2009

# Bilateral filter in image processing

Ming Zhang
*Louisiana State University and Agricultural and Mechanical College*

# BILATERAL FILTER IN IMAGE PROCESSING

A Thesis
Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering

in

The Department of Electrical and Computer Engineering

by
Ming Zhang
B.S., Beijing University of Posts and Telecommunications, 2006
August 2009

## ACKNOWLEDGEMENTS

I would like to thank Louisiana State University and the National Science Foundation, NSF for funding this research. I would also like to thank Professor Bahadir Gunturk for guiding me and sharing his wealth of experience and knowledge to further my education.

Thanks to fellow students Murat Gevrecki, Changzhu Wu and Stephen Bishop for their helps in general.

Thanks to my parents and girlfriend who always support me.

## TABLE OF CONTENTS

## ABSTRACT

The bilateral filter [1] is a nonlinear filter that does spatial averaging without smoothing edges. It has shown to be an effective image denoising technique. It also can be applied to the blocking artifacts reduction. An important issue with the application of the bilateral filter is the selection of the filter parameters, which affect the results significantly. Another research interest of bilateral filter is acceleration of the computation speed.

There are three main contributions of this thesis. The first contribution is an empirical study of the optimal bilateral filter parameter selection in image denoising. I propose an extension of the bilateral filter: multi resolution bilateral filter, where bilateral filtering is applied to the low-frequency sub-bands of a signal decomposed using a wavelet filter bank. The multi resolution bilateral filter is combined with wavelet thresholding to form a new image denoising framework, which turns out to be very effective in eliminating noise in real noisy images. The second contribution is that I present a spatially adaptive method to reduce compression artifacts. To avoid over-smoothing texture regions and to effectively eliminate blocking and ringing artifacts, in this paper, texture regions and block boundary discontinuities are first detected; these are then used to control/adapt the spatial and intensity parameters of the bilateral filter. The test results prove that the adaptive method can improve the quality of restored images significantly better than the standard bilateral filter. The third contribution is the improvement of the fast bilateral filter, in which I use a combination of multi windows to approximate the Gaussian filter more precisely.

# 1. INTRODUCTION

## 1.1 Image Denoising

Figure 1.1 Original image and its red, green, and blue channels are displayed in parallel scan order.

There are different sources of noise in a digital image. For example, dark current noise is due to the thermally generated electrons at sensing sites; it is proportional to the exposure time and highly dependent on the sensor temperature. Shot noise is due to the quantum uncertainty in photoelectron generation; and it is characterized by Poisson distribution. Amplifier noise and quantization noise occur during the conversion of the number of electrons generated to pixel intensities. The overall noise characteristics in an image depend on many factors, including sensor type, pixel dimensions, temperature, exposure time, and ISO speed. Noise is in general spatial position and channel dependent. Blue channel is typically the noisiest channel due to the low transmittance of blue filters. In single-chip digital cameras, demosaicking algorithms are used to interpolate missing color components; therefore, noise is not uncorrelated for different pixels. An often neglected characteristic of image noise is the spatial frequency. Figure 1.1 shows the portion of an image captured with

a Sony DCR-TRV27, and its red, green, and blue channels are shown in parallel scan order. The blue channel is the most degraded channel; it has a coarse-grain noise characteristics. The red and green channels have finer-grain noise characteristics. Referring to Figure 1, noise may have low-frequency (coarse-grain) and high-frequency (fine-grain) fluctuations. High-frequency noise is relatively easier to remove; on the other hand, it is difficult to distinguish between real signal and low-frequency noise.

**1.2 Bilateral Filter**

Bilateral filter [1] is firstly presented by Tomasi and Manduchi in 1998. The concept of the bilateral filter was also presented in [2] as the SUSAN filter and in [3] as the neighborhood filter. It is mentionable that the Beltrami flow algorithm is considered as the theoretical origin of the bilateral filter [4] [5] [6], which produces a spectrum of image enhancing algorithms ranging from the $L_2$ linear diffusion to the $L_1$ non-linear flows. The bilateral filter takes a weighted sum of the pixels in a local neighborhood; the weights depend on both the spatial distance and the intensity distance. In this way, edges are preserved well while noise is averaged out. Mathematically, at a pixel location x, the output of a bilateral filter is calculated as follows，

