BAYESIAN MULTI FRAME SUPER RESOLUTION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

EMRE TURGAY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OFf DOCTOR OF PHILOSOPHY
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

JANUARY 2014

Approval of the thesis:

BAYESIAN MULTI FRAME SUPER RESOLUTION

submitted by **EMRE TURGAY** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences** ________________

Prof. Dr. Gönül Turan Sayan
Head of Department, **Electrical and Electronics Engineering** ________________

Prof. Dr. Gözde Bozdağı Akar
Supervisor, **Electrical and Electronics Eng. Dept., METU** ________________

Assoc. Prof. Dr. Nail Akar
Co-supervisor, **Electrical and Electronics Eng. Dept., Bilkent Univ.** ________________

Examining Committee Members:

Prof. Dr. Aydın Alatan
Electrical and Electronics Engineering, METU ________________

Prof. Dr. Gözde Bozdağı Akar
Electrical and Electronics Engineering, METU ________________

Prof. Dr. Mübeccel Demirekler
Electrical and Electronics Engineering, METU ________________

Prof. Dr. Orhan Arıkan
Electrical and Electronics Engineering, Bilkent University ________________

Assistant Prof. Alptekin Temizel
Graduate School of Informatics, METU ________________

Date: ________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: EMRE TURGAY

Signature :

ABSTRACT

BAYESIAN MULTI FRAME SUPER RESOLUTION

Turgay, Emre

Ph. D., Department of Electrical and Electronics Engineering

Supervisor : Prof. Dr. Gözde Bozdağı Akar

Co-Supervisor : Assoc. Prof. Dr. Nail Akar

January 2014, 155 pages

This thesis aims at increasing the effective resolution of an image using a set of low resolution images. This process is referred to as super resolution (SR) image reconstruction in the literature. This work proposes maximum a-posteriori (MAP) based iterative reconstruction methods for this problem.

The first contribution of the thesis is a novel edge preserving SR image reconstruction method. The proposed MAP based estimator uses local gradient direction and amplitude for optimal noise reduction while preserving edges.

The second contribution of the thesis is a novel texture prior for maximum a posteriori (MAP) based super resolution (SR) image reconstruction. The prior is based on a multiscale compound Markov Random Field (MRF) model. Gabor filters are utilized for subband decomposition. Each subband is modeled by a compound MRF that inherits a binary texture process. The texture process at each pixel location at each subband is estimated iteratively along with the unknown high-resolution image pixels.

Finally, a two stage SR method comprising a Bayesian reconstruction step followed by a restoration step is proposed. In the first stage, two MAP based SR estimators with different regularizations are employed. In the second stage, pixel-to-pixel difference between these two estimates is post-processed to restore edges and textures while eliminating noise.

Experiments on synthetically generated images and real experiments on visual CCD cameras and thermal cameras demonstrate that the proposed methods are more favorable compared to state-of-the-art SR methods especially on textures and edges.

Keywords: Bayesian, Super Resolution, Texture, Edge, Reconstruction

ÖZ

BAYES TABANLI ÇOKLU ÇERÇEVELİ SUPER ÇÖZÜNÜRLÜK

Turgay, Emre

Doktora, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi : Prof. Dr. Gözde Bozdağı Akar

Ortak Tez Yöneticisi : Doç. Dr. Nail Akar

Ocak 2014, 155 sayfa

Bu tez, düşük çözünürlüklü görüntüler kullanarak bir görüntünün etkin çözünürlüğünü artırmayı hedeflemektedir. Bu işlem, literatürde süper çözünürlükte (SR) görüntü oluşturma olarak adlandırılır. Bu çalışma bu problemin çözümü için iteratif maksimum sonsal (MAP) yöntemler önermektedir.

İlk olarak, görüntüdeki kenar yapılarını koruyan yeni bir SR görüntü geri çatılama yöntemi sunulmuştur. Önerilen yöntem görüntüdeki yerel değişimlerin yönünü ve genliğini kullanarak kenar yapılarına zarar vermeden optimum gürültü giderme yapmaktadır.

Tezin literatüre ikinci katkısı, MAP tabanlı süper çözünürlükte kullanılmak üzere geliştirilmiş yeni bir desen görüntü öncüsüdür. Önerilen görüntü öncüsü çok katmanlı Markov rastgele alanlarını temel almaktadır. Katman ayrıştırılması amacıyla Gabor süzgeçleri kullanılmıştır. Her alt katman içerisinde bir desen rastgele süreci bulunur. Desen süreci, yüksek çözünürlük görüntü pikselleri ile birlikte döngüsel olarak bulunur.

Son olarak, MAP tabanlı bir geri çatılama yöntemi ve akabinde bir restorasyon adımını içeren iki aşamalı bir yöntem önerilmiştir. İlk aşamada, farklı seviyede düzenlileyici içeren iki MAP SR yöntem koşturulmuştur. İkinci aşamada, bu iki tahmin arasındaki piksel farkını içeren görüntü restore edilerek kenar ve desenler korunmuştur.

Sentetik görüntüler, gerçek kamera ve termal kamera görüntüleri üzerinde yapılan deneyler, önerilen yöntemlerin özellikle kenar ve desen içeren görüntülerde litertürdeki eş yöntemlere kıyasla daha iyi sonuç verdiğini göstermektedir.

Anahtar Kelimeler: Bayesçi, Süper Çözünürlük, Desen, Kenar, Geri çatılama

To the woman in red dress...

ACKNOWLEDGMENTS

I would like to express my deep appreciation and gratitude to my advisor Dr. Gözde B. Akar for her guidance and encouragement throughout this work.

I would like to thank Dr. Mübeccel Demireker and Dr. Nail Akar for their valuable comments.

It is my duty to thank to ASELSAN Inc. for the thermal image data set used in some of my experiments.

It should also be noted that the budget for the conferences and all other expenses for this thesis is covered just by myself.

Finally, I'd be remiss if I did not acknowledge the friendly environment provided by Starbucks Cafe and its staff. Most of the thesis work has been completed in Starbucks in Beşevler Ankara.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

GLOSSARY

SR	Super resolution
HR	High resolution
LR	Low resolution
ML	Maximum likelihood
MAP	Maximum a posteriori
PDF	Probability Density Function
CS	Compressive sensing
MSE	Mean square error
SNR	Signal to noise ratio
PSNR	Peak signal to noise ratio
MRF	Markov random field
GMRF	Gaussian Markov random field
HMRF	Huber Markov random field
SSIM	Structural similarity index
POCS	Projection onto convex set
DFT	Discrete Fourier transform
TV	Total variation
BTV	Bilateral total variation
RLS	Recursive least squares
LMS	Least mean squares

CHAPTER 1

INTRODUCTION

Image resolution enhancement is a critical need in several fields. Increasing the number of detector pixels in imaging devices can not always satisfy this demand due to the limitations in optics, manufacturing, and cost. Super resolution (SR) image reconstruction provides software solutions for this need. SR image reconstruction refers to methods that use a set of low resolution (LR) images to construct a higher resolution image of the same scene. Since the first formulation of the problem in the 1980s [1], it has found many application areas in both civil and military applications such as: surveillance [2], remote sensing [3, 4], plate reading, face recognition [5], target detection, identification, medical imaging (MRI [6, 7], PET [8]), video standard conversion, e.g. from NTSC video signal to HDTV signal.

Higher resolution does not always mean increasing only the number of pixels in an image. SR image reconstruction aims at retrieving the lost high frequency components, that is to say *details* in the scene. This can only be achieved if the input LR images contain non-redundant information, in other words, a different *look* at the same scene. This can be achieved by either changing the camera focus [9, 10, 11], changing the zoom [12], the lightening conditions or relative motion. By relative motion, we refer to using multiple cameras with slightly different angles or a single camera under relative motion. For example, in military applications a surveillance camera on a moving platform is under enough vibration so that successive images provide slightly different information [13]. Interestingly, the same type of image enhancement algorithm is also present in human eye. Assume we look at a far stationary target, which is barely visible. If this target moves, suddenly the edges and the silhouette of the target becomes apparent. Although the mechanism in the retina and human brain is unknown, the similarity of the need for motion is worth noting.

The most critical stage in SR image reconstruction is the estimation of relative motion among LR images. A survey of image registration algorithms have been given in [14]. Registered LR pixels are processed to reconstruct an SR image. Various methods have been proposed for SR image reconstruction in the last three decades. A survey of SR methods is presented later in Chapter 2.

In this chapter, we describe an efficient model for SR image reconstruction. This model will be used throughout the thesis for constructing the SR algorithms. Although theoretically image fusion to reconstruct an SR image using multiple cameras and using a single camera under motion are the same problems, we will consider single camera image reconstruction for the ease of the presentation. Furthermore the following assumptions have been made:

- Each image has undergone the same optical blur (camera focus is not altered during image acquisition).
- Optical blur is linear and spatially invariant. The proposed methods can be extended for a linear and spatially variant case.
- Lightening conditions are consistent.
- Each LR image contains non-redundant information (e. g., sub-pixel motion is present among LR images and LR images are aliased.).
- Only affine transformation among LR images are considered. Perspective motion or moving object within the background image are not considered for simplifying the discussion.

1.1 FORWARD PROBLEM: IMAGE FORMATION PROCESS

Figure 1.1 illustrates the image formation process for a typical CCD camera [15, 16]. Scene irradiance goes through an optical system, where a blurred version of the scene is projected onto an array of finite size detector elements. The detector elements produce a discrete image, whose pixel intensity values are proportional to the amount of flux projected on to each detector element. During this process, the pixel intensity values are contaminated by noise from several sources. This mechanism is simplified in the block diagram in Figure 1.1. Here, I is image irradiance, H_{Optics} term models optic effects (lens blur etc.), and $H_{DET,LR}$ represents detector pixel physics (pixel size, shape, sensitivity etc.). Finally, additive noise is represented by $\mathbf{n}$.

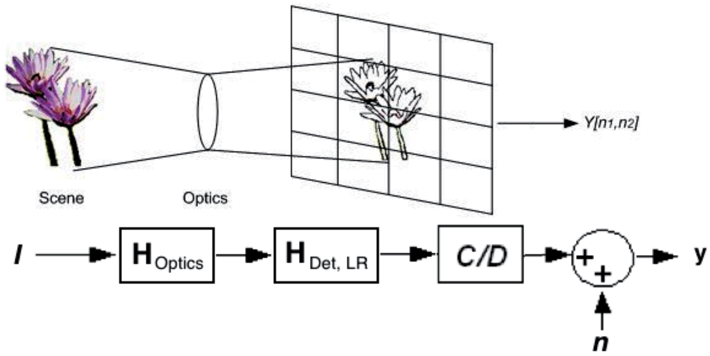


Figure 1.1: Image formation process.

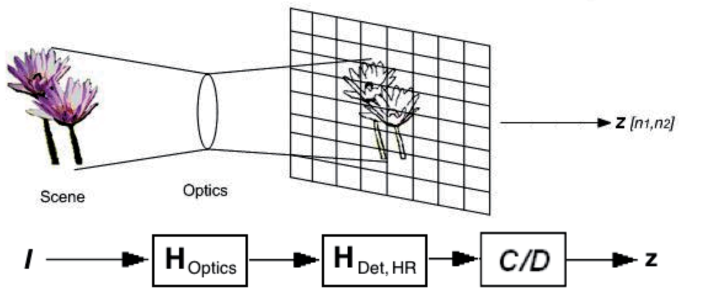


Figure 1.2: Camera model with a high-resolution and noise free detector array. The effect of the finite size high-resolution detector sensor is modeled as $H_{DET,HR}$.

Assume that the camera lens, modeled as H_{Optics} in Figure 1.1, restricts the bandwidth of the input irradiance, and let $I_{BandLimited}$ be the resultant irradiance. Suppose that an imaginary high-resolution detector array has replaced the low-resolution and noisy detector array as depicted in Figure 1.2. If this imaginary detector array is dense enough and noise free, a high-resolution image $\mathbf{z}$ can be obtained, whose pixels can be used to recover $I_{BandLimited}$ perfectly.

Super resolution methods try to estimate this high-resolution image $\mathbf{z}$ from the low-resolution image $\mathbf{y}$. The block diagram in Figure 1.3 illustrates the relation between the continuous image I, high-resolution image, $\mathbf{z}$, and low-resolution image, $\mathbf{y}$.

Lexicographical representation of the images are used for convenience in express-

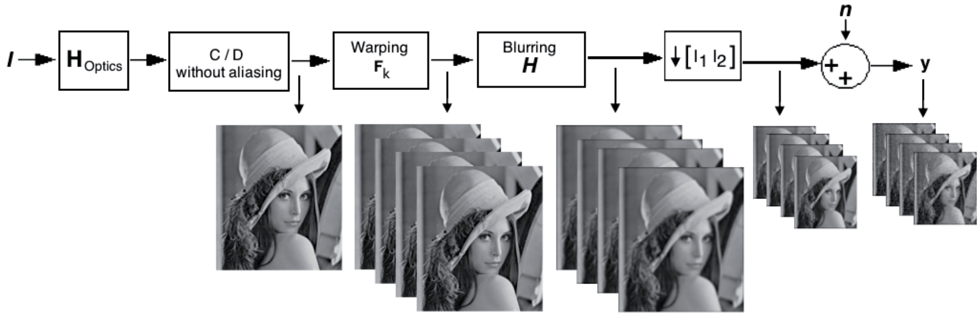


Figure 1.3: Relationship between the high-resolution image $\mathbf{z}$ and a set of low-resolution images $\mathbf{y_k}$, where $k = 1, \ldots, c$, and c is the number of LR images. l_1 and l_2 are downsampling factors in vertical and horizontal directions respectively.

ing the mathematical relationship between $\mathbf{y}$ and $\mathbf{z}$. In this representation, all the rows of an image is attached to each other to form a long column vector. By this way, the $M \times N$ size low-resolution image $\mathbf{y}$ is represented as an $MN \times 1$ size column vector. Similarly, the high-resolution image $\mathbf{z}$, is a vector of size $l_1M\,l_2N \times 1$, where l_1 and l_2 are downsampling factors in vertical and horizontal directions, respectively. The relationship between $\mathbf{z}$ and $\mathbf{y}$, shown in Figure 1.3, can be formulated as follows:

$$\mathbf{y} = DH \cdot \mathbf{z} + \mathbf{n}, \tag{1.1}$$

where D is an $MN \times l_1M\,l_2N$ matrix representing the down sampling operation, H is a $l_1Ml_2N \times l_1Ml_2N$ matrix representing the effect of finite detector size, and $\mathbf{n}$ is additive noise vector of size $MN \times 1$. Blurring an $l_1M \times l_2N$ image by a 3 by 3 kernel requires $l_1M \times l_2N \times 3 \times 3$ multiplications and additions in 2 dimensional image domain. However, the same operation $(H \cdot \mathbf{z})$ requires $l_1Ml_2N \times l_1Ml_2N \times l_1Ml_2N \times l_1Ml_2N$ multiplications and additions in lexicographical representations. Although the number of operations are vastly different in both domains, the results are exactly the same. This difference is due to the sparsity of operator matrices defined in lexicographical form as shown in Appendix A. The key operations such as blurring H, decimation D, and affine transformations F, are given explicitly in lexicographical form for 4×4 images in Appendix A. In practice, all the operations are conducted in 2-dimensional image domain due to efficiency. Lexicographical representation is used for all the equations throughout this thesis, unless otherwise stated.

The problem of finding $\mathbf{z}$ given $\mathbf{y}$ is an ill-posed inverse problem [1]. Firstly, the number of unknown high-resolution pixels are higher than the known low-resolution pixels. Secondly, observation model errors and additive noise creates instability during the iterative reconstructions. The equation (1.1) can be made over-determined by adding more LR images to the equation set. This idea is illustrated in Figure 1.4, where 3 LR images with different spatial orientation with respect to a reference grid are aligned on a higher resolution grid. If these images have non-redundant information, such as presence of aliasing and sub-pixel shifts with respect to a reference frame, then a higher resolution image can be reconstructed.

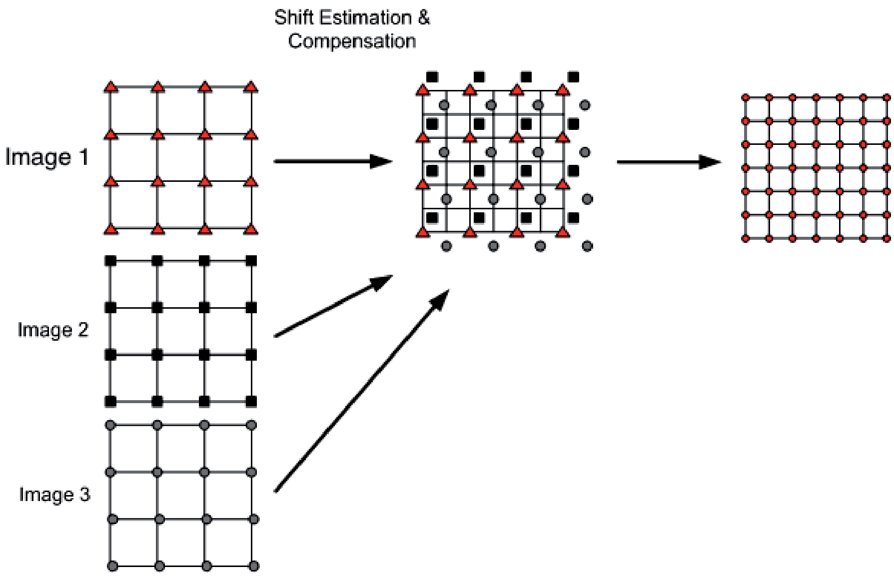


Figure 1.4: LR images with non-redundant information can cover a higher resolution grid to make higher resolution image reconstruction possible.

The sub-pixel shifts among LR images provide the necessary information to make the system of equations overdetermined. The equation (1.1) takes the following form with multiple LR images:

$$\mathbf{y_k} = \underbrace{DHF_k}_{W_k} \cdot \mathbf{z} + \mathbf{n_k} \qquad k = 1, \ldots, c \tag{1.2}$$

Here, LR images are denoted as $\mathbf{y}_k$, where k is the index of the LR images, and c is the number of LR images. F_k represents the warping operation in matrix form. Sub-pixel shifts are inserted into the equation set through this operator. Therefore, a sound estimation for F_k is the most critical part of the SR image reconstruction process. This thesis does not focus on the image registration problem. Image registration values are known *a priori* in most of the simulations. In (1.2), sampling, blur, and warping operators are merged into a single matrix W_k.

1.2 STATEMENT OF THE PROBLEM

Given the model in (1.2), the SR problem statement is given as follows.

Given a set of low-resolution images with dimensions $M \times N$, reconstruct a higher resolution image with an expansion factor l_1 and l_2 in vertical and horizontal directions respectively, so that the lost details of the original scene is restored.

Related problems are:

Image restoration: Given a single or a set of images with dimensions $M \times N$, reconstruct a restored image with the same dimensions as the degraded image.

Single frame interpolation: Given a single low-resolution image with dimensions $M \times N$, reconstruct a higher resolution image with an expansion factor l_1 and l_2 in vertical and horizontal directions respectively.

Unlike single frame interpolation methods, SR image reconstruction tries to increase the effective resolution of the image data. As mentioned before, just increasing the number of samples through single frame linear or bicubic interpolation do not necessarily increase the effective resolution of the image. The details are lost through imaging process due to optical blur, finite detector size, and down sampling. In contrast to the single frame interpolation methods, SR reconstruction methods retrieve the lost high frequency information. If the optical blur filter in Figure 1.1 were a perfect anti-aliasing filter, it would not be possible to retrieve any of the lost details. In such a case, SR reconstruction would only upsample and restore the LR images. Therefore, SR reconstruction is not suitable for imaging devices that outputs perfectly anti-aliased LR images.

1.3 CONTRIBUTION OF THE THESIS

- The first contribution of the thesis is a novel edge preserving SR image reconstruction method. The proposed Bayesian estimator uses local gradient direction and amplitude for optimal noise reduction while preserving edges. The proposed method estimates gradient amplitude and direction at each iteration. This gradient map guides the SR reconstruction stage through iterations.

- The second contribution of the thesis is a novel texture prior for maximum a posteriori (MAP) based super resolution (SR) image reconstruction. The prior is based on a multiscale compound Markov Random Field (MRF) model. Gabor filters are utilized for subband decomposition. Each subband is modeled by a compound MRF that inherits a binary texture process. The texture process at each pixel location at each subband is estimated iteratively along with the unknown high-resolution image pixels.

- Finally, A two stage method is proposed, comprising multiple SR reconstructions with different regularization parameters followed by a restoration step for preserving edges and textures. In the first stage, two maximum-a-posteriori (MAP) estimators with two different amounts of regularizations are employed. In the second stage, pixel-to-pixel difference between these two estimates is post-processed to restore edges and textures. Frequency selective characteristics of discrete cosine transform (DCT) and Gabor filters are utilized in the post-processing step.

1.4 THESIS OUTLINE

In Chapter 2, a literature survey on SR methods is presented.

In Chapter 3, a brief analysis is done on distance measures used in the cost function expressions in MAP estimators.

In Chapter 4, a novel gradient adaptive edge preserving Bayesian method for SR reconstruction is proposed.

In Chapter 5, a novel texture prior for Bayesian estimation of SR image pixels is proposed.

In Chapter 6, a two stage edge and texture preserving SR method suitable for real time applications is proposed.

In Chapter 7, conclusion and future directions are given.

CHAPTER 2

LITERATURE SURVEY

Super resolution image reconstruction has been discussed in image processing literature for over three decades. Although the problem is first stated in the 1980s [1], similar problems such as image and video interpolation, restoration, and algebraic image reconstruction for computed tomography have constructed the basis for many SR approaches. In this chapter, pioneering studies published in the literature are summarized under various topics to give a structured presentation for a reader who is not familiar with the field. We also briefly consider video SR problem at the end of the chapter. A special attention is given to stochastic methods since the proposed approaches belong to this category.

2.1 FREQUENCY DOMAIN METHODS

The first formulation of the SR problem is given in frequency domain by Tsai and Huang [1]. Their solution is based on the Fourier shift theorem and the aliasing relation between high resolution (HR) image, **z**, and corresponding low resolution (LR) images, **y**. Assume that **y** is obtained by downsampling **z** by a factor of r in both directions, then the aliasing relation between 2D discrete Fourier transforms (DFT) of **y** and **z** is given as, [17]

$$\mathbf{Y}[n_1, n_2] = \frac{1}{r^2} \sum_{p=0}^{r-1} \sum_{p=0}^{r-1} \mathbf{Z}[\,(n_1 - Mp)_{Mr}\,, (n_2 - Np)_{Nr}] \quad , \tag{2.1}$$

where $(\cdot)_N$ is the mod operation, M and N are the number of LR image pixels in vertical and horizontal directions. $\boldsymbol{Z}$ and $\boldsymbol{Y}$ are DFTs of **z** and **y** respectively. The aim of SR image reconstruction is to obtain **z**, given a set of low resolution images

$\mathbf{y}_k$, which are related to each other by global translational motion. Assume that each $\mathbf{y}_k$ is obtained by down sampling a separate HR image $\mathbf{z}_k$, and assume each HR image $\mathbf{z}_k$ is obtained from a reference image $\mathbf{z}_1$ by globally shifting it by α_k and β_k pixels in vertical and horizontal directions. The relationship between the DFTs of $\mathbf{z}_1$ and $\mathbf{z}_k$ is given in (2.2). It is assumed that no aliasing occurred while obtaining $\mathbf{z}$ through the samples of the continuous image [17],

$$\mathbf{Z}_k[n_1, n_2] = \mathbf{Z}[n_1, n_2] \cdot e^{-j\frac{2\pi}{Mr}\cdot n_1\cdot\alpha_k} e^{-j\frac{2\pi}{Nr}\cdot n_2\cdot\beta_k}. \tag{2.2}$$

Substituting (2.2) into (2.1) gives the following relation between DFTs of $\mathbf{z}$ and $\mathbf{y}_k$:

$$\begin{aligned}\mathbf{Y}_k[n_1, n_2] = \frac{1}{r^2}\sum_{p=0}^{r-1}\sum_{p=0}^{r-1} \Big(& \mathbf{Z}[\,(n_1 - Mp)_{Mr}\,, (n_2 - Np)_{Nr}] \quad \text{x} \\ & e^{-j\frac{2\pi}{Mr}\cdot(n_1 - Mp)_{Mr}\cdot\alpha_1} e^{-j\frac{2\pi}{Nr}\cdot(n_2 - Np)_{Nr}\cdot\beta_1} \Big),\end{aligned} \tag{2.3}$$

where k is the index for each LR image. For multiple LR images the above equation set can be made overdetermined to solve the missing DFT coefficients of $\mathbf{z}$. Once all of the DFT coefficients $\mathbf{Z}[m_1, m_2]$ are estimated, the inverse DFT of $\mathbf{Z}$ gives the HR image. This method directly aims at finding the missing aliased high frequency components. Later, this approach is extended to include image blur, and noise [18], [19]:

$$\begin{aligned}\mathbf{Y}[n_1, n_2] = \frac{1}{r^2}\sum_{p=0}^{r-1}\sum_{p=0}^{r-1} \Big(& \mathbf{Z}[\,(n_1 - Mp)_{Mr}\,, (n_2 - Np)_{Nr}] \quad \text{x} \quad H[\,(n_1 - Mp)_{Mr}\,, (n_2 - Np)_{Nr}\,] \\ & \text{x} \quad e^{-j\frac{2\pi}{Mr}\cdot(n_1 - Mp)_{Mr}\cdot\alpha_1} e^{-j\frac{2\pi}{Nr}\cdot(n_2 - Np)_{Nr}\cdot\beta_1} \Big) + \mathbf{V}[n_1, n_2],\end{aligned} \tag{2.4}$$

where H is the DFT of image blur operation, and $\mathbf{V}[n_1, n_2]$ is the DFT of additive noise. Frequency domain methods are convenient for hardware implementations due to the use of fast Fourier transforms. However, these methods only assume global translational motion among the LR images. Moreover, it is usually hard to apply *a priori* spatial domain information to the solution procedure.

2.2 INTERPOLATION-RESTORATION BASED METHODS

Interpolation-restoration type methods mainly consist of three stages: i) Registration of LR image pixels onto an HR image grid according to the motion estimate, ii) Non-uniform interpolation on the HR grid, iii) Restoration for blur removal. The first two stages are depicted in Figure 2.1. During the second stage direct [20] or iterative methods [21] can be used to map the interpolated data onto the HR grid.

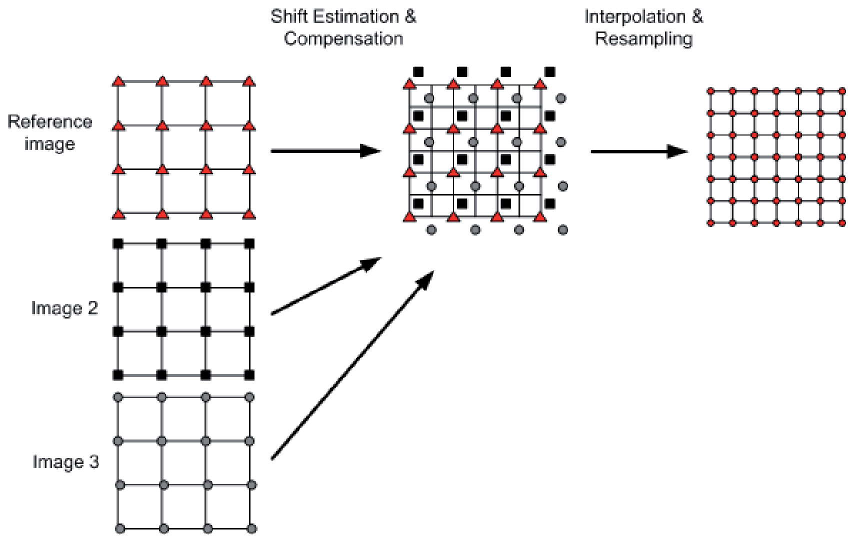


Figure 2.1: Once the motion is found this information is mapped on to the HR grid The HR image pixel points are found by interpolating this data around the LR image pixel values depicted as "△", "□ ", "○".

The simplest form of such methods is given in [20], where image registration step is followed by interpolation and motion compensation with respect to a reference frame. The interpolated images are then averaged pixel-wise to obtain an SR image. The outlier pixels among the LR image set is discarded during the averaging stage. Alam *et al.* [22] has employed the same approach for infrared images. After subpixel registration, a Wienner filter is designed according to the optics of the camera to deblur the resultant image. In [21], the low resolution image pixels are mapped to a high resolution grid and non-uniformly spaced samples are interpolated iteratively. However, sensor blur is not considered in their approach. In [19], local thin-plate spline method is used for interpolation, which is followed by a Wienner filter restoration for blur and noise.

These algorithms are simple, easy to implement, suitable for implementing on FP-

GAs, and computationally cheap [23]. however, these methods do not use an imaging model nor noise model. The methodology assumes that the LR image samples are obtained through impulse field at 2D sampling grids, however they are actually spatial averages [24]. Separation of non-uniform interpolation and restoration steps is only possible, if the underlying motion among LR images is pure translational motion [25]. Furthermore, it is not straightforward to incorporate *a priori* information to the solution.

2.3 ITERATIVE BACK PROJECTION METHOD

Iterative back projection method is adapted from algebraic reconstruction methods in computer aided tomography, [26]. The idea is to simulate the imaging process (forward problem) using an initial HR estimate and to back project the error between the simulated and real low resolution images to the new HR estimate. The method is given in the following equation, [27]:

$$\begin{aligned} ysim_k^n[m_1, m_2] &= \sum_{n_1, n_2} W_k[m_1, m_2; n_1, n_2] \cdot \hat{z}^n[n_1, n_2] \\ \hat{z}^{n+1}[n_1, n_2] &= \hat{z}^n[n_1, n_2] + \sum_{n_1, n_2} h_{BP}[m_1, m_2; n_1, n_2] \cdot \left(y_k[m_1, m_2] - ysim_k^n[m_1, m_2] \right), \end{aligned} \quad (2.5)$$

where $W_k[m_1, m_2, n_1, n_2]$ is related to the observation model in Chapter 1. $ysim_k^n$ is the k^{th} simulated low resolution image at n^{th} iteration. $\hat{z}^n$ is the HR estimate at n^{th} iteration and h_{BP} is the back projection operator. Various h_{BP} operators can be used for regularizing the solution as well. This method is easy to implement, however it is difficult to apply *a priori* constraints [24] through the back projection operator.

2.4 PROJECTION ONTO CONVEX SETS METHOD

The projection onto convex sets (POCS) method is an iterative method that has been employed in digital image restoration field [28, 29, 30]. The method has been first used in SR image reconstruction in [31]. In POCS method, each *a priori* information, such as an LR image pixel value, constrains the solution into a convex set, C_m. The solution belongs to the intersection of all sets C_m ($m = 1 \cdots c$). This intersection is found iteratively using projection operators, P_m, defined for each constraint, C_m, as follows:

$$z^{n+1} = P_c P_{c-1} P_{m-2} \cdots P_2 P_1 z^n. \tag{2.6}$$

The relationship between each LR image pixel value, $y_k[m_1, m_2]$, and corresponding HR image pixel value, $z[n_1, n_2]$, is referred to as the "fidelity to data" or the "data consistency" constraint. Each LR image pixel $y_k[m_1, m_2]$ defines the following convex set according to the data consistency constraint:

$$C_{m_1,m_2,k} = \left\{ z[n_1, n_2] : y_k[m_1, m_2] = \sum_{n_1,n_2} W_k[m_1, m_2; n_1, n_2] \cdot z[n_1, n_2] \right\}, \tag{2.7}$$

where $W_k[m_1, m_2, n_1, n_2]$ is related to the observation model in Chapter 1. The iterative solution for the above constraining convex set can be given as follows:

$$z^{(n+1)}[n_1, n_2] = z^{(n)}[n_1, n_2] + \frac{r^{(z)}[m_1, m_2] \cdot W_k[m_1, m_2, n_1, n_2]}{\sum_{p,q} W_k^2[m_1, m_2, p, q]}, \tag{2.8}$$

where

$$r^{(z)}[m_1, m_2] = y_k[m_1, m_2] - \sum_{n_1,n_2} W_k[m_1, m_2; n_1, n_2] \cdot z^{(n)}[n_1, n_2]. \tag{2.9}$$

The update equation (2.8) is a special case of (2.5), where $h_{BP}[m_1, m_2, n_1, n_2]$ is replaced by $W_k[m_1, m_2, n_1, n_2] \Big/ \sum_{p,q} W_k^2[m_1, m_2, p, q]$.

Tekalp *et al.* [19] has extended the POCS method to include observation noise as follows:

$$C_{m_1,m_2,k} = \left\{ z[n_1, n_2] : \left| y_k[m_1, m_2] - \sum_{n_1,n_2} W_k[m_1, m_2; n_1, n_2] \cdot z[n_1, n_2] \right| < \delta \right\}, \tag{2.10}$$

where δ is related to the observation noise and represents the confidence in the data. The update term with the confidence term, δ is:

$$z^{(n+1)}[n_1,n_2] = z^{(n)}[n_1,n_2] + \begin{cases} \frac{\left(r^{(z)}[m_1,m_2]-\delta[m_1,m_2]\right)\cdot W_k[m_1,m_2,n_1,n_2]}{\sum_{p,q} W_k^2[m_1,m_2,p,q]} & r^{(z)}[m_1,m_2] > \delta[m_1,m_2] \\ 0 & \left|r^{(z)}[m_1,m_2]\right| < \delta[m_1,m_2] \\ \frac{\left(r^{(z)}[m_1,m_2]+\delta[m_1,m_2]\right)\cdot W_k[m_1,m_2,n_1,n_2]}{\sum_{p,q} W_k^2[m_1,m_2,p,q]} & r^{(z)}[m_1,m_2] < -\delta[m_1,m_2] \end{cases} \tag{2.11}$$

Additional constraints, such as the amplitude constraint [19], can also be added to narrow the search space as follows:

$$C_a = \{z[n_1,n_2] : a < z[n_1,n_2] < b\}. \tag{2.12}$$

Later, smoothness constraint [9] has been added to penalize the image spatial derivative terms in Tikhonov type regularization sense as follows:

$$C_s = \left\{z[n_1,n_2] : \Big| [S\,z][n_1,n_2] \leq \delta \Big| \right\}, \tag{2.13}$$

where S is an high pass filter operator, and δ is related to the amount of smoothness.

Motion blur effects has been added to the POCS formulation in [32]. Later, POCS formulation has been expanded to include local object motions in the image by using a validity map and a segmentation map [33]. The main advantage of POCS method is the ease of using spatial domain *a priori* information. The method is robust against inconsistent or missing data. Possible difficulty is finding operators for projections [23].

2.5 DETERMINISTIC METHODS

The methods in this category is based on the following forward problem equation given in Chapter 1:

$$\mathbf{y_k} = W_k \cdot \mathbf{z} + \mathbf{n_k} \qquad k = 1 \cdots c, \tag{2.14}$$

where $\mathbf{z}$ and $\mathbf{y}_k$ are the the HR image and the k^{th} LR image respectively. A natural solution to the system of equations given in (2.14) is obtained by minimizing the following cost function:

$$Cost = \sum_{k=1}^{c} ||W_k \mathbf{z} - \mathbf{y_k}||_p^p. \tag{2.15}$$

ℓ_1 norm or ℓ_2 norm is used if the noise is assumed to have Laplacian distribution, [25] or Gaussian distribution [13] respectively. High resolution image can be estimated iteratively by minimizing this cost function using several methods such as: Gradient descent, steepest descent, conjugate gradient [34] etc. Gradient descent derivation is given here for its simplicity. For both noise cases, the gradients of the cost function with respect to $\mathbf{z}$ are;

$$\begin{aligned}
\nabla Cost &= \sum_{k=1}^{c} 2(W_k^T) \cdot (W_k \mathbf{z} - \mathbf{y_k}) \quad for \quad ||\cdot||_2^2, \\
\nabla Cost &= \sum_{k=1}^{c} (W_k^T) \cdot sign(W_k \mathbf{z} - \mathbf{y_k}) \quad for \quad ||\cdot||_1^1.
\end{aligned} \tag{2.16}$$

Substituting $\nabla Cost$ into, $\mathbf{z}^{n+1} = \mathbf{z}^n - \beta \cdot \nabla Cost$,

$$\begin{aligned}
\mathbf{z}^{n+1} &= \mathbf{z}^n - \beta \sum_{k=1}^{c} \left(W_k^T\right) \cdot \left(W_k \hat{\mathbf{z}}^n - \mathbf{y_k}\right) \quad for \quad ||\cdot||_2^2 \\
\mathbf{z}^{n+1} &= \mathbf{z}^n - \beta \sum_{k=1}^{c} \left(W_k^T\right) \cdot sign\left(W_k \hat{\mathbf{z}}^n - \mathbf{y_k}\right) \quad for \quad ||\cdot||_1^1
\end{aligned} \tag{2.17}$$

In SR problem, even for over-determined cases, modeling errors and noise in the process create unstable solutions. Ill-posed nature of the SR problem makes regularization necessary for achieving a stable solution. Tikhonov type regularization is very common and expressed as follows, [13]

$$Cost = \sum_{k=1}^{c} ||W_k \mathbf{z} - \mathbf{y_k}||_2^2 + \lambda ||C\mathbf{z}||_2^2 \tag{2.18}$$

where C refers to the regularization operator in matrix form. Selecting C as Laplacian operator in matrix form penalizes solutions with derivative values in both direction, hence favors smoother solutions, [13]. But in SR literature over-smoothing of edges is an important issue. Recently, total variation (TV) method [35] is proposed for edge preserving regularization in deblurring and denoising applications. The cost function to be minimized for TV type regularization is,

$$Cost = \sum_{k=1}^{c} ||W_k \mathbf{z} - \mathbf{y}_k||_p^p + \lambda ||\nabla \mathbf{z}||_1^1. \tag{2.19}$$

With this approach, ℓ_1 norm of the pixel gradient is penalized instead of the ℓ_2 norm. ℓ_1 norm tends to penalize edges less severely compared to ℓ_2 norm [25]. In [25] a variation of TV criterion, bilateral total variation filter (BTV) is proposed as the regularization term for edge preservation. The cost function to be minimized for BTV type regularization is,

$$Cost = \sum_{k=1}^{c} ||W_k \mathbf{z} - \mathbf{y}_k||_p^p + \lambda \sum_{l=-p}^{p} \sum_{m=-r}^{r} \alpha^{|l+m|} || \mathbf{z} - S_h^l S_v^m \mathbf{z}||_1^1 \tag{2.20}$$

where S_h^m shifts the image in horizontal direction by m pixels and S_v^m shifts image in vertical direction by m pixels. Note that, $||\frac{\partial f}{\partial h}||_1^1 \approx ||S_h^1 f||_1^1$ and $||\frac{\partial f}{\partial v}||_1^1 \approx ||S_v^1 f||_1^1$. So BTV can be considered as a more general derivative operator. α (between 0 and 1) decreases the effect of far pixels to regularization exponentially. As a special case, for $p = 1$, $r = 1$, and $\alpha = 1$, BTV approximates the TV prior [36].

Deterministic methods has the flexibility to vary the cost function in several ways to make the solution converge to any desired solution. It will be shown that stochastic methods given in the next section are similar in many ways to the deterministic methods. Many of the solutions given in this section can be derived by the stochastic approaches.

2.6 STOCHASTIC METHODS

Stochastic SR methods propose ways to insert probabilistic information to the solution procedure. These methods use image and noise probability density functions (PDF) to estimate the most probable SR image. Bayesian methods have been applied to image restoration [37], and interpolation problems [38, 39].

Stochastic SR methods are based on the Bayes' theorem which states that:

$$p(A|B) = \frac{p(B|A) \cdot p(A)}{p(B)}. \tag{2.21}$$

Application of this theorem to the SR problem is simply asking "*What is the most* *probable estimate for the high resolution image, given low resolution images and the image prior?*". Maximum a posteriori (MAP) solution to this problem maximizes a posteriori PDF, $p\left(\mathbf{z}\,|\,\mathbf{y}_1,\mathbf{y}_2,\mathbf{y}_3\cdots\mathbf{y}_c\right)$, with respect to $\mathbf{z}$. Applying the Bayes' theorem to this conditional probability results in the following MAP solution, [40, 13],

$$\hat{\mathbf{z}}_{MAP} = \underset{z}{argmax}\left[p(\mathbf{y}_1,\mathbf{y}_2,\mathbf{y}_3\cdots\mathbf{y}_c|\mathbf{z})\cdot p(\mathbf{z})\cdot p(\mathbf{y}_1,\mathbf{y}_2,\mathbf{y}_3\cdots\mathbf{y}_c)^{-1}\right], \tag{2.22}$$

where $p(\mathbf{z})$ is the PDF of the solution, and $p(\mathbf{y}_1,\mathbf{y}_2,\mathbf{y}_3\cdots\mathbf{y}_c|\mathbf{z})$ is the joint PDF of LR images given $\mathbf{z}$. Joint PDF of LR images, $p(\mathbf{y}_1,\mathbf{y}_2,\mathbf{y}_3\cdots\mathbf{y}_c)$, has no effect to the above maximization. Removing this term results:

$$\hat{\mathbf{z}}_{MAP} = \underset{z}{argmax}\left[p(\mathbf{y}_1,\mathbf{y}_2,\mathbf{y}_3\cdots\mathbf{y}_c|\mathbf{z})\cdot p(\mathbf{z})\right]. \tag{2.23}$$

In the above maximization, $p(\mathbf{y}_1,\mathbf{y}_2,\mathbf{y}_3\cdots\mathbf{y}_c\,|\,\mathbf{z})$ term is related to the observation model formulated in (2.14). Assuming each LR image $\mathbf{y}_k$ is independent observations:

$$\hat{\mathbf{z}}_{MAP} = \underset{z}{argmax}\left[p(\mathbf{z})\cdot\prod_{k=1}^{c}p(\mathbf{y}_k|\mathbf{z})\right]. \tag{2.24}$$

2.6.1 Maximum Likelihood Solution

If no a priori information is available about the solution, the image prior term, $p(\mathbf{z})$, can be considered to have uniform distribution. In this case, it has no effect to the maximization in (2.24). This case is referred to as the maximum likelihood (ML) solution:

$$\hat{\mathbf{z}}_{ML} = \underset{z}{argmax}\prod_{k=1}^{c}p(\mathbf{y}_k|\mathbf{z}). \tag{2.25}$$

The ML solution depends on the PDF of the observation noise in the model given in (2.14). The PDF of noise at each pixel is assumed to be zero mean, identical, and independent of each other. Both Gaussian and Laplacian noise cases are investigated.

Under i. i. d. assumptions, the multivariate Gaussian distribution for the noise vector is given as follows:

$$p(\mathbf{n_k}) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left(-\frac{\mathbf{n_k}^T\mathbf{n_k}}{2\sigma^2}\right). \tag{2.26}$$

Here, noise term, $\mathbf{n_k}$, is an $N \times 1$ column vector, where each element refers to the additive noise added to the specific pixel value and N is the number of LR image pixels. σ is the standard deviation of noise at one pixel. Conditional PDF of low resolution images, $p(\mathbf{y}_k|\mathbf{z})$, can be written in terms of noise PDF by substituting the noise term $\mathbf{n_k}$ with $(\mathbf{y}_k - W_k\mathbf{z})$ according to (2.14).

$$p(\mathbf{y}_k|\mathbf{z}) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\Big[-\frac{1}{2\sigma^2}(\mathbf{y}_k - W_k\mathbf{z})^T \cdot (\mathbf{y}_k - W_k\mathbf{z})\Big]$$

In this case ML solution takes the following form

$$\hat{\mathbf{z}}_{ML} = \underset{z}{argmax}\left(\prod_{k=1}^{c} \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\Big[-\frac{1}{2\sigma^2}(\mathbf{y}_k - W_k\mathbf{z})^T \cdot (\mathbf{y}_k - W_k\mathbf{z})\Big]\right). \tag{2.27}$$

Maximizing the logarithm of the PDF gives the same solution since the logarithm is a positive monotonic function. Product operation is replaced by summation term and the minus sign in each probability term reverses the maximization problem into minimization problem as follows:

$$\hat{\mathbf{z}}_{ML} = \underset{z}{argmin}\left(\sum_{k=1}^{n} \frac{1}{2\sigma^2}(\mathbf{y}_k - W_k\mathbf{z})^T(\mathbf{y}_k - W_k\mathbf{z}) + \ln\frac{c}{(2\pi\sigma^2)^{N/2}}\right). \tag{2.28}$$

The ML solution therefore can be found by minimizing the following cost function for Gaussian noise distribution:

$$Cost = \sum_{k=1}^{n} ||W_k\mathbf{z} - \mathbf{y}_k||_2^2\,. \tag{2.29}$$

Now assume noise has a zero mean, identical and independent Laplacian distribution. The probability density function for multivariate Laplacian noise is [41]

$$p(\mathbf{n}) = \frac{1}{(2\sigma)^N} \cdot \exp\left(\frac{||\mathbf{n}||_1^1}{\sigma}\right). \tag{2.30}$$

Conditional PDF of low resolution images are derived from the multivariate Laplacian noise distribution equation above with a similar approach to the normal distribution case. ML solution in this case is,

$$\hat{\mathbf{z}}_{ML} = \underset{z}{argmax} \prod_{i=1}^{n} \frac{1}{(2\sigma)^N} \cdot \exp \frac{1}{\sigma} \|\mathbf{y}_k - W_k\mathbf{z}\|_1^1 . \tag{2.31}$$

Maximizing the above expression gives the same estimate as maximizing the logarithm of the PDF since the probability is a positive monotonic function.

$$\hat{\mathbf{z}}_{ML} = \underset{z}{argmin} \left(\sum_{k=1}^{n} \frac{1}{\sigma} |W_k\mathbf{z} - \mathbf{y}_k\|_1^1 + \frac{1}{(2\sigma)^N} \right). \tag{2.32}$$

Therefore, if the noise is Laplacian the cost function becomes

$$Cost = \sum_{k=1}^{n} \|W_k\mathbf{z} - \mathbf{y}_k\|_1^1 . \tag{2.33}$$

It should be noted that the derived cost functions in (2.29) and (2.33) are exactly the same as the one given in the previous section in (2.15).

