse problem such as SISR.

REFERENCES

- [1] Elad, M., & Aharon, M. (2006). Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12), 3736-3745.

- [2] Aharon, M., Elad, M., & Bruckstein, A. (2006). K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing*, 54(11), 4311-4322.

- [3] Dabov, K., Foi, A., Katkovnik, V., & Egiazarian, K. (2006, January). Image denoising with block-matching and 3 D filtering. In *Proceedings of SPIE* (Vol. 6064, No. 30, pp. 606414-606414).

- [4] Yue, H., Sun, X., Yang, J., & Wu, F. (2015). Image denoising by exploring external and internal correlations. *IEEE Transactions on Image Processing*, 24(6), 1967-1982.

- [5] He, Y., Gan, T., Chen, W., & Wang, H. (2012). Multi-stage image denoising based on correlation coefficient matching and sparse dictionary pruning. *Signal Processing*, 92(1), 139-149.

- [6] Dabov, K., Foi, A., Katkovnik, V., & Egiazarian, K. (2007). Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8), 2080-2095.
- [7] Brunet, D., Vrscay, E. R., & Wang, Z. (2009, July). The use of residuals in image denoising. In *International Conference Image Analysis and Recognition* (pp. 1-12). Springer, Berlin, Heidelberg.
- [8] Om, H., & Biswas, M. (2015). A generalized image denoising method using neighbouring wavelet coefficients. *Signal, Image and Video Processing*, 9(1), 191-200.
- [9] Cai, N., Zhou, Y., Wang, S., Ling, B. W. K., & Weng, S. (2016). Image denoising via patch-based adaptive Gaussian mixture prior method. *Signal, Image and Video Processing*, 10(6), 993-999.
- [10] Shahdoosti, H. R., & Khayat, O. (2016). Image denoising using sparse representation classification and non-subsampled shearlet transform. *Signal, Image and Video Processing*, 10(6), 1081-1087
- [11] Sajjad, M., Mehmood, I., Abbas, N., & Baik, S. W. (2016). Basis pursuit denoising-based image superresolution using a redundant set of atoms. *Signal, Image and Video Processing*, 10(1), 181-188.

- [12] Skretting, K., & Engan, K. (2010). Recursive least squares dictionary learning algorithm. *IEEE Transactions on Signal Processing*, 58(4), 2121-2130.
- [13] Mairal, J., Bach, F., Ponce, J., & Sapiro, G. (2009, June). Online dictionary learning for sparse coding. In *Proceedings of the 26th annual international conference on machine learning* (pp. 689-696). ACM.
- [14] Zoran, D., & Weiss, Y. (2011, November). From learning models of natural image patches to whole image restoration. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (pp. 479-486). IEEE.
- [15] Franzen, R. (1999). Kodak lossless true color image suite. *source: <http://r0k.us/graphics/kodak>, 4.*
- [16] Bruckstein, A. M., Donoho, D. L., & Elad, M. (2009). From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM review*, 51(1), 34-81.
- [17] Dong, W., Zhang, L., Shi, G., & Wu, X. (2011). Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *IEEE Transactions on Image Processing*, 20(7), 1838-1857.
- [18] Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on information theory*, 52(4), 1289-1306.

- [19] Daubechies, I. (1992). Ten lectures on wavelets, vol. 61 of CBMS-NSF Regional Conference Series in Applied Mathematics.
- [20] Elad, M., & Yavneh, I. (2009). A plurality of sparse representations is better than the sparsest one alone. *IEEE Transactions on Information Theory*, 55(10), 4701-4714.
- [21] Farsiu, S., Robinson, M. D., Elad, M., & Milanfar, P. (2004). Fast and robust multiframe super resolution. *IEEE transactions on image processing*, 13(10), 1327-1344.
- [22] Mairal, J., Ponce, J., Sapiro, G., Zisserman, A., & Bach, F. R. (2009). Supervised dictionary learning. In *Advances in neural information processing systems* (pp. 1033-1040).
- [23] Nazzal, M., & Ozkaramanli, H. (2015). Wavelet domain dictionary learning-based single image superresolution. *Signal, Image and Video Processing*, 9(7), 1491-1501.
- [24] Nazzal, M., & Ozkaramanli, H. (2013, April). Improved single image super-resolution using sparsity and structured dictionary learning in wavelet domain. In *Signal Processing and Communications Applications Conference (SIU), 2013 21st* (pp. 1-4). IEEE.