$$\tilde{I}(\mathbf{x}) = \frac{1}{C} \sum_{y \in N(x)} e^{\frac{-\|\mathbf{y}-\mathbf{x}\|^2}{2\sigma_d^2}} e^{\frac{-|I(\mathbf{y})-I(\mathbf{x})|^2}{2\sigma_r^2}} I(y) \tag{1.1}$$

where $\sigma_d$ and $\sigma_r$ are parameters controlling the fall-off of weights in spatial and intensity domains, respectively, $N(x)$ is a spatial neighborhood of pixel $I(x)$, and C is the normalization constant:

$$C = \sum_{y \in N(x)} e^{\frac{-\|\mathbf{y}-\mathbf{x}\|^2}{2\sigma_d^2}} e^{\frac{-|I(\mathbf{y})-I(\mathbf{x})|^2}{2\sigma_r^2}} \quad (1.2)$$

Figure 1.2 shows the illustration of 1D bilateral filter. The top right image is the input noisy signal. The top left image shows the intensity Gaussian while the middle image shows the special Gaussian. The bilateral response is shown at the bottom.

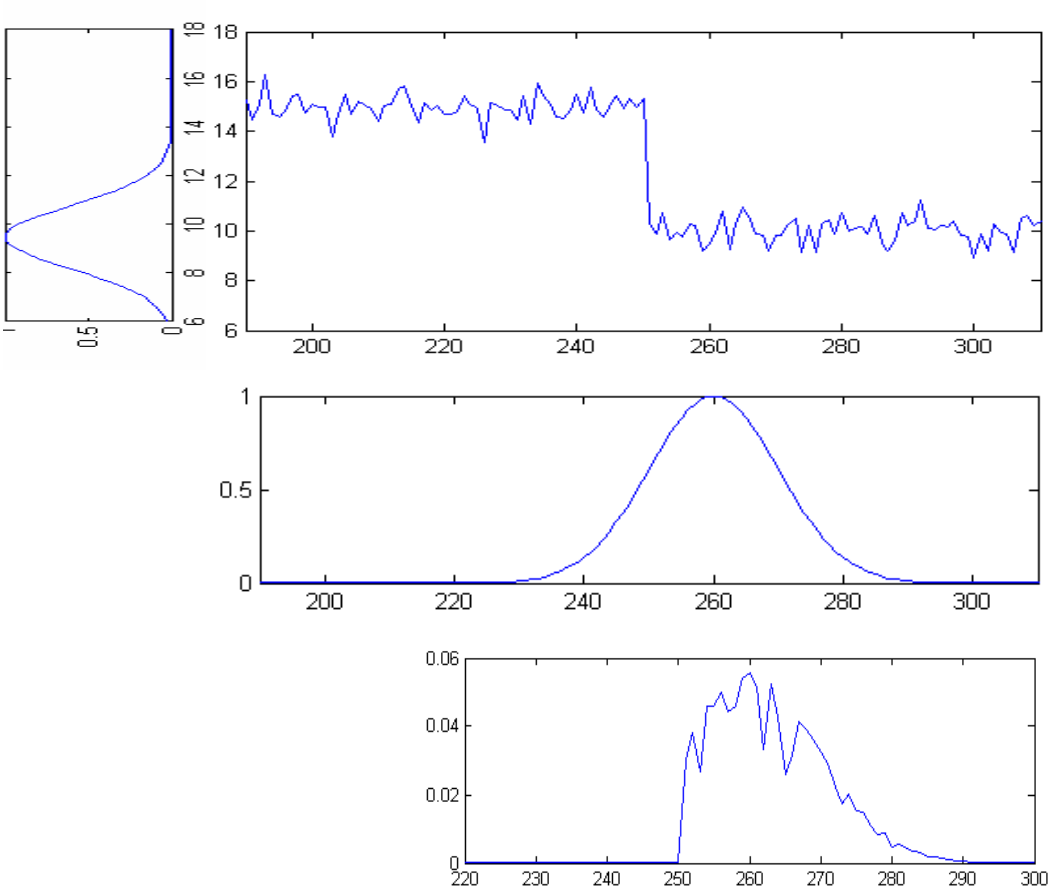



Figure 1.2 Illustration of 1-D bilateral filter. The filter is applied on a 1-D input step signal noised by random Gaussian noise. The output of the filter is shown in the bottom figure.

Another parameter during the running of the bilateral filter is the window size of how many pixels should be computed on time. The window size is related to the spatial Gaussian. Basically, based on the property of the Gaussian distribution, window size should be around

2 to 3 times the standard deviation of the Gaussian, since when it's over 3 times sigma, the output of Gaussian almost equals to zero.