2.6.2 Incorporating the Image Prior

Formulating an image prior is a challenging task in MAP estimation. Natural images have abrupt changes within neighboring pixels, due to occlusions, shadows, and textures. Therefore, slow variations as well as discontinuities may exist in a pixel neighborhood. Modeling this relation within a Bayesian framework is achieved by constructing an image prior model, $p(\mathbf{z})$. Markov Random Fields are widely used in achieving this goal. Markovian process is defined as the process, where the probability of having a certain outcome given all the previous outcomes is equivalent to the probability of having a certain outcome given only the previous outcome. This can be formulated as follows:

$$p(x_n|x_{n-1}) = p(x_n|x_{n-1}, x_{n-2}, x_{n-3}, x_{n-4}, \cdots),$$

where n shows the time index. This assumption is fairly true for real images since the dependencies of pixels to each other are mostly local. Markov random field (MRF) with the Gibbs density function is given in a general form as follows:

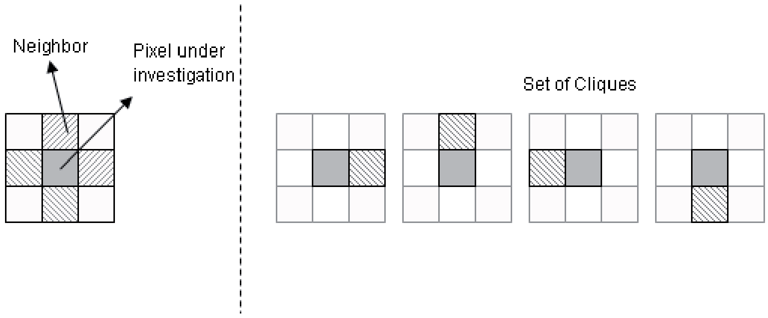


Figure 2.2: On the left, a pixel with its four neighbors is shown. On the right, 4 cliques defined for the neighborhood system are depicted with texture.

$$p(\mathbf{z}) = \frac{1}{H} \exp\left\{ -\sum_{c \in C} V_c(\mathbf{z}) \right\}, \tag{2.34}$$

where $V_c(\mathbf{z})$ is a clique potential function and C is the set of cliques. Cliques define certain pairs in images. For example, a pixel and its left neighbor is a clique. Clique potentials define the relationship within a clique. For a hypothetical case, if it is given that the probable value of a pixel only depends on its cardinal neighbors (up, down, left, right) as shown in Figure 2.2, then $p(\mathbf{z})$ is formed by using (2.34) as follows:

$$p(\mathbf{z}) = \frac{1}{Z} \exp\left\{ -\sum_{i \sim j} V_c(z_i - z_j) \right\} \tag{2.35}$$

where $\sum_{i \sim j}$ represents all pixel pairs that are neighbors to each other. z_i and z_j are i^{th} and j^{th} pixels of image $\mathbf{z}$. Note that the joint probability density function peaks when the image is uniform (*i. e.* $z_i - z_j = 0$).

Clique potentials takes the quadratic form for Gaussian prior case. The following relation penalizes the differences within neighboring pixels.

$$p(\mathbf{z}) = \frac{1}{Z} \exp\left\{ -\sum_{i \sim j} \lambda \cdot (z_i - z_j)^2 \right\} \tag{2.36}$$

Based on [13], the above PDF can be written in a multivariate Gaussian form:

$$p(\mathbf{z}) = |A|^{1/2} \left(\frac{1}{2\pi} \right)^{N/2} \exp\left\{ -\frac{1}{2} \mathbf{z}^T A \mathbf{z} \right\} \tag{2.37}$$

where A is the inverse covariance matrix, whose elements are:

$$A(i,j) = \begin{cases} 4\lambda & i = j \\ -\lambda & i \sim j \\ 0 & else \end{cases} . \tag{2.38}$$

here $i \sim j$ refers to the indices of neighboring pairs. λ is a tuning parameter affecting the contribution of the prior to the overall solution. Note that in (2.38) only cardinal neighbors are considered. Inserting the prior model in the MAP solution is straightforward:

$$\hat{\mathbf{z}}_{MAP} = \underset{z}{argmax}\left(\prod_{k=1}^{c} \frac{1}{(2\pi\sigma)^{N/2}} \exp\left[-\frac{1}{2\sigma^2}(\mathbf{y}_k - W_k\mathbf{z})^T \cdot (\mathbf{y}_k - W_k\mathbf{z})\right] \cdot \frac{1}{Z}\exp\left\{-\sum_{i\sim j}\lambda \cdot (z_i - z_j)^2\right\}\right). \tag{2.39}$$

Following the same argument to derive (2.28),

$$\hat{\mathbf{z}}_{MAP} = \underset{z}{argmin}\left(\sum_{k=1}^{n}(\mathbf{y}_k - W_k\mathbf{z})^T(\mathbf{y}_k - W_k\mathbf{z}) + \sum_{i\sim j}\lambda \cdot (z_i - z_j)^2\right), \tag{2.40}$$

where the constant terms are removed from the minimization problem. In [39, 40] Huber-Markov Random Field Model is used as image prior. The clique potentials are defined using Huber function as follows

$$p(\mathbf{z}) = \frac{1}{Z}\exp \lambda \sum_{i\sim j}\rho_\kappa(z_i - z_j) . \tag{2.41}$$

where,

$$\rho_\kappa(x) = \begin{cases} x^2 & |x| \leq \kappa \\ 2T|x| - \kappa^2 & |x| > \kappa \end{cases} \tag{2.42}$$

where κ is a threshold determining the linear and quadratic regions in the cost function. If the discontinuity between neighboring pixels is less than κ, a quadratic penalty is applied otherwise a less severe linearly varying penalty is applied. MAP estimate for HMRF case is:

$$\hat{\mathbf{z}}_{MAP} = \underset{z}{argmin}\left(\sum_{k=1}^{n}(\mathbf{y}_k - W_k\mathbf{z})^T(\mathbf{y}_k - W_k\mathbf{z}) + \lambda\sum_{i\sim j}\rho_\kappa(z_i - z_j)\right). \tag{2.43}$$

The nonlinear nature of this function apply fewer penalties to strong edges while smoothing the low amplitude noise.

Deterministic and stochastic methods are both based on the same forward problem equation given in (1.2). The methods proposed in this thesis belong to this family of methods.

2.7 LEARNING BASED METHODS

Learning based approaches construct an SR image from LR images by using a data base, which consists of low resolution and high resolution image patch pairs. The main idea is to divide an input LR image into small overlapping patches and finding the best matching LR patch in the data base. Once a good match is found, the corresponding high frequency image patch is used to construct an SR image. In theory, for each matching LR patch, infinitely many HR patches can exists. However, images are not random signals, and the data base contains similar images. Moreover, input LR image is divided into overlapping patches to limit the possible HR image patch candidates according to a continuity constraint.

The data base could either consist of small size blocks of high resolution and corresponding low resolution image pixel values, [42, 43, 44], or mid-frequency and high frequency blocks obtained from training the low and high resolution images [45, 46], or a feature space consisting of Gaussian, Laplacian and spatial derivative pyramids, [47]. In [42, 43, 44, 47], training sets are used to form an image prior to be used in the MAP estimator. In [43] the image prior term is formed directly from the best matching high resolution pixel value in the training data. A Gaussian probability distribution whose mean is the gray scale value of the selected pixels is formed. This prior image is used to solve the multi-frame reconstruction problem. The method is applied on textured images. In [44], a similar idea is used for constructing the image prior to solve SR problem from a single low resolution image. In [42] an extension of this idea is proposed to be used in facial images. Training sets are also used to estimate directly the image posterior instead of an image prior in [46, 45, 48, 49]. Although these are the best available methods for a texture preserving framework, each LR patch is searched in an enormous databases of millions of high-resolution and low-resolution patch pairs. The need for a training set and searching the best possible solution for every pixel or image block do not make these methods good candidates for real time applications.

Recent progress in compressive sensing (CS) theory has made SR image reconstruction possible from a single LR observation using an over complete dictionary, which is trained for similar images. The CS theory states that if the imaging process and the representation basis are incoherent enough, a sparse representation and a good recovery of the signal is possible [50, 51]. CS based methods approach the problem from the same perspective as the learning based methods. The LR image is first divided into overlapping LR patches. At this stage, it is assumed that there exists a coupled-image representing basis, in which the high resolution and corresponding low resolution image has a sparse representation. For each LR patch the coefficients for a sparse representation is extracted using an LR dictionary. These coefficients are used to construct the HR patch using an HR dictionary. This requires training for two coupled dictionaries in such a way that if an LR patch is represented by a certain set of coefficients in the LR dictionary, exactly the same coefficients can be used to construct a corresponding high resolution patch using a HR dictionary. In [52, 53, 54] a single sparse representation with respect to both LR and corresponding HR dictionary is formed. An input LR image is divided into overlapping patches. For each patch a sparse coefficient set with respect to a LR dictionary is formed. The coefficients for this LR patch are used to construct a HR patch using the HR dictionary. Additional constraints for forcing continuity are also included. Later in [53, 54], the dictionaries are optimized for compactness to speed up computations. The same concept is also extended for face hallucination problem in [53]. In a similar work [55], instead of a coupled dictionary, a single dictionary in HR image domain is constructed. A sparse coefficient set for each LR patch is extracted from the blurred and down-sampled version of this dictionary. These coefficients are than used to construct HR image patches using the original dictionary. This approach has been applied on SR reconstruction of satellite images.

2.8 VIDEO SUPER RESOLUTION

In the previously discussed SR problem, the whole low resolution image set is assumed to be available for high resolution image reconstruction. However, in video frame up-sampling, low resolution images arrive sequentially for the reconstruction. Starting from the first frame a high resolution image is constructed and updated after each new frame. The need for video SR reconstruction arises in applications such as; real time target tracking, zooming, or standard to HD video conversion.

In video super resolution, the last reconstructed SR frame should be projected onto the most recent frame of reference determined by the last low resolution image to achieve continuity in the video, [56, 57]. The forward problem for video super resolution is:

$$
\begin{aligned}
\mathbf{y}_t &= DB \cdot \mathbf{z}_t + \mathbf{n}_t \\
\mathbf{z}_t &= F_{t-1}^{t} \cdot \mathbf{z}_{t-1} + \mathbf{v}_t.
\end{aligned}
$$

Here, D and B are the decimation and blur operators respectively, defined in lexicographical representation. Subindex denotes the time index, where $\mathbf{z}_t$ and $\mathbf{z}_{t-1}$ refers to the HR image at time t and $t-1$ respectively. F_{t-1}^{t} operator warps image at time $t-1$ to time t. Kalman filtering is a natural approach for tackling the above problem, since it will result in an optimal solution in the mean square sense [57]. However because of the large dimensions involved in the computations, approximations are preferred, [58]. Recursive least square and least mean square derivations are given in [57, 56] (See Appendix B). However, a direct application of the RLS approach is not possible due to large matrix sizes. A pseudo-RLS approach is given in [57, 56] (See Appendix B). Another important issue in video SR is the necessity of faster and parallel methods for high video frame rates. Least mean square approach is feasible in real time applications [57, 56], a statistical analysis of LMS algorithms in SR is given in [59]. Another challenge in video SR is the motion blur [60]. Ben *et al.* proposed a special camera that minimizes motion blur during image acquisition by sampling the space-time volume in a special manner, [60].

CHAPTER 3

ANALYSIS OF DISTANCE MEASURES IN SUPERRESOLUTION

3.1 INTRODUCTION

Derivations of various MAP estimators based on (1.2) and Bayes' Theorem have been presented in Chapter 2. In this chapter, following that discussion, a general MAP estimator is formulated as follows,

$$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \rho(W_k\mathbf{z}, \mathbf{y_k}) + \lambda\Gamma(\mathbf{z}), \tag{3.1}$$

where $\rho(\cdot)$ is a function that measures the distance between the data (low resolution image pixels) and the model. Norm functions, $||\cdot||_R^R$, are usually used as distance measures. ℓ_2 norm is employed if the noise is assumed to have a Gaussian distribution [40, 13] and ℓ_1 norm for Laplacian distribution [25]. The derivations for both cases have been discussed in Chapter 2. In Section 3.2, we further investigate other possible distance functions $\rho(\cdot)$ in (3.1). $\Gamma(.)$ represents the regularization term, which is related to the image prior term. $\Gamma(.)$ is usually expressed in the form of a spatial finite difference operator. Effect of various norm functions in the $\Gamma(.)$ term is investigated in Section 3.3.

3.2 DISTANCE MEASURES FOR DATA AND THE MODEL

3.2.1 Norm Function as a Distance Measure

Substituting the distance measure, $\rho(\cdot)$, in (3.1) with a general norm function, results in the following expression,

$$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} ||W_k\mathbf{z} - \mathbf{y_k}||_R^R + \lambda\Gamma(\mathbf{z}). \tag{3.2}$$

This cost function can be minimized iteratively using the gradient descent method, $\mathbf{z}^{n+1} = \mathbf{z}^n - \beta \cdot \nabla Cost$. Ignoring the regularization term, $\nabla Cost$ term is given as,

$$\nabla Cost = \sum_{k=1}^{c} W_k^T sign(W_k\mathbf{z} - \mathbf{y_k}) \odot |W_k\mathbf{z} - \mathbf{y_k}|^{R-1}. \tag{3.3}$$

Here $\odot$ operator represents pixel wise multiplication among vectors, and $|W_k\mathbf{z} - \mathbf{y_k}|^{R-1}$ term is a pixel wise operation, which applies to the pixels of each vector element. $\nabla Cost$ term takes the following form for ℓ_1 and ℓ_2 cases:

$$\nabla Cost_{\ell_1} = \sum_{k=1}^{c} W_k^T sign(W_k\mathbf{z} - \mathbf{y_k}), \tag{3.4}$$

$$\nabla Cost_{\ell_2} = \sum_{k=1}^{c} W_k^T (W_k\mathbf{z} - \mathbf{y_k}). \tag{3.5}$$

A statistical analysis presented in [41] states that noise in the image model has a more Laplacian characteristics than a Gaussian one. For more general cases, where both Laplacian and Gaussian noise are present, R can be selected between 1 and 2. If the character of the noise is unknown or have both Gaussian and Laplacian components, a mixture of ℓ_1 and ℓ_2 norms can be used, [61]. In [61], the ratio of Laplacian and Gaussian noise components has been estimated at each iteration. This ratio is used in reconstructing the cost function, which has a similar form to (3.6).

$$Cost = (1 - \beta) \cdot \sum_{k=1}^{c} ||W_k\mathbf{z} - \mathbf{y_k}||_2^2 + \beta \cdot Z \left\{ \sum_{k=1}^{c} ||W_k\mathbf{z} - \mathbf{y_k}||_1^1 \right\}^2 + \lambda\Gamma(\mathbf{z}). \tag{3.6}$$

Here β term determines the ratio of ℓ_1 and ℓ_2 norms in the overall cost function. Square of the ℓ_1 cost is taken to match the units of ℓ_1 cost and ℓ_2 cost. Moreover, a normalizing constant, Z, is introduced to balance the magnitudes of the two parts of the cost function.

3.2.2 Log-Sum Function as a Distance Measure

In compressive sensing literature [50, 51], ℓ_0 norm of the unknown sparse signal is minimized subject to an underdetermined linear equation set. However, due to the non-convexity of ℓ_0 norm, ℓ_1 norm approximations are preferred in estimating the sparse signal [62]. In [62], it is stated that a log-sum penalty function shown in (3.7) approximates ℓ_0 norm better than the ℓ_1 norm.

$$Cost = \sum_i log(|x_i + \epsilon|), \tag{3.7}$$

where ϵ is kept as a constant, and x_i is the elements of a sparse vector $\mathbf{x}$.

In this section, this idea is adapted to the super resolution problem. First a residue vector, $\mathbf{r_k}$, is defined:

$$\mathbf{r_k} = W_k\mathbf{z} - \mathbf{y_k}. \tag{3.8}$$

Here $k = 1 \dots c$ is the LR image index. A cost function is redefined using the elements of $\mathbf{r_k}$ in (3.9):

$$Cost = \sum_{k=1}^{c} \sum_i log(1 + |r_{ki}|), \tag{3.9}$$

where r_{ki} represents the i^{th} pixel of the k^{th} LR image. The cost can be minimized to solve for $\mathbf{z}$ via a gradient descent algorithm, where $\nabla Cost$ term is:

$$\nabla Cost = \sum_{k=1}^{c} W_k^T sign(W_k\mathbf{z} - \mathbf{y_k}) \oslash (\boldsymbol{\eta} + |W_k\mathbf{z} - \mathbf{y_k}|) \tag{3.10}$$

Here $\oslash$ represents pixel wise division, $\boldsymbol{\eta}$ is a column vector of ones, and operator $|\cdot|$ takes pixel-wise absolute value of a vector.

3.2.3 Simulation Results

In this section ℓ_1, ℓ_2, mixture of ℓ_1 and ℓ_2 norms and log-sum functions are compared as penalty functions in SR image reconstruction in (3.1). Regularization term is ignored during the simulations to simplify the discussion.

An HR image shown in Figure 4.3(e) is used to obtain 16 LR images according to the observation model of (1.2). Each image is a translated, blurred and down-sampled version of the original HR image. A 2×2 moving average filter is used for blurring the high resolution images.

Peak signal-to-noise ratio and mean-square error (MSE) are used as performance comparison metrics:

$$PSNR = 20\, log_{10} \left\{ \frac{255\, N}{||\hat{\mathbf{z}} - \mathbf{z}||} \right\}, \tag{3.11}$$

$$MSE = \left[(\hat{\mathbf{z}} - \mathbf{z})^T \cdot (\hat{\mathbf{z}} - \mathbf{z}) \right] /N, \tag{3.12}$$

where $\hat{\mathbf{z}}$ and $\mathbf{z}$ is reconstructed and the original images respectively. N is the number of image pixels. Each experiment is repeated 10 times at each noise level. The mean and the standard deviation of the metrics are given in Table 3.1. In the first experiment Gaussian white noise at various SNR levels are added to simulate the observation noise in (1.2). True motion estimates are used during reconstructions. In the second experiment, salt & pepper noise is added as the observation noise in (1.2). Again true motion estimates are used during reconstructions. In the third experiment additive noise in (1.2) is set to zero. W_k term is corrupted by adding various amounts of Gaussian noise to the motion estimates. The PSNR of the reconstructions are given in Table 3.1.

Table 3.1 shows that ℓ_2 norm outperforms other two penalty functions under additive Gaussian noise. Figure 3.2 shows that ℓ_1 norm reconstruction is visually more successful compared to ℓ_2 reconstruction under salt and pepper noise consistent with the results in Table 3.1. For the cost functions under discussion, the estimation is updated at each iteration according to the residue defined in (3.8). The gradient of the cost function in ℓ_2 norm is proportional to the magnitude of the residue. However, it is proportional to the $sign(\cdot)$ of the residue for ℓ_1 norm, (See (3.4)). Lets say during

Table 3.1: PSNR (dB) values for the reconstructed image in Figure 4.3(a) under noise.

Gaussian Noise	ℓ_1 norm	ℓ_2 norm	log-sum
40 dB	34.89 ±0.16	35.29 ±0.19	33.95 ±0.18
35 dB	34.48 ±0.15	35.09 ±0.17	33.13 ±0.13
30 dB	32.74 ±0.09	33.37 ±0.08	31.04 ±0.09
25 dB	30.53 ±0.1	31.24 ±0.12	28.22 ±0.09
% Salt & Pepper Noise	ℓ_1 norm	ℓ_2 norm	log-sum
0.1	36.63±0.96	33.56±0.73	33.07 ±1.7
0.2	35.86±1.86	32.12±0.82	31.92 ±2.51
0.3	35.48±1.46	30.61±0.81	30.14 ±1.75
0.4	35.52±1.47	30.05±0.77	30 ±1.69
0.5	35.26±1.66	29.43±1.01	28.94 ±2.27
Std. of Motion Error	ℓ_1 norm	ℓ_2 norm	log-sum
0.1	34.87 ±0.79	35.29 ±0.81	32.78 ±0.89
0.2	33.29 ±0.59	33.48 ±0.83	31.29 ±0.49
0.3	32.06 ±0.93	31.64 ±0.85	30.65 ±0.38
0.4	30.23 ±0.7	29.36 ±0.52	29.59 ±0.31

(a) ℓ_1 norm (b) ℓ_2 norm (c) $log(\cdot)$-sum function

Figure 3.1: Effect of additive Gaussian noise on reconstructions.

(a) ℓ_1 norm (b) ℓ_2 norm (c) $log(\cdot)$-sum function

Figure 3.2: Effect of salt & pepper noise on reconstructions.

the iterations 20 LR pixels are affecting a particular SR image pixel. In ℓ_2 norm case, these 20 pixels have a vote in proportion to their difference to the model (residue). This results in averaging the residue from the 20 LR pixels. In ℓ_1 norm case, only the $sign(\cdot)$ of each residue corresponding to these 20 LR pixels are affecting the fate of the SR pixel on the next iteration. This corresponds to a median operation. The salt and pepper noise introduces huge errors from certain LR image pixels, but the effect of this huge error disappears during median operation in ℓ_1 norm case. However, this huge error effects the overall average in ℓ_2 norm case.

$log(\cdot)$-sum function behaves in a way inverse to the behavior of ℓ_2 norm in terms of updating SR pixels at each iteration. Derivative of the log function, $log(x)$, is $1/x$. If the magnitude of the residue is small, its effect to the SR image pixel is higher at each iteration. This helps recovering the salt and pepper noise during reconstruction (See Figure 3.2). The mean square error (MSE) of the reconstructed image gets lower during iterations. This lowers the residue (discrepancy between the data and the model). As a results, both the gradient of the cost and the change of the reconstructed pixel increase at each iteration as seen in Figure 3.3(i). This behavior is different in ℓ_1 and ℓ_2 reconstructions, where a nice exponential decay is observed, Figure 3.3(c) and (f). The undesirable trend of $log(\cdot)$-sum function in Figure 3.3(i) is the result of the concave cost function.

Motion estimation errors in SR reconstruction behaves more like Laplacian noise in the overall model. As the motion estimation error increases, ℓ_1 norm outperforms ℓ_2 norm reconstruction. $log(\cdot)$ function reconstructions have the worst performance under both Gaussian noise and motion errors.

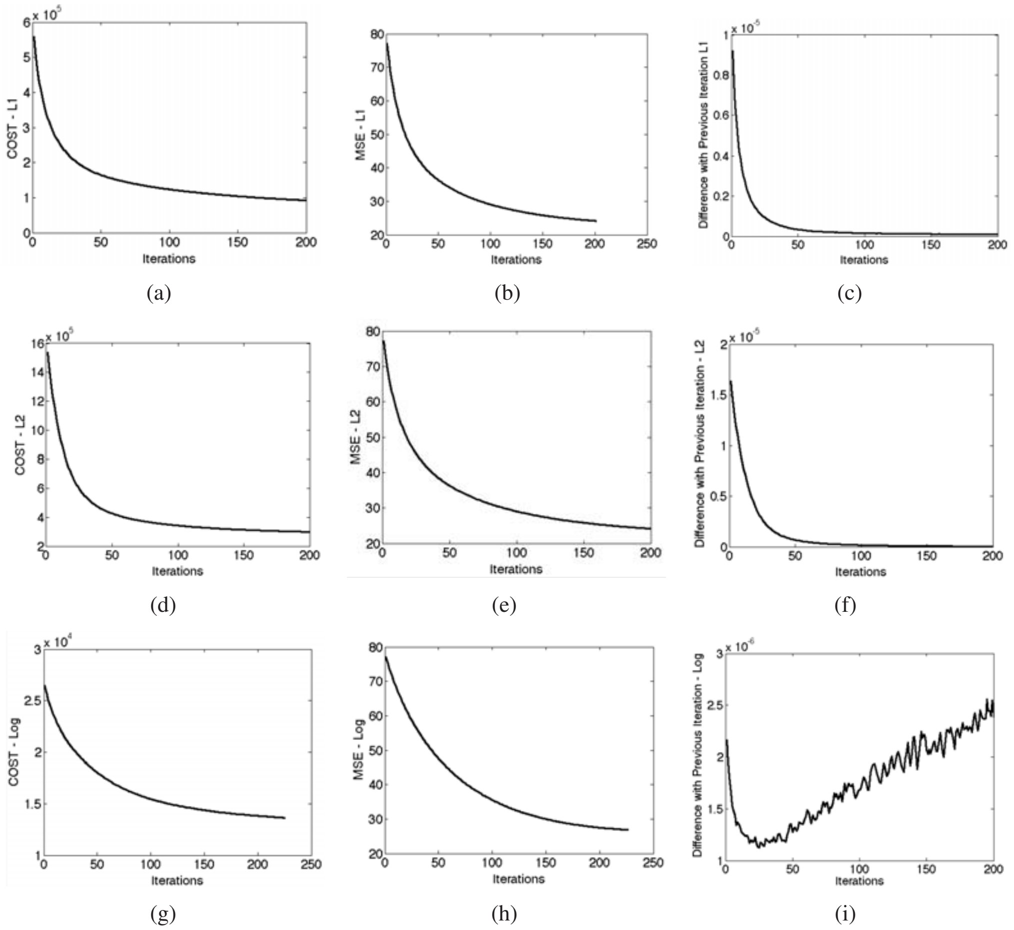


Figure 3.3: Evolution of Cost function, MSE, and average change in pixel value at each iteration for ℓ_1, ℓ_2, $log(\cdot)$-sum penalty functions.

The convergence properties of $log(\cdot)$ function are compared to ℓ_1 and ℓ_2 norms. In a standard approach at every iteration the "defined cost' is expected to drop in an exponential manner. If the cost function is properly defined, MSE value is also expected to drop at every iteration. It is observed in Figure 3.3 that MSE and Cost functions are monotonically decreasing with each iteration. However, the square of the average change at a pixel at each reconstruction is increasing unlike ℓ_1 and ℓ_2 norm reconstructions due to the concave "log-sum" cost function.

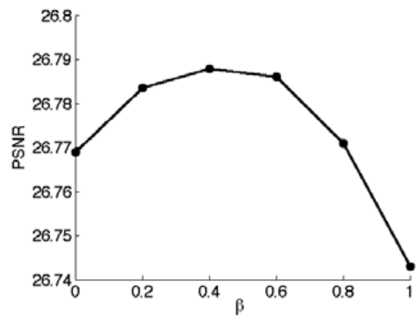


Figure 3.4: Variation of β parameter for additive Gaussian noise and motion errors.

As a final experiment, both additive Gaussian noise and motion error are added at the same time. PSNR performances as a function of β in (3.6) are investigated. Gaussian noise is added at each LR image so that the SNR value for each LR image is 25 dB. True motion vectors are corrupted by adding random Gaussian noise at a standard deviation of 0.3 pixels. Mixture of norms approach gives higher PSNR value under two different sources of noise as expected. However, the PSNR difference is not as high as to make this method an alternative in practical applications.

In this thesis, experiments are conducted in a controlled environment, i.e. i) Image sets with only global motion is selected, ii) Images that have poor motion estimation quality are discarded. Therefore, ℓ_2 norm is preferred. It should be noted that ℓ_1 norm should be preferred in a practical scenario, where the quality of motion estimation is poor or local object motion is present.

3.3 REGULARIZATION FUNCTIONS

In the following Bayesian solution, we focus on the $\Gamma(\,\cdot\,)$ function in (3.1), which controls regularization.

$$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \rho(W_k\mathbf{z}, \mathbf{y_k}) + \lambda\Gamma(\mathbf{z})$$

Natural images inherit abrupt changes within neighboring pixels, due to occlusions, shadows, and textures. $\Gamma(\,\cdot\,)$ term should be designed to model image discontinuities as well as smoothness in a natural scene. In this section, we consider ℓ_1 norm, ℓ_2 norm, $log(\cdot)$-sum function, and Huber function in the regularization term.

We restrict the distance measure in the first part of (3.1) to ℓ_2 norm to make a fair comparison and to simplify the discussion. The cost functions under consideration are:

$$
\begin{aligned}
Cost_{\ell_2} &= \sum_{k=1}^{c} \|W_k\mathbf{z} - \mathbf{y_k}\|_2^2 + \lambda\Big[\|Q_x\mathbf{z}\|_2^2 + \|Q_y\mathbf{z}\|_2^2\Big] \\
Cost_{\ell_1} &= \sum_{k=1}^{c} \|W_k\mathbf{z} - \mathbf{y_k}\|_2^2 + \lambda\Big[\|Q_x\mathbf{z}\|_1^1 + \|Q_y\mathbf{z}\|_1^1\Big]^2 \\
Cost_{huber} &= \sum_{k=1}^{c} \|W_k\mathbf{z} - \mathbf{y_k}\|_2^2 + \lambda\Big[\rho_T(Q_x\mathbf{z}) + \rho_T(Q_y\mathbf{z})\Big] \\
Cost_{logsum} &= \sum_{k=1}^{c} \|W_k\mathbf{z} - \mathbf{y_k}\|_2^2 + \lambda\left\{\sum_{i=1}^{N}\Big[log(|(Q_x\mathbf{z})_i| + 1) + log(|(Q_y\mathbf{z})_i| + 1)\Big]\right\}^2
\end{aligned}
$$

Here, Q_x and Q_y are image derivative operators in horizontal and vertical directions respectively. $Q_x\mathbf{z} = \mathbf{z} - S_x^1\mathbf{z}$ and $Q_y\mathbf{z} = \mathbf{z} - S_y^1\mathbf{z}$, where S_x^1 and S_y^1 shift the entire image by 1 pixels in horizontal and vertical directions respectively. c denotes the number of LR images and N denotes the number of pixels in the HR image. $(Q_x\mathbf{z})_i$ and $(Q_y\mathbf{z})_i$ refer to the i^{th} pixel of the derivative images. Huber function, $\rho_T(\cdot)$, is

$$
\rho_T(x) = \begin{cases} x^2 & |x| \le T \\ 2T|x| - T^2 & |x| > T \end{cases}, \tag{3.13}
$$

where T is a threshold determining the linear and quadratic regions in the cost function. If the discontinuity between neighboring pixels is less than T, a quadratic penalty is applied otherwise a less severe, linearly varying penalty is applied. The gradients of the cost functions have the following form:

$$
\begin{aligned}
\nabla Cost_{\ell_2} &= \sum_{k=1}^{c} W_k^T(W_k\mathbf{z} - \mathbf{y_k}) + \lambda\left[(I - S_x^{-1})(\mathbf{z} - S_x^1\mathbf{z}) + (I - S_y^{-1})(\mathbf{z} - S_y^1\mathbf{z})\right] \\
\nabla Cost_{\ell_1} &= \sum_{k=1}^{c} W_k^T(W_k\mathbf{z} - \mathbf{y_k}) + \lambda\left[(I - S_x^{-1})sign(\mathbf{z} - S_x^1\mathbf{z}) + (I - S_y^{-1})sign(\mathbf{z} - S_y^1\mathbf{z})\right] \cdot \Lambda(\mathbf{z}) \\
\nabla Cost_{huber} &= \sum_{k=1}^{c} W_k^T(W_k\mathbf{z} - \mathbf{y_k}) + \lambda\left[(I - S_x^{-1})CLIP(\mathbf{z} - S_x^1\mathbf{z}) + (I - S_y^{-1})CLIP(\mathbf{z} - S_y^1\mathbf{z})\right] \\
\nabla Cost_{logsum} &= \sum_{k=1}^{c} W_k^T(W_k\mathbf{z} - \mathbf{y_k}) + \lambda\left[(I - S_x^{-1})sign(\mathbf{z} - S_x^1\mathbf{z}) \oslash (I + |\mathbf{z} - S_x^1\mathbf{z}|)\right. \\
&\qquad \left. + (I - S_y^{-1})sign(\mathbf{z} - S_y^1\mathbf{z}) \oslash (I + |\mathbf{z} - S_y^1\mathbf{z}|)\right] \cdot \Phi(\mathbf{z})
\end{aligned}
$$

Here

$$\Lambda(\mathbf{z}) \triangleq ||Q_x\mathbf{z}||_1^1 + ||Q_y\mathbf{z}||_1^1$$

and

$$\Phi(\mathbf{z}) \triangleq \sum_{i=1}^{N} \Big[log(|(Q_x\mathbf{z})_i| + 1) + log(|(Q_y\mathbf{z})_i| + 1)\Big].$$

$\oslash$ represents pixel wise division, $\boldsymbol{I}$ is a column vector of ones. Operator $|\cdot|$ takes pixel-wise absolute value of a vector. Transpose of S_x^1 and S_y^1 are S_x^{-1} and S_y^{-1} respectively, and

$$CLIP(\mathbf{v}) \triangleq \begin{bmatrix} clip(v_1) \\ clip(v_2) \\ \vdots \\ clip(v_n) \end{bmatrix}. \tag{3.14}$$

Here, $clip(x) \triangleq \frac{\partial\rho_\kappa(x)}{\partial x}$ and is

$$\frac{\partial\rho_\kappa(x)}{\partial x} = \begin{cases} x & |x| \leq T \\ 2T sign(x) & |x| > T \end{cases}. \tag{3.15}$$

3.3.1 Simulation Results

An HR image shown in Figure 4.3(a) and (e) is used to obtain 16 LR images according to the observation model of (1.2). Each image is a translated, blurred and down-sampled version of the original HR image. A 2x2 moving average filter is used for blurring the high resolution images.

The first experiment investigates the effect of T in the huber function $\rho_T(\cdot)$. λ is set to 0.8, a large value, to see the effect of change in T more clearly in Figure 3.5. The edges are completely lost at $T = 50$, due to the quadratic penalty applied to the whole image. It can be observed that strong edges along the hat of lena is preserved while other edges are blurred at T = 3 and 10 in Figure 3.5. For T very large, Huber function behaves like ℓ_2 norm function and for $T = 1$ it acts similar to ℓ_1 norm function. This is due to the definition of Huber function, where a quadratic penalty is used for differences among pixels smaller than T and a fixed penalty for differences higher than T.

Figure 3.5: Effect of varying T in the huber function. From left to right T is set to 1, 3, 10, and 50.

We visually inspect the effect of blur to reconstructions for ℓ_1 norm, ℓ_2 norm, $log(\cdot)$-sum function, and Huber function, where T is set to 5. λ term is increased for each reconstruction in Figures 3.6 and 3.7 starting from (a) to (f). The differences of the regularization terms are clearer for large λ.

3.3.2 Discussion

Huber function has the characteristics of both ℓ_1 and ℓ_2 norms. However, the nonlinear behavior increases the computational complexity. Furthermore, determination of parameter T according to the image type in a heuristic way is a major drawback. ℓ_1 norm reconstructions, on the other hand, are satisfactory especially for the artificial image, in Figure 3.7. However, ℓ_1 norm adds a *cartoon* effect to the reconstructions for natural images. Visual inspection of log-sum function reveals similarity to ℓ_1 norm reconstructions. Unfortunately, the computational complexity is high and convergence is slow compared to other norms. The proposed methods in the next sections employs ℓ_1 norm, ℓ_2 norm, and Huber function for the regularization term.

(a) $\lambda = 0$

(b) $\lambda = 0.2$

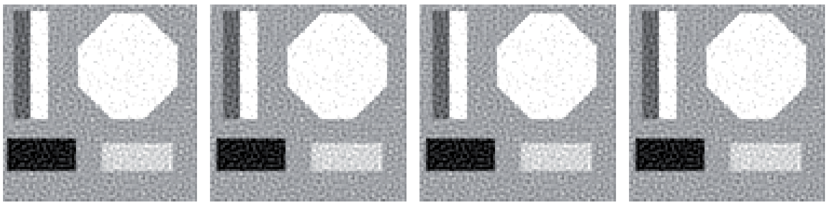
(c) $\lambda = 0.4$

(d) $\lambda = 0.6$

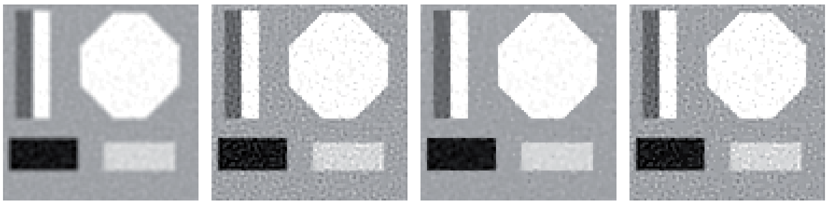
(e) $\lambda = 0.8$

(f) $\lambda = 0.9$

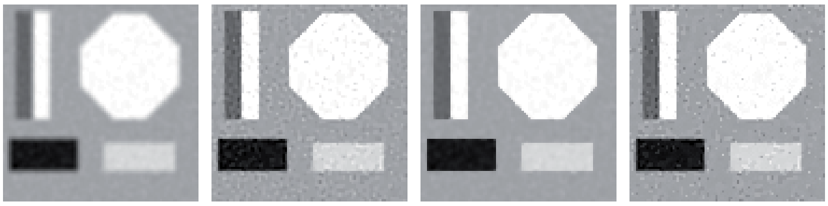
Figure 3.6: Effect of increasing λ. Starting from the first to 4^{th} column, reconstructions with ℓ_2 norm, ℓ_1 norm, Huber function, and $log(\cdot)$-sum function in the regularization term are given.



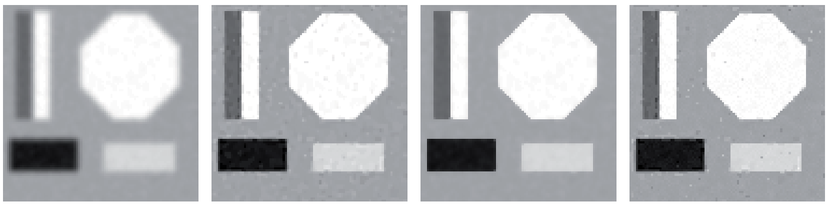
(a) $\lambda = 0$



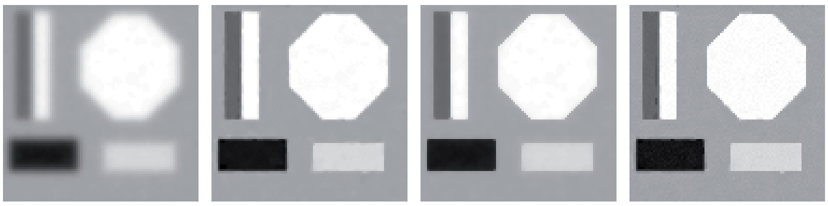
(b) $\lambda = 0.2$



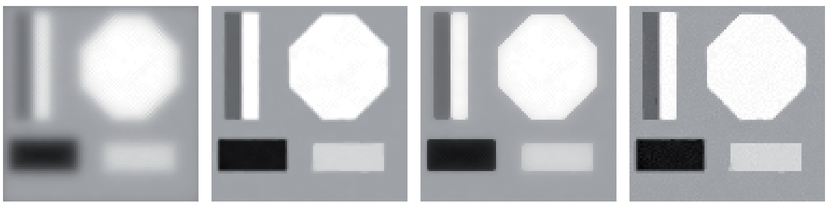
(c) $\lambda = 0.4$



(d) $\lambda = 0.6$



(e) $\lambda = 0.8$



(f) $\lambda = 0.9$

Figure 3.7: Effect of increasing λ. Starting from the first to 4^{th} column, reconstructions with ℓ_2 norm, ℓ_1 norm, Huber function, and $log(\cdot)$-sum function in the regularization term are given.

CHAPTER 4

GRADIENT DIRECTION ADAPTIVE SUPER RESOLUTION

4.1 INTRODUCTION

In this chapter, a maximum *a posteriori* (MAP) based SR estimator is proposed, where image gradient direction and magnitude are used for aiding the SR image pixel estimation. The aim of this method is to improve edge preservation in MAP estimators. The proposed method is based on the following observation model given in Chapter 1.

$$\mathbf{y_k} = W_k \cdot \mathbf{z} + \mathbf{n_k} \qquad k = 1 \cdots c, \tag{4.1}$$

where $\mathbf{z}$ and $\mathbf{y_k}$ are the HR image and the k^{th} LR image respectively, and $\mathbf{n_k}$ is the observation noise, where both Gaussian and Laplacian distributions are considered. Each LR image, $\mathbf{y}_k$, is obtained from $\mathbf{z}$ through warping, blurring, and down-sampling operations modeled in the system matrix $\mathbf{W}_k$, of size $LN \times m^2LN$.

4.2 PROPOSED DIRECTIONAL REGULARIZATION

The main focus of this chapter is edge preservation in SR image reconstruction. We start with a brief comparison of three state-of-the-art SR methods and construct a direction adaptive regularizer in relation to these methods. In [13], observation noise in (4.1) is assumed to have i. i. d. Gaussian distribution and image prior term is based on GMRF. The MAP estimator under these conditions is

$$\hat{\mathbf{z}}_{MAP} = \underset{z}{argmin}\left(\sum_{k=1}^{n}||W_k\mathbf{z} - \mathbf{y_k}||_2^2 + \lambda \sum_{i\sim j}(z_i - z_j)^2\right), \tag{4.2}$$

here $i \sim j$ refers to the indices of neighboring pairs, and z_i and z_j are the i^{th} and j^{th} pixels respectively. The well known disadvantage of this type of regularization is over-smoothing. This problem is addressed in [40], where edge preserving regularization is applied using Huber Markov Random Fields as image priors. The nonlinear nature of this regularizer applies fewer penalties to strong edges while smoothing low amplitude noise. Replacing the GMRF prior with HMRF in (4.2) results:

$$\hat{\mathbf{z}}_{MAP} = \underset{z}{argmin}\left(\sum_{k=1}^{n}||W_k\mathbf{z} - \mathbf{y_k}||_2^2 + \lambda\sum_{i\sim j}\rho_T(z_i - z_j)\right). \tag{4.3}$$

where $\rho_T(\cdot)$ is defined in 3.13.

Recently, the distribution of the additive noise in the SR observation model is assumed to be Laplacian [25]. This has introduced ℓ_1 norm to be used in the minimization problem, which yields very successful edge preservation. The robustness of this method is also achieved by using bilateral total variation (BTV) regularization [25], where the differences within the neighboring pixels are penalized by ℓ_1 norm instead of ℓ_2 norm as used in [13], [40].

$$\hat{\mathbf{z}} = \underset{\mathbf{z}}{argmin}\sum_{k=1}^{c}||\mathbf{W}_k\mathbf{z} - \mathbf{y}_k||_1^1 + \lambda\sum_{q=-2}^{2}\sum_{p=-2}^{2}\alpha^{|q|+|p|}||\mathbf{z} - S_h^pS_v^q\mathbf{z}||_1^1 \tag{4.4}$$

In this equation, S_h^p and S_v^p operators shift the image by p pixels horizontally and vertically respectively, and α is a decay term to decrease the effect of distant pixels in the difference operation.