- [25] Rauhut, H., Schnass, K., & Vandergheynst, P. (2008). Compressed sensing and redundant dictionaries. *IEEE Transactions on Information Theory*, 54(5), 2210-2219.
- [26] Tipping, M. E., & Bishop, C. M. (2003). Bayesian image super-resolution. In *Advances in neural information processing systems* (pp. 1303-1310).
- [27] Wang, S., Zhang, L., Liang, Y., & Pan, Q. (2012, June). Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (pp. 2216-2223).
- [28] Yang, J., Wright, J., Huang, T. S., & Ma, Y. (2010). Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11), 2861-2873.
- [29] Yang, J., Wang, Z., Lin, Z., Cohen, S., & Huang, T. (2012). Coupled dictionary training for image super-resolution. *IEEE transactions on image processing*, 21(8), 3467-3478.
- [30] Zeyde, R., Elad, M., & Protter, M. (2010, June). On single image scale-up using sparse-representations. In *International conference on curves and surfaces* (pp. 711-730). Springer, Berlin, Heidelberg.

- [31] Zhang, K., Gao, X., Tao, D., & Li, X. (2012, June). Multi-scale dictionary for single image super-resolution. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (pp. 1114-1121).IEEE.
- [32] Hardie, R. C., Barnard, K. J., & Armstrong, E. E. (1997). Joint MAP registration and high-resolution image estimation using a sequence of undersampled images. *IEEE Transactions on Image Processing*, 6(12), 1621-1633.
- [33] Jia, K., Wang, X., & Tang, X. (2013). Image transformation based on learning dictionaries across image spaces. *IEEE transactions on pattern analysis and machine intelligence*, 35(2), 367-380.
- [34] Ophir, B., Lustig, M., & Elad, M. (2011). Multi-scale dictionary learning using wavelets. *IEEE Journal of Selected Topics in Signal Processing*, 5(5), 1014-1024.
- [35] Xu, J., Qi, C., & Chang, Z. (2014, October). Coupled K-SVD dictionary training for super-resolution. In *Image Processing (ICIP), 2014 IEEE International Conference on* (pp. 3910-3914).
- [36] Baloch, G., & Ozkaramanli, H. (2017). Image denoising via correlation-based sparse representation. *Signal, Image and Video Processing*, 1-8.
- [37] Eksioğlu, E. M., & Bayir, O. (2014). K-SVD meets transform learning: transform K-SVD. *IEEE signal processing letters*, 21(3), 347-351.

- [38] Riot, P., Almansa, A., Gousseau, Y., & Tupin, F. (2016, August). Penalizing local correlations in the residual improves image denoising performance. In *Signal Processing Conference (EUSIPCO), 2016 24th European* (pp. 1867-1871). IEEE.
- [39] Dong, W., Zhang, L., Shi, G., & Li, X. (2013). Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing*, 22(4), 1620-1630.
- [40] Elad, M., & Aharon, M. (2006, June). Image denoising via learned dictionaries and sparse representation. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on* (Vol. 1, pp. 895-900). IEEE.
- [41] Engan, K., Aase, S. O., & Husoy, J. H. (1999). Method of optimal directions for frame design. In *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on* (Vol. 5, pp. 2443-2446).
- [42] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4), 600-612.
- [43] Mallat, S. G., & Zhang, Z. (1993). Matching pursuits with time-frequency dictionaries. *Signal Processing, IEEE Transactions on*, 41(12), 3397-3415.