In some research, it is shown that the bilateral filter is identical to the first iteration of the Jacobi algorithm (diagonal normalized steepest descent) with a specific cost function. Elad et al. [7] related the bilateral filter with the anisotropic diffusion.

**1.3 Wavelet Decomposition and Thresholding**

Wavelet is a mathematical function used to divide a given function or continuous-time signal into different scale components. One can assign a frequency range to each scale component. Each scale component can then be studied with a resolution that matches its scale. Thus the Wavelet is a multi resolution representation function.

Wavelet transform is the discrete sampling of the wavelets. Based on the recurrence relations property of wavelet, the most common wavelet transforms, such as Daubechies wavelet transform, generate progressively finer discrete samplings of an implicit mother wavelet function; each resolution is twice that of the previous scale down-sampled by 2.

Therefore, using the one level wavelet transform, the input signal can be decomposed into two frequency coefficients, the approximation coefficients as the low frequency part and the detail coefficients as the high frequency part. This is the so called wavelet decomposition.

With higher level decompositions, multi resolution representation of the signal can be achieved. Figure 1.3 shows the wavelet decomposition of an image. The left picture is the original image and the right one using 1-level wavelet decomposition. We can see from the right image that the top left small picture is the low frequency part which keeps the energy

mostly while the others are detail information.

Figure 1.3 Right one shows the wavelet decomposition of the left picture. It uses 1 level Db4 wavelet decomposition in the Matlab. The image size is $512 \times 512$

Wavelet thresholding is a denoising method that applies the thresholding shrinkage upon the high frequency components after the wavelet decomposition. There are two basic thresholding methods, the hard thresholding and the soft thresholding, by which the threshold value is computed.

Generally speaking, the soft thresholding is used for the wavelet thresholding application. The most common wavelet thresholding methods are Bayes Shrink [10], Visu Shrink[9] and SURE Shrink [8].

### 1.4 Compression Artifacts

Block-based discrete cosine transform (BDCT) is adopted by widely used image/video compression standards, such JPEG, MPEG, and H-263, due to its high energy compaction and low computational complexity.

Before BDCT, the image must be split into $8 \times 8$ blocks of pixels. If the data does not represent an integer number of blocks then the encoder must fill the remaining area of the incomplete blocks with some form of dummy data. Filling the edge pixels with a fixed color

(typically black) creates ringing artifacts along the visible part of the border; repeating the edge pixels is a common technique that reduces the visible border, but it can still create artifacts. DCT is applied on such $8 \times 8$ blocks.

Figure 1.4 Illustration of blocking artifacts. Left is the original image with size of $512 \times 512$, while the right one is the compressed output of left with bit-rate=0.18.

After the DCT, quantization is used to reduce the amount of information in the high frequency components. This is done by simply dividing each component in the frequency domain by a constant for that component, and then rounding to the nearest integer, which is the main lossy operation in the whole process. As a result of this, it is typically the case that many of the higher frequency components are rounded to zero, and many of the rest become small positive or negative numbers, which take many fewer bits to store. When using quantization with block-based coding, several types of artifacts can appear, including staircase noise along curving edges, "mosquito noise" around edges, and blocking artifacts.

The major problem here is the blocking artifacts, the discontinuities along the block boundaries. The blocking artifacts and other compression artifacts become more severe with increasing compression rates. Figure 1.4 shows the illustration of an image with blocking

artifacts after over compressed

### 1.5 Overview of the Thesis

In order to solve the image denoising and compression artifacts reduction problems, this thesis consists of five parts. In Chapter 2 is given a comprehensive literature review, which covers the most popular and advanced researches in the certain fields. In Chapter 3, I will explain the elaborate proposed methods for image denoising. A multi resolution bilateral filter is applied to the image for real noise elimination. In Chapter 4, I will give the full results of the experiments in image denoising. Then, in chapter 5, I will discuss the framework of my method in compression artifacts reduction, which uses a spatial adaptive bilateral filter. In Chapter 6, all the experiment results and data for compression artifacts reduction are presented. Chapter 7 makes the summary and conclusion.