Although edge preservation is achieved in (4.3),([40]), (4.4)([25]), these methods do not use the direction of the local variation for further optimizing the regularization. A more computationally cumbersome approach for solving the over-smoothing problem in SR is using anisotropic diffusion in regularization [63]. In this approach the regularization parameter in the MAP estimator is replaced by a modified version of Perona-Malik diffusion operator, [64], such that smoothing works along the edge. Direction of the gradient has also been used in single frame image interpolation problem. In [65], multiple-direction wavelets, directionlets, are used to do directionally adaptive interpolation. In [66], HR image local covariance estimates, which comprise edge orientation information around an edge pixel, are used to adapt interpolation through edges.

In the proposed method, the gradient amplitude and direction at each pixel is used to improve the regularization in the MAP solution. The proposed method is based on the MAP formulation expressed in (3.2). The regularization term $\Gamma(\cdot)$ is modified to include both the gradient amplitude and gradient direction into the solution. The main idea is to run a regularization Kernel perpendicular to the gradient direction as depicted in Figure 4.1. The parallel arrows show the direction of regularization for 4 different directions. Regularization works on neighboring pixels that are aligned perpendicular to the gradient direction. This is formulated in (4.5).

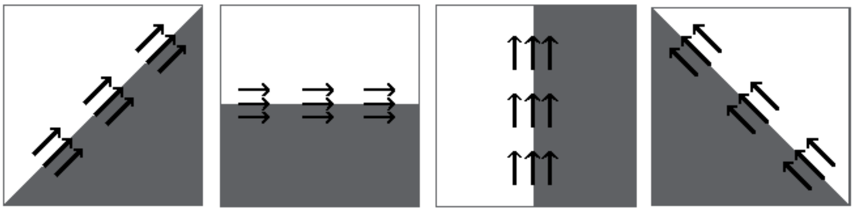


Figure 4.1: Four images with white to gray transitions are shown. The arrows are perpendicular to gradient direction.

$$\Gamma_{DIR}(\mathbf{z}) = \begin{cases} \sum_{p=-2}^{2} \alpha^{|p|}\rho(\mathbf{z}^n - S_v^p\mathbf{z}^n) & \theta \in \left\{(\frac{15\pi}{8}, \frac{\pi}{8}] \cup (\frac{7\pi}{8}, \frac{9\pi}{8}]\right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|}\rho(\mathbf{z}^n - S_h^p S_v^{-p}\mathbf{z}^n) & \theta \in \left\{(\frac{\pi}{8}, \frac{3\pi}{8}] \cup (\frac{9\pi}{8}, \frac{11\pi}{8}]\right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|}\rho(\mathbf{z}^n - S_h^p\mathbf{z}^n) & \theta \in \left\{(\frac{3\pi}{8}, \frac{5\pi}{8}] \cup (\frac{11\pi}{8}, \frac{13\pi}{8}]\right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|}\rho(\mathbf{z}^n - S_h^p S_v^p\mathbf{z}^n) & \theta \in \left\{(\frac{5\pi}{8}, \frac{7\pi}{8}] \cup (\frac{13\pi}{8}, \frac{15\pi}{8}]\right\} \quad |\vec{g}| > T \\ \sum_{q=-2}^{2}\sum_{p=-2}^{2} \alpha^{|p|+|q|}\rho(\mathbf{z}^n - S_h^p S_v^q\mathbf{z}^n) & \qquad\qquad\qquad\qquad\qquad\quad |\vec{g}| < T \end{cases} \tag{4.5}$$

Similar to the regularizers previously discussed, the differences between neighboring pixels are penalized by a symmetric nonnegative function $\rho(\cdot)$, and α is a weighting term in the difference operation.

$\vec{g}$ is the gradient vector defined for each pixel in the image. θ is the gradient angle for each gradient vector. A Scharr operator in the following form is used for obtaining the gradient at each pixel.

$$G_x = I * \begin{bmatrix} -1 & 0 & 1 \\ -5 & 0 & 5 \\ -1 & 0 & 1 \end{bmatrix}, \qquad G_y = I * \begin{bmatrix} -1 & -5 & -1 \\ 0 & 0 & 0 \\ 1 & 5 & 1 \end{bmatrix} \tag{4.6}$$

and,

$$G_{mag}(m_1, m_2) = \sqrt{G_x^2(m_1, m_2) + G_y^2(m_1, m_2)}, \qquad G_\theta = atan\frac{G_x(m_1, m_2)}{G_y(m_1, m_2)}, \tag{4.7}$$

where I is an image in matrix form. G_x and G_y are images, where each pixel is an approximation for gradients in x and y directions respectively, and $*$ is convolution operator. Magnitude and angle of a gradient at an image pixel $[m_1, m_2]$ is $G_{mag}(m_1, m_2)$ and $G_\theta(m_1, m_2)$ respectively. Gradient angle θ is quantized to four directions given in (4.5). T is a threshold for eliminating weak gradients. A gradient map is obtained out of the reconstructed image as shown in Figure 4.2.

Figure 4.2: (a) Zebra image used in simulations. (b) Gradient map of Zebra. White regions show pixels with small gradient activity. Four directions are coded as shown on the right.

Accordingly, if there is a small gradient activity in the region the regularization term works in all directions the same way as Tikhonov [13] or BTV type regularizers [25], depending on the selection of $\rho(\cdot)$. If the gradient has a vertical direction, the regularization is towards horizontal direction. Similarly if the gradient is near $-\pi/4$ to $3\pi/4$ diagonal direction, the regularizer works along the perpendicular direction, $\pi/4$ to $5\pi/4$. Same idea applies to other directions.

At every iteration the gradient map is updated using the current reconstructed image. The most general case for the proposed method can be formulated as

$$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} ||W_k\mathbf{z} - \mathbf{y_k}||_R^R + \lambda\Gamma_{DIR}(\mathbf{z}). \tag{4.8}$$

Three variations of the proposed regularization are formulated in the next sections.

4.2.1 Directional Huber-Markov Random Field Method

In this method, additive observation noise is assumed to have identical independent Gaussian distribution. MAP estimate for directional HMRF method is given as follows:

$$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} ||W_k\mathbf{z} - \mathbf{y_k}||_2^2 + \lambda\Gamma_{DIR}(\mathbf{z}) \tag{4.9}$$

The penalty function, $\rho(\cdot)$, in (4.5) is selected as Huber function (3.13), [40], for further edge-preservation. According to (4.5), if the gradient is lower than a certain threshold, (4.9) reduces to a similar form of (4.3) ([40]). (4.9) can be solved iteratively using the gradient descent method, $\mathbf{z}^{n+1} = \mathbf{z}^n - \beta \cdot \nabla Cost$, as follows.

$$\hat{\mathbf{z}}^{n+1} = \hat{\mathbf{z}}^n - \beta\left(\sum_{k=1}^{c}\left(W_k^T\right)\cdot\left(W_k\hat{\mathbf{z}}^n - \mathbf{y}_k\right) + \lambda\nabla\Gamma_{DIR}(\hat{\mathbf{z}}^n)\right) \tag{4.10}$$

The derivative of the huber function is also known as the clip function $\frac{\partial\rho_\kappa(x)}{\partial x} \triangleq clip(x)$. Clip function for column vector $\mathbf{v}$ with elements $v_i \quad i = 1\cdots n$ is defined as $CLIP(\mathbf{v})$ in (3.14). Substituting (3.14) in $\nabla\Gamma_{DIR}$ term of directional HMRF solution in (4.10) results;

$$\nabla\Gamma_{DIR}(\mathbf{z}) = \begin{cases} \sum_{p=-2}^{2} \alpha^{|p|}(I - S_v^{-p})\cdot CLIP(\mathbf{z}^n - S_v^p\mathbf{z}^n) & \theta \in \left\{(\frac{15\pi}{8},\frac{\pi}{8}] \cup (\frac{7\pi}{8},\frac{9\pi}{8}]\right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|}(I - S_v^{-p}S_h^{p})\cdot CLIP(\mathbf{z}^n - S_h^{-p}S_v^p\mathbf{z}^n) & \theta \in \left\{(\frac{\pi}{8},\frac{3\pi}{8}] \cup (\frac{9\pi}{8},\frac{11\pi}{8}]\right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|}(I - S_h^{-p})\cdot CLIP(\mathbf{z}^n - S_h^p\mathbf{z}^n) & \theta \in \left\{(\frac{3\pi}{8},\frac{5\pi}{8}] \cup (\frac{11\pi}{8},\frac{13\pi}{8}]\right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|}(I - S_v^{-p}S_h^{-p})\cdot CLIP(\mathbf{z}^n - S_h^pS_v^p\mathbf{z}^n) & \theta \in \left\{(\frac{5\pi}{8},\frac{7\pi}{8}] \cup (\frac{13\pi}{8},\frac{15\pi}{8}]\right\} \quad |\vec{g}| > T \\ \sum_{q=-2}^{2}\sum_{p=-2}^{2} \alpha^{|p|+|q|}(I - S_v^{-q}S_h^{-p})CLIP(\mathbf{z}^n - S_h^pS_v^q\mathbf{z}^n) & |\vec{g}| < T \end{cases} \tag{4.11}$$

4.2.2 Directional Bilateral Total Variation Method under Laplacian Noise

In this section, $\rho(\cdot)$ in the regularization term in (4.5) is selected as $\|\cdot\|_1^1$. MAP estimate for directional BTV assuming Laplacian distribution for observation noise is

$$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \|W_k \mathbf{z} - y_k\|_1^1 + \lambda \Gamma_{DIR}(\mathbf{z}), \tag{4.12}$$

where

$$\Gamma_{DIR}(\mathbf{z}) = \begin{cases} \sum_{p=-2}^{2} \alpha^{|p|} \|(\mathbf{z}^n - S_v^p \mathbf{z}^n)\|_1^1 & \theta \in \left\{ (\frac{15\pi}{8}, \frac{\pi}{8}] \cup (\frac{7\pi}{8}, \frac{9\pi}{8}] \right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|} \|(\mathbf{z}^n - S_h^{-p} S_v^p \mathbf{z}^n)\|_1^1 & \theta \in \left\{ (\frac{\pi}{8}, \frac{3\pi}{8}] \cup (\frac{9\pi}{8}, \frac{11\pi}{8}] \right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|} \|(\mathbf{z}^n - S_h^p \mathbf{z}^n)\|_1^1 & \theta \in \left\{ (\frac{3\pi}{8}, \frac{5\pi}{8}] \cup (\frac{11\pi}{8}, \frac{13\pi}{8}] \right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|} \|(\mathbf{z}^n - S_h^p S_v^p \mathbf{z}^n)\|_1^1 & \theta \in \left\{ (\frac{5\pi}{8}, \frac{7\pi}{8}] \cup (\frac{13\pi}{8}, \frac{15\pi}{8}] \right\} \quad |\vec{g}| > T \\ \sum_{q=-2}^{2} \sum_{p=-2}^{2} \alpha^{|p|+|q|} \|(\mathbf{z}^n - S_h^p S_v^q \mathbf{z}^n)\|_1^1 & |\vec{g}| < T \end{cases} \tag{4.13}$$

If the gradient values are smaller than a certain threshold, resultant Γ_{DIR} term reduces to the BTV expression given in [25].

(4.12) can be solved iteratively using gradient descent method as follows:

$$\hat{\mathbf{z}}^{n+1} = \hat{\mathbf{z}}^n - \beta \left(\sum_{k=1}^{c} \left(W_k^T \right) \cdot sign\left(W_k \hat{\mathbf{z}}^n - \mathbf{y}_k \right) + \lambda \nabla \Gamma_{DIR}(\hat{\mathbf{z}}^n) \right) \tag{4.14}$$

$\nabla\Gamma_{DIR}$ term for directional BTV solution in (4.14) is;

$$\nabla\Gamma_{DIR}(\mathbf{z}) = \begin{cases} \sum_{p=-2}^{2} \alpha^{|p|} (I - S_v^{-p}) \cdot sign(\mathbf{z}^n - S_v^p \mathbf{z}^n) & \theta \in \left\{ (\frac{15\pi}{8}, \frac{\pi}{8}] \cup (\frac{7\pi}{8}, \frac{9\pi}{8}] \right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|} (I - S_v^{-p} S_h^{p}) \cdot sign(\mathbf{z}^n - S_h^{-p} S_v^p \mathbf{z}^n) & \theta \in \left\{ (\frac{\pi}{8}, \frac{3\pi}{8}] \cup (\frac{9\pi}{8}, \frac{11\pi}{8}] \right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|} (I - S_h^{-p}) \cdot sign(\mathbf{z}^n - S_h^p \mathbf{z}^n) & \theta \in \left\{ (\frac{3\pi}{8}, \frac{5\pi}{8}] \cup (\frac{11\pi}{8}, \frac{13\pi}{8}] \right\} \quad |\vec{g}| > T \\ \sum_{p=-2}^{2} \alpha^{|p|} (I - S_v^{-p} S_h^{-p}) \cdot sign(\mathbf{z}^n - S_h^p S_v^p \mathbf{z}^n) & \theta \in \left\{ (\frac{5\pi}{8}, \frac{7\pi}{8}] \cup (\frac{13\pi}{8}, \frac{15\pi}{8}] \right\} \quad |\vec{g}| > T \\ \sum_{q=-2}^{2} \sum_{p=-2}^{2} \alpha^{|p|+|q|} (I - S_v^{-q} S_h^{-p}) sign(\mathbf{z}^n - S_h^p S_v^q \mathbf{z}^n) & |\vec{g}| < T \end{cases} \tag{4.15}$$

4.2.3 Directional Bilateral Total Variation Method under Gaussian Noise

If noise is assumed to have Gaussian distribution and regularization is selected as in (4.13), then the MAP solution is

$$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} ||W_k\mathbf{z} - \mathbf{y}_k||_2^2 + \lambda \left[\Gamma_{DIR}(\mathbf{z})\right]^2 \tag{4.16}$$

The iterative solution is

$$\hat{\mathbf{z}}^{n+1} = \hat{\mathbf{z}}^n - \beta \left(\sum_{k=1}^{c} \left(W_k^T\right) \cdot \left(W_k\hat{\mathbf{z}}^n - \mathbf{y}_k\right) + \lambda \nabla\Gamma_{DIR}(\hat{\mathbf{z}}^n) \cdot \Gamma_{DIR}(\hat{\mathbf{z}}^n) \right), \tag{4.17}$$

where Γ_{DIR} term is the same as in (4.15).

4.3 SIMULATION RESULTS

The performance of the proposed methods are evaluated through simulations. HR images shown in Figure 4.3 are used to obtain three LR image sets according to the observation model of (4.1). Each set consists of 8 LR images, which are translated, blurred and down-sampled by a factor of two. A 2x2 moving average filter is used for blurring the high resolution images. Gaussian white noise at SNR level of 25 dB is added to simulate the observation noise in (4.1). Original images and corresponding LR images used in reconstruction is given in Figure 4.3. True motion information is used during reconstructions.

Proposed methods are compared against bicubic interpolation and edge preserving methods given in [39, 40, 25]. Method described in [39] is a single frame interpolation method with a regularization term comprising a Huber function. The abbreviations of the methods along with their formulations is summarized in Table 4.1. The proposed methods are denoted by MTD1, MTD2 and MTD3. SR method with a GMRF prior is also added to the simulations, where the regularization term in (4.2) is approximated by $||C\mathbf{z}||_2^2$. Here, C is a matrix operator in lexicographic form, which is equivalent to a Laplace operator with parameters [1 1 1 ; 1 $-$8 1 ; 1 1 1].

Table 4.1: Reconstruction methods in simulations.

Abbreviation	Equation	Reference
BICUBIC	Reference frame is interpolated using bi-cubic interpolation	
INT HUBER	$\hat{\mathbf{z}} = \underset{z}{argmin} \, \|\|W_k\mathbf{z} - \mathbf{y}_1\|\|_2^2 + \lambda \sum_{q=-1}^{1} \sum_{p=-1}^{1} \alpha^{\|p\|+\|q\|} \rho_\kappa \left(z - S_x^p S_y^q z \right)$	[39]
GMRF	$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \|\|W_k\mathbf{z} - \mathbf{y}_k\|\|_2^2 + \lambda \|\|C\mathbf{z}\|\|_2^2$	[13]
RSR	$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \|\|W_k\mathbf{z} - \mathbf{y}_k\|\|_1^1 + \lambda \sum_{q=-2}^{2} \sum_{p=-2}^{2} \alpha^{\|p\|+\|q\|} \|\|z - S_x^p S_y^q z\|\|_1^1$	[25]
HUBER	$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \|\|W_k\mathbf{z} - \mathbf{y}_k\|\|_2^2 + \lambda \sum_{q=-1}^{1} \sum_{p=-1}^{1} \alpha^{\|p\|+\|q\|} \rho_\kappa \left(z - S_x^p S_y^q z \right)$	[40]
MTD1	$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \|\|W_k\mathbf{z} - \mathbf{y}_k\|\|_2^2 + \lambda \Gamma_{DIR}(\mathbf{z}) \quad \rho_\kappa(\cdot) = HuberF(\cdot)$	Proposed method in Section 4.2.1
MTD2	$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \|\|W_k\mathbf{z} - \mathbf{y}_k\|\|_1^1 + \lambda \Gamma_{DIR}(\mathbf{z}) \quad \rho(\cdot) = \|\|\cdot\|\|_1^1$	Proposed method in Section 4.2.2
MTD3	$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \|\|W_k\mathbf{z} - \mathbf{y}_k\|\|_2^2 + \lambda \left[\Gamma_{DIR}(\mathbf{z}) \right]^2 \quad \rho(\cdot) = \|\|\cdot\|\|_1^1$	Proposed method in Section 4.2.3

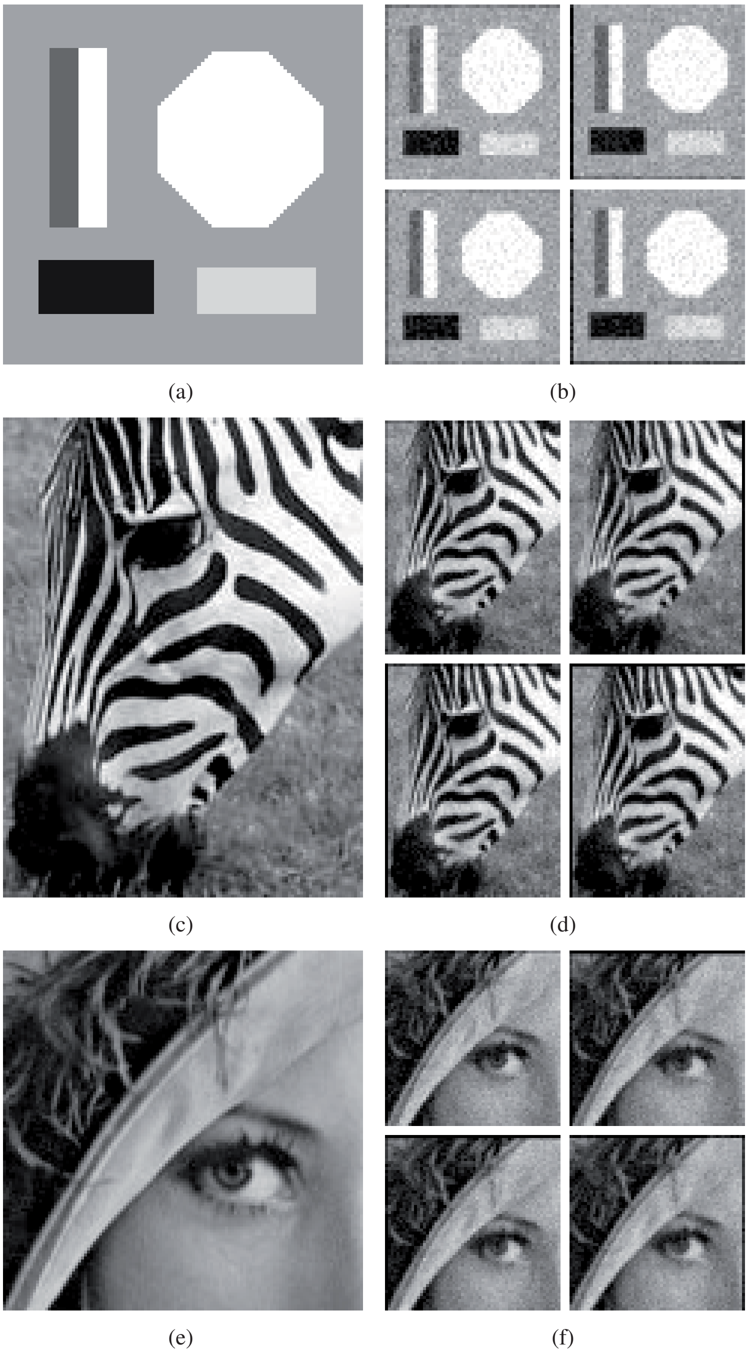
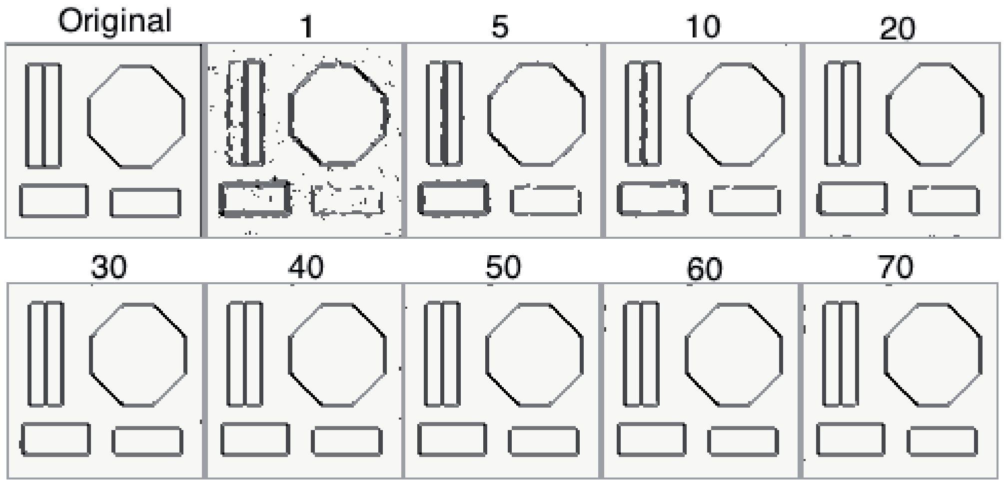


Figure 4.3: Original HR images and 4 of 8 corresponding LR images.

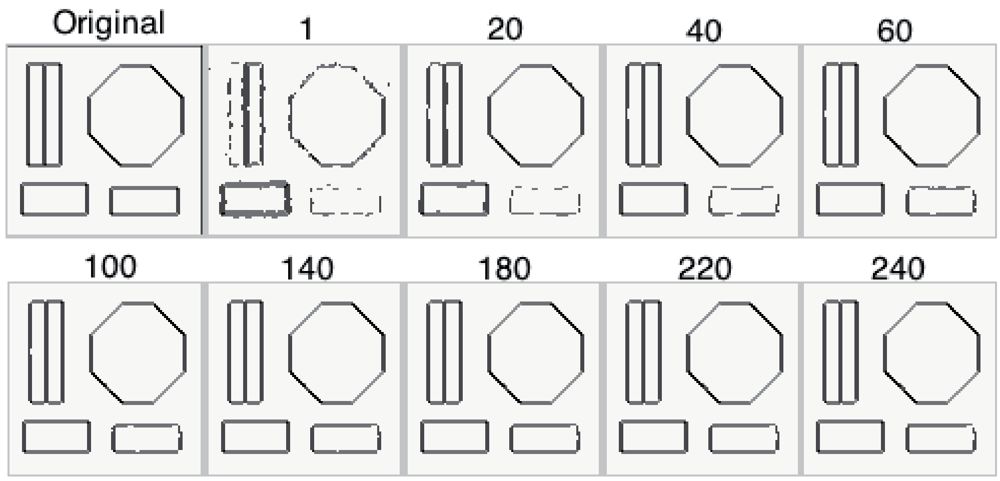
4.3.1 Gradient Map Estimation

It is assumed that the gradient map is fixed during the derivations of the proposed methods. However, the gradient map is estimated out of the reconstructed image at each iteration. The gradient map modifies the reconstructed image and the resultant reconstructed image modifies the gradient map for the next iteration. The validity of fixed gradient map assumption is tested during the simulations. The evolution of the gradient map through iterations are shown in Figures 4.4, 4.5, and 4.6. Gradient map obtained from the original image is also shown in these figures for comparison. It is observed that after early iterations the map converges. Although the map is quite similar to the original gradient map, the map slightly varies for the three methods.

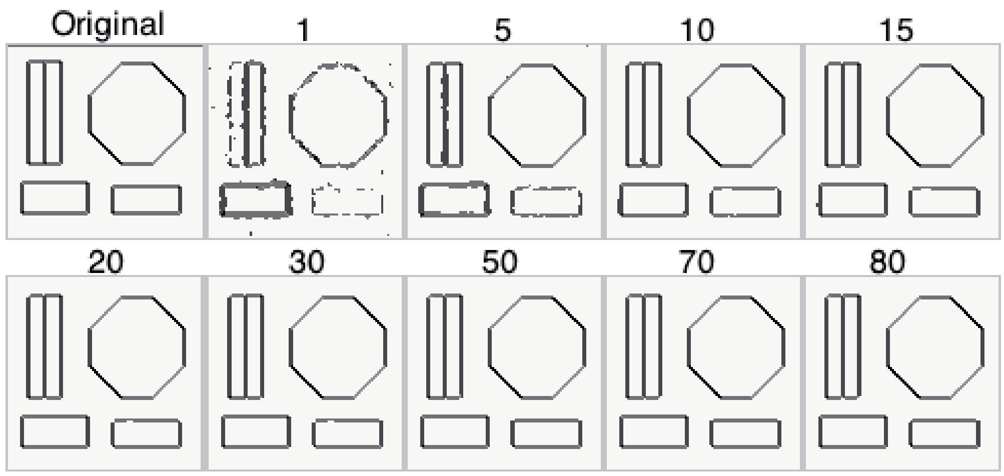
Total number of edge pixels, in other words, number of pixels whose gradient value is greater than the threshold T of Equation 4.5 is counted for every iteration. Percentage of total edge pixels versus the iteration number is plotted in Figure 4.7. For all three methods, although the number of total edge pixels are different, the overall percentage does not vary after early iterations in accordance with Figures 4.4, 4.5, and 4.6. Fixed gradient map assumption is also necessary for convergence analysis. Under this assumption, the proposed cost function is convex following the arguments discussed in [25, 40].



(a) Directional HMRF Method in Section 4.2.1

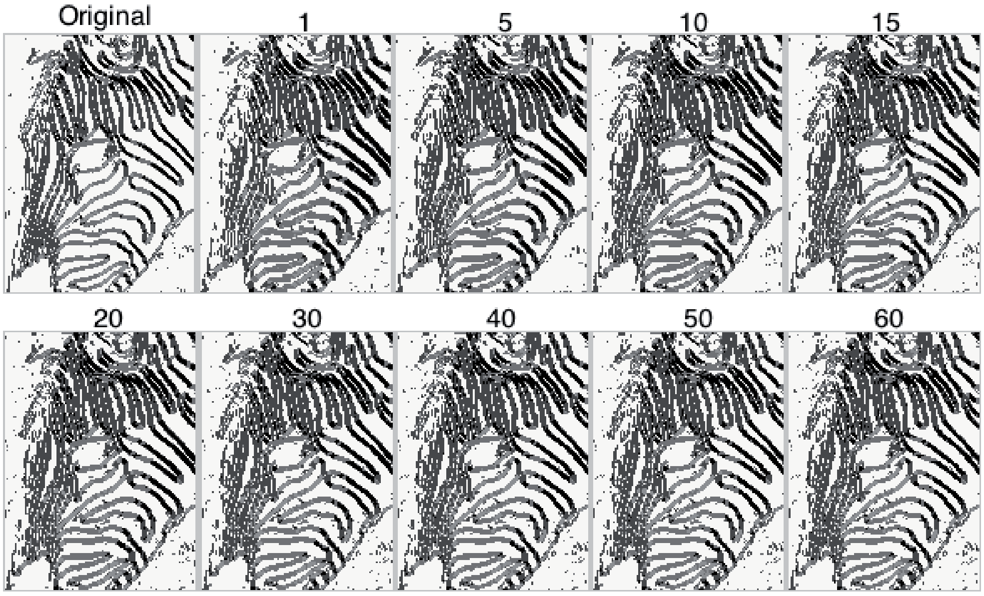


(b) Directional BTV with Laplacian Noise Assumption in Section 4.2.2

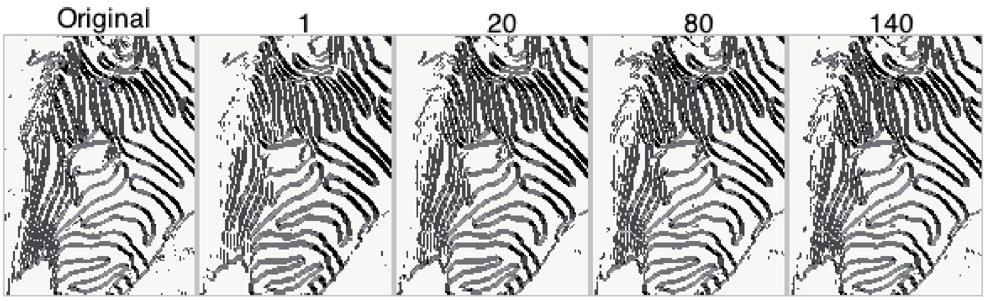


(c) Directional BTV with Gaussian Noise Assumption in Section 4.2.3

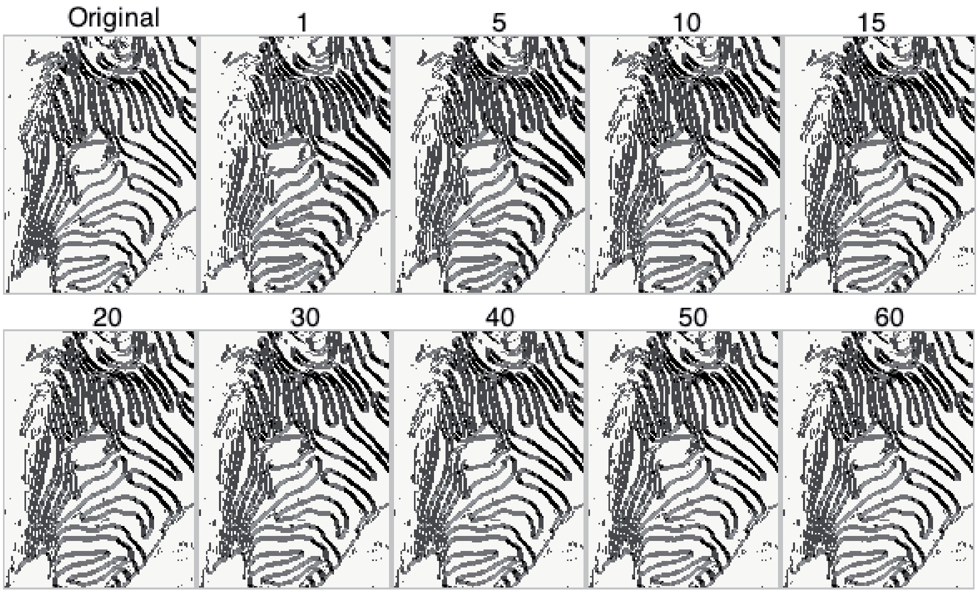
Figure 4.4: Evolution of gradient map estimate through iterations. The iteration number is tagged at every map image.



(a) Directional HMRF Method in Section 4.2.1

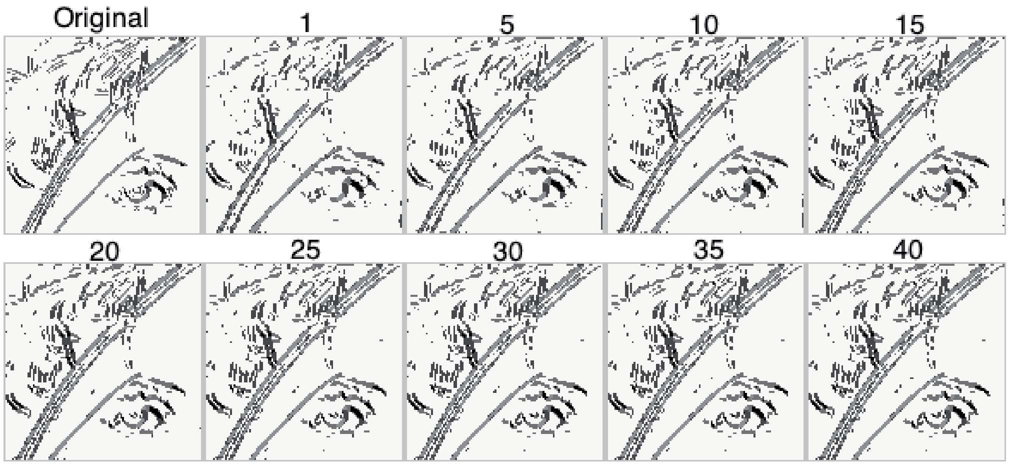


(b) Directional BTV with Laplacian Noise Assumption in Section 4.2.2

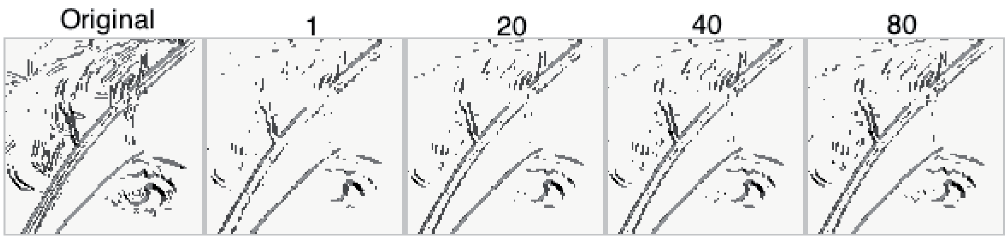


(c) Directional BTV with Gaussian Noise Assumption in Section 4.2.3

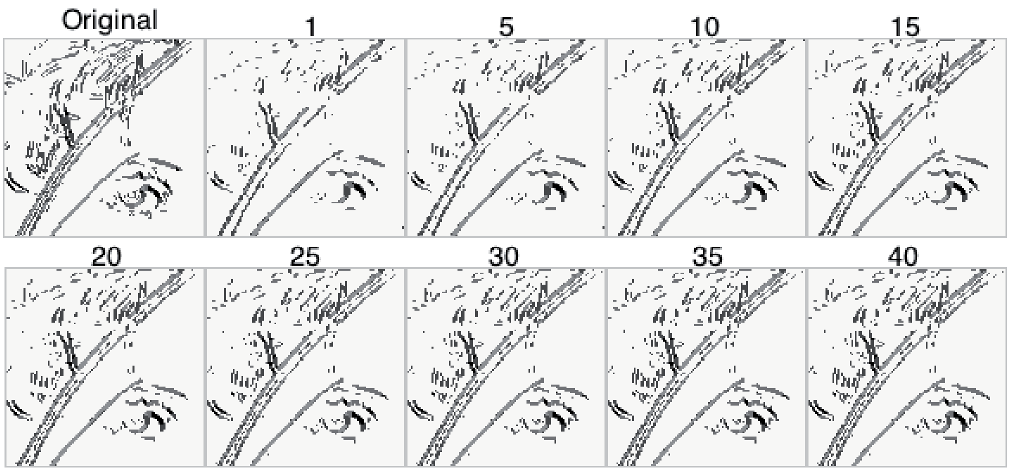
Figure 4.5: Evolution of gradient map estimate through iterations. The iteration number is tagged at every map image.



(a) Directional HMRF Method in Section 4.2.1

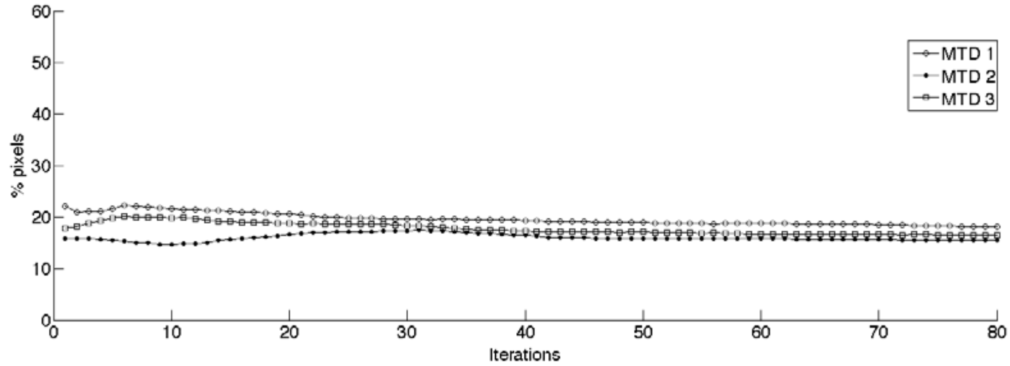


(b) Directional BTV with Laplacian Noise Assumption in Section 4.2.2

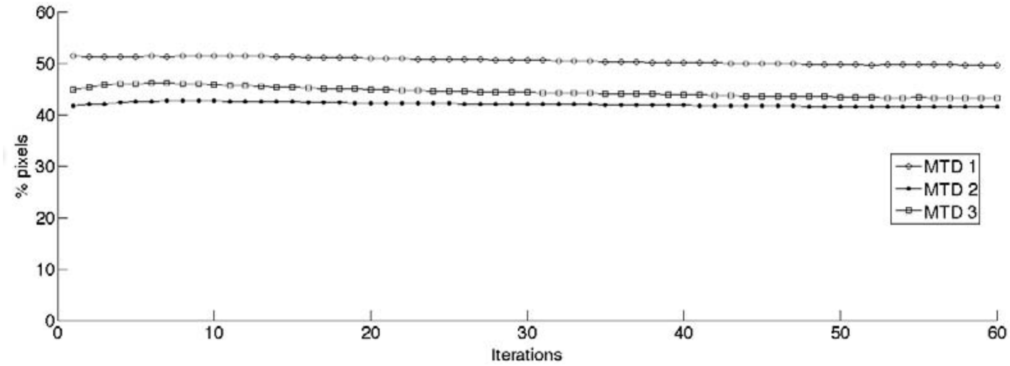


(c) Directional BTV with Gaussian Noise Assumption in Section 4.2.3

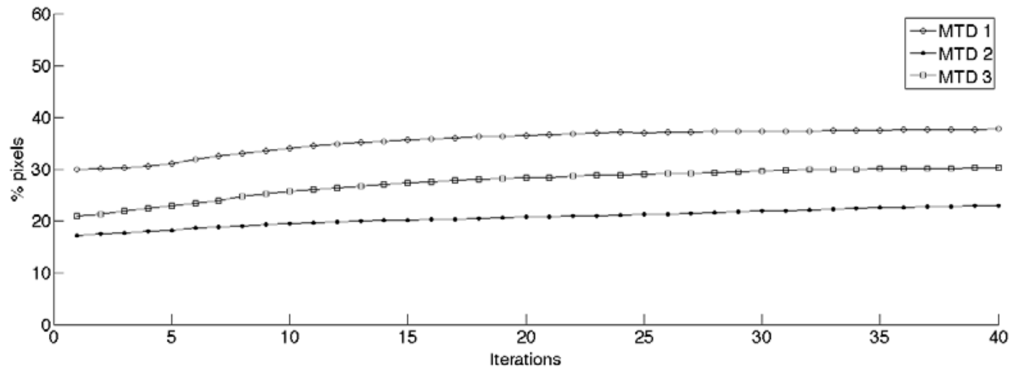
Figure 4.6: Evolution of gradient map estimate through iterations. The iteration number is tagged at every map image.



(a) Image set in Figure 4.3(a)



(b) Image set in Figure 4.3(b)

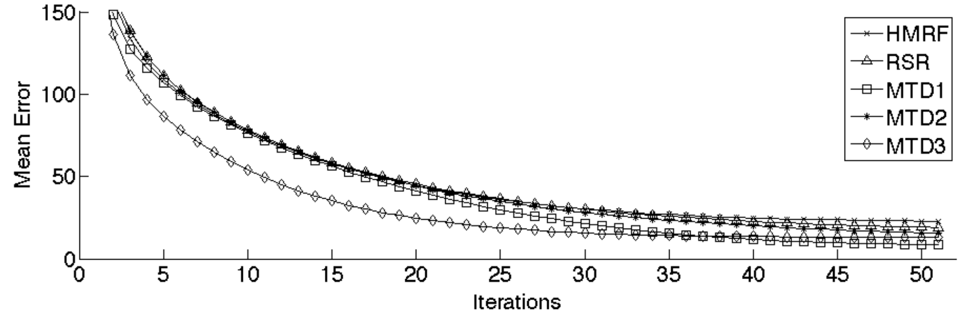


(c) Image set in Figure 4.3(c)

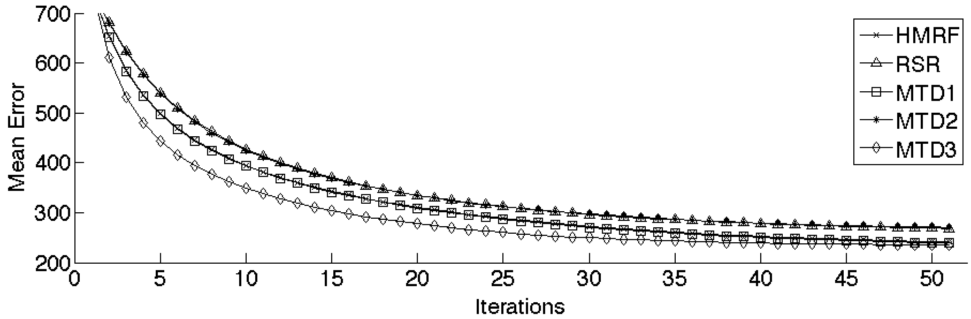
Figure 4.7: Percentage of pixels whose gradient value is greater than the threshold T of Equation 4.5 is plotted for every iteration.

4.3.2 Convergence of the Proposed Methods

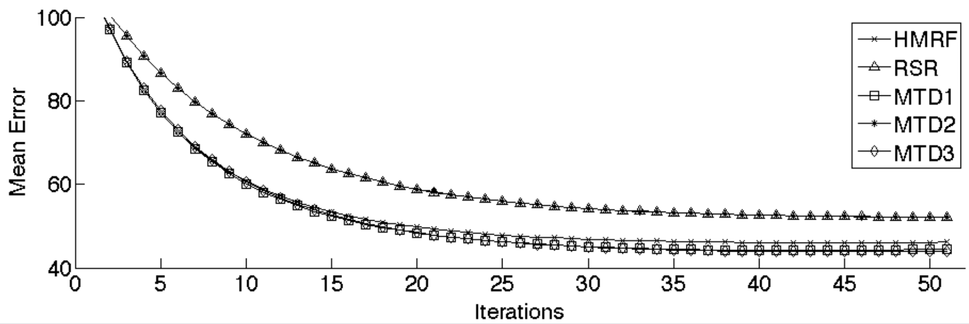
The convergence of each method is investigated experimentally. Number of iterations and the step size parameter for the gradient descent is determined according to the convergence of the algorithms. In Figure 4.8, the change in mean square error at each iteration is plotted. As seen, the methods converge for each image set.



(a) Image set in Figure 4.3(a)



(b) Image set in Figure 4.3(b)



(c) Image set in Figure 4.3(c)

Figure 4.8: Mean error at each iteration.

4.3.3 Performance Comparison

The quantitative performance comparison is achieved using PSNR (3.11) and mean structural similarity index measure (SSIM) measurements. SSIM is also used in addition to the PSNR measures, since human visual perception is more sensitive to structural information in a scene [67]. SSIM is calculated as follows:

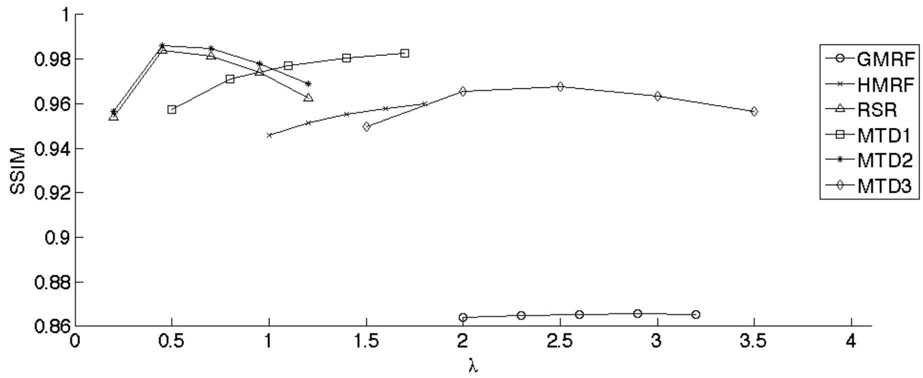
$$SSIM(x,y) = \frac{\left(2\mu_x\mu_y + (255 \cdot K_1)^2\right)\left(2\sigma_{xy} + (255 \cdot K_2)^2\right)}{\left(\mu_x^2 + \mu_y^2 + (255 \cdot K_1)^2\right)\left(\sigma_x^2 + \sigma_y^2 + (255 \cdot K_2)^2\right)}, \tag{4.18}$$

where μ_x and μ_y are the mean value of the pixels in a window for images x and y. σ_x^2, σ_y^2, and σ_{xy} are the variance of x, y and covariance of x and y respectively. K_1 and K_2 are two constants set to 0.01 and 0.03 respectively in [67]. This calculation is repeated at each pixel within a windowed neighborhood. For each image the mean values are calculated to obtain mean SSIM. An SSIM value close to one refers to perfect match.