- [44] Pati, Y. C., Rezaifar, R., & Krishnaprasad, P. S. (1993). Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. *In Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on* (pp. 40-44). IEEE.
- [45] Chen, S. S., Donoho, D. L., & Saunders, M. A. (1998). Atomic decomposition by basis pursuit. *SIAM journal on scientific computing*, 20(1), 33-61.
- [46] Gorodnitsky, I. F., & Rao, B. D. (1997). Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm. *Signal Processing, IEEE Transactions on*, 45(3), 600-616.
- [47] Cotter, S. F., Rao, B. D., Kreutz-Delgado, K., & Adler, J. (1999). Forward sequential algorithms for best basis selection. *IEE Proceedings-Vision, Image and Signal Processing*, 146(5), 235-244.
- [48] Skretting, K., & Husoy, J. H. (2008). Partial search vector selection for sparse signal representation.
- [49] Engan, K., Aase, S. O., & HakonHusoy, J. (1999). Method of optimal directions for frame design. In *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on* (Vol. 5, pp. 2443-2446). IEEE.
- [50] Davis, G. (1994). Adaptive nonlinear approximations (*Doctoral dissertation, Courant Institute of Mathematical Sciences New York*).

- [51] Engan, K., Aase, S. O., & Husoy, J. H. (1998). Designing frames for matching pursuit algorithms. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on* (Vol. 3, pp. 1817-1820). IEEE.
- [52] Gharavi-Alkhansari, M., & Huang, T. S. (1998). A fast orthogonal matching pursuit algorithm. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on* (Vol. 3, pp. 1389-1392). IEEE.
- [53] Engan, K., Rao, B. D., & Kreutz-Delgado, K. (2000). Regularized FOCUSS for subset selection in noise. In *Proc. Nordic Signal Processi. Symp* (pp. 247-250).
- [54] Aharon, M., Elad, M., & Bruckstein, A. M. K-SVD (2005): An algorithm for designing of overcomplete dictionaries for sparse representation Technion—*Israel Inst. of Technology*. Tech. Ref.
- [55] Tropp, J. (2004). Greed is good: Algorithmic results for sparse approximation. *Information Theory, IEEE Transactions on*, 50(10), 2231-2242.
- [56] Mairal, J., Bach, F., Ponce, J., & Sapiro, G. (2010). Online learning for matrix factorization and sparse coding. *The Journal of Machine Learning Research*, 11, 19-60.

- [57] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 267-288.
- [58] Lee, H., Battle, A., Raina, R., & Ng, A. Y. (2007). Efficient sparse coding algorithms. In *Advances in neural information processing systems* (pp. 801-808).
- [59] Farhadifard, F., Abar, E., Nazzal, M., & Ozkaraman, H. (2014, April). Single image super resolution based on sparse representation via directionally structured dictionaries. In *Signal Processing and Communications Applications Conference (SIU), 2014 22nd* (pp. 1718-1721). IEEE.
- [60] Franzen, R. (2013). Kodak lossless true color image suite. *source: <http://r0k.us/graphics/kodak>, 4.*
- [61] Klette, R. (2014). *Concise computer vision*. Springer, London.
- [62] Liu, D., & Klette, R. (2015, November). Sharpness and contrast measures on videos. In *Image and Vision Computing New Zealand (IVCNZ), 2015 International Conference on* (pp. 1-6). IEEE.
- [63] Zhang, L., Zhang, L., Mou, X., & Zhang, D. (2011). FSIM: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8), 2378-2386.

- [64] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4), 600-612.
- [65] Combettes, P. L., & Müller, C. L. (2016). Perspective functions: Proximal calculus and applications in high-dimensional statistics. *Journal of Mathematical Analysis and Applications*.
- [66] Boyat, A. K., & Joshi, B. K. (2015). A review paper: noise models in digital image processing. *An International Journal (SIPIJ)*, Vol.6, No.2, 63-73.
- [67] Aster, R. C., Borchers, B., & Thurber, C. H. (2011). *Parameter estimation and inverse problems* (Vol. 90). Academic Press.
- [68] Haghigat, M. B. A., Aghagolzadeh, A., & Seyedarabi, H. (2011). A non-reference image fusion metric based on mutual information of image features. *Computers & Electrical Engineering*, 37(5), 744-756.
- [69] Belouchrani, A., Abed-Meraim, K., Cardoso, J. F., & Moulines, E. (1997). A blind source separation technique using second-order statistics. *IEEE Transactions on signal processing*, 45(2), 434-444.