# 2. LITERATURE REVIEW

## 2.1 Image Denoising Applications

### 2.1.1 Wavelet Approaches

Donoho and Johnstone [8][9] provided an ideal spatial adaptive wavelet shrinkage. With ideal spatial adaptation, they described a new principle for spatially-adaptive estimation: selective wavelet reconstruction. It showed that variable-knot spline fits and piecewise-polynomial fits, when equipped with an oracle to select the knots, are not dramatically more powerful than selective wavelet reconstruction with an oracle. Then they developed a practical spatially adaptive method, SureShrink[9], which works by shrinkage of empirical wavelet coefficients. A new inequality in multivariate normal decision theory which they called the oracle inequality showed that attained performance differs from ideal performance.

Chang and Vetterli [10] proposed an adaptive, data-driven threshold for image denoising using the wavelet soft-thresholding. The threshold is derived in a Bayesian framework, and the prior used on the wavelet coefficients is the generalized Gaussian distribution (GGD) widely used in image processing applications. The proposed threshold is closed-form and adaptive to each sub-band. This method, so called BayesShrink [10], outperforms Donoho and Johnstone's SureShrink [9] most of the time.

Since wavelet coefficients of real images have significant dependencies, Sendur et al. [11] considered the dependencies between the coefficients and their parents in the detail coefficients part. For this purpose, the non-Gaussian bivariate distributions are proposed, and corresponding nonlinear threshold functions are derived from the models using

Bayesian estimation theory. The new shrinkage functions do not assume the independence of wavelet coefficients. However, the performance of this method is not very well.

Pezurica [12] et al. developed three wavelet domain denoising methods for sub-band adaptive, spatially adaptive and multi-valued image denoising. The core of his approach is the estimation of the probability that a given coefficient contains a significant noise-free component, which is called "signal of interest." In this respect, he analyzed cases where the probability of signal presence is 1) fixed per sub-band, 2) conditioned on a local spatial context, and 3) conditioned on information from multiple image bands. All the probabilities are estimated assuming a generalized Laplacian prior for noise-free sub-band data and additive white Gaussian noise. His sub-band adaptive shrinkage function outperforms Bayesian thresholding approaches in terms of MSE (Mean-Squared Error).

Portilla [13] et al. developed a model for neighborhoods of oriented pyramid coefficients based on a Gaussian scale mixture: the product of a Gaussian random vector, and an independent hidden random scalar multiplier. This model, called BLS-GSM, can account for both marginal and pair-wise joint distributions of wavelet coefficients. Then he showed a local denoising solution as a Bayesian least squares estimator, and demonstrated the performance of this method on images corrupted by simulated additive white Gaussian noise of known variance. Portilla's methods

One of the best wavelet thresholding methods recently is the SureShrink based on the inter-scale orthonormal wavelet transform. Instead of postulating a statistical model for the wavelet coefficients, Luisier et al. [14] directly parameterized the denoising process as a sum of elementary nonlinear process with unknown weights. Then minimize an estimate of

the mean square error between the clean image and the denoised one. He use the statistically unbiased, MSE estimate—Stein's unbiased risk estimate—that depends on the noisy image alone, not on the clean one. Like the MSE, this estimate is quadratic in the unknown weights, and its minimization amounts to solving a linear system of equations. The existence of this a priori estimate makes it unnecessary to devise a specific statistical model for the wavelet coefficients. Instead, and contrary to the custom in the literature, these coefficients are not considered random anymore.

### 2.1.2 Non-wavelet Approaches

Denoising images can be achieved by a spatial averaging of nearby pixels. This method removes noise but creates blur. Henceforth, neighborhood filters, which perform an average of neighboring pixels under the condition that their grey level is close enough to the one of the pixel in restoration, creates shocks and staircasing effects. Buades et al. [15] performed an asymptotic analysis of neighborhood filters as the size of the neighborhood shrinks to zero. His paper proved that these filters are asymptotically equivalent to the Perona-Malik equation [16], one of the first nonlinear PDE proposed for image restoration. In continuation, he proposed an extremely simple variant of the neighborhood filter using a linear regression instead of an average. By analyzing its subjacent PDE, the artifacts can be eliminated.

Elad et al. [17][18][19] addressed his approach based on sparse and redundant representations over a trained dictionary. The proposed algorithm denoised the image, while simultaneously training a dictionary on its corrupted content using the K-SVD algorithm. As the dictionary training algorithm is limited in handling small image patches, the author extended its deployment to arbitrary image sizes by defining a global image prior that forces

sparsity over patches in every location in the image.