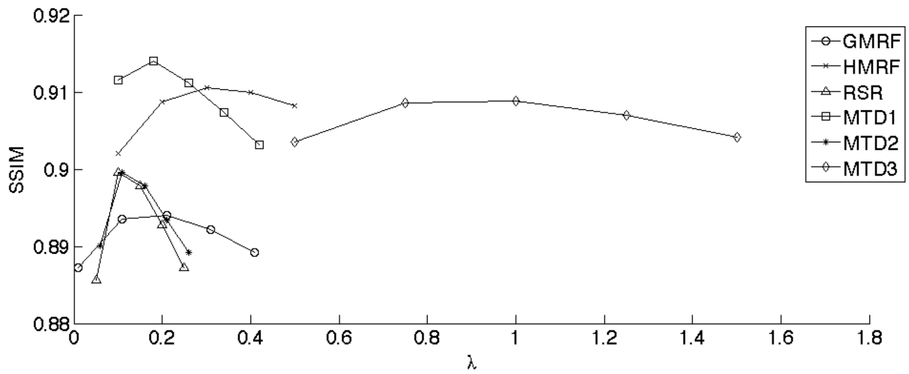
To have a fair comparison, the optimal value of λ is searched for the highest SSIM score heuristically for each algorithm in Figure 4.9. The effect of the change in λ is given for the methods given in Table 4.1.

The best achievable SSIM and corresponding PSNR values are given in Table 4.2. The proposed methods effects only around 15% of the pixels as given in Figure 4.7. According to (4.5), in regions with small gradient activity, MTD1 reduces to the method in [40] , and MTD2 reduces to (4.4). The edge thresholds for the proposed methods are selected as 250, 280, 250 for MTD1, MTD2 and MTD3 respectively. The PSNR values for edge pixels are given in Table 4.3 .

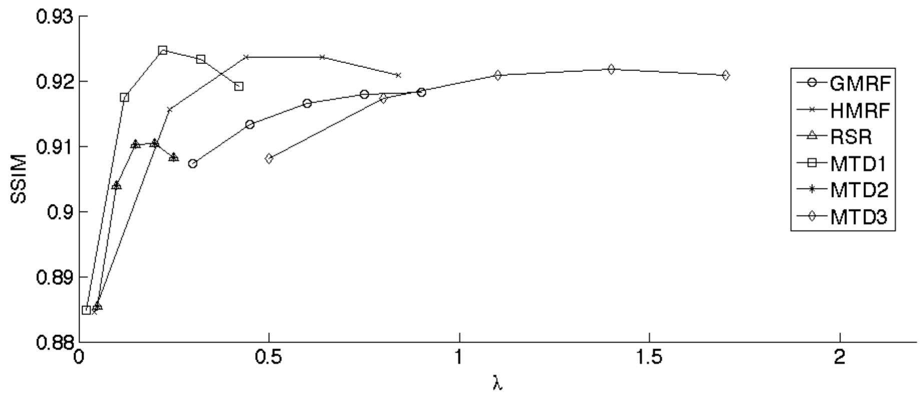
The reconstructions are given in Figures 4.10(a), 4.11(a), and 4.12(a). These reconstructions are filtered by a Laplacian filter in the form of [-1 -1 -1; -1 8 -1; -1 -1 -1] for better visual assessment. The resultant high frequency images for each reconstruction are given in Figures 4.10(b), 4.11(b), and 4.12(b).



(a) Image set in Figure 4.3(a)



(b) Image set in Figure 4.3(b)



(c) Image set in Figure 4.3(c)

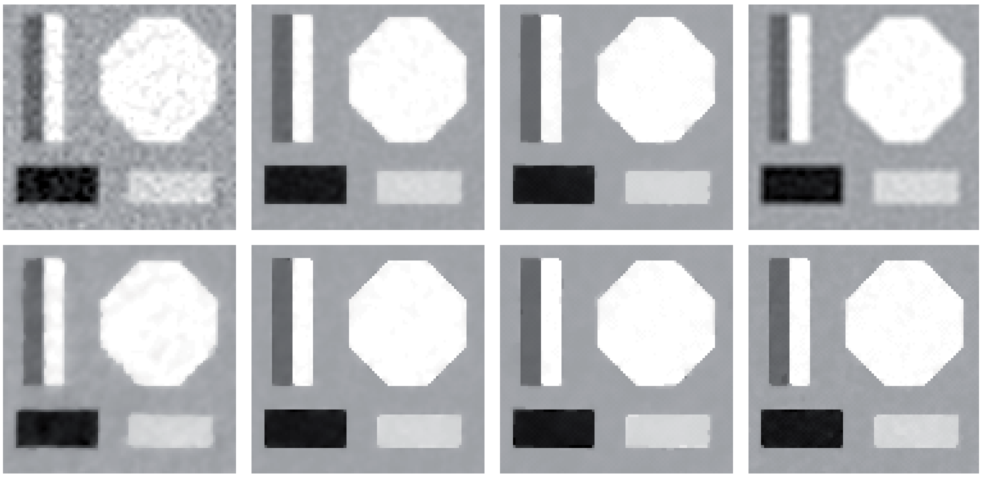
Figure 4.9: Effect of varying the regularization amount, λ. SSIM values for each reconstruction is given for each method.

Table 4.2: SSIM / PSNR (dB) values for the SR images obtained from the LR data set of the images in Figure 4.3.

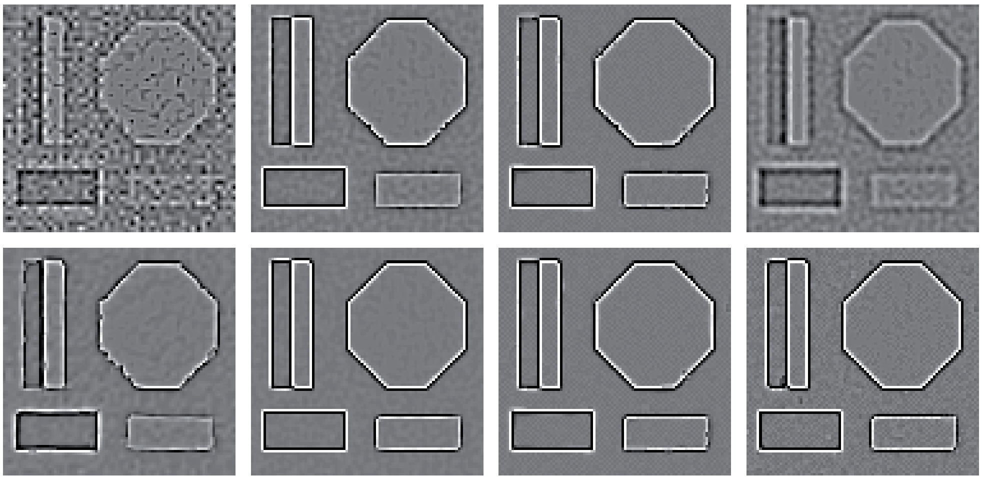
	Figure 4.3(a)	Figure 4.3(b)	Figure 4.3(c)
BICUBIC	0.709 / 25.2	0.805 / 20.6	0.784 / 27.5
INT HUBER	0.912 / 28.3	0.845 / 22.0	0.826 / 28.2
GMRF	0.871 / 26.9	0.904 / 24.5	0.900 / 31.0
HUBER	0.961 / 34.3	0.920 / 25.5	0.911 / 31.4
RSR	0.983 / 35.9	0.914 / 25.5	0.898 / 30.8
MTD1	0.988 / 38.5	0.926 / 26.0	0.911 / 31.5
MTD2	0.987 / 37.2	0.914 / 25.5	0.898 / 30.8
MTD3	0.978 / 37.7	0.923 / 26.0	0.909 / 31.5

Table 4.3: PSNR (dB) values for the edge pixels of the images in Figure 4.3.

	Fig 4.3(a)	Fig 4.3(b)	Fig 4.3(c)
BICUBIC	26.7	21.9	31.8
INT HUBER	29.1	23.5	32.5
GMRF	27.3	26.3	34.7
HUBER	35.4	27.4	35.7
RSR	36.6	27.6	35.1
MTD1	40.0	28.1	35.8
MTD2	38.1	27.5	35.1
MTD3	39.4	28.0	35.9



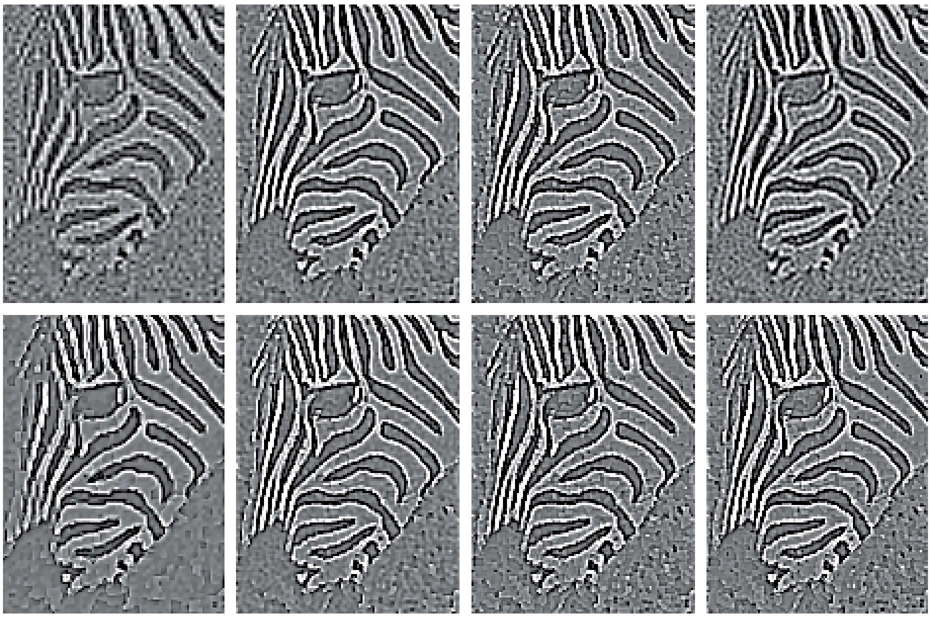
(a)



(b)

Figure 4.10: (a) Reconstructions for Figure 4.3(a). In the first row: Bicubic interpolation, HUBER, RSR, and GMRF are given respectively; In the second row: HUBER INT, MTD1, MTD2, and MTD3 are given. (b) High frequency components of the reconstructions are given. In the first row: Bicubic interpolation, HUBER, RSR, and GMRF are given; In the second row: HUBER INT, MTD1, MTD2, and MTD3 are given.

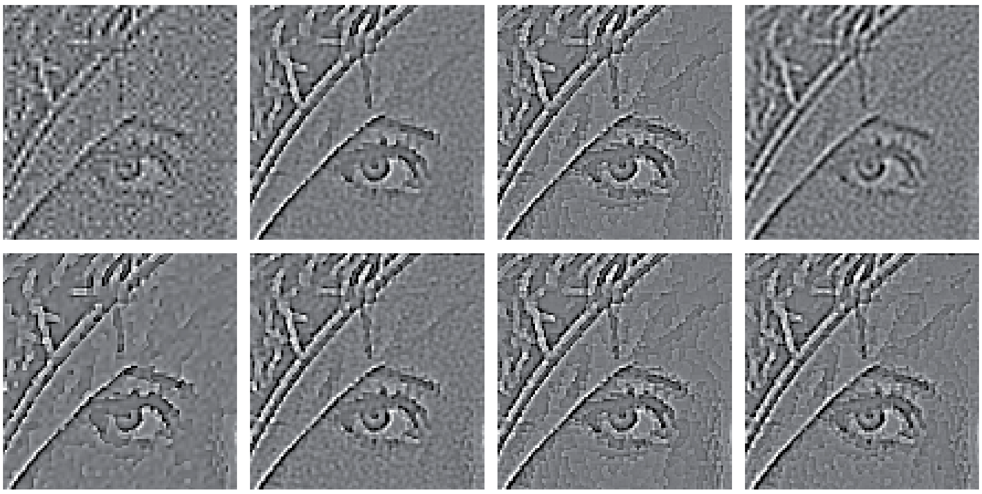
(a)



(b)

Figure 4.11: (a) Reconstructions for Figure 4.3(b). In the first row: Bicubic interpolation, HUBER, RSR, and GMRF are given respectively; In the second row: HUBER INT, MTD1, MTD2, and MTD3 are given. (b) High frequency components of the reconstructions are given. In the first row: Bicubic interpolation, HUBER, RSR, and GMRF are given; In the second row: HUBER INT, MTD1, MTD2, and MTD3 are given.

(a)



(b)

Figure 4.12: (a) Reconstructions for Figure 4.3(c). In the first row: Bicubic interpolation, HUBER, RSR, and GMRF are given respectively; In the second row: HUBER INT, MTD1, MTD2, and MTD3 are given. (b) High frequency components of the reconstructions are given. In the first row: Bicubic interpolation, HUBER, RSR, and GMRF are given; In the second row: HUBER INT, MTD1, MTD2, and MTD3 are given.

4.4 DISCUSSION

In this chapter, a directionally adaptive SR image reconstruction method is presented. The basic idea behind this algorithm is determining the best regularization filter according to the magnitude and direction of the local gradient. Three variations of this algorithm are compared with previous work [25, 40]. The SSIM, PSNR values and illustrations are given for performance evaluations. Especially for highly degraded images with sharp edges, the proposed methods clearly preserve edges while applying high amounts of regularization. The performance is similar to [25, 40] in regions, where a strong edge is not present. The proposed approach has a computational overhead, which varies depending on the gradient distribution in the input LR image sets. In the simulations 10 - 15 % increase in computation time is observed compared to RSR and HUBER implementations.

CHAPTER 5

TEXTURE ADAPTIVE SUPER RESOLUTION

5.1 INTRODUCTION

This chapter targets texture preservation during SR reconstruction. Similar to the proposed direction adaptive MAP approach in Chapter 4, the proposed solution also depends on the general MAP estimator

$$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} ||W_k \mathbf{z} - \mathbf{y}_k||_R^R + \lambda \Gamma(\mathbf{z}) \tag{5.1}$$

where $\Gamma(.)$ represents the regularization term which depends on the image prior term. λ controls the influence of the regularization to the solution. The main focus in this chapter is the $\Gamma(\mathbf{z})$ term in Eqn. 5.1, which is directly related to the proposed texture prior.

Designing a texture prior in a MAP framework is not a straightforward task because of the variety of textures in nature, [68]. A region in an image is defined as textured if a set of local statistics or local properties of the region are constant, slowly varying, or approximately periodic [69]. Several successful texture models have been proposed in texture synthesis, analysis, and segmentation fields. Early models include auto regressive (AR) models [70], simultaneous auto regressive (SAR) models, and GMRF models. However, the major problem of these single-layer models is locality. Texture characteristics need to be defined at a large neighborhood. To solve this problem, multiscale approaches have been proposed. In [71], a rotation invariant SAR model has been proposed for texture. This model has been used in analyzing texture at multiple resolutions. In [72], texture has been modeled by fitting a separate AR model at each level of a Laplacian pyramid. This model has been utilized in texture segmentation. Later, multiresolution MRF models have been proposed in texture modeling. In [73], multidimensional histograms have been obtained at multiple scales of an example texture. These histograms have been considered as approximations of local conditional probability density functions. These density functions have been used to synthesize a texture from coarse to fine resolution. In [74], multiresolution images have been obtained by using Haar wavelets. Each subband has been modeled by an MRF, whose coefficients have been used in texture classification along with subband energy parameters. In [75], a GMRF model has been proposed for each level of a Gaussian pyramid, which has been used in a course to fine resolution segmentation procedure.

Although there is a large choice of multiresolution texture models in the literature, “to best of our knowledge”, there has not been any attempt to employ these models in SR problem. As mentioned previously, standard priors in SR literature such as TV, BTV, HMRF, and GMRF are all defined in an 8-pixel neighborhood aiming at edge preservation and noise suppression. However, texture preservation requires a multiscale approach to model complex interactions in a large neighborhood. We propose a Bayesian SR method with a multiscale compound MRF prior. This prior is the main contribution of this study. It has been reported that dominant scales and orientations representing both highly random and quasi-periodic textures are captured using Gabor filters [76], [77], [78]. Therefore, Gabor filters are specifically preferred for multiscale decomposition. The proposed model inherits a compound GMRF at each subband, similar to the “line process” [79]. This second hidden binary process models sharp variations due to high-frequency texture. We propose a joint estimation method for estimating the model parameters and high-resolution image pixel in a MAP framework.

The rest of the chapter is organized as follows. In Sections 5.2.1 and 5.2.2, the forward problem and proposed multiresolution texture prior are described. A MAP-based reconstruction method is given in Section 5.2.3. Section 5.2.4 presents prior estimation using Gabor filters. Lastly, in Section 5.3, the proposed methods are validated through simulations and experiments on visual CCD images and thermal images.

5.2 PROPOSED BAYESIAN SUPER RESOLUTION

5.2.1 The Imaging Model

The proposed method is based on the following observation model given in Chapter 1.

$$\mathbf{y}_k = \mathbf{W}_k \cdot \mathbf{z} + \mathbf{n}_k \qquad k = 1, \ldots, c. \tag{5.2}$$

Here, observation noise is assumed to be independent and identically distributed Gaussian noise. It is also assumed that each observation (LR image) is statistically independent. Under these conditions, the joint probability for all LR images can be written as follows [80]:

$$Pr(\mathbf{y}_1, \ldots, \mathbf{y}_c|\mathbf{z}, \beta) = \prod_{k=1}^{c} \left(\frac{\beta}{2\pi}\right)^{N/2} \exp\left\{-\frac{\beta}{2}||\mathbf{W}_k\mathbf{z} - \mathbf{y}_k||_2^2\right\}. \tag{5.3}$$

where β is the inverse variance of independent and identically distributed noise and N is the number of total pixels in each LR image. The above-mentioned equation is rearranged in the following form to be used in the next sections,

$$Pr(\mathbf{y}_1, \cdots \mathbf{y}_c|\mathbf{z}, \beta) = \left(\frac{\beta}{2\pi}\right)^{cN/2} \exp\left\{-\sum_{k=1}^{c} \frac{\beta}{2}||\mathbf{W}_k\mathbf{z} - \mathbf{y}_k||_2^2\right\}. \tag{5.4}$$

5.2.2 Multiresolution Compound Texture Prior

Gabor filters are used in many fields in image processing such as edge detection, texture classification, and data compression [81]. They are employed as a *family* of filters with multiple scales and orientations [81, 82]. The form of a 2D real symmetric Gabor filter is:

$$G(x, y) = \exp\left(\frac{x^2 + y^2}{\sigma^2}\right) \cos\left(\frac{2\pi}{\lambda}\left(x \cos\theta + y \sin\theta\right)\right), \tag{5.5}$$

where θ determines the direction of the filter. Usually, eight directions between 0^o and 180^o are used [83]. Defining the filter family on a half circle is sufficient owing to the symmetry of cosine function. λ is inversely proportional to the frequency of the carrier and σ is related to the spread of the Gaussian envelope. As the span of the Gaussian envelope increases, the frequency resolution of the filter increases and the spatial selectivity decreases.

Let F_k represent Gabor filters ($k = 1 \cdots K$) and $\mathbf{z}_{F_k}$ be the response of image $\mathbf{z}$ to k^{th} Gabor filter. The multiresolution MRF (MRMRF) prior given in [84] (pp. 40) has the following form:

$$Pr(\mathbf{z}) = \frac{1}{Z} \exp\left\{-\sum_{k=1}^{K}\sum_{c\in C} V_c^{(k)}(\mathbf{z}_{\mathbf{F_k}})\right\}, \tag{5.6}$$

where $V_c^{(k)}$ are clique potentials defined on the filtered image $\mathbf{z}_{F_k}$ and Z is the normalizing constant. In this section, we will derive the proposed compound MRMRF texture prior starting from (5.6). A quadratic potential function is defined for each clique, similar to that in [13, 80],

$$V_c(\mathbf{z}, \boldsymbol{\rho}) = \sum_{k=1}^{K}\sum_{i\sim j} \rho_k (z_i^k - z_j^k)^2, \tag{5.7}$$

Substituting the potential function term in the prior term gives:

$$p(\mathbf{z}|\boldsymbol{\rho}) = \frac{1}{Z} \exp\left\{-\sum_{k=1}^{K}\sum_{i\sim j} \rho_k (z_i^k - z_j^k)^2\right\}, \tag{5.8}$$

where z_i^k represents the value of the i^{th} pixel of $\mathbf{z}_{F_k}$ and $\sum_{i\sim j}$ represents all pixel pairs that are neighbors to each other. Clique potentials within the same subband have the same quadratic form. ρ_k determines the total effect of the sum of the clique potentials within a subband to the overall sum. ρ_k of each subband is stacked in vector $\boldsymbol{\rho}$ in (5.8).

We extend the MRMRF prior by inserting a latent variable into the potential function expression in (5.8) at each subband, referring it as the "texture process". Texture process, $\boldsymbol{\eta}$, is composed of binary variables $\eta_{i,j,k}$ similar to the line process defined in [79]. We follow the same naming convention as used in [85, 86], with the only difference being the additional index, k, referring to the corresponding subband. Inserting $\boldsymbol{\eta}$ into (5.8) gives

$$p(\mathbf{z}|\boldsymbol{\rho}, \boldsymbol{\eta}) = \frac{1}{Z} \exp\left\{-\sum_{k=1}^{K}\sum_{i\sim j} \rho_k \eta_{i,j,k} (z_i^k - z_j^k)^2\right\}. \tag{5.9}$$

Unlike the line process [79, 85, 86], the texture process occupies the same lattice as the image pixels. This makes the index j redundant for defining the process. In other words, for a pixel on a subband, e.g., z_i^k, if $\eta_{i,k}$ is one, then the cliques defined on z_i^k and its eight neighbors is turned on simultaneously. This results in a smoother solution around that pixel at the specified subband. Therefore, we omit the additional index j, having only $\eta_{i,k}$, where i stands for the pixel index and k denotes the subband index.

Based on [85, 86], (5.9) can be written in a multivariate Gaussian form. The resultant multiresolution multivariate Gaussian MRF image prior has the following form:

$$p(\mathbf{z}|\boldsymbol{\rho}, \boldsymbol{\eta}) = \prod_{k=1}^{K} |A_{\eta,k}|^{1/2} \left(\frac{\rho_k}{2\pi}\right)^{N/2} \exp\left\{-\frac{\rho_k}{2}\mathbf{z}_{F_k}^T A_{\eta,k}\mathbf{z}_{F_k}\right\} \tag{5.10}$$

where

$$A_{\eta,k}(i,j) = \begin{cases} 8\eta_{i,k} & i = j \\ -\eta_{i,k} & i \sim j \\ 0 & else \end{cases} .(5.11)$$

here $i \sim j$ refers to the indices of neighboring pairs.

5.2.3 MAP Solution

The standard MAP approach for estimating high-resolution image pixels is to maximize the conditional probability of $\mathbf{z}$, given low-resolution image pixels and unknown model parameters. The unknown parameter set consists of high-resolution image pixels, $\mathbf{z}$, inverse variance of observation noise, β, and texture prior parameters represented by $(\boldsymbol{\eta}, \boldsymbol{\rho})$. $\boldsymbol{\eta}$ is defined for each image pixel at each subband and $\boldsymbol{\rho}$ is a vector consisting of ρ_k's defined for each subband. The unknown parameter set could be increased by adding the unknown camera blur and registration parameters similar to that carried out in [87, 88].

Various methods have been proposed in SR literature to solve such problems with multidimensional unknowns. Hardie *et al.* proposed a joint MAP approach [13] to solve for both registration among LR pixels and high-resolution image pixels. An alternating minimization algorithm has been employed, in which one set of parameters are fixed while the other set is iteratively estimated for both sets in an alternating manner. Similarly, in [89, 85], a marginalized maximum likelihood (ML) approach has been proposed for solving the unknown high-resolution pixels under unknown registration and camera blur. A marginalized MAP formulation has been proposed for solving for unknown high-resolution image pixels along with model parameters in [88, 87, 86]. In the present study, we have simplified the discussion by assuming that the camera blur and registration were estimated before reconstruction. It should be noted that the main aim of this study is the compound texture prior and it is straightforward to include unknown registration and camera blur to the proposed MAP solution similar to that carried out in [88, 87, 86].

The proposed MAP approach is based on maximizing the following posterior probability for unknown high-resolution image $\mathbf{z}$,

$$p(\mathbf{z}|\mathbf{y}_1 \cdots \mathbf{y}_c, \boldsymbol{\eta}, \boldsymbol{\rho}, \beta) = \frac{p(\mathbf{y}_1 \cdots \mathbf{y}_c|\mathbf{z}, \beta)p(\mathbf{z}|\boldsymbol{\rho}, \boldsymbol{\eta})}{p(\mathbf{y}_1 \cdots \mathbf{y}_c|\boldsymbol{\rho}, \boldsymbol{\eta}, \beta)}. \tag{5.12}$$

The probability term in the denominator in (5.12) is independent of $\mathbf{z}$ and therefore, is removed from the maximization. Among the various approaches to solve this high-dimensional problem, we resort to a joint estimation method to maximize the following expression iteratively:

$$\mathbf{z}_{MAP} = \underset{\mathbf{z}}{argmax} p(\mathbf{y}_1 \cdots \mathbf{y}_c|\mathbf{z}, \boldsymbol{\eta}, \boldsymbol{\rho})p(\mathbf{z}|\boldsymbol{\rho}, \boldsymbol{\eta}), \tag{5.13}$$

where $\boldsymbol{\rho}, \boldsymbol{\eta}$ terms are extracted iteratively from the available estimate of $\mathbf{z}$. By substituting the conditional probabilities in (5.4) and (5.10) with (5.13), the following maximization problem is obtained:

$$\mathbf{z}_{MAP} = \underset{\mathbf{z}}{argmax} \; \exp\left\{-\sum_{k=1}^{c}\frac{\beta}{2}||\mathbf{W}_k\mathbf{z} - \mathbf{y}_k||_2^2 - \sum_{k=1}^{K}\frac{\rho_k}{2}\mathbf{z}_{F_k}^T(A_{\eta,k})\mathbf{z}_{F_k}\right\}, \tag{5.14}$$

where the constants are removed from maximization. The overall problem can be written as a minimization problem as follows:

$$\mathbf{z}_{MAP} = \underset{\mathbf{z}}{argmin} \sum_{k=1}^{c}\frac{\beta}{2}||\mathbf{W}_k\mathbf{z} - \mathbf{y}_k||_2^2 + \sum_{k=1}^{K}\frac{\rho_k}{2}\mathbf{z}_{F_k}^T(A_{\eta,k})\mathbf{z}_{F_k}. \tag{5.15}$$

As stated previously, $\mathbf{z}_{F_k}$ is obtained by filtering $\mathbf{z}$ by the k^{th} filter of the Gabor filter family. Let $\mathbf{F}_k$ be $N \times N$ matrix operator representing the k^{th} Gabor filter in lexicographical representation. Substitution of $\mathbf{z}_{F_k}$ with $\mathbf{F}_k\mathbf{z}$ results in minimization only with respect to $\mathbf{z}$:

$$\mathbf{z}_{MAP} = \underset{\mathbf{z}}{argmin} \sum_{k=1}^{c} \frac{\beta}{2} ||\mathbf{W}_k\mathbf{z} - \mathbf{y}_k||_2^2 + \sum_{k=1}^{K} \frac{\rho_k}{2} \mathbf{z}^T \mathbf{F}_k^T (A_{\eta,k}) \mathbf{F}_k \mathbf{z}\,. \tag{5.16}$$

Separation of $A_{\eta,k}$ term as $A_{\eta,k} = D_{\eta,k}C$ is possible, because the sites for the texture process occupy the same lattice as those for the image pixels. The definitions of $D_{\eta,k}$ and C are as follows:

$$C(i,j) = \begin{cases} 8 & i = j \\ -1 & i \sim j \\ 0 & else \end{cases} \tag{5.17}$$

and

$$D(i,j) = \begin{cases} \eta_i & i = j \\ 0 & else \end{cases} \tag{5.18}$$

Note that an 8-pixel neighborhood is considered. Substitution of these new terms with $A_{\eta,k}$ in (5.16) results in

$$\mathbf{z}_{MAP} = \underset{\mathbf{z}}{argmin} \sum_{k=1}^{c} \frac{\beta}{2} ||\mathbf{W}_k\mathbf{z} - \mathbf{y}_k||_2^2 + \sum_{k=1}^{K} \frac{\rho_k}{2} \mathbf{z}^T \mathbf{F}_k^T (D_{\eta,k}C) \mathbf{F}_k \mathbf{z}. \tag{5.19}$$

It can be shown that the inverse covariance matrix $\mathbf{C}$ acts as a Laplacian operator in the form of c = [-1 -1 -1 ; -1 8 -1; -1 -1 -1] in 2D image domain. Being block circulant matrices, $\mathbf{F}_k$ and $\mathbf{C}$ in (5.19) commute [90]. This results in

$$\mathbf{z}_{MAP} = \underset{\mathbf{z}}{argmin} \sum_{k=1}^{c} \frac{\beta}{2} ||\mathbf{W}_k\mathbf{z} - \mathbf{y}_k||_2^2 + \sum_{k=1}^{K} \frac{\rho_k}{2} \mathbf{z}^T \mathbf{F}_k^T (D_{\eta,k}) \mathbf{F}_k \mathbf{C}\, \mathbf{z}\,. \tag{5.20}$$

Application of gradient descent to (5.20) gives us

$$\hat{\mathbf{z}}^{n+1} = \hat{\mathbf{z}}^n - \lambda \left[\beta \sum_{k=1}^{c} \left(W_k^T\right) \cdot \left(W_k \hat{\mathbf{z}}^n - \mathbf{y}_k\right) + \sum_{k=1}^{K} \rho_k \mathbf{F}_k^T (D_{\eta,k}) \mathbf{F}_k \mathbf{C}\, \hat{\mathbf{z}}^n \right], \tag{5.21}$$

where λ is the step size at n^{th} iteration and $\hat{\mathbf{z}}^n$ is the estimate at n^{th} iteration. For

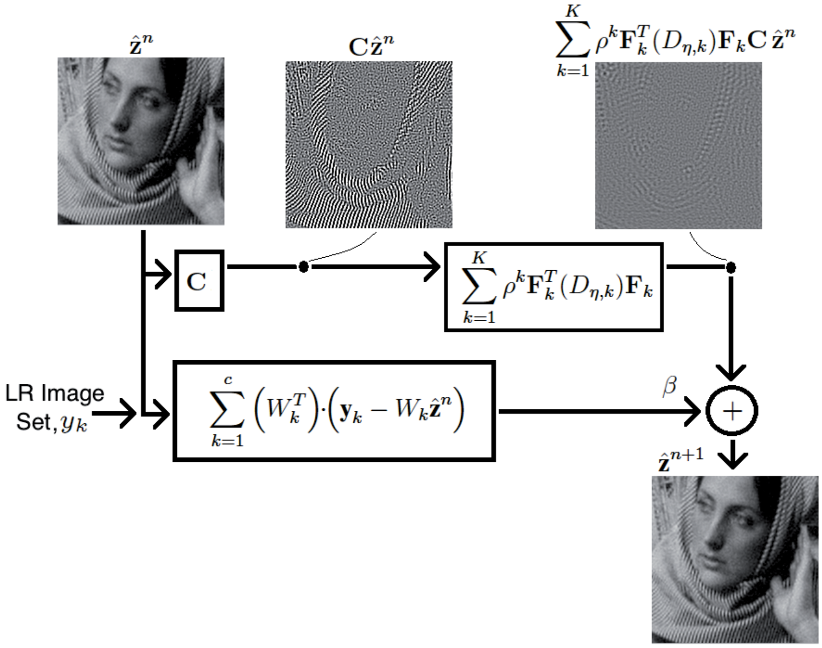


Figure 5.1: One iteration of the proposed method.

comparison, the gradient descent equation with GMRF prior is also given in (5.22),

$$\hat{\mathbf{z}}^{n+1} = \hat{\mathbf{z}}^n - \lambda \Big[\beta \sum_{k=1}^{c} \left(W_k^T\right) \cdot \left(W_k \hat{\mathbf{z}}^n - \mathbf{y}_k\right) + \mathbf{C}\,\hat{\mathbf{z}}^n \Big]. \tag{5.22}$$

The difference between (5.21) and (5.22) is clear on the block diagram given in Figure 5.1, where one iteration of (5.21) is given. The $\mathbf{C}\hat{\mathbf{z}}^n$ term represents the high-frequency content of the reconstructed image at n^{th} iteration. It can be observed that the $\mathbf{C}\hat{\mathbf{z}}^n$ term in Figure 5.1 comprises high-frequency noise and texture fragments. If GMRF prior is employed, the $\mathbf{C}\hat{\mathbf{z}}^n$ term is penalized and a smoother solution is obtained. However, with the use of the proposed texture prior, $\sum_{k=1}^{K} \rho^k \mathbf{F}_k^T (D_{\eta,k}) \mathbf{F}_k \mathbf{C}\,\hat{\mathbf{z}}^n$ term is penalized at each iteration. The penalty image does not contain texture on the scarf of Barbara in Figure 5.1.

An important challenge in the proposed solution in (5.21) is the estimation of $\boldsymbol{\rho}$ and $\boldsymbol{\eta}$ at each iteration so that texture preservation is achieved.

5.2.4 Prior Parameter Estimation

Gabor filter bank used in this study is composed of 24 filters with 3 scales and 8 directions. The size of each filter is 32×32. The spatial spread of each Gabor filter is dependent on the scale such that at each scale, 5 lobes are visible. As a result, a better span of the frequency domain with minimum overlap is achieved.

Filtering an image with a Gabor filter bank results in a series of filtered images. The magnitude of the response at a pixel location is proportional to the presence of an edge [81] or texture [82] oriented at the selected orientation. Using this idea, the $D_{\eta,k}$ term of (5.21) is estimated by thresholding the $\mathbf{F}_k\mathbf{C}\,\hat{\mathbf{z}}^n$ term of (5.21) at each band as follows:

$$D_{\eta,k}\mathbf{F}_k\mathbf{C}\,\hat{\mathbf{z}}^n(i) = \begin{cases} \mathbf{F}_k\mathbf{C}\,\hat{\mathbf{z}}^n(i) & \mathbf{F}_k\mathbf{C}\,\hat{\mathbf{z}}^n(i) < T \\ 0 & else \end{cases} \tag{5.23}$$

where i represents the index of each image pixel and ρ_k is set to 1 for all subbands to simplify the discussion. A heuristic approach is taken in determining T. T is selected to be proportional to the standard deviation of all pixels in the $\mathbf{C}\hat{\mathbf{z}}^n$, which results in successful reconstructions in the simulations.

The filtered and thresholded images are filtered again with their corresponding filters and summed up to form the residue $\sum_{k=1}^{K} \rho_k \mathbf{F}_k^T (D_{\eta,k})\mathbf{F}_k\mathbf{C}\,\hat{\mathbf{z}}^n$ at n^{th} iteration. This additional filtering corresponds to the $\mathbf{F}_k^T$ operation in the residue term. This operation is equivalent to convolving a 2D image with the k^{th} Gabor filter corresponding to $\mathbf{F}_k$, owing to the symmetry of $\mathbf{F}_k$. Proving the symmetry of $\mathbf{F}_k$ is strait-forward if corresponding Gabor filter in 2D image model is considered. Each Gabor filter has a spatial symmetry with respect to its center element. Any radial line crossing the center element is symmetric with respect to its center. Therefore, each row of the operator matrix $\mathbf{F}_k$ is symmetric with respect to the diagonal element. Henceforth, $\mathbf{F}_k$ is equal to $\mathbf{F}_k^T$, because $\mathbf{F}_k$ is also circulant. The reconstructed image is updated by adding the back-projected error and residue term, as described in (5.21).

5.3 RESULTS

In this section, we demonstrate the performance of the proposed method on both synthetically generated images and real images. The proposed method is compared with single frame bicubic interpolation, MAP estimate with Gaussian Markov Random Field image prior, and the method given in [25]. The reasons for selecting the method in [25] are its robustness to motion estimation errors and its edge-preserving nature. This method is abbreviated as RSR.

Gradient descent formulation for the MAP estimate with GMRF prior is

$$\hat{\mathbf{z}}^{n+1} = \hat{\mathbf{z}}^{n} - \beta^{n}\left[\sum_{k=1}^{c}\left(\mathbf{W}_k^T\right)\cdot\left(\mathbf{W}_k\hat{\mathbf{z}}^{n} - \mathbf{y}_k\right) + \lambda\mathbf{C}^T\mathbf{C}\cdot\hat{\mathbf{z}}^{n}\right], \tag{5.24}$$

$\mathbf{C}$ is selected as a Laplacian operator in the form of [-1 -1 -1; -1 8 -1; -1 -1 -1] to force smoothness, as in [13]. This method is abbreviated as GMRF through out the paper. Best possible λ is searched experimentally for the GMRF estimate.

The method described in [25] is implemented as follows:

$$\hat{\mathbf{z}} = \underset{\mathbf{z}}{argmin}\sum_{k=1}^{c}||\mathbf{W}_k\mathbf{z} - \mathbf{y}_k||_1^1 + \lambda\sum_{q=-2}^{2}\sum_{p=-2}^{2}\alpha^{|p|+|p|}||\mathbf{z} - S_x^pS_y^q\mathbf{z}||_1^1 \tag{5.25}$$

In this equation S_x^p and S_y^q operators shift the image by p pixels horizontally and q pixels vertically respectively. α (set to 0.3) is a decay term to decrease the effect of distant pixels in the difference operation. Gradient descent formulation in the following form is used during reconstruction.

$$\hat{\mathbf{z}}^{n+1} = \hat{\mathbf{z}}^{n} - \beta\left[\sum_{k=1}^{c}\left(W_k^T\right)\cdot sign\left(W_k\hat{\mathbf{z}}^{n} - \mathbf{y}_k\right) + \lambda\sum_{q=-2}^{2}\sum_{p=-2}^{2}\alpha^{|p|+|q|}sign\left(\hat{\mathbf{z}}^{n} - S_x^pS_y^q\hat{\mathbf{z}}^{n}\right)\right] \tag{5.26}$$

where $sign(x)$ operation results in 1 or -1 if the sign of x is positive or negative respectively. Best possible λ is searched experimentally. All SR methods are initialized with bicubic interpolation through the experiments. In the first section experiments on synthetically generated images are conducted. Real experiments on day and thermal cameras are conducted in the other two sections. The iterations are ceased after the percentage change in the reconstructed image with respect to the previous reconstruction is below a threshold. This condition is formulated as: $||\hat{\mathbf{z}}^{n+1} - \hat{\mathbf{z}}^{n}||/\,||\hat{\mathbf{z}}^{n}|| \leq 10^{-6}$, where $\mathbf{z}^n$ is the reconstructed image at n^{th} iteration.

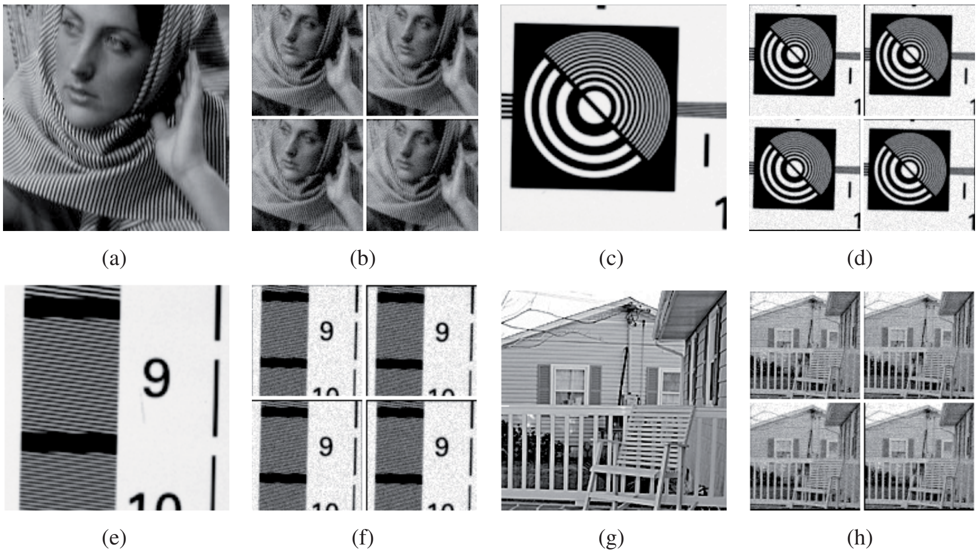


Figure 5.2: Original HR images for (a) "Barbara", (c) "Chart Section 1", (e) "Chart Section 2", and (g) "House", (b), (d), (f) and (h) Four of eight synthetically generated LR images at 25 dB noise.

5.3.1 Experiments on Synthetically Generated Images

In the following simulations, the proposed method is tested against synthetically generated images given in Figure 5.2. The "Barbara" and "House" images are selected for its textured structure. The other two images are cropped from the well-known "Chart" image. Each 200×200 high-resolution image is warped, blurred, and downsampled by a factor of 2 to obtain eight 100×100 LR images according to (5.2). The Gaussian noise at SNR levels of 25 and 30 dB are used to degrade the LR images. The example LR images are also given in Figure 5.2 for 25 dB noise case. Only translational motion is considered, which is randomly set within a 2-pixel range among the LR images. The true motion values are used during reconstruction. A $2{\times}2$ moving average filter is used to simulate blurring. The resolution is increased by a factor of 2.

The quantitative performance comparison is achieved using MSE and SSIM measurements. SSIM (4.18) is also used in addition to the MSE measures, because human visual perception is more sensitive to structural information in a scene [67].

The amount of regularization, λ, is varied to achieve the best possible SSIM value for all methods. By this way, the best possible performances are compared with the proposed method. The effect of varying λ for the GMRF, RSR, and proposed method is given for both 25 and 30 dB noise levels in Figure 5.3. The GMRF and proposed method converge to each other for λ close to 0 as expected. Figure 5.4 shows the convergence of the reconstructions for both 25 and 30 dB noise. The GMRF and proposed method converge after 20 iterations and RSR converges after 200 iterations. One of every 10 iterations is shown to plot RSR convergence on the same graph. The reconstructed images are given in Figures 5.5, 5.7, 5.9, and 5.11. These reconstructions are filtered by a Laplacian filter in the form of [-1 -1 -1; -1 8 -1; -1 -1 -1] for better visual assessment. The resultant high-frequency images for each reconstruction are given in Figures 5.6, 5.8, 5.10, and 5.12.

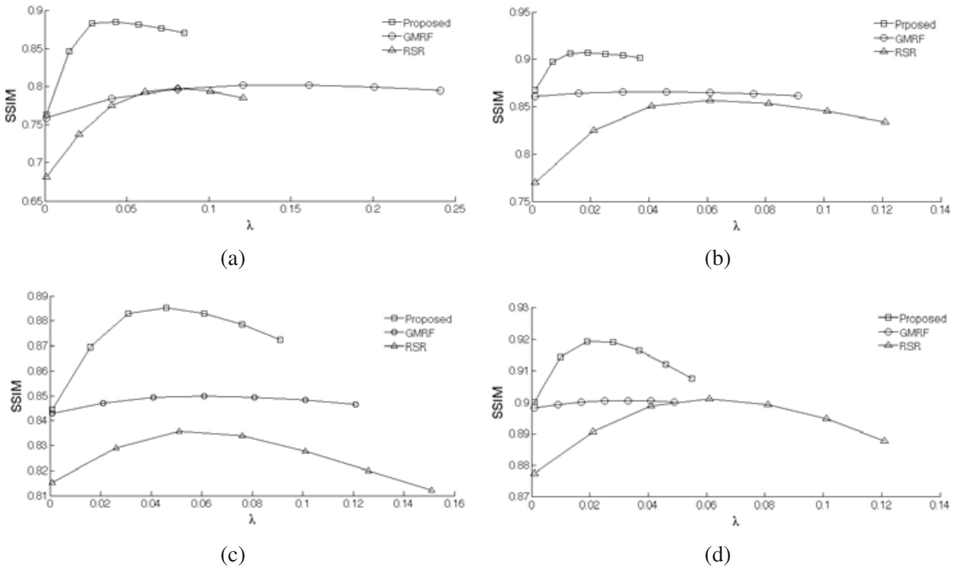


Figure 5.3: Effect of varying regularization amount λ versus SSIM values at different SNR levels (a) Barbara image at SNR = 25 dB, (b) SNR = 30 dB. (c) Chart image Section 1 at SNR = 25 dB, (d) SNR = 30 dB.