- [70] Yue, L., Shen, H., Li, J., Yuan, Q., Zhang, H., & Zhang, L. (2016). Image super-resolution: The techniques, applications, and future. *Signal Processing*, 128, 389-408.

APPENDICES

Appendix A: Derivation of expanded equation (4.6)

$$J_c(k_s, \alpha_s)$$

$$= \frac{1}{2} \left(\alpha_s^2 \mathbf{d}_{k_s}^T \mathbf{d}_{k_s} - 2\alpha_s (\mathbf{d}_{k_s})^T \mathbf{r}_{s-1} + \mathbf{r}_{s-1}^T \mathbf{r}_{s-1} \right) + \sum_{m=1}^M \lambda_m |(\mathbf{r}_{s-1} - \mathbf{d}_{k_s} \alpha_s)^T \mathbf{r}^m|$$

Solution:

Let $f = J_c(k_s, \alpha_s)$, then finding derivative (w.r.t.) α_s gives:

$$\frac{\partial f}{\partial \alpha_s} = \frac{1}{2} \left(2\alpha_s \mathbf{d}_{k_s}^T \mathbf{d}_{k_s} - 2(\mathbf{d}_{k_s})^T \mathbf{r}_{s-1} + 0 \right) + \left(0 - \sum_{m=1}^M \lambda_m \mathbf{r}^m \mathbf{d}_{k_s}^T \right)$$

$$\frac{\partial f}{\partial \alpha_s} = -\mathbf{d}_{k_s}^T (\mathbf{r}_{s-1} - \alpha_s \mathbf{d}_{k_s}) - \sum_{m=1}^M |s_m \lambda_m \mathbf{r}^m \mathbf{d}_{k_s}^T|$$

Letting derivative be equal to zero,

$$-\mathbf{d}_{k_s}^T (\mathbf{r}_{s-1} - \alpha_s \mathbf{d}_{k_s}) - \sum_{m=1}^M |s_m \lambda_m \mathbf{r}^m \mathbf{d}_{k_s}^T| = 0$$

$$-\mathbf{d}_{k_s}^T \mathbf{r}_{s-1} + \alpha_s \mathbf{d}_{k_s}^T \mathbf{d}_{k_s} - \sum_{m=1}^M |s_m \lambda_m \mathbf{r}^m \mathbf{d}_{k_s}^T| = 0$$

$$-\mathbf{d}_{k_s}^T \mathbf{r}_{s-1} + \alpha_s - \sum_{m=1}^M |s_m \lambda_m \mathbf{r}^m \mathbf{d}_{k_s}^T| = 0$$

$$\alpha_s = \mathbf{d}_{k_s}^T \mathbf{r}_{s-1} + \sum_{m=1}^M s_m \lambda_m \mathbf{d}_{k_s}^T \mathbf{r}^m$$

Appendix B: Simplifying equation (4.10)

$$\begin{aligned}
 J_d(D) &= \frac{1}{t} \left[\sum_{i=1}^t \|r_i\|_2^2 + \sum_{i=1}^{t-1} \lambda_i |r_t^T r_i| \right] \\
 &= \frac{1}{t} \left[\sum_{i=1}^t \|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \sum_{i=1}^{t-1} \lambda_i |(\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}_t)^T (\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i)| \right]
 \end{aligned}$$

Solution:

Let $f_a = \sum_{i=1}^t \|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2$ and $f_b = \sum_{i=1}^{t-1} \lambda_i |(\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}_t)^T (\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i)|$