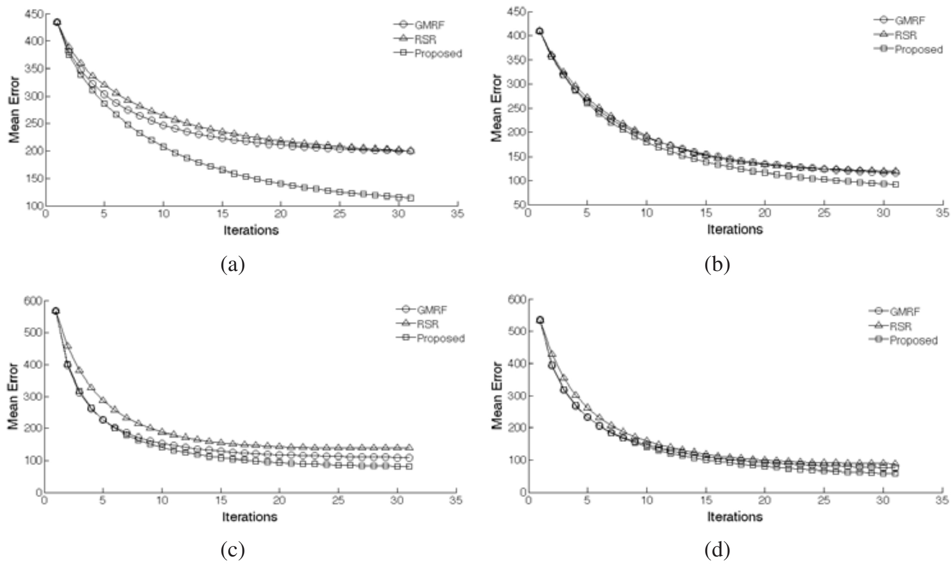


Figure 5.4: Convergence of the algorithms. (a) Barbara image at SNR = 25 dB, (b) SNR = 30 dB. (c) Chart image at SNR = 25 dB, (d) SNR = 30 dB.

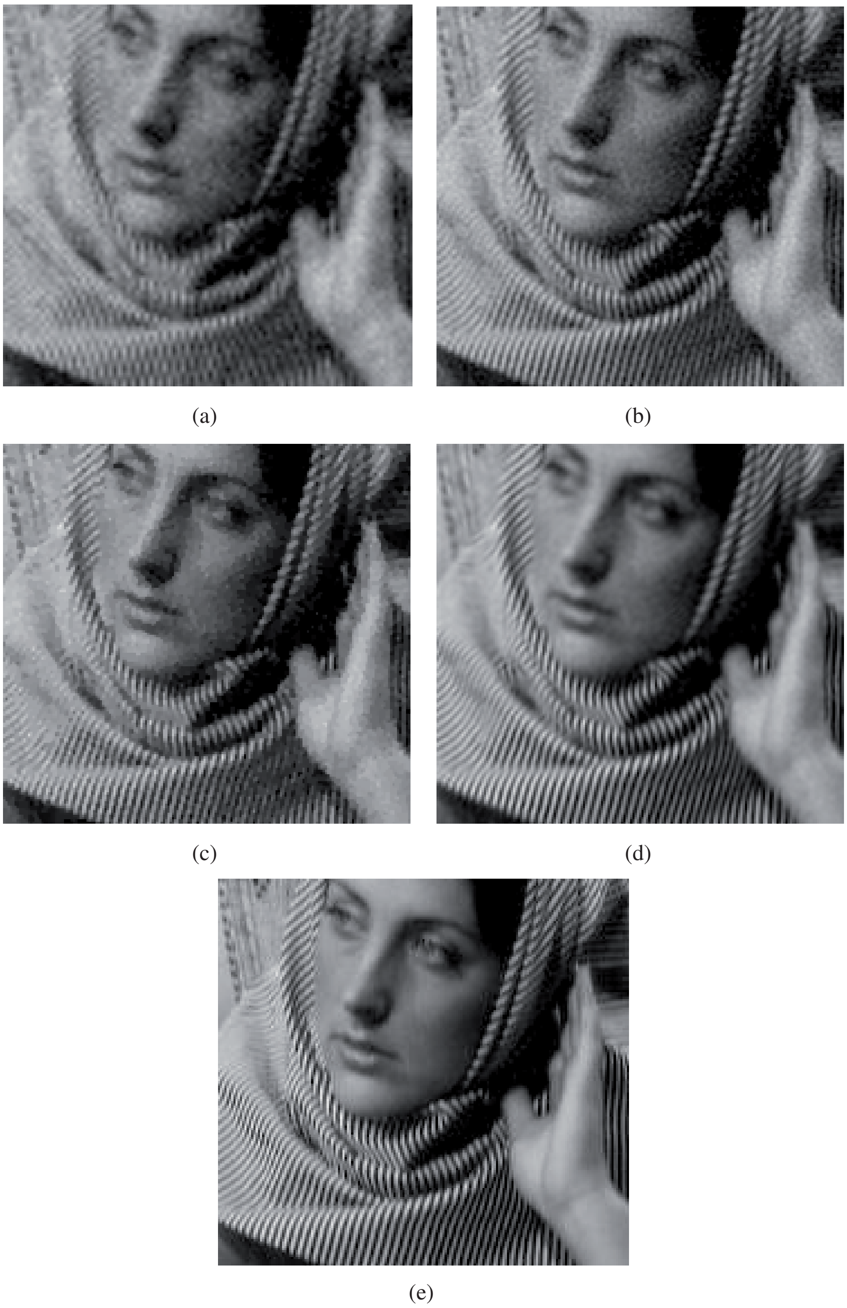


Figure 5.5: (a) Bicubic Interpolation, (b) GMRF, (c) RSR, (d) Proposed, (e) Original Image.

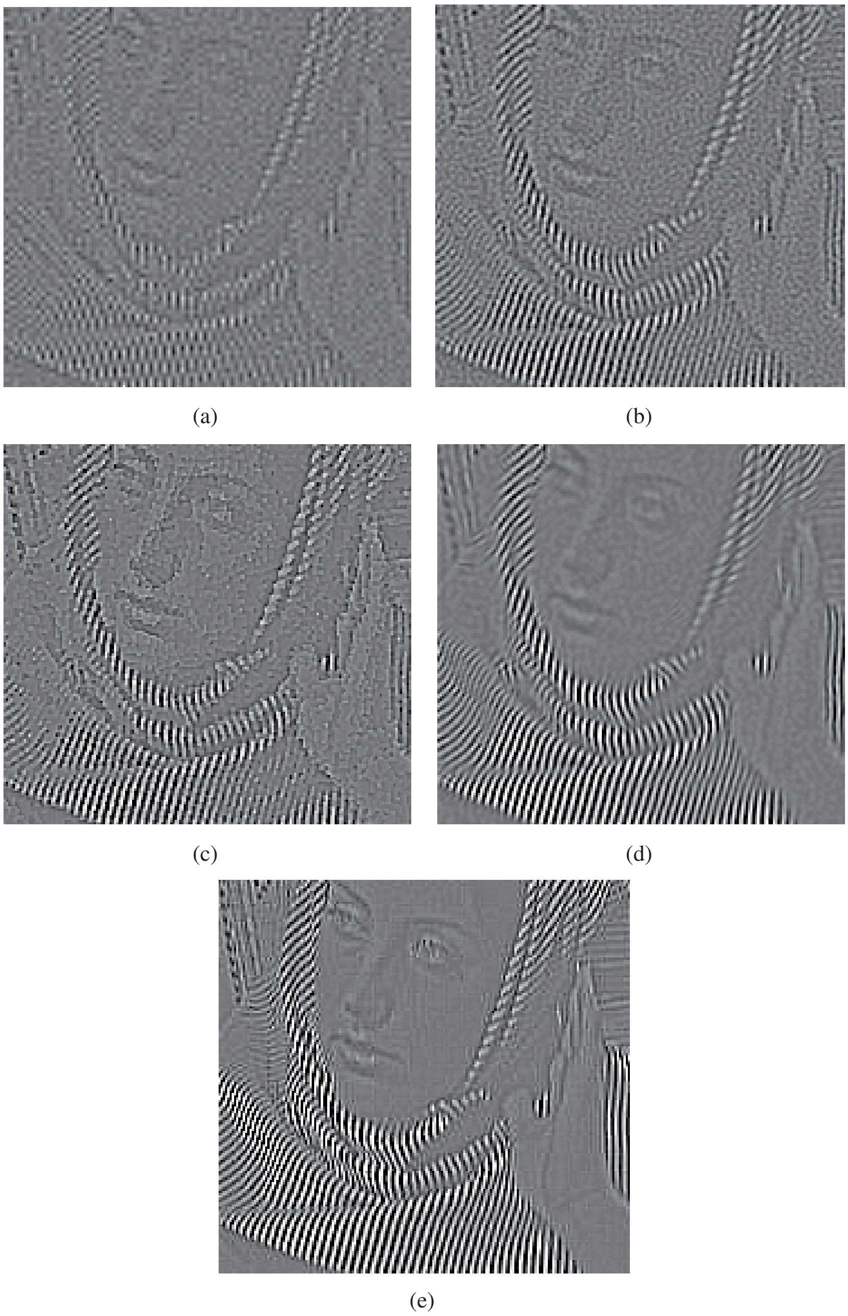
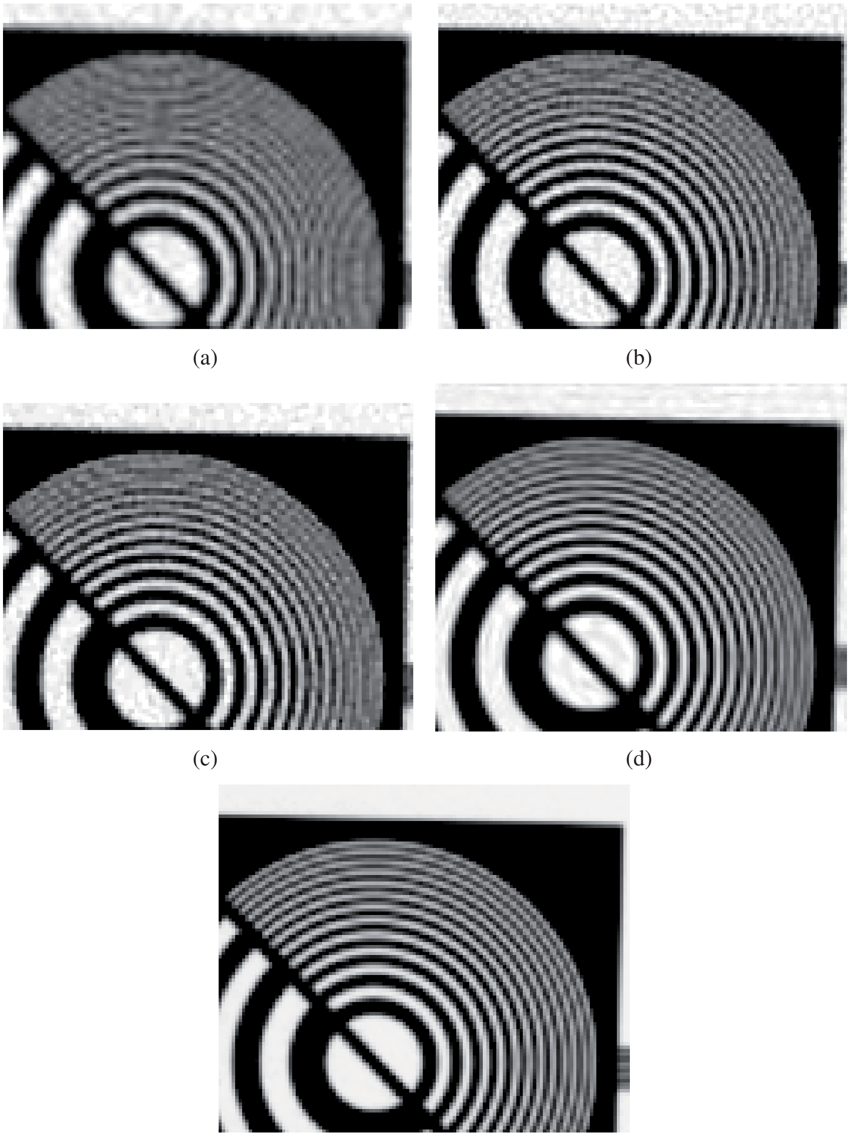


Figure 5.6: High-frequency contents of each reconstruction are given (a) Bicubic Interpolation, (b) GMRF, (c) RSR, (d) Proposed, (e) Original Image.



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

Figure 5.7: (a) Bicubic Interpolation, (b) GMRF, (c) RSR, (d) Proposed, (e) Original Image.

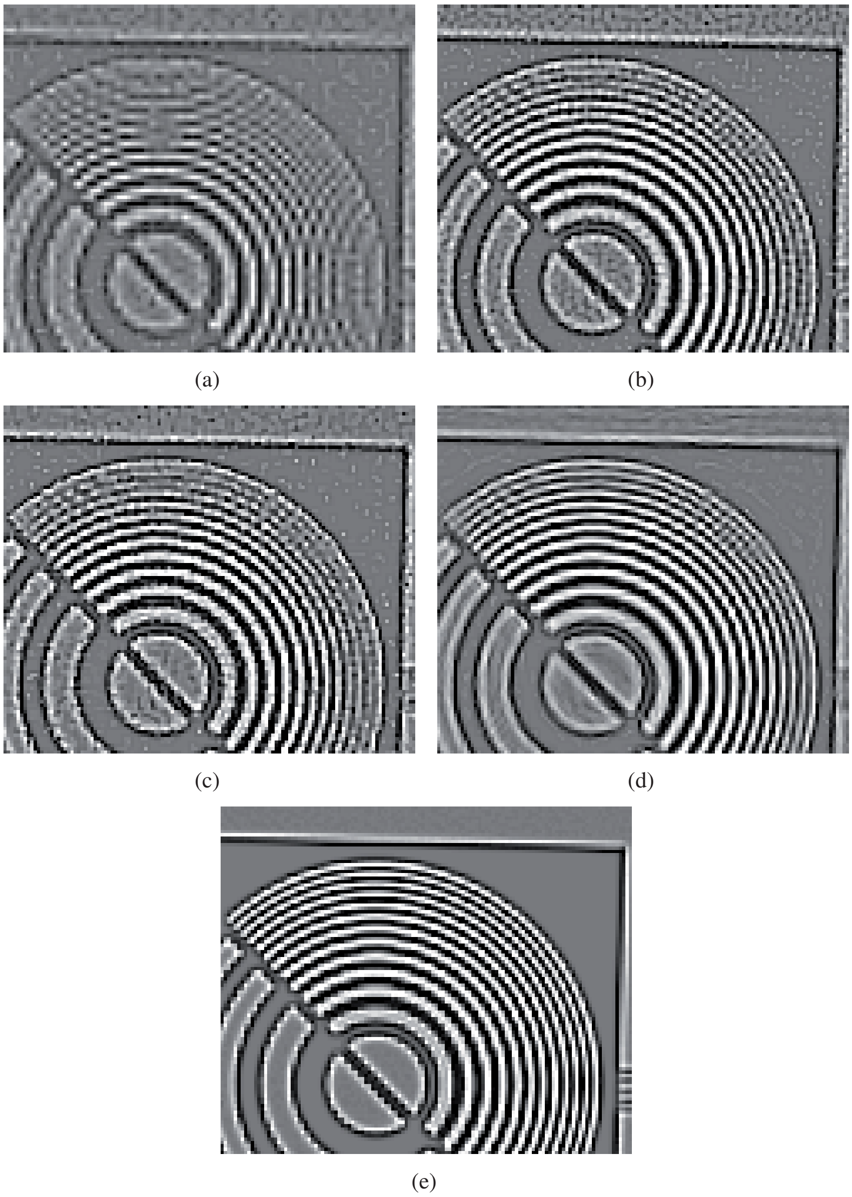
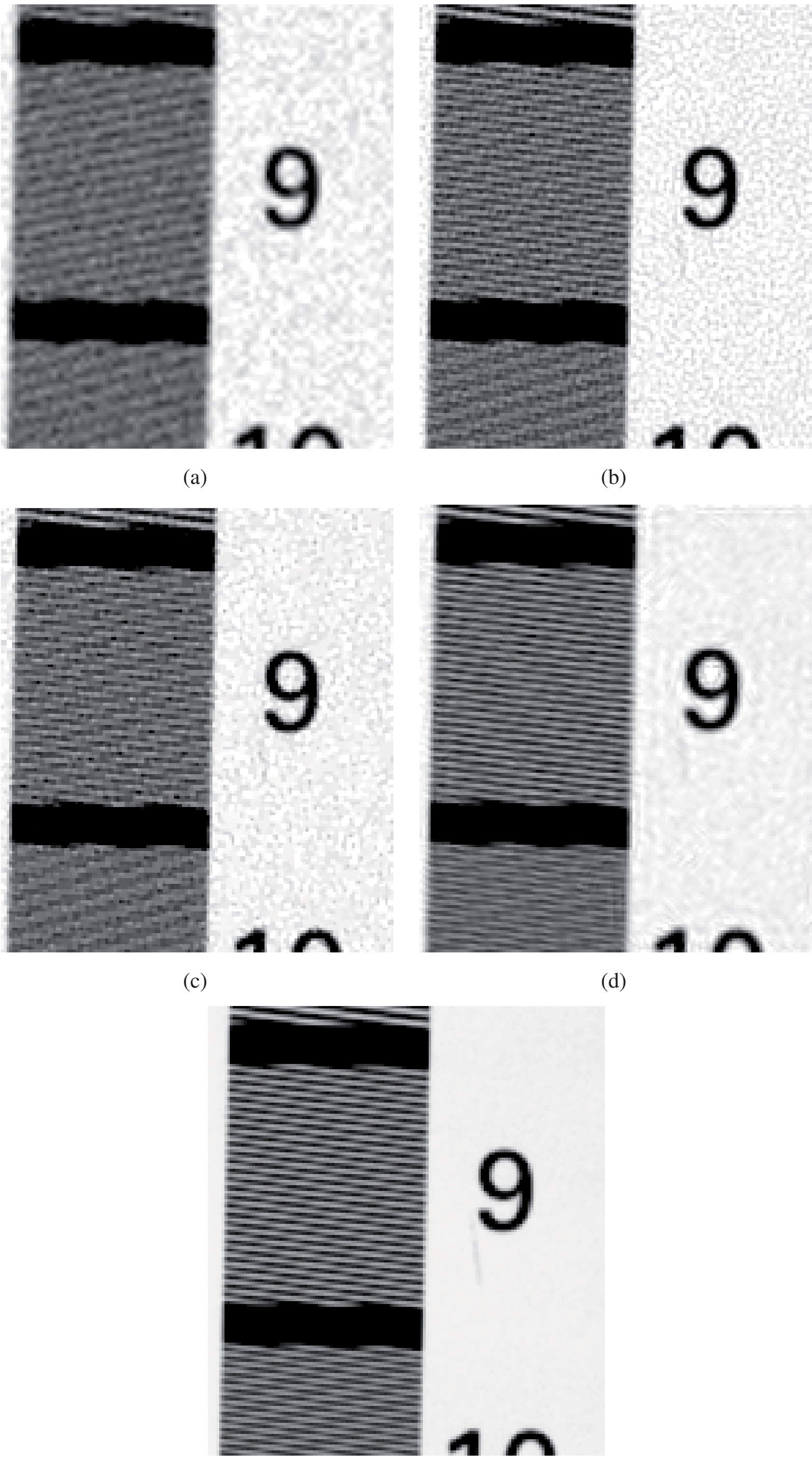
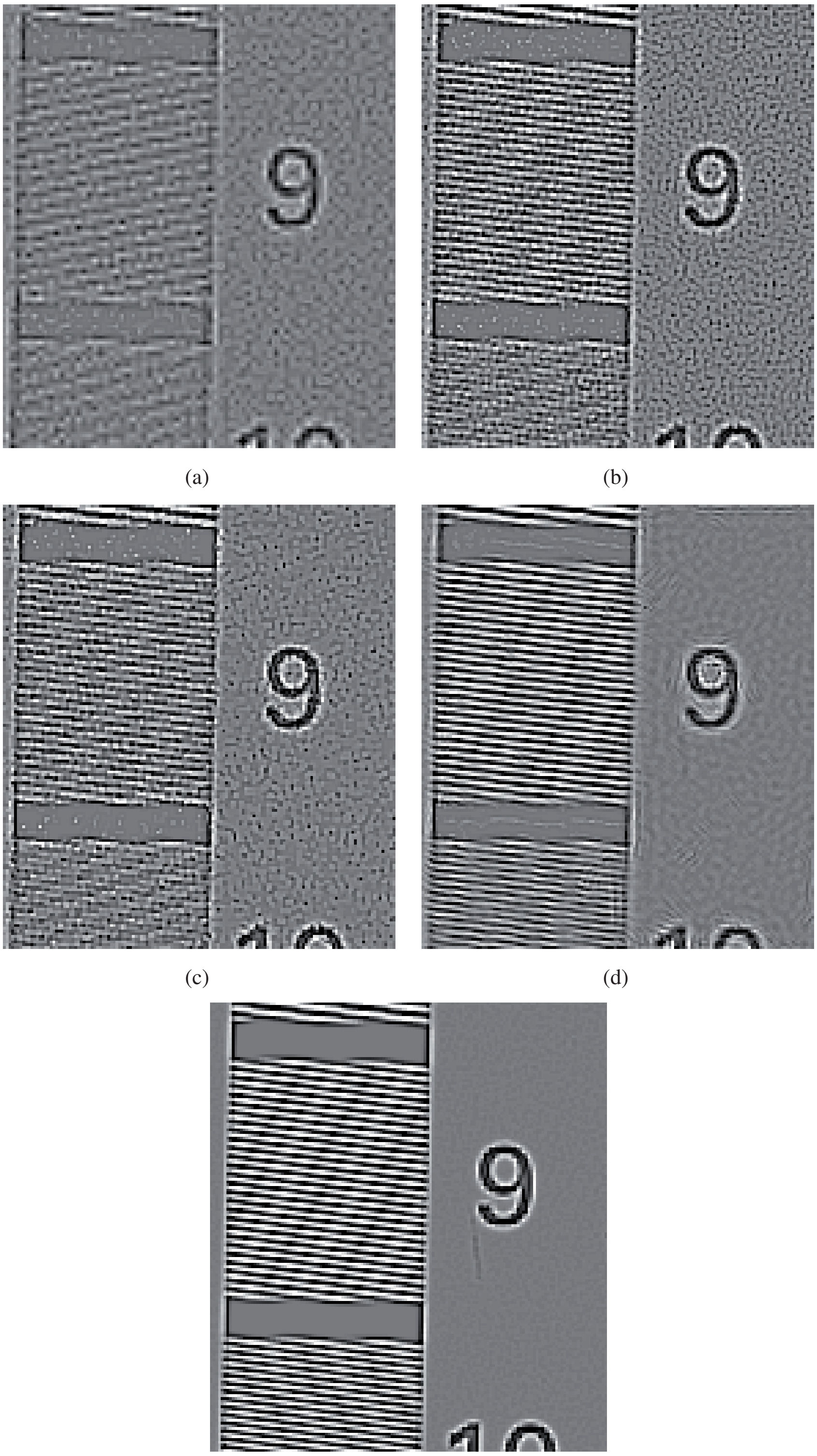


Figure 5.8: High-frequency contents of each reconstruction are given (a) Bicubic Interpolation, (b) GMRF, (c) RSR, (d) Proposed, (e) Original Image.



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

Figure 5.9: (a) Bicubic Interpolation, (b) GMRF, (c) RSR, (d) Proposed, (e) Original Image.



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

Figure 5.10: High-frequency contents of each reconstruction are given (a) Bicubic Interpolation, (b) GMRF, (c) RSR, (d) Proposed, (e) Original Image.

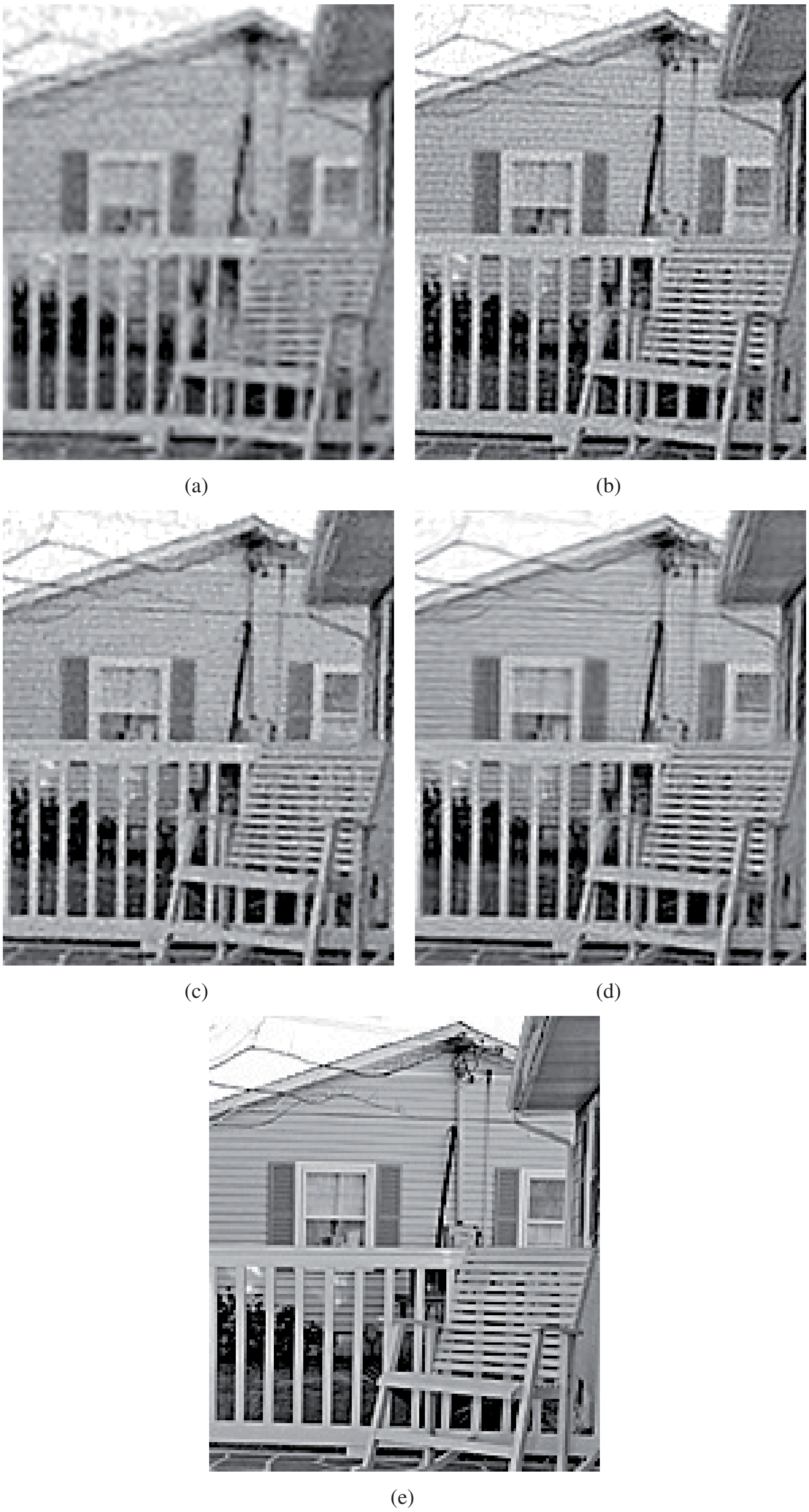
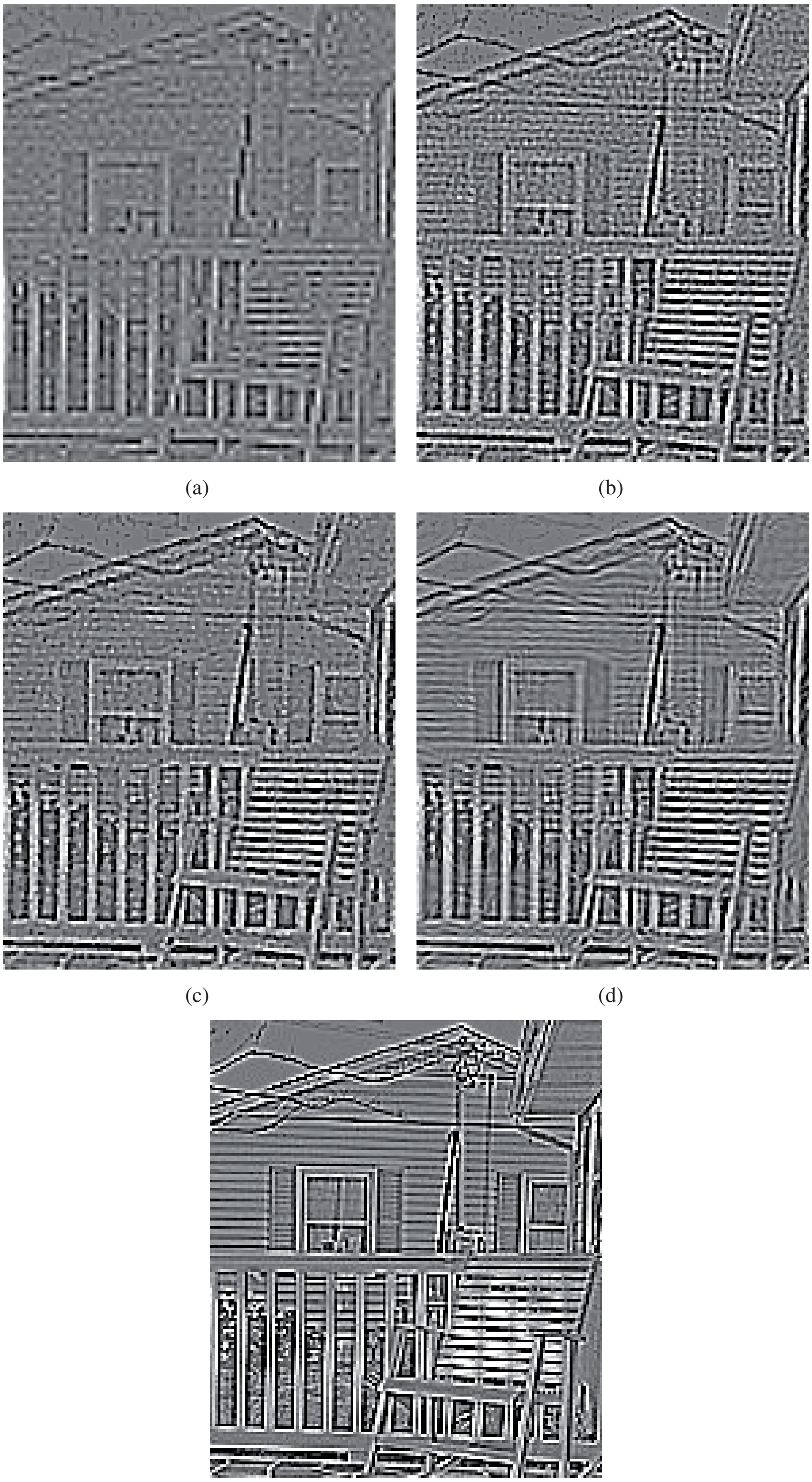
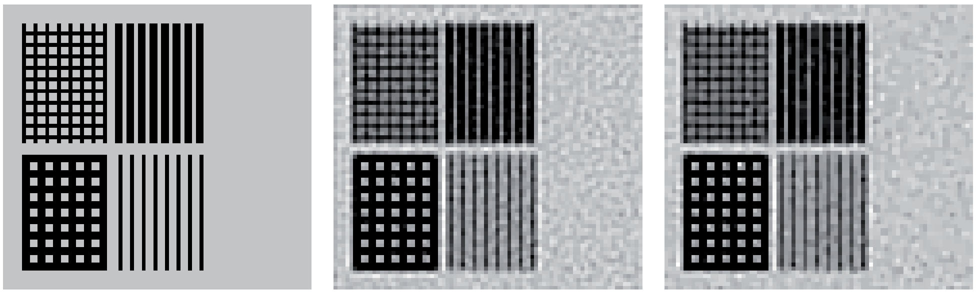


Figure 5.11: (a) Bicubic Interpolation, (b) GMRF, (c) RSR, (d) Proposed, (e) Original Image.

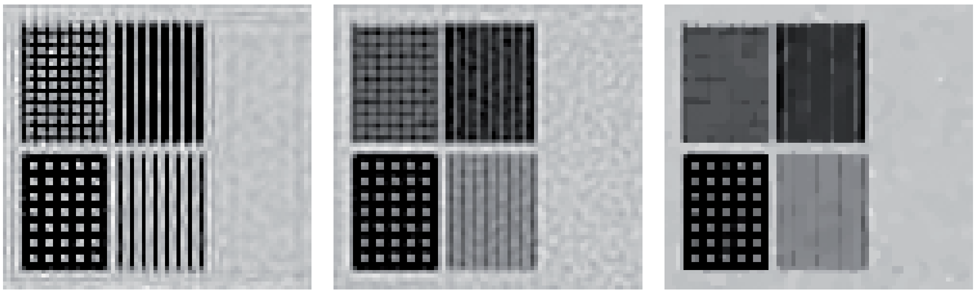


(a) (b) (c) (d) (e)

Figure 5.12: High-frequency contents of each reconstruction are given (a) Bicubic Interpolation, (b) GMRF, (c) RSR, (d) Proposed, (e) Original Image.



(a) Original Image (b) GMRF with $\lambda = 0.1$ (c) RSR with $\lambda = 0.1$,



(d) Proposed method (e) GMRF with $\lambda = 0.3$ (f) RSR with $\lambda = 0.3$

Figure 5.13: Effect of regularization on highly textured and uniform regions for the GMRF and RSR methods.

As expected, all reconstructed images have better visual quality, when compared with bicubic interpolation. The MSE and SSIM values for the best possible reconstructions are given in Tables 5.1 and 5.2. A total of 20 experiments have been conducted at each noise level. The mean and standard deviation of the metrics are given in Tables 5.1 and 5.2. The proposed method has higher SSIM and lower MSE values, when compared with GMRF and RSR estimates for both 25 and 30 dB noise levels. The success of the proposed method is visually apparent, especially on the textured regions, such as the texture on the clothing in Figure 5.5, the higher frequency lines in Figures 5.7 and 5.9, and the front surface of the background house in Figure 5.11.

The following simulation is conducted on a synthetic image shown in Figure 5.13(a). The image is composed of vertical and horizontal line patterns with periods: 3 pixels-per-cycle in horizontal and vertical directions in the top left quadrant; 3-pixels-per-cycle in horizontal direction in the top right and bottom right quadrants, and 4 pixels-per-cycle in two directions in the bottom left quadrant. The simulation setup is the same as the ones that are conducted before. For the GMRF and RSR methods, if λ is optimized for the textured region, MSE at the uniform region increases, whereas if λ is optimized for the uniform region, MSE at the textured region

Table 5.1: MSE Values at 25 and 30 dB SNR Levels

Image	SNR	MSE			
	(dB)	BICUBIC	GMRF	RSR	Proposed
Barbara	25	435.8±1	183.6±15	181.6±18	101.8±8
	30	411.1±0	113.2±24	115.9±20	87.7±20
Chart Sec1	25	570.9±3	117.3±11	151.2±13	87±13
	30	533.1±2	82.1±16	94.5±16	64.8±18
Chart Sec2	25	618.7±2	244.5±48dB	280.7±46	126.5±38
	30	560.7±1	142.5±41	152.5±42	72.2±33dB
House	25	949.8±2	378.6±28	374.4±34	359.9±33
	30	943.1±2	277.2±12	265.5±17	256.6±15

Table 5.2: SSIM Values at 25 and 30 dB SNR Levels

Image	SNR	MSE			
	(dB)	BICUBIC	GMRF	RSR	Proposed
Barbara	25	0.646±0.002	0.819±0.008	0.817±0.01	0.899±0.005
	30	0.693±0.001	0.884±0.013	0.881±0.012	0.921±0.012
Chart Sec1	25	0.685±0.005	0.893±0.004	0.886±0.006	0.926±0.005
	30	0.742±0.004	0.941±0.006	0.942±0.006	0.958±0.006
Chart Sec2	25	0.479±0.004	0.648±0.025	0.666±0.023	0.85±0.022
	30	0.595±0.003	0.773±0.021	0.828±0.02	0.912±0.018
House	25	0.575±0.002	0.822±0.015	0.81±0.017	0.841±0.015
	30	0.608±0.001	0.873±0.005	0.879±0.007	0.895±0.006

increases. This is demonstrated in Figure 5.13(b), (c), (e), and (f), where the GMRF and RSR reconstructions at λ set to 0.1 and 0.3 are given. Proposed method has a good balance at both the highly textured and uniform regions. Although the GMRF and RSR reconstructions in Figure 5.13(b) and (c) are optimized for the textured region, only the proposed method has recovered the bars having 3 pixel-per-cycle period in Figure 5.13(d). Standard regularizers work on a 3×3 neighborhood, which is not wide enough to differentiate the true high frequency variations and noise. The Gabor filters, on the other hand, can easily detect the underlying pattern under heavy noise, because these filters span a 32×32 pixel neighborhood at different scales in the proposed method.

5.3.2 Experiments on Real Image Data

A Samsung camera with 768×1024 pixel size is used in the experiments. Two image sequences, each having seven images of size 100×100, are obtained. The relative motion among images is estimated using the global optical flow based approach described in [20]. The blurring kernel in (5.2) is selected as 5×5 Gaussian, whose standard deviation is heuristically estimated to be 1.0. The resolution is increased by a factor of 2. Regularization parameters are manually tuned to obtain the most visually appealing reconstructions for the RSR, GMRF, and proposed method.

Four of the seven images are shown in Figure 5.14 as well as the pixel duplication results. Figures 5.15 and 5.17 presents the reconstructions for the GMRF, RSR, and proposed method. High-frequency components of the reconstructed images are also given in Figures 5.16 and 5.18 to compare the texture and edge recovery performances. The reconstruction of the proposed method has sharper textures and edges, when compared with RSR and GMRF. The success of the proposed method is clearer on the high-frequency texture at the bottom of each reconstruction in Figure 5.17. Although the proposed algorithm is not designed for images with text, the results in Figure 5.15 are sharper than the RSR and GMRF reconstructions.

The reconstructions are run on a laptop with a 1.8-GHz dual-core Intel Ivy Bridge Core i5. For GMRF method, one iteration takes around 0.5 s for GMRF method for seven 100×100 images and for the proposed method, It takes around 1 s.

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

Figure 5.14: (a) and (c) Four of the seven LR images, (b) and (d) 2× resolution increase by pixel duplication.

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

Figure 5.15: (a) Bicubic interpolation, (b) GMRF, (c) RSR, (d) Proposed Method.

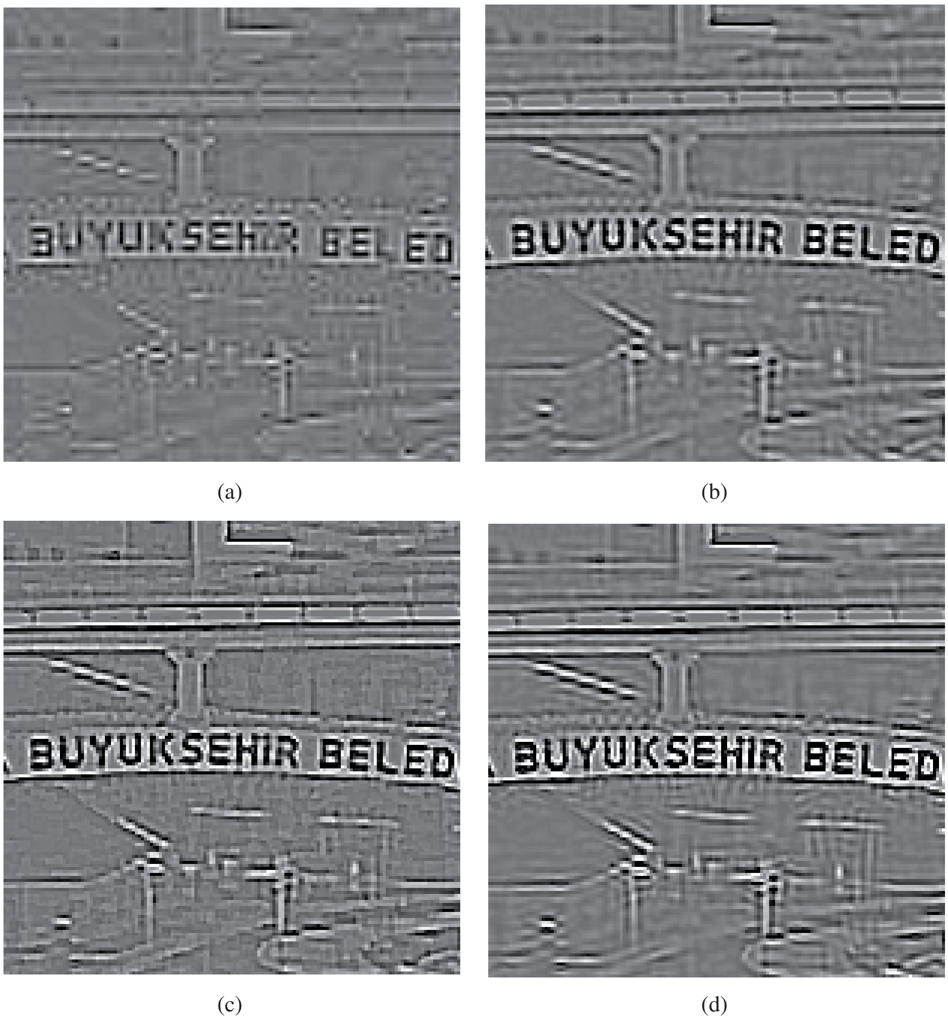
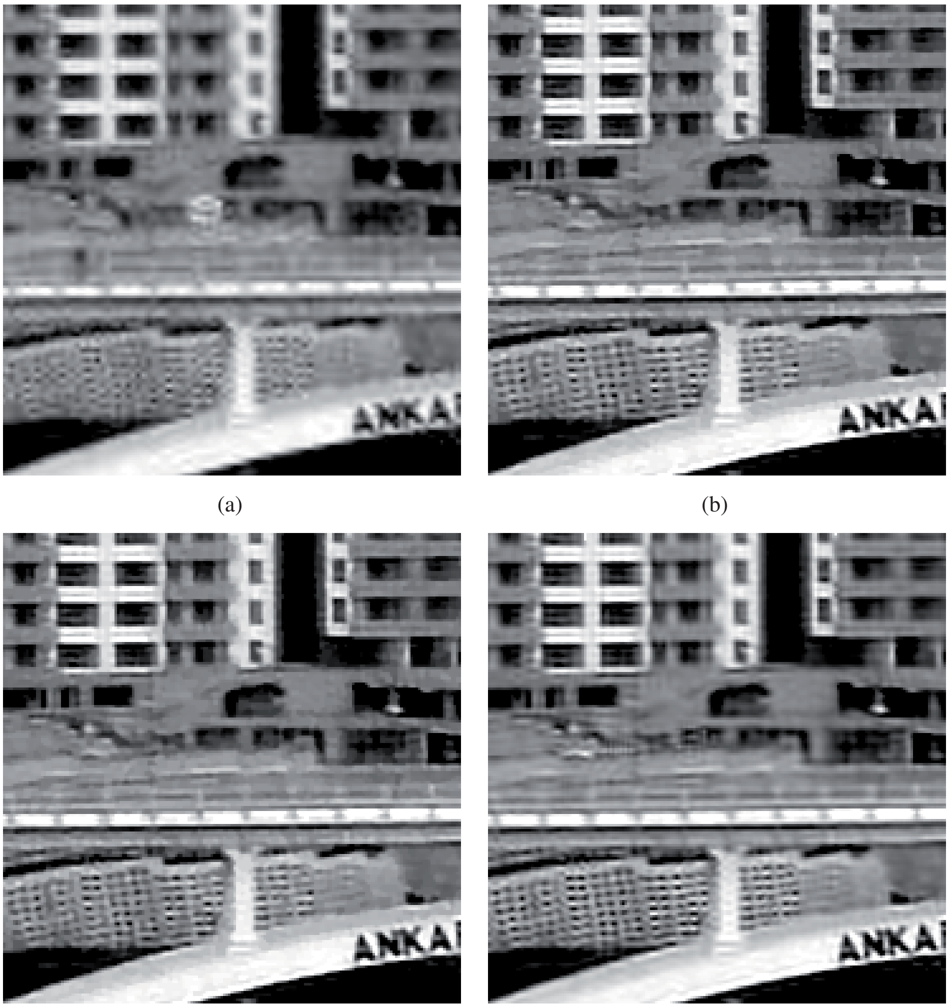


Figure 5.16: High-frequency content of reconstructions (a) Bicubic interpolation, (b) GMRF, (c) RSR, (d) Proposed Method.



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

Figure 5.17: (a) Bicubic interpolation, (b) GMRF, (c) RSR, (d) Proposed Method.

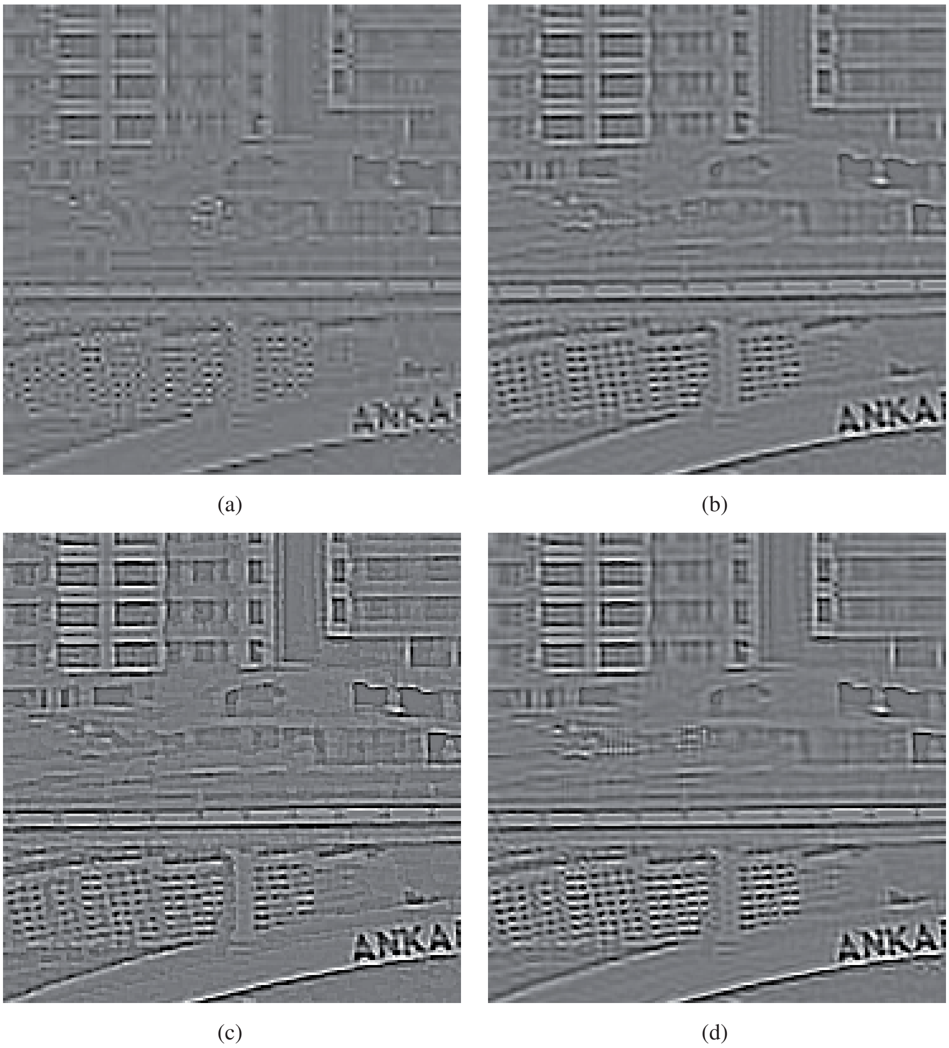


Figure 5.18: High-frequency content of reconstructions (a) Bicubic interpolation, (b) GMRF, (c) RSR, (d) Proposed Method.

5.3.3 Experiments on Thermal Image Data

An uncooled thermal camera with 144×176 pixel detector size is used in the experiments. 20 frames are used in the reconstruction. Relative motion is estimated as described in, [20]. The resolution is increased by a factor of 2 in horizontal direction and 4 in vertical direction. This asymmetric resolution increase is due to the image acquisition system. The output video is composed of odd and even lines of the generated images. The monitor combines odd and even lines into a full image. However when motion is present, odd and even lines shift with respect to each other. To overcome this problem, each frame is separated into its even and odd line images and treated as separate LR images during SR reconstruction. Some of these LR images are shown in Figure 5.19. The blurring kernel in Eq. 5.2 is selected as 5×10 Gaussian filter with heuristically estimated standard deviation of 1.0. The regularization parameter for RSR, GMRF, and the proposed method are manually tuned to obtain the most visually appealing reconstructions, given in Figures 5.20 and 5.22. High frequency components of the reconstruction are also given in Figures 5.21 and 5.23 to compare the reconstructions on regions with textures and edges.

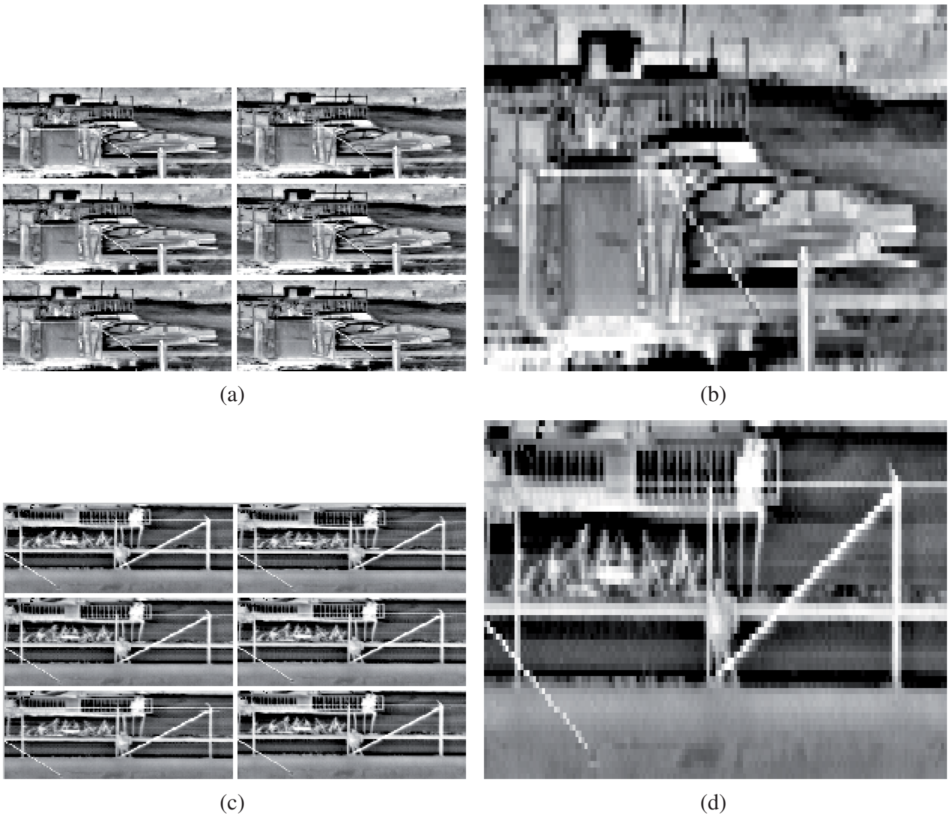
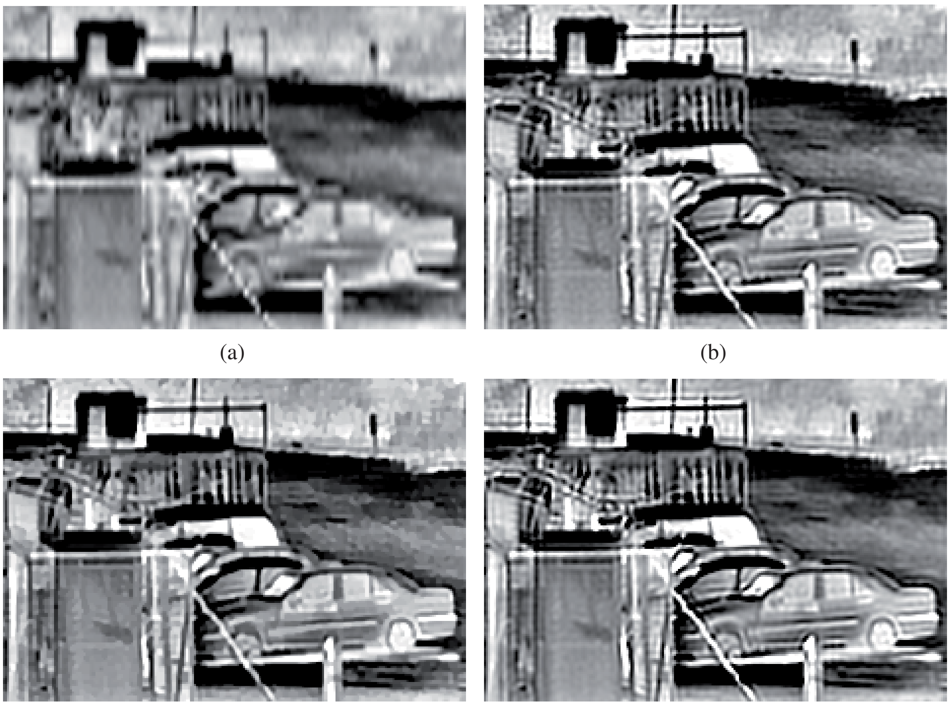


Figure 5.19: (a) 6 of 20 LR images of the image set 1, (b) 2x resolution increase by pixel duplication. (c) LR images of the image set 2, (d) 2x resolution increase by pixel duplication.



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

Figure 5.20: Reconstructions for the image set. A 120×150 window is selected for better visual inspection. (a) Bicubic, (b) GMRF, (c) RSR, (d) Proposed.

(a)

(b)

(c)

(d)

Figure 5.21: Reconstructions for the image set. A 120 × 150 window is selected for better visual inspection. High frequency content of (a) Bicubic, (b) GMRF, (c) RSR, (d) Proposed.

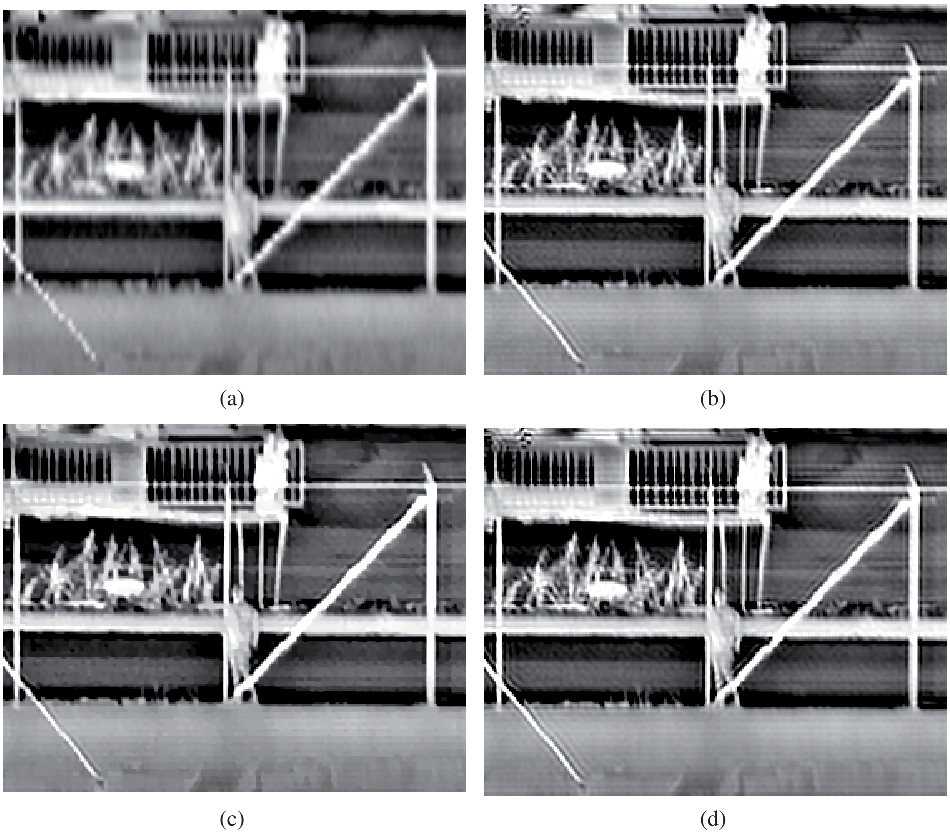


Figure 5.22: Reconstructions for the image set. A 120×150 window is selected for better visual inspection. (a) Bicubic, (b) GMRF, (c) RSR, (d) Proposed.

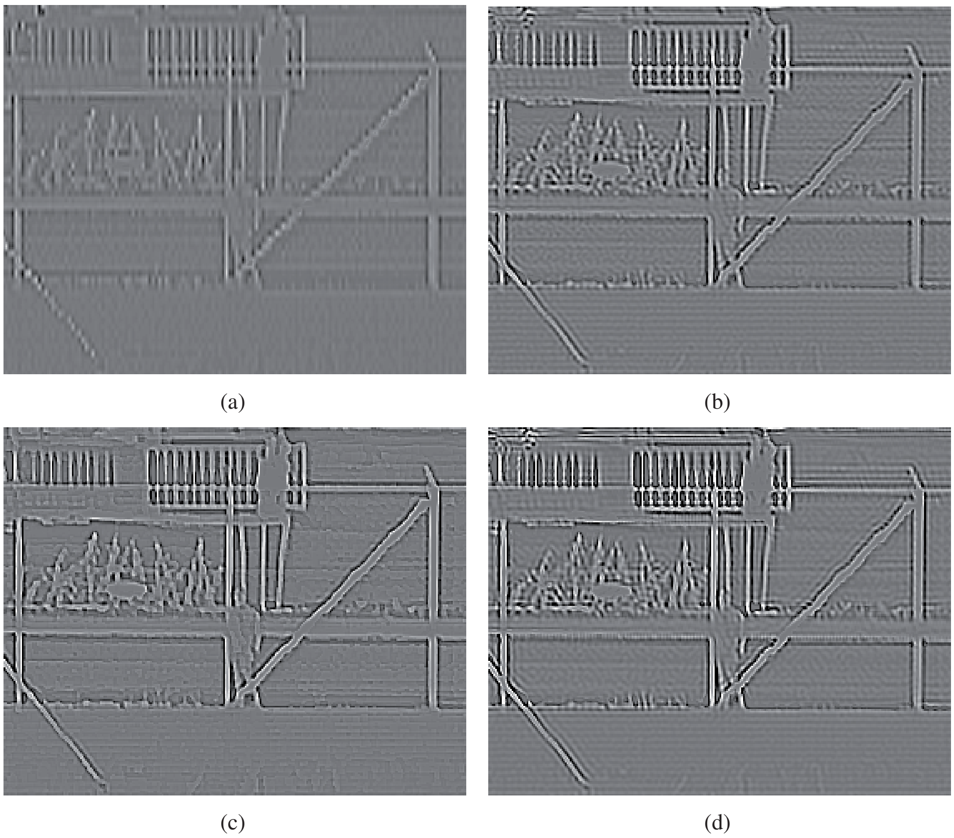


Figure 5.23: Reconstructions for the image set. A 120×150 window is selected for better visual inspection. High frequency content of (a) Bicubic, (b) GMRF, (c) RSR, (d) Proposed.

5.4 DISCUSSION

Various texture models have been proposed in texture segmentation and synthesis literature. Repetition of structures in a texture at various resolutions makes multiscale models good candidates in modeling texture behavior. Moreover, such models employ multiscale decomposition to model complex interactions on a large neighborhood without increasing the number of model parameters astronomically. In this study, a multiscale texture model was used for the first time in solving super-resolution image reconstruction problem. Furthermore, we extended the multiscale model by embedding a binary process in each scale of the model. The overall model is easy to implement and intuitive. It has been shown that the proposed model has sharper reconstructions, when compared with the standard BTV and GMRF priors, especially on images with high-frequency texture.

CHAPTER 6

POST PROCESSING BASED SUPER RESOLUTION

6.1 INTRODUCTION

In this chapter, texture preserving regularization is handled as a post-processing step. A two stage method is proposed, comprising multiple SR reconstructions with different regularization parameters followed by a restoration step for preserving edges and textures. In the first stage, two maximum-a-posteriori (MAP) estimators with two different amounts of regularization are employed. In the second stage, pixel-to-pixel difference between these two estimates is post-processed to restore edges and textures. Frequency selective characteristics of discrete cosine transform (DCT) and Gabor filters are utilized in the post-processing step.

The main contribution of this study is the idea of running two MAP estimators with different amount of regularization in parallel to obtain two solutions that can be considered as the two extremes in the SR solution space. Post processing the difference of these extremes to preserve edges and textures, to the best of the authors' knowledge, has not been considered in SR image reconstruction before. The second contribution of this study is using DCTs and Gabor filters to restore edge and texture information during SR reconstruction.

This chapter is organized as follows. Section 6.2 presents the proposed methods along with the standard GMRF solution. Section 6.3 presents experiments on synthetically generated images to demonstrate the mean squared error (MSE) and structural similarity index measure (SSIM) improvement compared to the Bayesian methods given in [13, 25]. Performances are also visually validated through real experiments on visual CCD images and thermal images. Finally, Section 6.4 has a discussion on the results.

6.2 METHODOLOGY

The general MAP solution given in Eq. 5.1 comprises a regularization term represented in $\Gamma(\cdot)$ function. It is previously mentioned that priors such as HMRF [40, 87], TV [91, 88], BTV[25], or compound MRFs with line process [85, 86] are used for sharper estimates. In this work, GMRF is considered as image prior. $\Gamma(\cdot)$ function comprises the square sum of local pixel differences, which can be simplified to a Laplacian operator matrix **C** as described in [13, 80]. As mentioned in the previous section, this leads to smoother estimates.

$$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} ||\mathbf{W}_k\mathbf{z} - \mathbf{y}_k||_2^2 + \lambda||\mathbf{C}\mathbf{z}||_2^2 \tag{6.1}$$

where λ controls the influence of the regularization to the solution. Setting λ to zero simplifies the above equation to maximum likelihood solution where the noise is amplified while the details are preserved. Increasing λ will increase the effect of the GMRF prior; as a result, the differences among neighboring pixels will be penalized more and the high frequency content of the reconstructed image will decrease.

Edge preserving regularizers [40, 25, 92] are known to corrupt high frequency textures in images [93]. Textures usually have rapid low amplitude variations. The classical regularizers are usually defined for 8-pixel-neighborhood system which is not sufficient to meet the periodicity and regularity of textures. As a result, classical regularizers usually smooth out texture information. In Chapter 5, a texture prior is proposed to solve this problem. In this chapter, a post-processing approach is preferred for texture and edge preservation as an alternative.

The proposed method comprises multiple SR reconstructions with different regularization parameters followed by a restoration step. In the first stage, two MAP estimates given in (6.1) are utilized with regularization parameters λ_1 and λ_2. Setting $\lambda_1 < \lambda_2$ creates a noisy and an over-smoothed SR estimates respectively, as depicted in Figure 6.1. These two solutions can be considered as two extremes in the SR solution space. The pixel-to-pixel difference of these solutions is a high frequency image that is composed of edges, texture fragments, and high frequency noise as seen in Figure 6.1.

$$\mathbf{z}_{DIFF} = \hat{\mathbf{z}}_{MAP1} - \hat{\mathbf{z}}_{MAP2}. \tag{6.2}$$

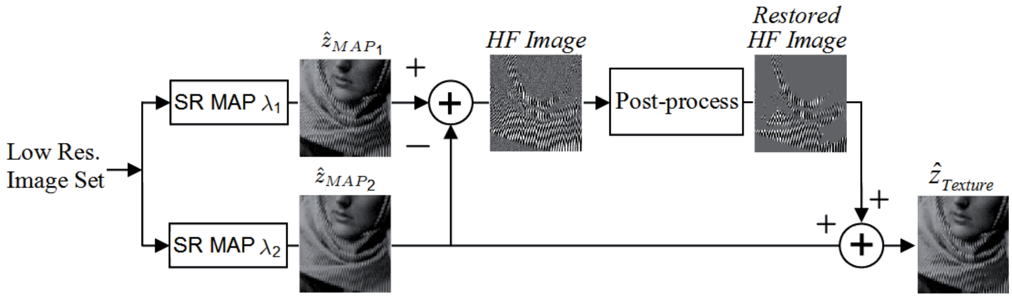


Figure 6.1: Block diagram of the proposed texture preserving super resolution reconstruction method. *SR MAP* λ_1, and *SR MAP* λ_2 refer to SR methods given in (6.1) with two different regularizations. According to the figure $\lambda_2 > \lambda_1$.

where $\hat{\mathbf{z}}_{MAP1}$ and $\hat{\mathbf{z}}_{MAP1}$ are MAP estimates with regularization parameters λ_1 and λ_2. The difference image is restored at multiple frequencies as given in (6.3).

$$\hat{\mathbf{z}} = \hat{\mathbf{z}}_{MAP2} + N_1 F_1 \mathbf{z}_{DIFF} + N_2 F_2 \mathbf{z}_{DIFF} + \cdots + N_c F_c \mathbf{z}_{DIFF} \quad k = 1, \ldots, c. \qquad (6.3)$$

Here F_k is a linear operation for extracting k^{th} frequency component of $\mathbf{z}_{DIFF}$. N_k is a diagonal matrix with entries n_i (i being the image pixel index) being either 1 or 0. N_k can be considered as a thresholding operator for selecting pixels corresponding to k^{th} frequency band. The proposed solution given in (6.3) is realized in two alternative ways: Gabor filter bank based or DCT based methods.

6.2.1 Gabor Filter Bank Based Restoration

Th same Gabor filter family used in Chapter 5 is also employed in the determination of local texture/edge activity. The magnitude of Gabor filter response at a pixel location is proportional to the presence of an edge [81] or texture [82] aligned with the filter orientation. Using this idea, (6.3) is rewritten as follows:

$$\hat{\mathbf{z}} = \hat{\mathbf{z}}_{MAP2} + \mathbf{z}_{F_1} + \mathbf{z}_{F_2} + \ldots + \mathbf{z}_{F_c} \quad k = 1, \ldots, c. \qquad (6.4)$$

where each pixel of $\mathbf{z}_{F_k}$ term is approximated as follows:

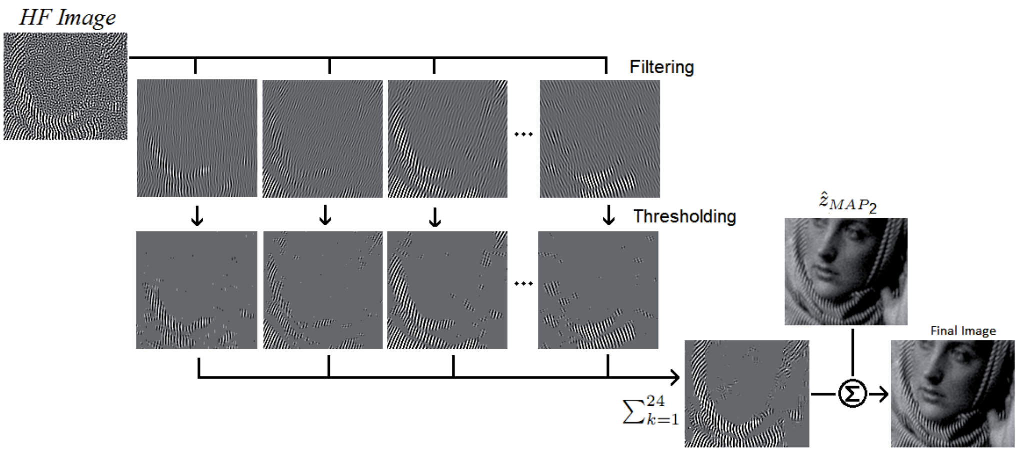


Figure 6.2: Intermediate images obtained through the proposed Gabor Filter Bank Based method.

$$\mathbf{z}_{F_k}(i) = \begin{cases} F_k \mathbf{z}_{DIFF}(i) & F_k \mathbf{z}_{DIFF}(i) > T \\ 0 & else \end{cases} \tag{6.5}$$

Here, i is the pixel index, F_k is a matrix operator ($k = 1 \cdots 24$) representing the Gabor filters. The size of each filter is 32 by 32. The extent of each Gabor filter is dependent on the scale such that 5 lobes are visible at each scale.

T term in (6.5) thresholds texture/edge activity at the corresponding pixel. It realizes N_k matrix referred to in (6.3). T is adaptively chosen according to the standard deviation of all pixels in $\mathbf{z}_{DIFF}$. Thresholding is also necessary to suppress ringing artifacts especially on strong edges after filtering. Figure 6.2 illustrates the effect of thresholding. Filtered and thresholded images are summed up in (6.6) to obtain restored $\hat{\mathbf{z}}_{DIFF}$, abbreviated with a hat.

$$\hat{\mathbf{z}}_{DIFF} = \sum_{k=1}^{24} \mathbf{z}_{F_k}. \tag{6.6}$$

Restored difference image $\hat{\mathbf{z}}_{DIFF}$ is added to the over-smoothed SR estimate $\hat{\mathbf{z}}_{MAP2}$ to obtain the final image.

$$\hat{\mathbf{z}} = \hat{\mathbf{z}}_{MAP2} + \hat{\mathbf{z}}_{DIFF}. \tag{6.7}$$

6.2.2 Discrete Cosine Transform Based Restoration

Discrete cosine transform is widely used in texture classification, texture feature extraction, and representation [94]. High frequency DCT coefficients are employed for analyzing variation, regularity, complexity, and directional information of textures [94]. High frequency DCT coefficients are also analyzed in the proposed method. The proposed DCT based method is based on two assumptions: Signal level of texture is stronger than noise; DCT coefficients of texture gather around close vicinities in DCT domain. Based on these assumptions, $\mathbf{z}_{DIFF}$ term is divided into non-overlapping blocks. Each block is restored by two band pass filters. These filters are defined in transform domain with center frequencies set as the locations of the peaking DCT coefficients. Practical reasons for selecting only two band-pass filters are discussed later in this section.

A similar process, called zone filtering [95], is used in DCT based image restoration. However, in Zone filtering, the mask is fixed for the whole image, it is connected, and it must include the DC coefficient. In the proposed method, the mask varies for each block, and it is centered around the strongest DCT coefficients. The zone filter in [95] discards the high frequency DCT coefficients. However, high frequency DCT coefficients are mostly preserved in the proposed method. Note that since $\mathbf{z}_{DIFF}$ is a high frequency image, the strong DCT coefficients are expected to gather in the high frequency region.

The proposed DCT based method approximates the realization of (6.3). Let $\mathbf{B}_k$ be the DCT of 8×8 sized non-overlapping blocks in $\mathbf{z}_{DIFF}$. The algorithm for modifying $\mathbf{B}_k$ is given as follows:

1. Let $\lambda_k^{max} = \mathbf{B}_k(o,l)$ *s.t.*

$$\forall (x,y) \neq (o,l)\,,\ |\mathbf{B}_k(o,l)| > |\mathbf{B}_k(x,y)|$$

   where (x,y) and (o,l) are coordinates in transform domain.

2. Define a region R_k^a in coefficient space such that

$$R_k^a = \{(x,y)_k \ \mid \|(x,y)_k - (o,l)_k\| \leq D\}\,.$$

D is related to the bandwidth of the band-pass filters.

3. Let $\lambda_k^{max2} = \mathbf{B}_k(m,n)$, $(m,n)_k \notin R_k^a$ $s.t.$

$$\forall(x,y) \neq (m,n) \; and \; (x,y) \notin R_k^a \; , \; |\mathbf{B}_k(m,n)| > |\mathbf{B}_k(x,y)|$$

4. Define R_k^b s. t.

$$R_k^b = \left\{ (x,y)_k \; \middle| \left| \begin{cases} ||(x,y)_k - (m,n)_k|| \leq D & \lambda_k^{max2} > \lambda_k^{max}/2 \\ \emptyset & else \end{cases} \right. \right\}$$

5. Set

$$\mathbf{B}_k^R(x,y) = \begin{cases} \mathbf{B}_k(x,y) & (x,y) \in \left(R_k^a \cup R_k^b\right) \\ 0 & else \end{cases} .$$

Here $\mathbf{B}_k^R$ represents the k^{th} restored block. Let $\mathbf{z}_{IDCT}$ be image obtained from $\mathbf{B}_k^R$. Thresholding is applied to suppress ringing artifacts in (6.8).

$$\hat{\mathbf{z}}_{DIFF}(i) = \begin{cases} \mathbf{z}_{IDCT}(i) & \sigma_{N_i} > \sigma_{all} \\ 0 & else \end{cases} \quad (6.8)$$

where σ_{all} refers to the standard deviation of all pixels in $\hat{\mathbf{z}}_{MAP2}$, and i refers to pixel index. σ_{N_i} is the standard deviation at a 3×3 neighborhood of the i^{th} pixel location. $\hat{\mathbf{z}}_{DIFF}$ is the restored version of $\mathbf{z}_{DIFF}$. Similarly, restored difference image $\hat{\mathbf{z}}_{DIFF}$ is added to the over-smoothed SR estimate $\hat{\mathbf{z}}_{MAP_2}$ to obtain the final image.

$$\hat{\mathbf{z}} = \hat{\mathbf{z}}_{MAP2} + \hat{\mathbf{z}}_{DIFF}. \tag{6.9}$$

Unlike the proposed Gabor based method, DCT based method is an approximation for (6.3). Using 8×8 window size is a common practice in video processing and video compression literature. The bandwidth of the band-pass filters, D, could take limited values (1, 2, 3). Setting $D = 3$ results in an all-pass filter if the center of the filter is selected as the center in the DCT domain. Setting $D = 2$ is a practical choice determined through simulations. With $D = 2$, the span of two filters in total can be

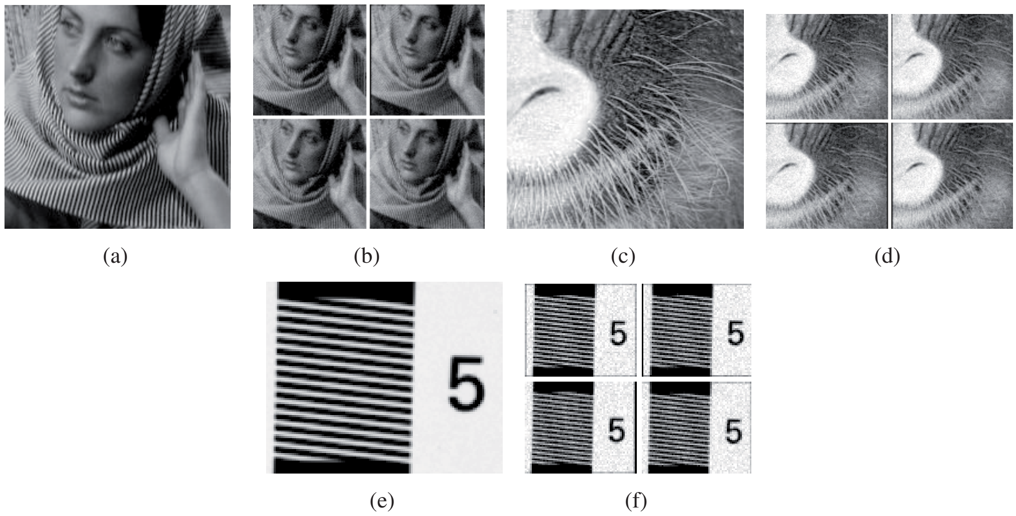


Figure 6.3: (a) Original high resolution image *Barbara* and (b) four of eight synthetically generated LR images. (c) Original high resolution image *Baboon* and (d) four of eight synthetically generated LR images. (e) Original high resolution image *Chart* and (f) four of eight synthetically generated LR images.

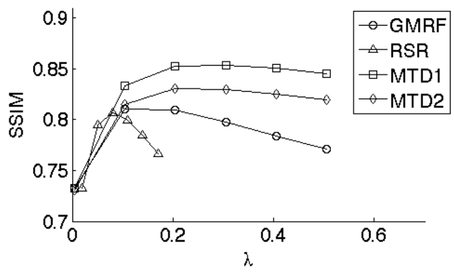
maximum 5×5×2. This number is already more than half of the DCT domain (8×8). Therefore, using one or two band-pass filters is optimum. This approach may cause blocking artifacts as discussed in the results section. Using overlapping windows is an option to deal with this artifact, which will compromise computation time.

6.3 RESULTS

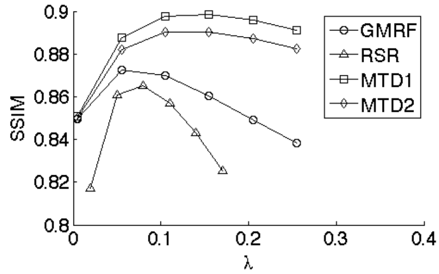
In this section, the performances of the proposed methods are demonstrated on both synthetically generated images and real images. The proposed methods are compared against single frame bicubic interpolation, MAP estimate with Gaussian Markov Random Field image prior, and method given in [25]. The reasons for selecting method in [25] are its robustness to motion estimation errors and its edge preservation property.

MAP estimate with GMRF prior is obtained as described in (5.24). This method is abbreviated as GMRF throughout the chapter. Best possible λ is searched experimentally for this estimate. The method described in [25] is implemented as described in (5.26). Best possible λ is searched experimentally. This method is denoted as RSR in the rest of the chapter.

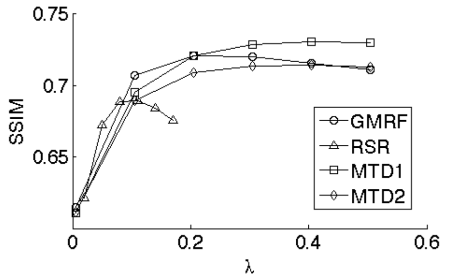
Proposed methods are based on restoring the difference of two MAP estimates with two different λ values. These two MAP estimators are implemented as described



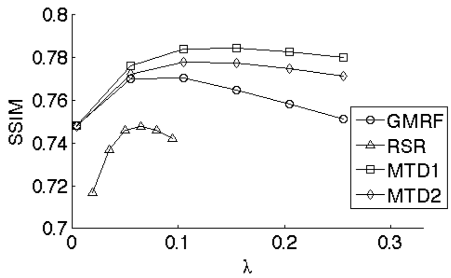
(a) ”Barbara” 25 dB noise



(b) ”Barbara” 30 dB noise



(c) ”Baboon” 25 dB noise



(d) ”Baboon” 30 dB noise

Figure 6.4: Effect of varying regularization amount λ versus SSIM values

in (5.24). Setting both λ_1 and λ_2 to a high value results in two smooth solutions. In this case, the difference image will have poor frequency content. On the other hand, as the difference between λ_1 and λ_2 increases, the content of the difference image increases. Intuitively, the best strategy is to maximize the high frequency content of the difference image. For this purpose, λ_1 is set to zero, simplifying the MAP estimator to maximum likelihood (ML) estimator. This selection does not risk the convergence of the solution, since the blur kernel in (6.1) acts as a regularizer during the reconstruction. After setting λ_1 to zero, the best possible λ_2 is searched experimentally. The proposed Gabor filter based and DCT based restoration methods are abbreviated as MTD1 and MTD2 respectively. The iterations are initialized with bicubic interpolation for all the SR methods. Experiments are conducted on synthetically generated images in the first section. Results for real experiments are given in the next section. The iterations are ceased if the following condition occurs:

$$\left\|\mathbf{z}^{n+1} - \mathbf{z}^{n}\right\| / \|\mathbf{z}^{n}\| \leq 10^{-6} \tag{6.10}$$

here, $\mathbf{z}^n$ is the reconstructed image at n^{th} iteration.

6.3.1 Experiments on Synthetically Generated Images

The proposed methods are tested against synthetically generated images given in Figure 6.3. These images are selected for their textured structure. Each 512 x 512 high resolution image is warped, blurred and down sampled by a factor of two to obtain eight 256 x 256 LR images according to Eq. 1.2. Gaussian noise at SNR levels of 25 dB and 30 dB are used to degrade the LR images. 4 of 8 LR images are also given in Figure 6.3 for 25 dB noise case. Only translational motion randomly set within 2 pixel range among LR images is used. True motion values are used during reconstructions. A 2x2 moving average filter is used to simulate blurring. The resolution is increased by a factor of 2. The quantitative performances are compared using MSE and SSIM measurements.

The amount of regularization, λ, is varied to achieve the highest possible SSIM value for both RSR and GMRF methods. By this way the best performances are compared against the proposed methods MTD1 and MTD2. The method proposed in Chapter 5 is also added to the tests and abbreviated as CHP5. The The effect of varying λ for GMRF, RSR, MTD1, and MTD2 reconstructions are given for both 25 dB and 30 dB noise levels in Figure 6.4. λ refers to the regularization amount given in (5.24) and (5.26) for GMRF and RSR respectively. For MTD1 and MTD2, λ refers to λ_2 of z_{MAP2}. It has been previously stated that λ_1 of z_{MAP1} is set to zero. The reconstructed images are given in Figures 6.5, 6.7, and 6.9. These reconstructions are filtered by a Laplacian filter in the form of [-1 -1 -1; -1 8 -1; -1 -1 -1] for better visual assessment. The resultant high frequency images for each reconstruction are given in the second row of Figures 6.6, 6.8, and 6.10. Both proposed methods have better visual quality than RSR and GMRF reconstructions when high frequency components are compared.

MSE and SSIM values for best possible reconstructions are given in Table 6.1. The proposed methods have higher SSIM and MSE values compared to GMRF and RSR estimates for both 25 dB and 30 dB noise levels. Block-wise post-processing step in MTD2 creates blocking artifacts in restored difference image $\hat{\mathbf{z}}_{MAP1} - \hat{\mathbf{z}}_{MAP2}$. This results in lower SSIM and higher MSE values compared to MTD1. Using overlapping blocks in MTD2 can reduce these artifacts. However, this increases the computational complexity of the method. MTD1 on the other hand restores the whole

Table 6.1: MSE and SSIM Values at 25 and 30 dB SNR Levels

Image	SNR (dB)	MSE					
		BICUBIC	GMRF	RSR	MTD1	MTD2	CHP5
Barbara	25	434.3±1dB	155.2±13dB	172.9±16dB	125.1±10dB	131.3±9dB	86.4±5dB
	30	414.9±1dB	108.6±8dB	117.1±8dB	91±7dB	89.6±7dB	60.6±4dB
Baboon	25	285.7±2dB	163.7±4dB	184.4±4dB	160.7±4dB	173.4±3dB	206.2±14dB
	30	254.2±0dB	126.4±4dB	135.3±4dB	115.2±4dB	119.6±3dB	120.9±6dB
Chart	25	817.6±6dB	104.3±8dB	162.1±11dB	81.9±5dB	104.9±9dB	86.3±8dB
	30	786.1±5dB	82.1±10dB	99.4±6dB	43.4±7dB	48.7±8dB	44.6±7dB
		SSIM					
		BICUBIC	GMRF	RSR	MTD1	MTD2	CHP5
Barbara	25	0.654±0.001	0.824±0.008	0.825±0.008	0.865±9.829	0.851±0.007	0.911±0.004
	30	0.696±0.001	0.889±0.005	0.887±0.005	0.911±7.263	0.908±0.005	0.941±0.003
Baboon	25	0.629±0.004	0.784±0.004	0.757±0.005	0.794±3.57	0.783±0.003	0.729±0.017
	30	0.675±0.001	0.841±0.004	0.825±0.004	0.852±3.794	0.848±0.004	0.843±0.007
Chart	25	0.808±0.003	0.924±0.002	0.939±0.004	0.958±4.586	0.955±0.001	0.933±0.003
	30	0.675±0.001	0.841±0.004	0.825±0.004	0.852±3.794	0.848±0.004	0.843±0.007

difference image hence it has smoother reconstructions. As λ approaches to zero, $\hat{\mathbf{z}}_{MAP2}$ approaches to the maximum likelihood estimate. As a result, their difference and the restored difference approach to a zero image. Therefore, MTD1 and MTD2 converges to GMRF as lambda approaches to zero. This can be observed in Figure 6.4. The method proposed in Chapter 5 has better performance on *Barbara* and *Chart* Image. However, the performance of CHP5 drops for the *Baboon* image. Visual inspection of the high frequency content of CHP5 reconstruction reveals ghost textures, which are formed due the Gabor filters. The patterns on the *fur* of the *baboon* are not regular enough for the Gabor filters.

(a) Bicubic Interpolation

(b) GMRF

(c) RSR

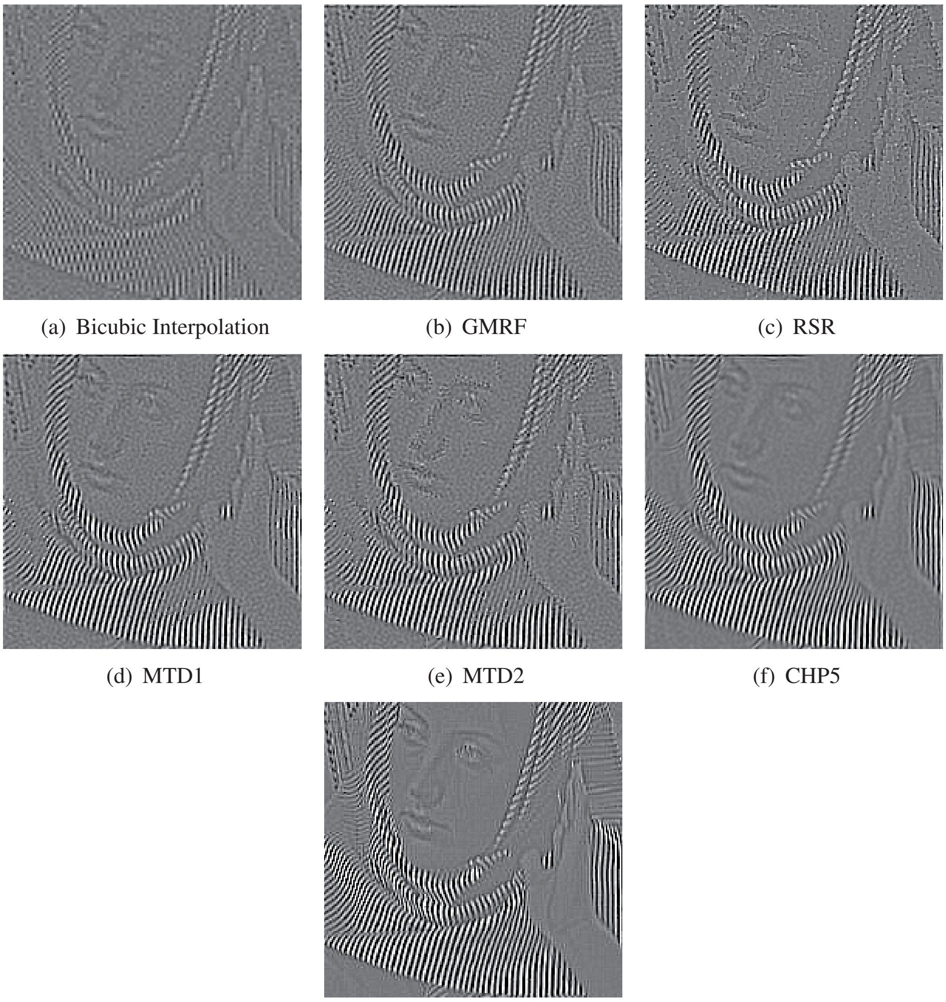
(d) MTD1

(e) MTD2

(f) CHP5

(g) Original Image

Figure 6.5: Reconstructed images.