Firstly, we simplify f_a ; (using Polarization property)

$$\begin{aligned}
 f_a &= \frac{1}{2} \sum_{i=1}^t [(\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i)^T (\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i)] \\
 f_a &= \frac{1}{2} \sum_{i=1}^t (\mathbf{x}_i^T \mathbf{x}_i - \mathbf{D} \mathbf{x}_i^T \boldsymbol{\alpha}_i - \mathbf{D} \boldsymbol{\alpha}_i^T \mathbf{x}_i + \mathbf{D} \mathbf{D}^T \boldsymbol{\alpha}_i^T \boldsymbol{\alpha}_i)
 \end{aligned}$$

Let

$$\mathbf{A}_t = \sum_{i=1}^t \boldsymbol{\alpha}_i \boldsymbol{\alpha}_i^T = \mathbf{A}_{t-1} + \boldsymbol{\alpha}_t \boldsymbol{\alpha}_t^T$$

$$\mathbf{B}_t = \sum_{i=1}^t \mathbf{x}_i \boldsymbol{\alpha}_i^T = \mathbf{B}_{t-1} + \mathbf{x}_t \boldsymbol{\alpha}_t^T$$

since $\text{Tr}(\mathbf{A}) = \text{Tr}(\mathbf{A}^T)$; Here Tr is a trace operator

$$\text{then } \text{Tr}(\mathbf{D}\mathbf{x}_i^T \boldsymbol{\alpha}_i) = \text{Tr}(\mathbf{D}^T \mathbf{x}_i \boldsymbol{\alpha}_i^T)$$

Trace comes from Euclidean inner product on the real space of real

matrices i.e., $\|\mathbf{AB}\|_2 = \langle \mathbf{A}, \mathbf{B} \rangle = \text{Tr}(\mathbf{A}^T \mathbf{B})$

Hence, in terms of trace of matrix,

$$f_a = \frac{1}{2} \left(\text{Tr}(\mathbf{x}_t^T \mathbf{x}_t - \mathbf{D}^T \mathbf{B}_t - \mathbf{D}^T \mathbf{B}_t + \mathbf{D} \mathbf{D}^T \mathbf{A}_t) \right)$$

$$f_a = \frac{1}{2} \left(\text{Tr}(\mathbf{x}_t^T \mathbf{x}_t - 2\mathbf{D}^T \mathbf{B}_t + \mathbf{D} \mathbf{D}^T \mathbf{A}_t) \right)$$

Now we simplify $f_b = \sum_{i=1}^{t-1} \lambda_i |(\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}_t)^T (\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i)|$

$$f_b = \left| (\mathbf{x}_t - \mathbf{D}\boldsymbol{\alpha}_t)^T \sum_{i=1}^{t-1} \lambda_i (\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i) \right|$$

$$f_b = \left| \mathbf{x}_t^T \sum_{i=1}^{t-1} \lambda_i \mathbf{x}_i - \boldsymbol{\alpha}_t^T \mathbf{D}^T \sum_{i=1}^{t-1} \lambda_i \mathbf{x}_i + \boldsymbol{\alpha}_t^T \mathbf{D}^T \mathbf{D} \sum_{i=1}^{t-1} \lambda_i \boldsymbol{\alpha}_i - \mathbf{x}_t^T \mathbf{D} \sum_{i=1}^{t-1} \lambda_i \boldsymbol{\alpha}_i \right|$$

$$\text{Let } \mathbf{g}_{t-1} = \sum_{i=1}^{t-1} \lambda_i \mathbf{x}_i \text{ and } \mathbf{f}_{t-1} = \sum_{i=1}^{t-1} \lambda_i \boldsymbol{\alpha}_i$$

$$f_b = \left(\text{Tr}(\mathbf{x}_t^T \mathbf{g}_{t-1} - \boldsymbol{\alpha}_{t-1}^T \mathbf{D}^T \mathbf{g}_{t-1} - \mathbf{x}_t^T \mathbf{D} \mathbf{f}_{t-1} + \boldsymbol{\alpha}_{t-1}^T \mathbf{D}^T \mathbf{D} \mathbf{f}_{t-1}) \right)$$