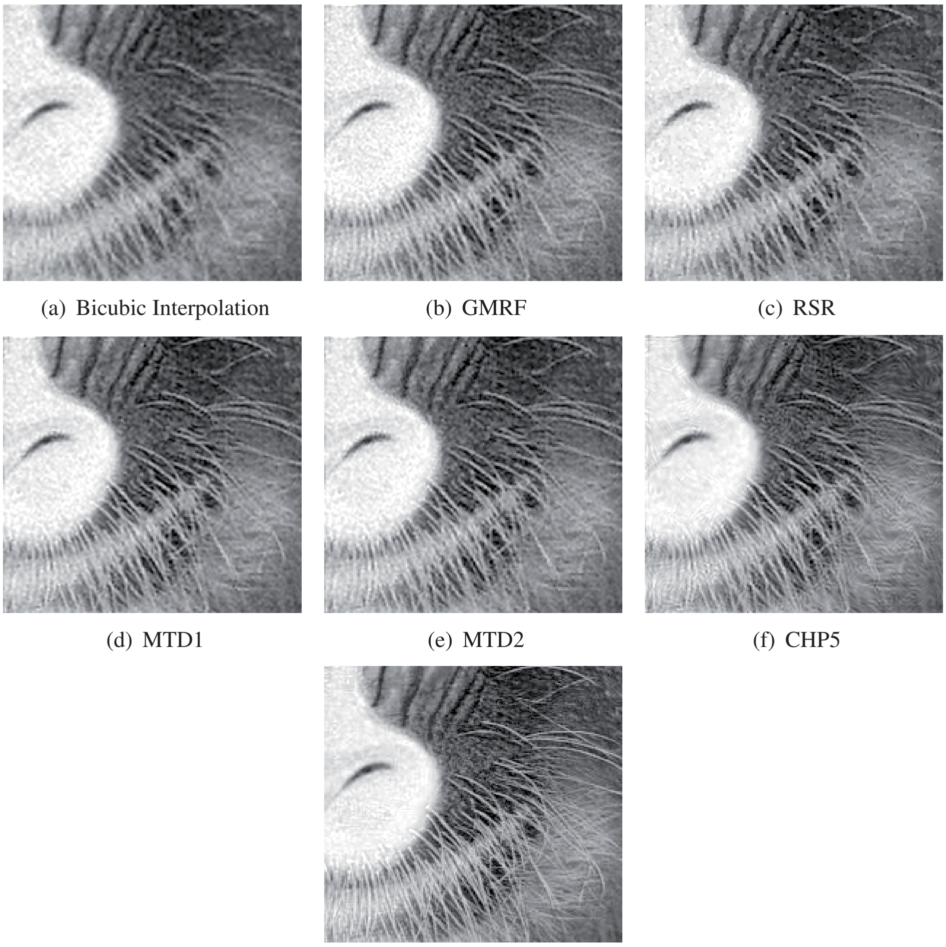


(a) Bicubic Interpolation (b) GMRF (c) RSR

(d) MTD1 (e) MTD2 (f) CHP5

(g) Original Image

Figure 6.6: High frequency content of the reconstructed images.

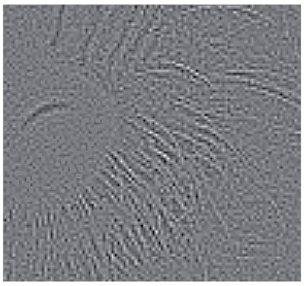


(a) Bicubic Interpolation (b) GMRF (c) RSR

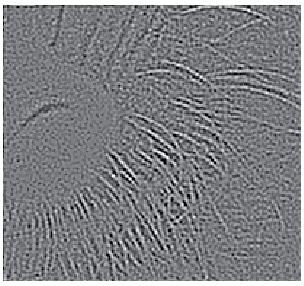
(d) MTD1 (e) MTD2 (f) CHP5

(g) Original Image

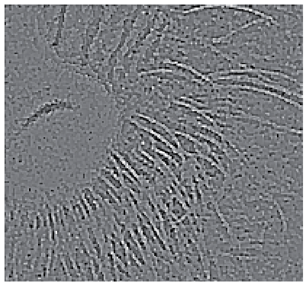
Figure 6.7: Reconstructed images.



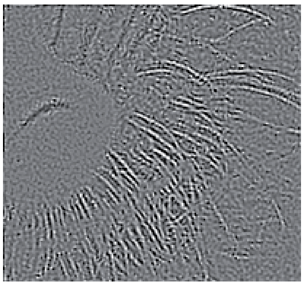
(a) Bicubic Interpolation



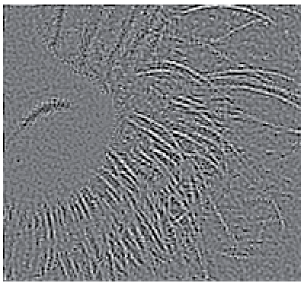
(b) GMRF



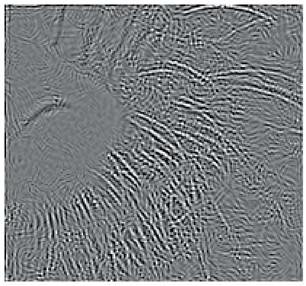
(c) RSR



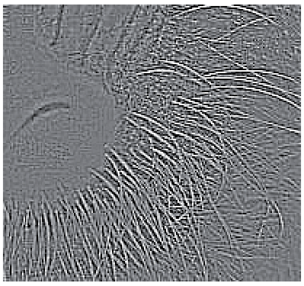
(d) MTD1



(e) MTD2

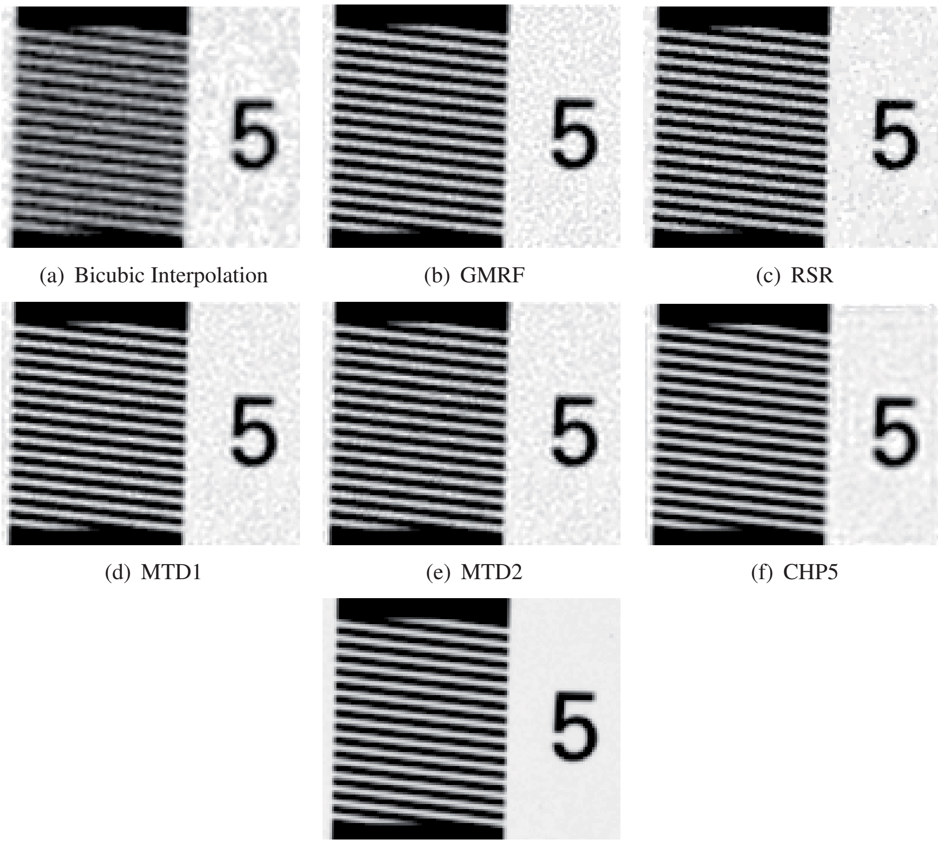


(f) CHP5



(g) Original Image

Figure 6.8: High frequency content of the reconstructed images.

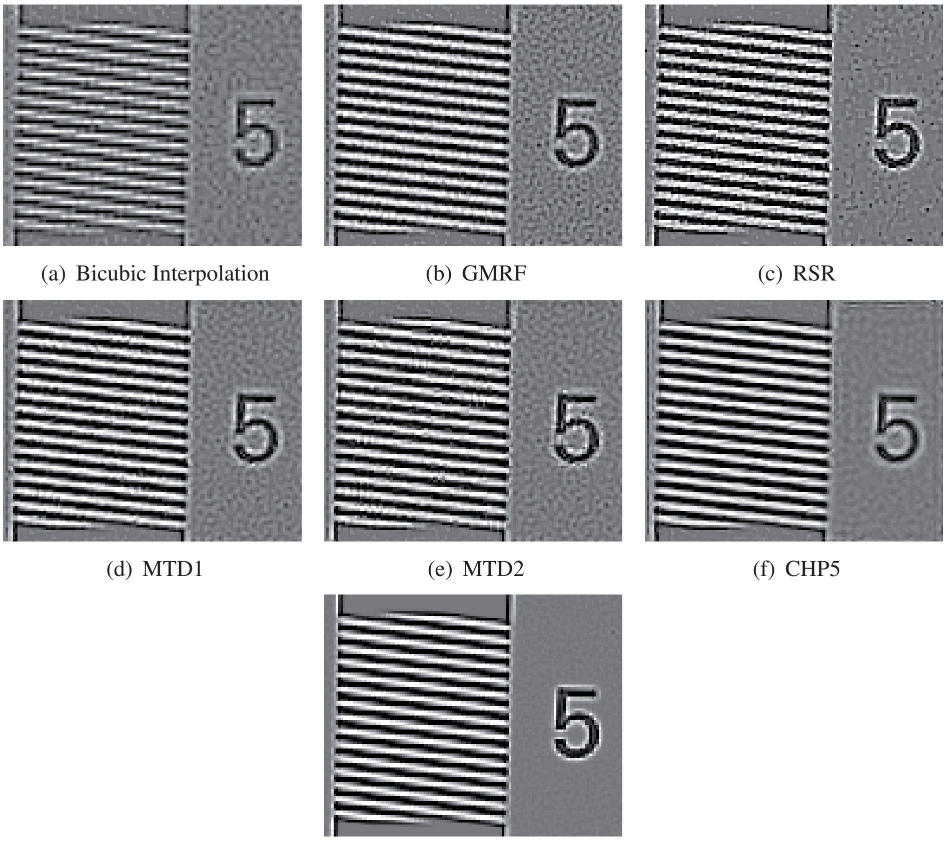


(a) Bicubic Interpolation (b) GMRF (c) RSR

(d) MTD1 (e) MTD2 (f) CHP5

(g) Original Image

Figure 6.9: Reconstructed images.



(a) Bicubic Interpolation (b) GMRF (c) RSR

(d) MTD1 (e) MTD2 (f) CHP5

(g) Original Image

Figure 6.10: High frequency content of the reconstructed images.

6.3.2 Experiments on Real Image Data

The same image set used in Chapter 5 Section 5.3.2 is also experimented in this section. The settings for the experiments are explained again for the sake of completeness. 7 LR images are used in the reconstructions. The relative motion among LR images is estimated using a global optical flow based approach described in [20]. Optimum value for the blurring kernel in Eq. 1.2 is selected as 5x5 Gaussian filter whose standard deviation is heuristically estimated to be 1.0. The resolution is increased by a factor of 2. Regularization parameters are manually tuned to obtain the most visually appealing reconstructions for RSR, GMRF, MTD1, and MTD2

Of the 7 LR images, 4 are shown in Figure 6.11(a) and 6.11(c). Pixel duplication results are also given in Figures 6.11(b) and 6.11(d). Reconstructions for GMRF, RSR, MTD1, and MTD2 methods are compared in Figures 6.12 and 6.13. High frequency components of the reconstructed images are also given so that the differences on textures and edges can be observed. It is observed that RSR results are sharper than GMRF results since RSR is more robust to motion estimation errors. The reconstructions of MTD1 and MTD2 have sharper textures and edges compared to RSR and GMRF. MTD1 is slightly better than MTD2 consistent with the simulation results. The discontinuity in textures and edges can be observed in MTD2 which is due to the block-wise post-processing stage. Although the proposed algorithms are not designed for images with text, the results in Figure 6.13 are very successful.

The reconstructions are run on a laptop with Intel core i5 M 460 processor at 2.53 GHz. One iteration takes around 1 s for GMRF method for seven 100 x 100 images. The post processing stages takes 3.11 and 5.15 s for MTD1 and MTD2 respectively. A 20 iteration reconstruction takes 20 s (1 s x 20) for GMRF method. The reconstruction time for the proposed methods is therefore 2 times this value plus the time for the post-processing stage.

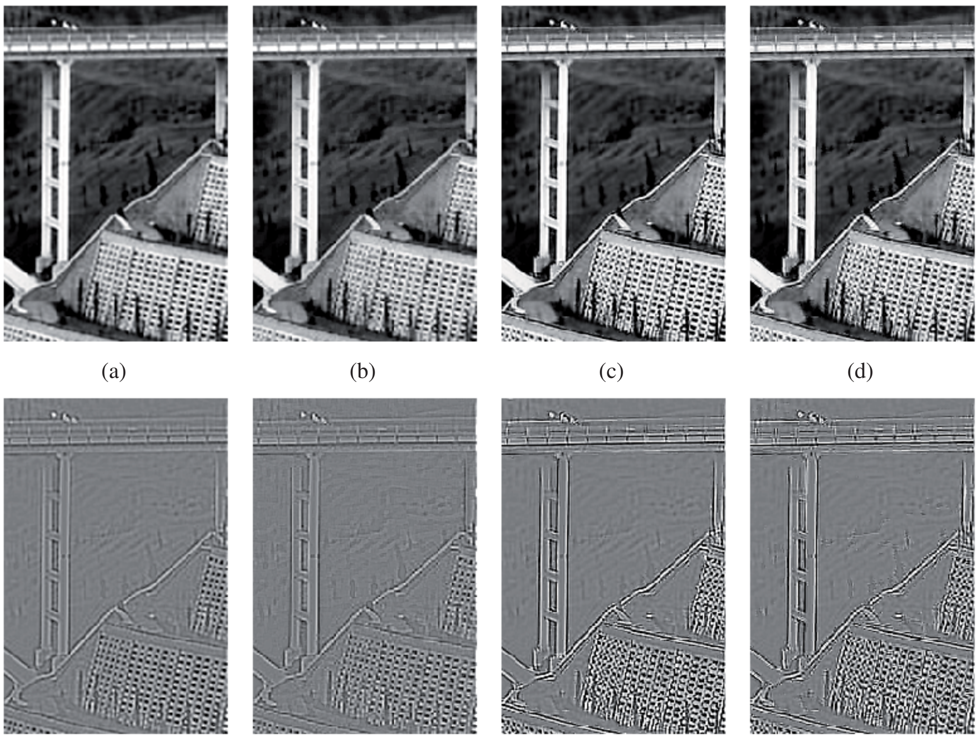
6.3.3 Experiments on Thermal Image Data

The same thermal image data set used in Chapter 5 is also experimented here. Relative motion is estimated as described in, [20]. The resolution is increased by a factor of 2 in horizontal direction and 4 in vertical direction. Some of these LR images are shown in Fig. 6.14. The blurring kernel in (1.2) is selected as 5×10 Gaussian filter with

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

Figure 6.11: (a) Four of seven LR images of the first image set, (b) 2x resolution increase by pixel duplication, (c) Four of seven LR images of the second image set, (d) 2x resolution increase by pixel duplication.

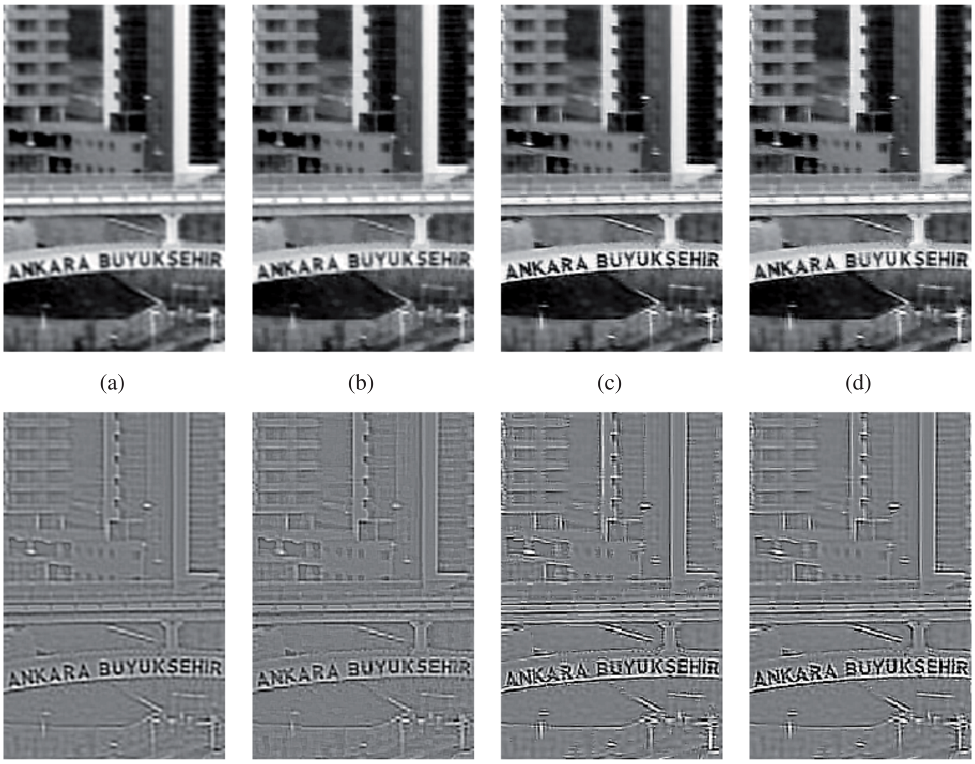
heuristically estimated standard deviation of 1.0. The method proposed in Chapter 5 is also added to the tests and abbreviated as CHP5. The regularization parameter for RSR, GMRF, and the proposed method are manually tuned to obtain the most visually appealing reconstructions, given in Figures 6.15 and 6.17. High frequency components of the reconstruction are also given in Figures 6.16 and 6.18 to compare the reconstructions on regions with textures and edges.



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

(e) (f) (g) (h)

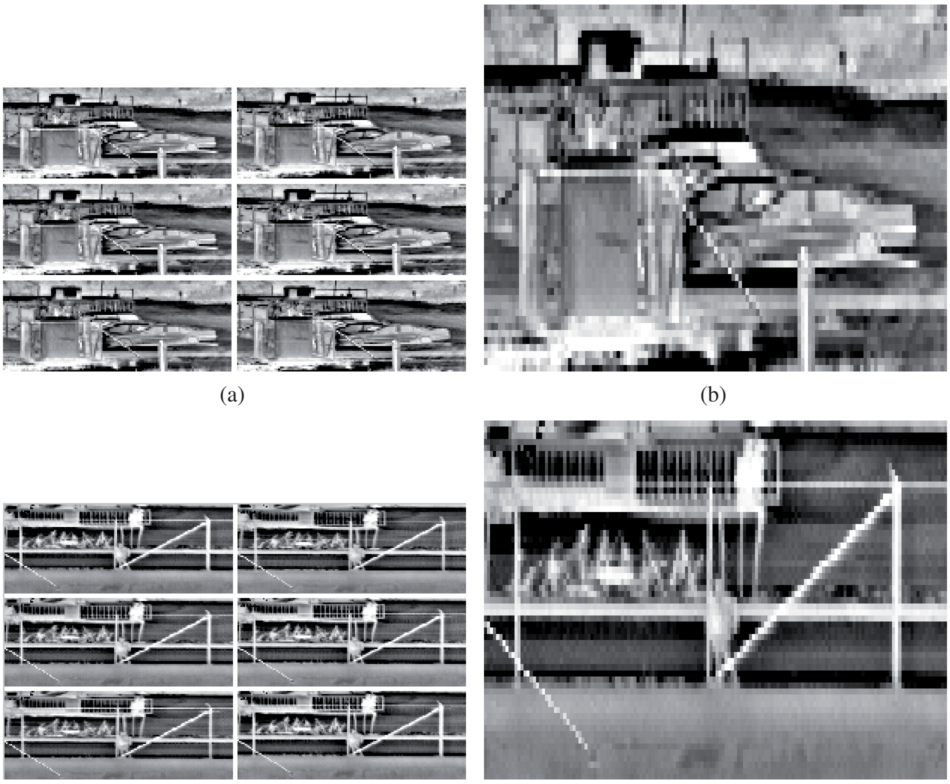
Figure 6.12: Reconstructions for first image set. A 270 x 180 window is selected for better visual inspection. (a) GMRF, (b) RSR, (c) MTD1, (d) MTD2. High frequency images are also given in the second row for (e) GMRF, (f) RSR, (g) MTD1, (h) MTD2.



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

(e) (f) (g) (h)

Figure 6.13: Reconstructions for second image set. A 280 x 180 window is selected for better visual inspection. (a) GMRF, (b) RSR, (c) MTD1, (d) MTD2. High frequency images are also given in the second row for (e) GMRF, (f) RSR, (g) MTD1, (h) MTD2.



(a) (b)

(c) (d)

Figure 6.14: (a) 6 of 20 LR images of the image set 1, (b) 2x resolution increase by pixel duplication. (c) LR images of the image set 2, (d) 2x resolution increase by pixel duplication.

(a) Bicubic Interpolation

(b) GMRF

(c) RSR

(d) MTD1

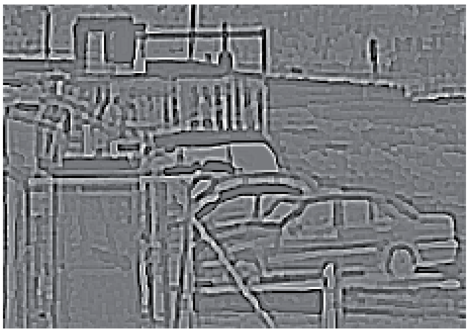
(e) MTD2

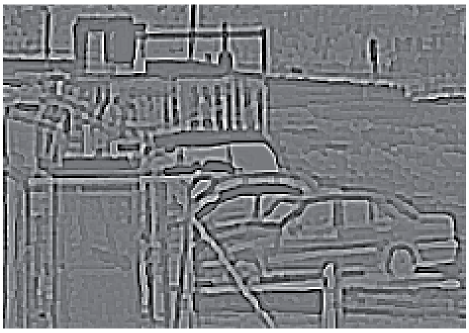
(f) CHP5

Figure 6.15: Reconstructions for the image set. A 120×150 window is selected for better visual inspection.


(a) Bicubic Interpolation


(b) GMRF

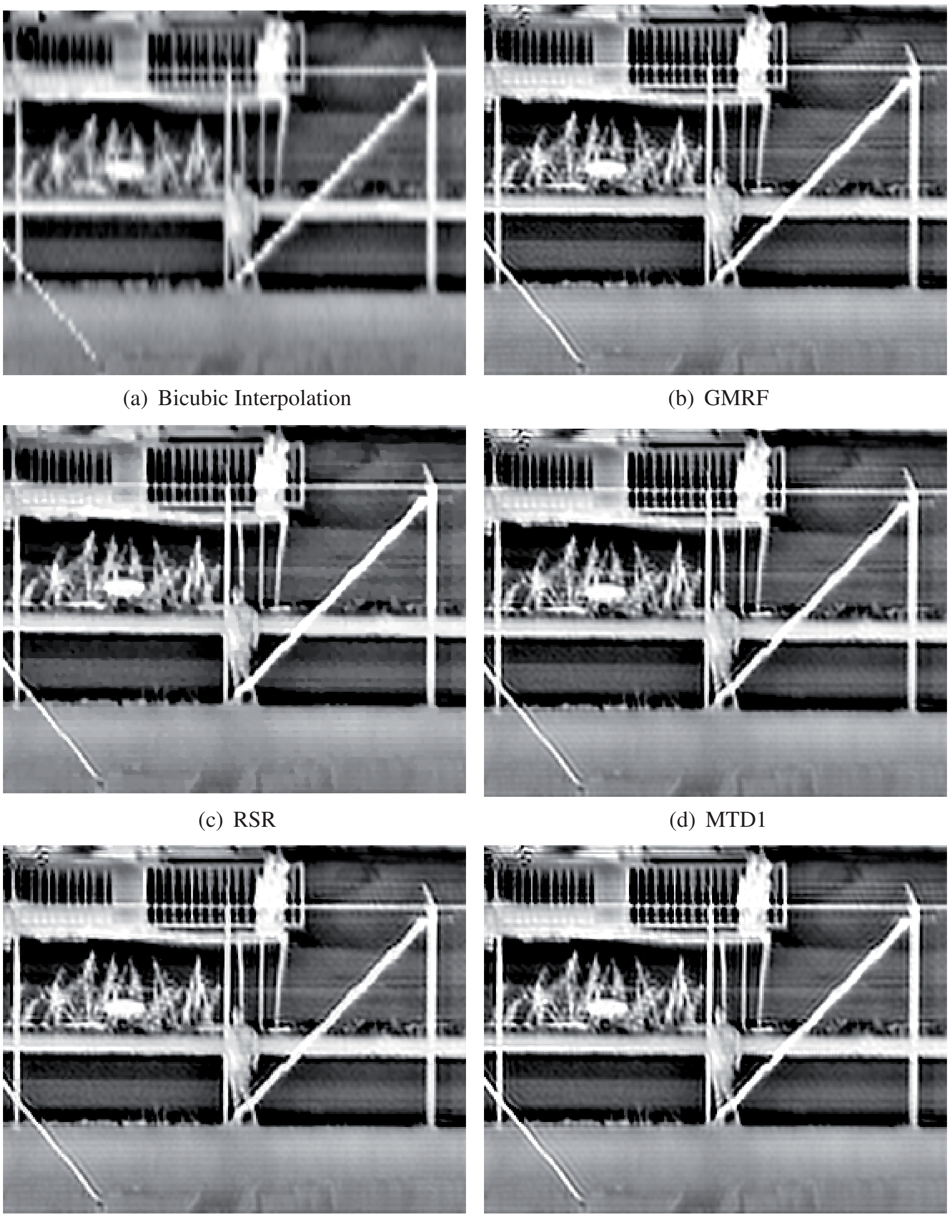

(c) RSR


(d) MTD1


(e) MTD2


(f) CHP5

Figure 6.16: Reconstructions for the image set. A 120×150 window is selected for better visual inspection. High frequency content of the reconstructed images.



(a) Bicubic Interpolation

(b) GMRF

(c) RSR

(d) MTD1

(e) MTD2

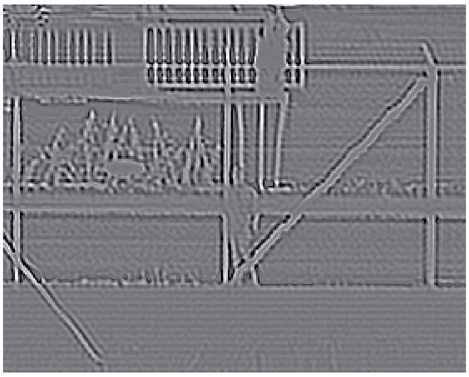
(f) CHP5

Figure 6.17: Reconstructions for the image set. A 120×150 window is selected for better visual inspection.

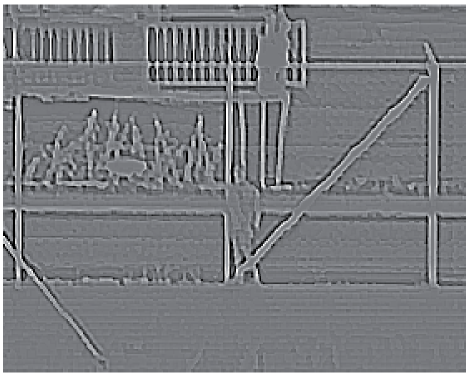
6.4 DISCUSSION

In this chapter, a novel SR image reconstruction algorithm is presented where texture regularization is treated as a post-processing step. Two methods based on Gabor filters and DCT are used in the post-processing step. Effect of regularization values and effect of noise to robustness of the proposed methods are investigated. For all regularization values and noise levels, proposed methods gave higher MSE and SSIM values compared to standard SR algorithms. The proposed approaches have significantly better results in real world scenarios, especially on textured areas and edges. It is noted that the methods reduce to MAP estimator for uniform regions. This property makes these methods flexible for different types of image sets. The proposed methods are also computationally very efficient since the two Bayesian estimators can run in parallel and the restoration stage is not iterative.

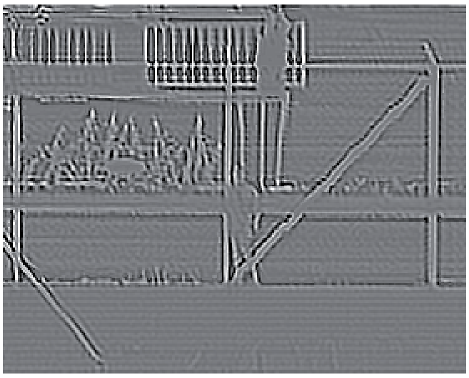
(a) Bicubic Interpolation



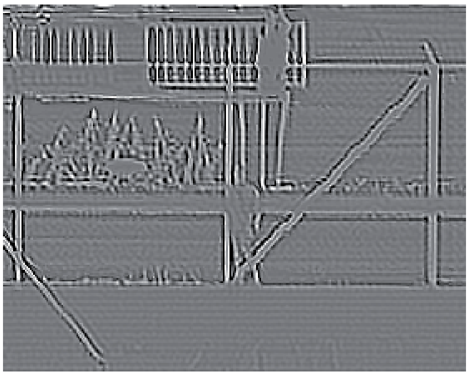
(b) GMRF



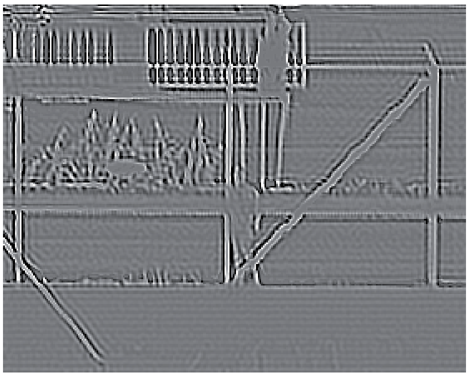
(c) RSR



(d) MTD1



(e) MTD2



(f) CHP5

Figure 6.18: Reconstructions for the image set. A 120×150 window is selected for better visual inspection. High frequency content of the reconstructed images.

CHAPTER 7

CONCLUSION

7.1 Summary

Superresolution image reconstruction refers to methods where a higher resolution image is reconstructed using a set of overlapping, aliased low resolution observations of the same scene. This thesis has proposed several maximum a posteriori based SR image reconstruction methods. Here, we summarize basic contributions of the thesis and give directions for future research.

Chapter 2 presents a survey of SR methods covering a 30 year period in literature. We have selected the pioneering studies related to each class of algorithms and summarized each category for a reader who is not familiar with the SR field. A special attention is given to iterative maximum *a posteriori* based methods since the proposed approaches in Chapter 4, 5, and 6 belong to this category.

The proposed methods in Chapter 4, 5, and 6 are based on minimizing the flowing cost function

$$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \rho(W_k\mathbf{z}, \mathbf{y_k}) + \lambda\Gamma(\mathbf{z}). \tag{7.1}$$

The main focus of this thesis is the image prior term, $\Gamma(\cdot)$, in (7.1). Chapter 4 presents a directionally adaptive SR image reconstruction method. The basic idea behind this algorithm is determining the best regularization filter according to the magnitude and direction of the local gradient. Three variations of this algorithm are compared with previous work [25, 40]. The SSIM, PSNR values and illustrations are given for performance evaluations. Especially for highly degraded images with sharp edges, the proposed methods clearly preserve edges while applying high amounts of regularization. The performance is however very similar to [25, 40] in regions where a strong edge is not present.

In Chapter 5, we propose a novel texture prior for maximum a posteriori (MAP) based super-resolution (SR) image reconstruction. The prior is based on a multiscale compound Markov Random Field (MRF) model. Gabor filters are utilized for subband decomposition. Each subband is modeled by a compound MRF that inherits a binary texture process. The texture process at each pixel location at each subband is estimated iteratively along with the unknown high resolution image pixels. The proposed method is novel in two ways: 1) Multiscale priors has never been used in SR literature. 2) Compound multiscale priors,"to the best of our knowledge", has not been proposed in image processing literature before. The overall model is easy to implement and intuitive. The proposed method is validated through simulations and real experiments, which clearly demonstrates significant visual improvements, especially on images with high frequency textures, when compared with state-of-the-art methods.

Incorporating more information to the solution results in better reconstructions. However, this increases the computational overhead. In Chapter 4, direction of gradients is used in addition to standard regularizers. The proposed approach has a computational overhead which varies depending on the gradient distribution in the input LR image sets. In our simulations we have observed 10 - 15 % increase in computation time compared with [25, 40]. In Chapter 5, Gabor filters are effectively used in determining underlying texture process. However, the computational overhead is 2-4 times compared with [25, 40].

In Chapter 6, a two stage method is proposed, comprising multiple SR reconstructions with different regularization parameters followed by a restoration step for preserving edges and textures. In the first stage, two maximum-a-posteriori (MAP) estimators with two different amounts of regularization are employed. In the second stage, pixel-to-pixel difference between these two estimates is post-processed to restore edges and textures. Frequency selective characteristics of discrete cosine transform (DCT) and Gabor filters are utilized in the post-processing step. The proposed approaches have significantly better results in real world scenarios, especially on textured areas and edges. It is noted that the methods reduce to MAP estimator for uniform regions. This property makes these methods flexible for different types of image sets. The proposed methods are also computationally very efficient since the two Bayesian estimators can run in parallel and the restoration stage is not iterative.

7.2 Future Directions

Proposed algorithms consider image registration values, camera blur and various parameters related to the image prior as known. In SR literature, several approaches have been proposed for simultaneously solving SR image pixels, image registration, sensor PSF, and model parameters in a Bayesian framework. The natural follow up is to adapt these simultaneous estimation methods to the proposed methods in the thesis. The second line of study is application of these methods for video super resolution. Two important challenges are: 1) optimizing the proposed methods in terms of speed; 2) finding an efficient way to deal with the ever increasing video frames in terms of memory and computation time. Some important derivations for video super resolution are given in Appendix B, which highlights these challenges.

REFERENCES

[1] R. Y. Tsai and T. S. Huang, "Multiframe image restoration and registration," *in Advances in Computer Vision and Image Processing*, vol. 1, no. 2, pp. 317–339, 1984.

[2] Marco Cristani, Dong Seon Cheng, Vittorio Murino, and Donato Pannullo, "Distilling information with super-resolution for video surveillance," in *Proceedings of the ACM 2nd international workshop on Video surveillance & sensor networks*. ACM, 2004, pp. 2–11.

[3] Toygar Akgun, Yucel Altunbasak, and Russell M Mersereau, "Super-resolution reconstruction of hyperspectral images," *Image Processing, IEEE Transactions on*, vol. 14, no. 11, pp. 1860–1875, 2005.

[4] Feng Li, Xiuping Jia, and Donald Fraser, "Universal hmt based super resolution for remote sensing images.," in *ICIP*. 2008, pp. 333–336, IEEE.

[5] Bahadir K Gunturk, Aziz Umit Batur, Yucel Altunbasak, Monson H Hayes III, and Russell M Mersereau, "Eigenface-domain super-resolution for face recognition," *Image Processing, IEEE Transactions on*, vol. 12, no. 5, pp. 597–606, 2003.

[6] Sharon Peled and Yehezkel Yeshurun, "Superresolution in mri: application to human white matter fiber tract visualization by diffusion tensor imaging," *Magnetic resonance in medicine*, vol. 45, no. 1, pp. 29–35, 2001.

[7] Hayit Greenspan, Gal Oz, Nahum Kiryati, and Sharon Peled, "Super-resolution in mri," in *Biomedical Imaging, 2002. Proceedings. 2002 IEEE International Symposium on*. IEEE, 2002, pp. 943–946.

[8] John A Kennedy, Ora Israel, Alex Frenkel, Rachel Bar-Shalom, and Haim Azhari, "Super-resolution in pet imaging," *Medical Imaging, IEEE Transactions on*, vol. 25, no. 2, pp. 137–147, 2006.

[9] M. Elad and A. A. Feuer, "Restoration of a single superresolution image from several blurred, noisy,and undersampled measured images," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 1646–1658, December 1997.

[10] Deepu Rajan and Subhasis Chaudhuri, "Generation of super-resolution images from blurred observations using markov random fields," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*. IEEE, 2001, vol. 3, pp. 1837–1840.

[11] Deepu Rajan and Subhasis Chaudhuri, "Simultaneous estimation of super-resolved scene and depth map from low resolution defocused observations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1102–1117, 2003.

[12] Manjunath V Joshi, Subhasis Chaudhuri, and Rajkiran Panuganti, "Super-resolution imaging: use of zoom as a cue," *Image and Vision Computing*, vol. 22, no. 14, pp. 1185–1196, 2004.

[13] R. C. Hardie, K. J. Barnard, and E. E. Armstrong, "Joint map registration and high-resolution image estimation using a sequence of undersampled images," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 1621–1633, December 1997.

[14] B. Zitov'a and J. Flusser, "Image registration methods: a survey," *Image and Vision Computing*, vol. 21, no. 11, pp. 977–1000, 2003.

[15] N. X. Nguyen, *Numerical Algorithms for Image Superresolution*, Ph.D. thesis, Stanford University, July 2000.

[16] Z. Lin and H. Y. Shum, "Fundamental limits of reconstruction-based super-resolution algorithms under local translation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 1, January 2004.

[17] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, Prentice Hall. 2nd Eddition, 1999.

[18] S. P. Kim, N. K. Bose, and H. M. Valenzuela, "Recursive reconstruction of high resolution image from noisy undersampled multiframes," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 1013–1027, June 1990.

[19] A. M. Tekalp, M. K. Ozkan, and M. I. Sezan, "High-resolution image reconstruction from lower-resolution image sequences and space varying image restoration," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*. IEEE San Francisco, CA, March 1992, vol. 3, pp. 169–172.

[20] D. Keren, S. Peleg, and R. Brada, "Image sequence enhancement using sub-pixel displacement," in *Computer Vision and Pattern Recognition*. IEEE, June 1988, pp. 742–746.

[21] K. D. Sauer and J. P. Allebach, "Iterative reconstruction of band-limited images from non-uniformly spaced samples," *IEEE Trans. Circuits Systs.*, vol. CAS-34, pp. 1497–1505, 1987.

[22] M. S. Alam, J. G. Bognar, R. C. Hardie, and B. C. Yasuda, "Infrared image registration and high-resolution reconstruction using multiple translationally shifted aliased video frames," *IEEE Transactions on Instrumentation and Measurement*, vol. 49, no. 5, pp. 915–923, October 2000.

[23] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: A technical overview," *IEEE Signal Processing Magazine*, May 2003.

[24] S. Borman and R. Stevenson, "Spatial resolution enhancement of low-resolution image sequences - a comprehensive review with directions for future research," January 1998, Technical Report.

[25] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1327–1344, October 2004.

[26] A. C. Kak and M. Slaney, *Principles of Computerized Topographic Imaging*, IEEE Press, New York. (Available in pdf format at http://slaney.org/pct/index.html), 1988.

[27] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graphical Models and Image Proc.*, vol. 53, pp. 231–239, May 1991.

[28] M.R. Banham and A.K. Katsaggelos, "Digital image restoration," *Signal Processing Magazine, IEEE*, vol. 14, no. 2, pp. 24–41, mar 1997.

[29] M. K. Özkan, A. M. Tekalp, and M. I. Sezan, "Pocs-based restoration of space-varying blurred images," *IEEE Trans. Image Process.*, vol. 3, no. 4, July 1994.

[30] A. K. Katsaggelos, "A multiple input image restoration approach," *Journal of Visual Communication and Image Representation*, vol. 1, pp. 93–103, 1990.

[31] H. Stark and P. Oskoui, "High resolution image recovery from image-plane arrays, using convex projections," *Opt. Soc. Am.*, vol. 6, pp. 1715–1726, 1989.

[32] M. I. Sezan A. J. Patti and A.M. Tekalp, "Super resolution video reconstruction with arbitrary sampling lattices and nonzero aperture time," *IEEE Trans. Image Process.*, vol. 6, no. 8, pp. 1064–1076, August 1997.

[33] P. E. Eren, M. I. Sezan, and A. M. Tekalp, "Robust, object-based high-resolution image reconstruction from low-resolution video," *IEEE Trans. Image Process.*, vol. 6, no. 10, July 1997.

[34] J. R. Shewchuk, "An introduction to the conjugate gradient method without the agonizing pain," Tech. Rep., Pittsburgh, PA, USA, 1994.

[35] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, pp. 259–268, November 2004.

[36] Yuying Li and Fadil Santosa, "A computational algorithm for minimizing total variation in image restoration," *Image Processing, IEEE Transactions on*, vol. 5, no. 6, pp. 987–995, 1996.

[37] D. Brzakovic L. Hong, "Bayesian restoration of image sequences using 3-d markov random fields," *IEEE*, 1989.

[38] R. R. Schultz and R. L. Stevenson, "Improved definition image expansion," in *Int. Conf. Acoust. Speech Signal Processing*, 1992, pp. III–173 – III–176.

[39] R. R. Schultz and R. L. Stevenson, "A bayesian approach to image expansion for improved definition," *IEEE Trans. Image Process.*, vol. 3, no. 3, pp. 996–1011, May 1994.

[40] R. R. Schultz and R. L. Stevenson, "Extraction of high-resolution frames from video sequences," *IEEE Trans. Image Process.*, vol. 5, no. 6, pp. 996–1011, June 1996.

[41] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Robust shift and add approach to super-resolution," in *Proc. SPIE Conf. Applications of Digital Signal and Image Processing*, August 2003, pp. 121–130.

[42] M. Elad and D. Datsenko, "Example-based regularization deployed to super-resolution reconstruction of a single image," *Computer Journal*, vol. 52, pp. 15–30, 2007.

[43] L. C. Pickup, S. J. Roberts, and A. Zisserman, "A sampled texture prior for image super-resolution," in *Proceedings of Advances in Neural Information Processing Systems 16 (NIPS '03)*, December 2004, pp. 1587–1594.

[44] D. Datsenko and M. Elad, "Example-based single document image super-resolution: a global map approach with outlier rejection," *Multidimensional Systems and Signal Processing*, vol. 18, pp. 103–121, September 2007.

[45] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael, "Learning low-level vision," *Int. J. Comput. Vis.*, vol. 40, no. 1, pp. 25–47, 2000.

[46] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Comp. Graphics And Appl.*, vol. 22, no. 2, pp. 56–65, 2002.

[47] S. Baker and T. Kanade, "Limits on super-resolution and how to break them," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1167–1183, 2002.

[48] R. Nakagaki and A. K. Katsaggelos, "Vq-based blind image restoration algorithm," *IEEE Trans. Image Process.*, vol. 12, no. 9, pp. 1044–1053, 2003.

[49] Christopher M Bishop, Andrew Blake, and Bhaskara Marthi, "Super-resolution enhancement of video," in *Proc. Artificial Intelligence and Statistics*. Key West, FL, USA, 2003, vol. 2.

[50] Richard Baraniuk, "Compressive sensing," *IEEE Signal Processing Mag*, pp. 118–120, 2007.

[51] Emmanuel J Candès and Michael B Wakin, "An introduction to compressive sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21–30, 2008.

[52] Jianchao Yang, John Wright, Thomas Huang, and Yi Ma, "Image super-resolution as sparse representation of raw image patches," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[53] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma, "Image super-resolution via sparse representation," *Image Processing, IEEE Transactions on*, vol. 19, no. 11, pp. 2861–2873, 2010.

[54] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, "Coupled dictionary training for image super-resolution," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3467–3478, 2012.

[55] Shuyuan Yang, Fenghua Sun, Min Wang, Zhizhou Liu, and Licheng Jiao, "Novel super resolution restoration of remote sensing images based on compressive sensing and example patches-aided dictionary learning," in *Multi-Platform/Multi-Sensor Remote Sensing and Mapping (M2RSM), 2011 International Workshop on*. IEEE, 2011, pp. 1–6.

[56] M. Elad and A. Feuer, "Super-resolution reconstruction of image sequences," *IEEE Transactions on Pattern Analysis and machine Intelligence*, vol. 21, no. 9, pp. 817–834, September 1999.

[57] M. Elad and A. Feuer, "Superresolution restoration of an image sequence: adaptive filtering approach," *IEEE Trans. Image Process.*, vol. 8, no. 3, pp. 387–395, March 1999.

[58] C Newland, D Gray, and Daniel Gibbins, "Modified kalman filtering for image super-resolution," in *In Proc. of the 2006 Conf. on Image and Vision Computing (IVCNZ06)*, 2006, pp. 79–84.

[59] Guilherme Holsbach Costa and José Carlos M Bermudez, "Statistical analysis of the lms algorithm applied to super-resolution image reconstruction," *Signal Processing, IEEE Transactions on*, vol. 55, no. 5, pp. 2084–2095, 2007.

[60] Moshe Ben-Ezra, Assaf Zomet, and Shree K Nayar, "Video super-resolution using controlled subpixel detector shifts," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 6, pp. 977–987, 2005.

[61] Huihui Song, Lei Zhang 0006, Peikang Wang, Kaihua Zhang, and Xin Li, "An adaptive l1-l2 hybrid error model to super-resolution.," in *ICIP*. 2010, pp. 2821–2824, IEEE.

[62] Emmanuel J Candes, Michael B Wakin, and Stephen P Boyd, "Enhancing sparsity by reweighted l1 minimization," 2007.

[63] H. Kim, J. H. Jang, and K. S. Hong, "Edge-enhancing superresolution using anisotropic diffusion," in *Proc. IEEE Int. Conf. Image Processing*. IEEE, 2001, pp. 130–133.

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

[65] V. Velisavljevic, "Edge-preservation resolution enhancement with oriented wavelets," in *Proceedings of the IEEE International Conference on Image Processing*, October 2008, pp. 1252–55.

[66] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1521–27, October 1997.

[67] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, April 2004.

[68] T. Trandon, "Brodatz textures," http://www.ux.uis.no/~tranden/brodatz.html.

[69] J. Sklansky, "Image segmentation and feature extraction," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 8, no. 4, pp. 237 –247, april 1978.

[70] Koichiro Deguchi, "Two-dimensional auto-regressive model for analysis and sythesis of gray-level textures," *Proc. of the 1st Int. Sym. for Science on Form, General Ed. S. Ishizaka, Eds. Y. Kato, R. Takaki, and J. Toriwaki*, pp. 441–449, 1986.

[71] Wan Jiaruo, Mao Jianchang, and Wang Cheng Dao, "Multiresolution rotation invariant simultaneous auto regressive model for texture analysis," in *Pattern Recognition, 1988., 9th International Conference on*. IEEE, 1988, pp. 845–847.

[72] Pierre M Tardif and Andre Zaccarin, "Multiscale autoregressive image representation for texture segmentation," in *Electronic Imaging'97*. International Society for Optics and Photonics, 1997, pp. 327–337.

[73] Rupert Paget and I Dennis Longstaff, "Texture synthesis via a noncausal nonparametric multiscale markov random field," *Image Processing, IEEE Transactions on*, vol. 7, no. 6, pp. 925–931, 1998.

[74] Lei Wang and Jun Liu, "Texture classification using multiresolution markov random field models," *Pattern Recognition Letters*, vol. 20, no. 2, pp. 171–182, 1999.

[75] Santhana Krishnamachari and Rama Chellappa, "Multiresolution gauss-markov random field models for texture segmentation," *Image Processing, IEEE Transactions on*, vol. 6, no. 2, pp. 251–267, 1997.

[76] J. Portilla and E. P. Simoncelli, "A parametric texture model based on joint statistics of complex wavelet coefficients," *International Journal of Computer Vision*, vol. 40, no. 1, pp. 49–71, October 2000.

[77] J. Malik, S. Belongie, T. Leung, and J Shi, "Contour and texture analysis for image segmentation," *International Journal of Computer Vision*, vol. 43, no. 1, pp. 7–27, February 2001.

[78] Song Chun Zhu, Yingnian Wu, and David Mumford, "Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling," *International Journal of Computer Vision*, vol. 27, no. 2, pp. 107–126, 1998.

[79] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-6, no. 6, pp. 721–741, 1984.

[80] N. A. Woods, N. P. Galatsanos, and A. K. Katsaggelos, "Stochastic methods for joint registration, restoration, and interpolation of multiple undersampled images," *IEEE Trans. Image Process.*, vol. 15, no. 1, pp. 201–213, January 2006.

[81] R. Mehrotra, K. R. Namuduri, and N. Ranganathan, "Gabor filter-based edge detection," *Pattern Recognition*, vol. 25, no. 12, pp. 1479–1494, 1992.

[82] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using gabor filters," *Pattern Recognition*, vol. 24, no. 12, pp. 1167–1186, 1991.

[83] S.E. Grigorescu, N. Petkov, and P. Kruizinga, "Comparison of texture features based on gabor filters," *IEEE Trans. Image Process.*, vol. 11, no. 10, pp. 1160 – 1167, oct 2002.

[84] Stan Z Li, *Markov random field modeling in image analysis*, Springer, 2009.

[85] A. Kanemura, S. Maeda, and S Ishii, "Superresolution with compound markov random fields via the variational em algorithm," *Neural Networks*, vol. 22, no. 7, pp. 1025–1034, 2009.

[86] T. Katsuki, A. Torii, and M. Inoue, "Posterior-mean super-resolution with a causal gaussian markov random field prior," *IEEE Trans. Image Process.*, vol. 21, no. 7, pp. 3182–3193, 2012.

[87] L. C. Pickup, Capel D. P., S. J. Roberts, and A. Zisserman, "Bayesian image super-resolution, continued," in *Advances in Neural Information Processing Systems*. 2006, pp. 1089–1096, MIT Press.

[88] S. D. Babacan, D. Molina, and A. K. Katseggelos, "Variational bayesian super resolution," *IEEE Trans. Image Process.*, vol. 20, no. 4, pp. 984 – 999, april 2011.

[89] M. E. Tipping and C. M. Bishop, "Bayesian image super-resolution," in *Advances in Neural Information Processing Systems*. 2003, pp. 1303–1310, MIT Press.

[90] M. Elad and Y. Hel-Or, "A fast super-resolution reconstruction algorithm for pure translational motion and common space-invariant blur," *IEEE Trans. Image Process.*, vol. 10, no. 8, pp. 1187–1193, August 2001.

[91] G. Chantas, N. P. Galatsanos, R. Molina, and A. K. Katsaggelos, "Variational bayesian image restoration with a product of spatially weighted total variation image priors," *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 351–362, 2010.

[92] E. Turgay and G. B. Akar, "Directionally adaptive super-resolution," in *Proc. IEEE Int. Conf. Image Processing*, November 2009, pp. 1201–04.

[93] E. Turgay and G. B. Akar, "Context based super resolution image reconstruction," in *Proc. IEEE Local and Non-Local Approximation in Image Processing*, August 2009, pp. 54–61.

[94] G. Sorwar and A. Abraham, "Dct based texture classification using a soft computing approach," *Malaysian Journal of Computer Science*, vol. 17, no. 1, pp. 12–23, June 2004.

[95] L. Kumar and C. N. R. Kumar, "Novel super resolution reconstruction of low reoslution images progressively using dct and zonal filter based denoising," *International Journal of Computer Science and information Technology*, vol. 3, no. 1, February 2011.

APPENDIX A

STRUCTURE OF THE OPERATIONS IN MATRIX FORM

Structures of the matrix operators and system matrix used in super resolution formulation are investigated. Decimation, blur, warping operators are defined in lexicographical representation. The lexicographical form is constructed by adding the rows of the image to each other to construct a long column vector. For example the following 2x2 matrix is rewritten in lexicographical form as a 4×1 vector.

$$Im = \begin{bmatrix} 1 & 5 \\ 6 & 8 \end{bmatrix} \rightarrow \begin{bmatrix} 1 \\ 5 \\ 6 \\ 8 \end{bmatrix}$$

Assume $\mathbf{y}_k$ and $\mathbf{z}$ are the low and high resolution images in lexicographical form with sizes L × L and M × M respectively. The forward problem of obtaining a low resolution image out of the high resolution image is given in (A.1)

$$\mathbf{y}_k = DBF_k \cdot \mathbf{z} + \mathbf{n}_k \qquad k = 1 \cdots c. \tag{A.1}$$

where D is decimation, B is blur and F_k is the spatial warping operator. F_k and B are $M^2 \times M^2$ matrices and D is a $L^2 \times M^2$ matrix. Their structures are written explicitly for small images in the following sections.

A.1 Decimation Operation

The following operator creates a 2×2 image out of a 4×4 image by sampling one pixel every two pixels in the 4×4 image.

$$D = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{A.2}$$

A.2 Shift Operation

The following operator shifts a 4×4 image toward left by one pixel.

$$S_x(1) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{A.3}$$

The flowing matrix shifts a 4×4 image one pixel down.

$$
S_y(1) = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0
\end{bmatrix} \tag{A.4}
$$

The following matrix is obtained by multiplying the two matrices given above. It shifts a 4×4 image one pixel left and one pixel down.

$$
S_x(1) \cdot S_y(1) = \begin{bmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0
\end{bmatrix} \tag{A.5}
$$

The two matrices given next shifts an image 0.5 pixels to left, 0.5 pixels down respectively. The third matrix shifts an image 0.5 pixels left and 0.5 pixels down at the same time

$$
S_x(0.5) = \begin{bmatrix}
0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5
\end{bmatrix} \tag{A.6}
$$

$$
S_y(0.5) = \begin{bmatrix}
0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5
\end{bmatrix} \tag{A.7}
$$

$$
S_x(0.5) \cdot S_y(0.5) = \begin{bmatrix}
0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.25 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.25 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0.25 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0.25
\end{bmatrix} \tag{A.8}
$$

A.3 Rotation Operation

For a 2×2 matrix rotation with bilinear approximation is defined as follows

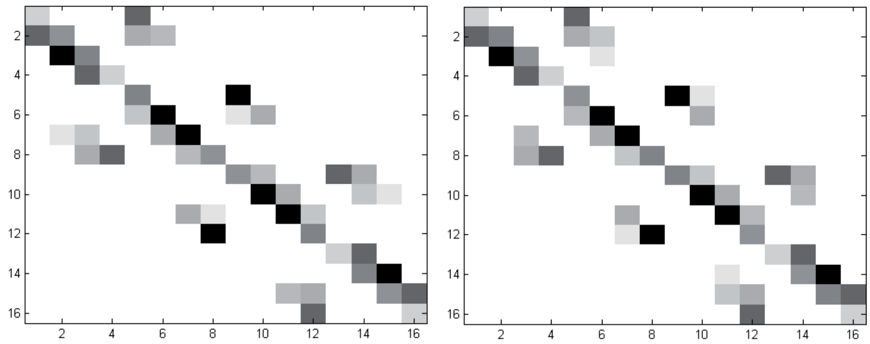


Figure A.1: For a 4×4 image 25^o clockwise rotation matrix is as shown on the left. Transpose of a 25^o counter clock wise rotation is shown on the right. The two matrices are not exactly same however can be considered similar.

$$Rot(-45^{\text{o}}) = \begin{pmatrix} 0.4 & 0 & 0.4 & 0 \\ 0.4 & 0.4 & 0 & 0 \\ 0 & 0 & 0.4 & 0.4 \\ 0 & 0.4 & 0 & 0.4 \end{pmatrix}, \quad Rot(45^{\text{o}}) = \begin{pmatrix} 0.4 & 0.4 & 0 & 0 \\ 0 & 0.4 & 0 & 0.4 \\ 0.4 & 0 & 0.4 & 0 \\ 0 & 0 & 0.4 & 0.4 \end{pmatrix} \tag{A.9}$$

Note that $Rot(45^o)^T = Rot(-45^{\text{o}})$, however it is shown in Figure A.1 that for larger images $Rot(\theta^{\text{o}})^T \approx Rot(-\theta^{\text{o}})$. Note also that $Rot(\theta^{\text{o}})^{-1} \neq Rot(-\theta^{\text{o}})$. Structure of a larger rotation matrix is shown in Figure A.1.

A.4 Blur Operation

For a 3×3 circulary symmetric blur operator such as $b = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ The blur operator matrix becomes

$$
B = \begin{bmatrix}
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1
\end{bmatrix} \times 1/9 \tag{A.10}
$$

The above matrix is symmetric, $B^T = B$.

A.5 System Matrix

System Matrix $W = D \times B \times F_k$ is obtained by multiplying the above matrices. The following system matrix creates a 2×2 low resolution image from a 4×4 high resolution image according to the observation model in (A.1).

$$
D{\cdot}B{\cdot}F(0.5, 0.5) = \begin{bmatrix}
1 & 0.5 & 0 & 0 & 0.5 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.5 & 1 & 0.5 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.5 & 0.25 & 0 & 0 & 1 & 0.5 & 0 & 0 & 0.5 & 0.25 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0.5 & 1 & 0.5 & 0 & 0.25 & 0.5 & 0.25
\end{bmatrix} \tag{A.11}
$$

The pattern can be observed better in a system where a 4×4 low resolution image is obtained from a 16×16 image where the system matrix is 64×256 in Figure A.2.

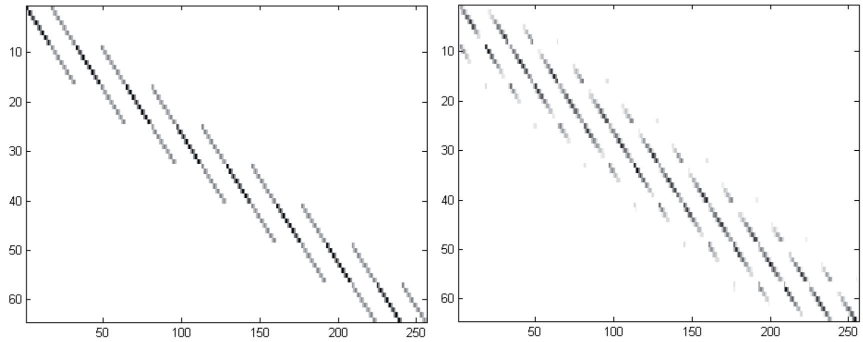


Figure A.2: For a 16×16 high resolution image and 8×8 low revolution image the system matrix W has the above forms. Here the motion in x and y directions are 0.5 pixels each (the right system matrix also comprises 10^o of counter clock wise rotation), zoom factor is two, and the blur operation is a 2×2 averaging operator the white pixels are zeros.

A.6 Observations on Matrix Operators in Super Resolution

- $S_x(dx) \cdot S_y(dy) = S_y(dy) \cdot S_x(dx)$ translation matrices are commutative.
- $S_x(dx) = S_x(-dx)^T$ but $S_x(dx)^{-1} \neq S_x(-dx)$.
- Let $F(dx, dy) = S_x(dx) \cdot S_y(dy)$ then $F(dx, dy)^T = F(-dx, -dy)$ but $F(dx, dy)^{-1} \neq F(-dx, -dy)$.
- $F(dx_1, dy_1) \cdot F(dx_2, dy_2) = F(dx_2, dy_2) \cdot F(dx_1, dy_1)$
- $Rot(\theta^o)^T \approx Rot(-\theta^o)$ (as shown in Figure A.1) a small difference can be observed.
- $Rot(\theta^o)^{-1} \neq Rot(-\theta^o)$
- $Rot(\theta_1^o) \cdot Rot(\theta_2^o) \approx Rot(\theta_2^o) \cdot Rot(\theta_1^o) \approx Rot(\theta_1^o + \theta_2^o)$.
- D down-samples Z by $Y = D \cdot Z$ operation, and D^T upsamples Y by zero padding.
- $D \cdot D^T = I$ is identity however $D^T \cdot D$ is a diagonal matrix only.
- If the blurring Kernel is circulary symmetric having a center pixel, then the blur matrix B is symmetric, $B = B^T$.

- If warping only consists of transformation then $D \cdot W \cdot F = D \cdot F \cdot W$ where W is the blur and F is the warping operator. The two image obtained through the system matrix given at both sides of the equality only differs at the pixels located at the edges where translational motion introduces zero padding.

APPENDIX B

VIDEO SUPERRESOLUTION

In the previously discussed SR problem, the whole low resolution image set is assumed to be available for high resolution image reconstruction. However, in video frame up-sampling, low resolution images arrive sequentially for the reconstruction. Starting from the first frame a high resolution image is constructed and updated after each new frame. This type of application is named as *sequential super resolution* throughout this chapter. The classical SR problem where low resolution input images are available before the reconstruction are referred to as *batch super resolution*.

B.1 SEQUENTIAL VERSUS BATCH SUPER RESOLUTION

For both batch and sequential SR problem the relationship between SR image and LR image, namely the forward problem, is quite similar. This relationship is constructed through a camera model where image formation process is simplified as blurring, down sampling and introduction of additive noise. The formulation for both batch and sequential SR is given in (B.1).

Batch SR	$\mathbf{y}_k = DBF_k \cdot \mathbf{z} + \mathbf{n}_k$	$k = 1 \cdots c$
Sequential SR	$\mathbf{y}_t = DB \cdot \mathbf{z}_t + \mathbf{n}_t$	
	$\mathbf{z}_t = F^t_{t-i} \cdot \mathbf{z}_{t-i} + \mathbf{v}_t$	

(B.1)

For batch SR, $\mathbf{y}_k$ represents the k^{th} LxN low resolution image. $\mathbf{z}$ is the corresponding HR image of size $mL \times mN$ where m is the down sampling factor, and c is the number of LR images. All images are represented in lexicographical ordering. Each LR image is obtained from $\mathbf{z}$ through warping (F_k), blurring (B), and down-sampling (D) operations. $\mathbf{n_k}$ represents the additive observation noise in the process. For sequential

SR, LR image at time t ($\mathbf{y}_t$) is obtained from $\mathbf{z}_t$ through blurring (B), and down-sampling (D) operations only. The relationship between two SR images at time t and $t-i$ is given through warping operator F_i^t where the notation in sub and superscripts denotes warping from time $t-i$ to t. Finally, $\mathbf{n}_t$ and $\mathbf{v}_t$ represents the additive noise in the processes.

A MAP estimator for both methods are formulated using Bayes' theory as follows;

Batch SR	$\hat{\mathbf{z}} = \underset{z}{argmax} p(\mathbf{y}_1, \cdots \mathbf{y}_c \vert \mathbf{z}) p(\mathbf{z})$	(B.2)
Sequential SR	$\hat{\mathbf{z}}_t = \underset{z_t}{argmax} p(\mathbf{y}_t, \mathbf{y}_{t-1} \cdots \mathbf{y}_{t-c} \vert \mathbf{z}_t) p(\mathbf{z}_t)$	

where $p(\mathbf{y}_k|\mathbf{z})$ is the conditional probability density of $\mathbf{y}_k$ given $\mathbf{z}$, $p(\mathbf{y}_{t-i}|\mathbf{z}_t)$ is the conditional probability density of $\mathbf{y}_{t-i}$ given $\mathbf{z}_t$ (t being the time tag). $p(\mathbf{z})$ and $p(\mathbf{z}_t)$ is related to the image prior, and c is the number of LR images. In relation with (B.1), the above estimation problem can be written as a minimization problem in the following form assuming: 1) noise has independent identical Gaussian distribution and 2) image prior is based on GMRF.

Batch SR	$\hat{\mathbf{z}} = \underset{z}{argmin} \sum_{k=1}^{c} \Vert DBF_k \mathbf{z} - \mathbf{y}_k \Vert_2^2 + \lambda \Vert C\mathbf{z} \Vert_2^2$	(B.3)
Sequential SR	$\hat{\mathbf{z}}_t = \underset{z_t}{argmin} \sum_{i=1}^{c} \Vert DBF_t^{t-i} \mathbf{z}_t - \mathbf{y}_{t-i} \Vert_2^2 + \lambda \Vert C\mathbf{z} \Vert_2^2$	

where $||C\mathbf{z}||_2^2$ represents the regularization term which depends on the image prior term. λ controls the influence of the regularization to the solution. C is usually expressed in the form of a spatial finite difference operator.

Steepest descent formulation for the cost function of sequential SR problem in (B.3) is as follows.

$$\hat{\mathbf{z}}_t^{n+1} = \hat{\mathbf{z}}_t^n - \beta^n \left(\sum_{i=0}^{c} \left[(DBF_t^{t-i})^T \cdot (DBF_t^{t-i} \hat{\mathbf{z}}_t^n - \mathbf{y}_{t-i}) \right] + \lambda C^T C \cdot \hat{\mathbf{z}}_t^n \right) \tag{B.4}$$

The above solution is not different than batch SR solution if number of images (c) is limited. However, as c grows, it gets computationally impossible to calculate $\sum_{i=0}^{c} \left[(DBF_t^{t-i})^T \cdot (DBF_t^{t-i} \hat{\mathbf{z}}_t^n - \mathbf{y}_{t-i}) \right]$ term directly.

A common practice is using a forgetting factor, μ ($0 \leq \mu < 1$), to decrease the effect of old observations to the reconstruction.

$$\hat{\mathbf{z}}_t^{n+1} = \hat{\mathbf{z}}_t^n - \beta^n \left(\sum_{i=0}^{\infty} \mu^i \Big[(DBF_t^{t-i})^T \cdot (DBF_t^{t-i} \hat{\mathbf{z}}_t^n - \mathbf{y}_{t-i}) \Big] + \lambda C^T C \cdot \hat{\mathbf{z}}_t^n \right) \tag{B.5}$$

Solving $\nabla Cost_t$ term directly requires all previous low resolution images to be buffered. Adaptive filtering theory has recursive solutions for this type of problems. In the next section recursive least squares (RLS) estimate is detailed which gives a direct solution to (B.5). Then pseudo RLS formulation is given where the forgetting factor is used to ignore very old observations completely. Finally, least mean squares (LMS) estimate is given.

B.2 Recursive Least Squares (RLS) Approach

Gradient of the cost function with infinitely many images has the following form. λ in (B.5) is set to zero for simplicity during the derivations.

$$\nabla Cost_t = \sum_{i=0}^{\infty} \mu^i \Big[(DBF_t^{t-i})^T \cdot (DBF_t^{t-i} \hat{\mathbf{z}}_t^n - \mathbf{y}_{t-i}) \Big] \tag{B.6}$$

The gradient term is rewritten in terms of two new variables $\mathbf{R}_t$ and $\mathbf{P}_t$. The aim is to rewrite the gradient at time t in terms of $\mathbf{R}_{t-1}$ and $\mathbf{P}_{t-1}$. Instead of buffering all the past images, these two variables will be buffered. Sizes of $\mathbf{P}_t$ and $\mathbf{R}_t$ are $LNm^2 \times 1$ and $LNm^2 \times LNm^2$ respectively. Resolution increase is set to m in both directions.

$$\nabla Cost_t = \underbrace{\sum_{i=0}^{\infty} \mu^i (DBF_t^{t-i})^T \cdot \mathbf{y}_{t-i}}_{\mathbf{P}_t} - \underbrace{\sum_{i=0}^{\infty} \mu^i (DBF_t^{t-i})^T (DBF_t^{t-i})}_{\mathbf{R}_t} \hat{\mathbf{z}}_t^n$$

$$\mathbf{R}_t \triangleq \sum_{i=0}^{\infty} \mu^i (DBF_t^{t-i})^T (DBF_t^{t-i})$$

$$\mathbf{P}_t \triangleq \sum_{i=0}^{\infty} \mu^i (DBF_t^{t-i})^T (DBF_t^{t-i}) \mathbf{y}_{t-i}$$

$$\mathbf{R}_t = (DB)^T (DB) + \sum_{i=1}^{\infty} \mu^i (DBF_t^{t-i})^T (DBF_t^{t-i})$$

$$\mathbf{P}_t = (DB)^T \mathbf{y}_t + \sum_{i=1}^{\infty} \mu^i (DBF_t^{t-i})^T \mathbf{y}_{t-i}$$

Replace the index i with $i+1$,

$$\mathbf{R}_t = (DB)^T (DB) + \sum_{i=0}^{\infty} \mu^{i+1} (DBF_t^{t-1-i})^T (DBF_t^{t-1-i})$$

$$\mathbf{P}_t = (DB)^T \mathbf{y}_t + \sum_{i=0}^{\infty} \mu^{i+1} (DBF_t^{t-1-i})^T \mathbf{y}_{t-1-i}$$

It is straightforward to see that transformation from time t to time $t-1-i$, F_t^{t-1-i}, is equivalent to transforming from time t to $t-1$ then from $t-1$ to $t-1-i$. Thus F_t^{t-1-i} is equivalent to $F_t^{t-1} \cdot F_{t-1}^{t-1-i}$ using this equality in the formulation of $\mathbf{R}_t$

$$\mathbf{R}_t = (DB)^T (DB) + \mu (F_t^{t-1})^T \underbrace{\sum_{i=0}^{\infty} \mu^i (DBF_{t-1}^{t-1-i})^T (DBF_{t-1}^{t-1-i})}_{\mathbf{R}_{t-1}} F_t^{t-1}$$

$$\mathbf{P}_t = (DB)^T \mathbf{y}_t + \mu (F_t^{t-1})^T \underbrace{\sum_{i=0}^{\infty} \mu^i (DBF_{t-1}^{t-1-i})^T \mathbf{y}_{t-1-i}}_{\mathbf{P}_{t-1}}$$

Steepest descent formulation given in (B.5) is calculated in two steps. *Step 1*: Update $\mathbf{P}_t$ and $\mathbf{R}_t$ in terms of its previous values

$$\mathbf{P}_t = \mu (F_t^{t-1})^T \mathbf{P}_{t-1} + (DB)^T \cdot \mathbf{y}_t$$

$$\mathbf{R}_t = \mu (F_t^{t-1})^T \mathbf{R}_{t-1} F_t^{t-1} + (DB)^T DB$$

Step 2: Update $\hat{\mathbf{z}}_t$ iteratively.

$$\hat{\mathbf{z}}_t^{n+1} = \hat{\mathbf{z}}_t^n - \beta^n \left[\mathbf{R}_t \hat{\mathbf{z}}_t^n - \mathbf{P}_t \right]$$

Although SR problem and its solutions are expressed in lexicographical format, the implementations are preferred to be in the 2 dimensional image domain. For example calculating back projection $D^T B^T F^T \mathbf{z}_t$ requires multiplying LN^2 by LNm^2 matrix D with LNm^2 by LNm^2 matrix B then with with LNm^2 by LNm^2 matrix F and

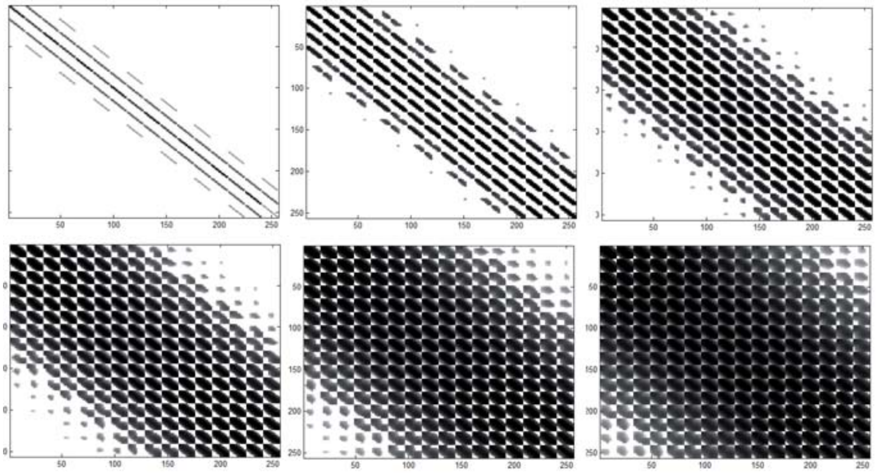


Figure B.1: Change in **R** as new images are added in time. The non zero elements along the diagonal increases away from the diagonal.

LNm^2 by 1 vector $\mathbf{z}_t$. Instead, B^T can be expressed as a 3 by 3 matrix convolution, and D^T as upsampling with zero padding, and F as image affine transform in 2D image domain. Mathematically both domains give exactly the same solution. This easy conversion is due to the structures of the matrices D, B, F. These matrices are sparse matrices as shown in Appendix A. For this reason computational complexity is calculated as the sum of operations involving non zero elements of the key matrices D, B, F.

Considering the elements of RLS calculations updating $\mathbf{P}_t$ requires, warping $\mathbf{P}_{t-1}$ and adding $(DB)^T \cdot \mathbf{y}_t$ term. These operations can easily be transferred to 2 dimensional image domain. Convert $\mathbf{P}_{t-1}$ as M by M image. Apply affine transform expressed in F_t^{t-1} in image domain. $(DB)^T \cdot \mathbf{y}_t$ term can be expressed as upsampling and zeropadding $\mathbf{y}_t$ in image domain and blurring it with simple convolution operation.

However updating $\mathbf{R}_t$ has difficulties, since as t increases $\mathbf{R}_t$ develops such that all elements of $\mathbf{R}_t$ is non-zero. In Figure B.1 for a 16 by 16 image the development of $\mathbf{R}_t$ is given. Unlike batch SR, calculating matrix $\mathbf{R}_t$ increases the complexity of the calculations dramatically.

B.3 Pseudo Recursive Least Squares (Pseudo-RLS) Approach

To overcome this problem a pseudo RLS approach is proposed. The gradient function in (B.6) is re-written in the following form.

$$\nabla Cost_t = \sum_{i=0}^{c} \mu^i \Big((DBF_t^{t-i})^T \cdot (DBF_t^{t-i} \hat{\mathbf{z}}_t{}^n - \mathbf{y}_{t-i}) \Big) + \sum_{i=c}^{\infty} \mu^i \Big((DBF_t^{t-i})^T \cdot (DBF_t^{t-i} \hat{\mathbf{z}}_t^n - \mathbf{y}_{t-i}) \Big)$$

The past observations can be selected to be forgotten completely according to the selection of the forgetting factor μ and image index c, depending on the application.

$$\nabla Cost_t = \sum_{i=0}^{c} \mu^i \Big((DBF_t^{t-i})^T \cdot (DBF_t^{t-i} \hat{\mathbf{z}}_t^n - \mathbf{y}_{t-i}) \Big) + \underbrace{\mu^c}_{0} \sum_{i=0}^{\infty} \mu^i \Big((DBF_t^{t-i})^T \cdot (DBF_t^{t-i} \hat{\mathbf{z}}_t^n - \mathbf{y}_{t-i}) \Big)$$

With the above simplification of the cost term, $\nabla Cost_t$, SR problem simplifies to the standard SR formulation given in the following equation,

$$\hat{\mathbf{z}}_t^{n+1} = \hat{\mathbf{z}}_t^n - \beta^n \sum_{i=0}^{c} \mu^i \Big((DBF_t^{t-i})^T \cdot (DBF_t^{t-i} \hat{\mathbf{z}}_t^n - \mathbf{y}_{t-i}) \Big) \tag{B.7}$$

B.4 Least Mean Squares Estimate

Setting c in (B.7) to 0 leads to the least mean squares (LMS) estimate given in the following equation,

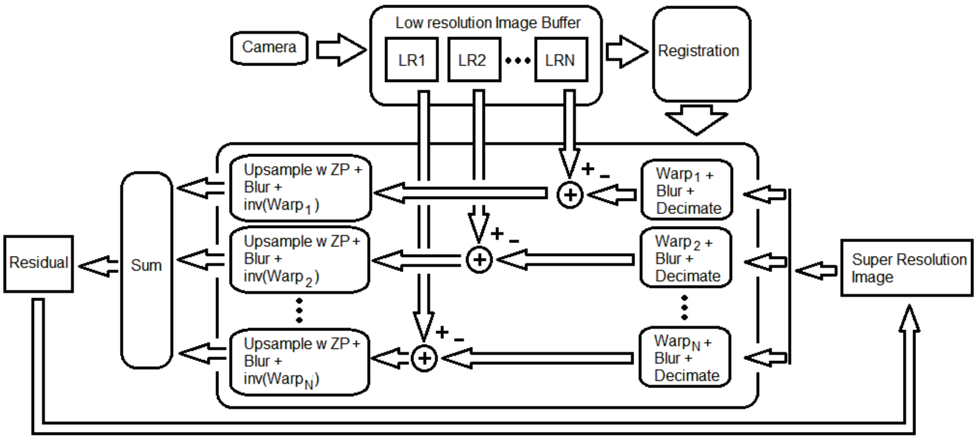
$$\hat{\mathbf{z}}_t^{n+1} = \hat{\mathbf{z}}_t^n - \beta^n \Big[(DB)^T \cdot (DB\, \hat{\mathbf{z}}_t^n - \mathbf{y}_t) \Big] \tag{B.8}$$

Adding a regularization term with a GMRF prior results in the following formulation.

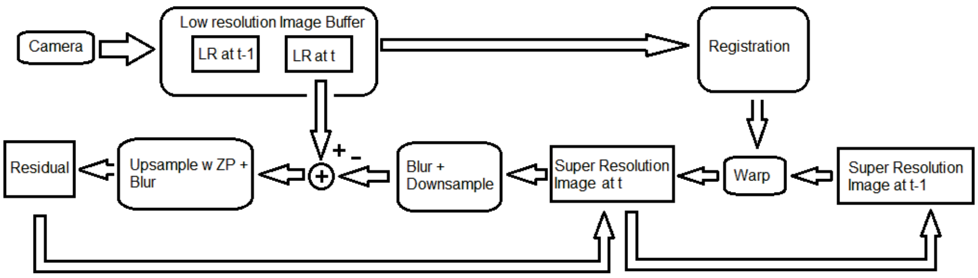
$$\hat{\mathbf{z}}_t^{n+1} = \hat{\mathbf{z}}_t^n - \beta^n \Big[(DB)^T \cdot (DB\, \hat{\mathbf{z}}_t^n - \mathbf{y}_t) + \lambda C^T C \hat{\mathbf{z}}_t^n \Big] \tag{B.9}$$

A comparison of batch SR and LMS SR methods are given In Figure B.2.

The video flow for RLS, pseudo-RLS, LMS and video interpolation methods are graphically given in Figures B.3 and B.4 to summarize the sequential methods derived in this section.

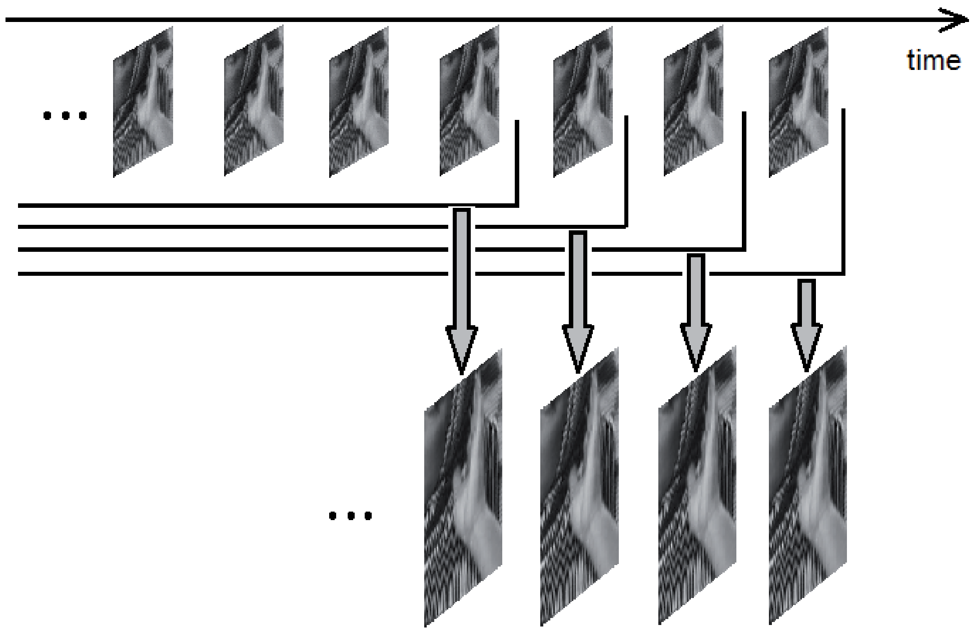


(a) Batch Super Resolution

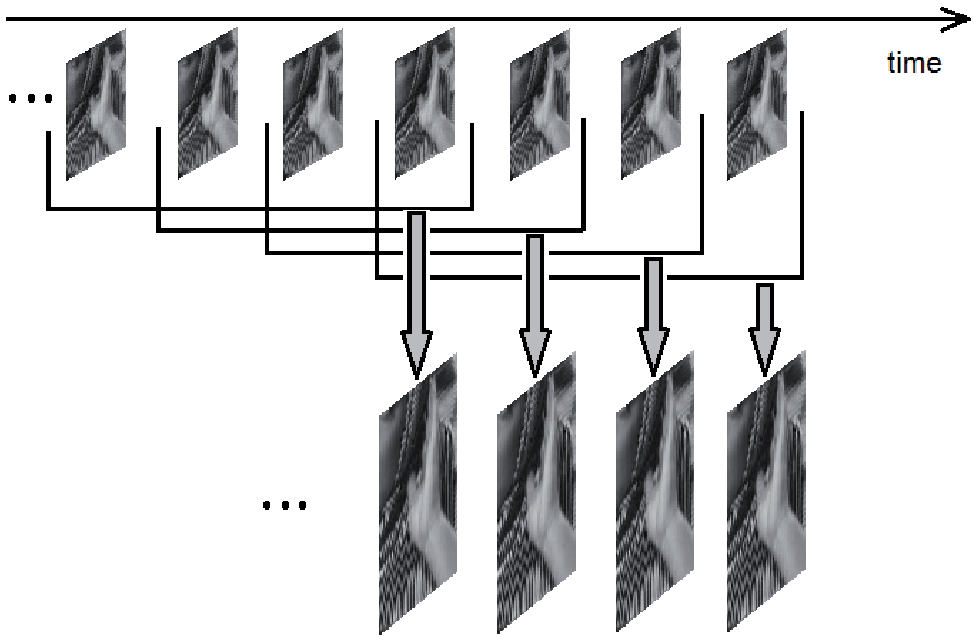


(b) LMS Super Resolution

Figure B.2: Batch SR versus Sequential SR with LMS method is given.

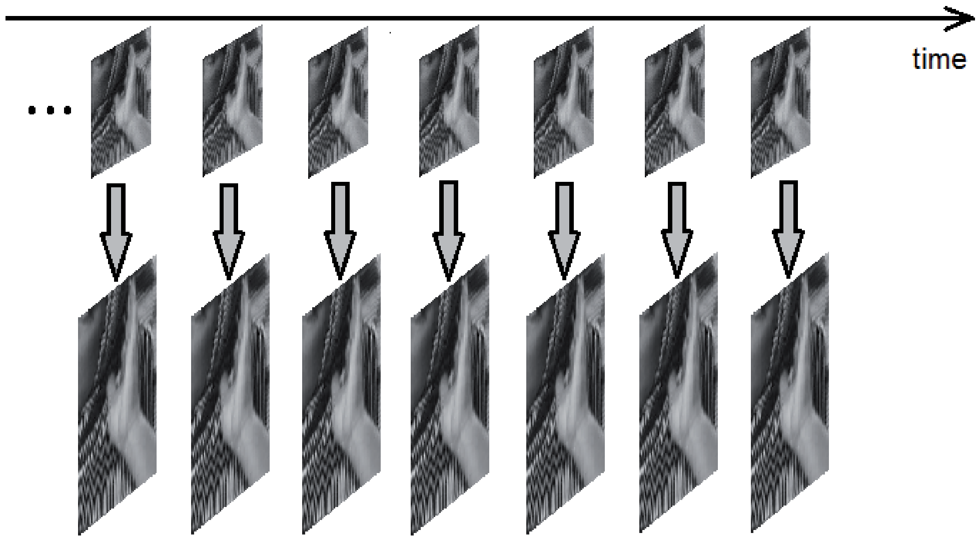


(a) Video SR with RLS Estimation

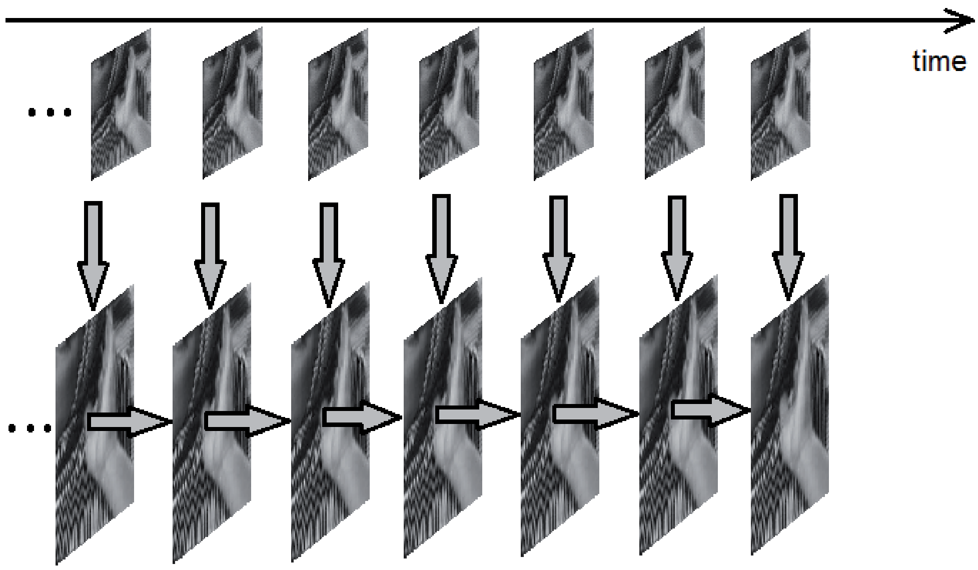


(b) Video SR with pseudo-RLS estimation.

Figure B.3: Flow of RLS and Pseudo-RLS algorithms.



(a) Video Upsampling with Bicubic interpolation



(b) Video SR with LMS Estimation

Figure B.4: Flow of Bicubic interpolation and LMS methods.

CURRICULUM VITAE

Emre Turgay was born in Aydın, Turkey in 25.12.1979. He received the B.Sc. degree in Electrical and Electronics Engineering Department from Middle East Technical University, Ankara, Turkey, in 2001 and the M.A.Sc. degree in the Department of Electrical and Computer Engineering from the University of British Columbia, Vancouver, BC, Canada, in 2004.

He has been working in Aselsan Military Electronics Company as a senior software engineer since November 2004. His primary research interests are, image enhancement, image super resolution, thermal image restoration, and software management.

His publications are listed as follows:

- E. Turgay and G. B. Akar, "Texture and Edge Preserving Multiframe Super-Resolution", *IET Image Process.*, in press 2013.
- E. Turgay and G. B. Akar, "Texture Preserving Multi Frame Super Resolution With Spatially Varying Image Prior" in *International Conference on Image Processing*. IEEE, October 2012.
- F. Erbay, E. Turgay and G. B. Akar, "Color super resolution in HSV domain," in *Signal Processing and Communications Applications - SIU*. IEEE, 2011 April, pp. 1064–1067.
- E. Turgay and G. B. Akar, "Texture preserving super resolution in thermal images," in *Signal Processing and Communications Applications - SIU*. IEEE, 2011 April, pp. 578–581.
- E. Turgay and O. Teke, "Autofocus method in thermal cameras based on image histogram," in *Signal Processing and Communications Applications - SIU*. IEEE, 2011 April, pp. 462–465.
- E. Turgay and G. B. Akar, "Directionally adaptive super-resolution" in *International Conference on Image Processing*. IEEE, October 2009, pp. 1201–1204.

- E. Turgay and G. B. Akar, “Context based super resolution image reconstruction,” in *Local and Non-Local Approximation in Image Processing*. IEEE, August 2009, pp. 54–61.
- E. Turgay and G. B. Akar, “Direction Adaptive Super Resolution Imaging,” in *Signal Processing and Communications Applications - SIU*. IEEE, 2009 April, pp.37–40.
- E. Turgay, S. Salcudean and R. Rohling. “Identifying the mechanical properties of tissure by ultrasound strain imaging”. *Ultrasound in Medicine and Biology* . Elsevier Science Inc. New York, NY. 32(2) pp. 221–235. 2005.
- Salcudean, S. E., R. N. Rohling, and E. Turgay, “Method for imaging the mechanical properties of tissue” , no. 20050119568, US, allowed May 2010, 2004.
- E. Turgay, S. MacIntosh, R. N. Rohling, and S. E. Salcudean, “Parameter identification of tissue lumped models based on sequences of ultrasonic strain images”, *International Conference on the Ultrasonic Measurement and Imaging of Tissue Elasticity*, Corpus Christi, Texas, pp. 29, October 2003.