Now $f = f_a + f_b$

$$\begin{aligned} f &= \frac{1}{2} \left(\text{Tr}(\mathbf{x}_t^T \mathbf{x}_t - 2\mathbf{D}^T \mathbf{B}_t + \mathbf{D} \mathbf{D}^T \mathbf{A}_t) \right) \\ &+ \left(\text{Tr}(\mathbf{x}_t^T \mathbf{g}_{t-1} - \boldsymbol{\alpha}_{t-1}^T \mathbf{D}^T \mathbf{g}_{t-1} - \mathbf{x}_t^T \mathbf{D} \mathbf{f}_{t-1} \right. \\ &\left. + \boldsymbol{\alpha}_{t-1}^T \mathbf{D}^T \mathbf{D} \mathbf{f}_{t-1}) \right) \end{aligned}$$

**Appendix C: Derivation of equation (4.17) to obtain
equation (4.21)**

$$f = \frac{1}{2} \left(\text{Tr}(\mathbf{x}_t^T \mathbf{x}_t - 2\mathbf{D}^T \mathbf{B}_t + \mathbf{D} \mathbf{D}^T \mathbf{A}_t) \right) \left(\text{Tr}(\mathbf{x}_t^T \mathbf{g}_{t-1} - \boldsymbol{\alpha}_{t-1}^T \mathbf{D}^T \mathbf{g}_{t-1} - \mathbf{x}_t^T \mathbf{D} \mathbf{f}_{t-1} + \boldsymbol{\alpha}_{t-1}^T \mathbf{D}^T \mathbf{D} \mathbf{f}_{t-1}) \right)$$

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{D}} &= \frac{1}{2} (2\mathbf{D} \mathbf{A}_t - 2\mathbf{B}_t + 0) \\ &\quad + \sigma_\lambda (2\mathbf{D} \mathbf{f}_{t-1} \boldsymbol{\alpha}_{t-1}^T - \mathbf{g}_{t-1} \boldsymbol{\alpha}_{t-1}^T - \mathbf{f}_{t-1} \mathbf{x}_t^T + 0) \end{aligned}$$

Note that σ_λ is signum function.

$$\frac{\partial f}{\partial \mathbf{D}} = \mathbf{D} (\mathbf{A}_t + 2\sigma_\lambda \mathbf{f}_{t-1} \boldsymbol{\alpha}_{t-1}^T) - [\mathbf{B}_t + \sigma_\lambda (\mathbf{g}_{t-1} \boldsymbol{\alpha}_{t-1}^T + \mathbf{f}_{t-1} \mathbf{x}_t^T)]$$

Let $\mathbf{A}_C = \mathbf{A}_t + 2\sigma_\lambda \mathbf{f}_{t-1} \boldsymbol{\alpha}_{t-1}^T$ and $\mathbf{B}_C = \mathbf{B}_t + \sigma_\lambda (\mathbf{g}_{t-1} \boldsymbol{\alpha}_{t-1}^T + \mathbf{f}_{t-1} \mathbf{x}_t^T)$

Hence;

$$\frac{\partial f}{\partial \mathbf{D}} = \mathbf{D} \mathbf{A}_C - \mathbf{B}_C$$

Let derivative be equal to zero, we have

$$\mathbf{D} \mathbf{A}_C = \mathbf{B}_C$$

The approximate solution of above equation can be obtained by:

$$\mathbf{D} = \mathbf{B}_C \mathbf{A}_C^+ + \mathbf{D}_{t-1} (\mathbf{I} - \mathbf{A}_C \mathbf{A}_C^+)$$

$$\mathbf{D} = \mathbf{B}_C \mathbf{A}_C^+ + \mathbf{D}_{t-1} - \mathbf{D}_{t-1} \mathbf{A}_C \mathbf{A}_C^+$$

$$\mathbf{D} = \mathbf{D}_{t-1} - \mathbf{A}_C^+ (\mathbf{B}_C - \mathbf{D}_{t-1} \mathbf{A}_C)$$