Kernel regression is also a popular state-of-the-art method for image denoising. Takeda et al. [20] made contact with the field of nonparametric statistics and adapt kernel regression ideas for use in image denoising, upscaling, interpolation, fusion, and more. They established key relationships with some popular existing methods and show how several of these algorithms, including the recently popularized bilateral filter, are special cases of the proposed framework. Especially they proposed the iterative steering regression which has a better performance than the bilateral filter for the elimination of both Gaussian white noises and real noise.

Patch-based approach [21] is proposed by Kervrann et al. The method is based on a point-wise selection of small image patches of fixed size in the variable neighborhood of each pixel. Associate with each pixel the weighted sum of data points within an adaptive neighborhood in a manner that it balances the accuracy of approximation and the stochastic error at each spatial position. By introducing spatial adaptivity, they extend the Non-local means filter which can be considered as an extension of bilateral filtering to image patches. So they propose a nearly parameter-free algorithm for image denoising.

One of the best methods in non-wavelet pattern is called sparse 3D transform domain collaborative filtering (BM3D) [22] by Dabov et al. Their strategy is based on an enhanced sparse representation in transform domain. The enhancement of the sparsity is achieved by grouping similar 2D image fragments (e.g. blocks) into 3D data arrays called "groups". Collaborative filtering is a special procedure developed to deal with these 3D groups. The result is a 3D estimate that consists of the jointly filtered grouped image blocks. By

attenuating the noise, the collaborative filtering reveals even the finest details shared by grouped blocks and at the same time it preserves the essential unique features of each individual block.

**2.2 Blocking Artifacts Reduction**

In the literature, there are numerous methods proposed to reduce compression artifacts. Some methods are introduced as a part of the encoding process, such as the lapped transform. Since these methods require modification of the codec, alternative post-processing methods, which do not require any codec changes, have become main focus in the area. The post-processing methods can be categorized into two: enhancement based algorithms and restoration based algorithms. Enhancement based algorithms try to improve the perceptual quality without an explicit optimization process; on the other hand, restoration based algorithms try to recover the original image based on some optimization criteria. Another way of categorizing these methods is spatial domain vs. transform domain, depending on which domain the image is processed. There are methods that use both domains.

An example of the enhancement based algorithms is by Apostolopoulos et al. [23], where the blockiness is first detected based on the number of zero DCT coefficients in each block, and then applying 1D median filter to reduce block discontinuities and 2D median filter to reduce mosquito artifacts.

A restoration based algorithm is proposed by Katsaggelos [24] via the Bayesian approach. They used the hierarchical Bayesian paradigm to the reconstruction of block discrete cosine transform (BDCT) compressed images and the estimation of the required parameters. Then derive expressions for the iterative evaluation of these parameters applying the evidence

analysis within the hierarchical Bayesian paradigm. This method allows for the combination of parameters estimated at the coder and decoder.

Another restoration base method is POCS (projection onto convex sets) by Liew et al. [25]. POCS method is presented for the suppression of blocking and ringing artifacts in a compressed image that contains homogeneous regions. In their paper, a new family of convex smoothness constraint sets is introduced, using the uniformity property of image regions. This set of constraints allows different degrees of smoothing in different regions of the image, while preserving the image edges. The regions are segmented using the fuzzy c-means algorithm, which allows ambiguous pixels to be left unclassified.

Wu et al. [26] proposed the post-filter using the DCT coefficients of shifted blocks to deblock and preserve the details. For each block, its DC value and DC values of the surrounding eight neighbor blocks are exploited to predict low frequency AC coefficients. Those predicted AC coefficients allow inferring spatial characteristics of a block before quantization stage in the encoding system. They are used to classify each block into either of two categories, low-activity and high-activity block. In the following post-processing stage, two kinds of low pass filters are adaptively applied according to the classified result on each block. It allows for strong low pass filtering in low-activity regions where the blocking artifacts are most noticeable, whereas it allows for weak low pass filtering in high-activity regions to reduce ringing noise as well as blocking artifacts without introducing undesired blur.

Bovik et al. [27] proposed the fast and blind measurement of detection and reduction to the blocks in the DCT domain. In the algorithm, blocking artifacts are modeled as 2-D step

functions. A fast DCT-domain algorithm extracts all parameters needed to detect the presence of, and estimate the amplitude of blocking artifacts, by exploiting several properties of the human vision system. Using the estimate of blockiness, a novel DCT-domain method is then developed which adaptively reduces detected blocking artifacts.

Delp et al. [28] presented the DCT-domain Markov Random Field model. It is called transform-domain Markov random field (TD-MRF) model, which used two block artifact reduction post-processing methods. The first method, referred to as TD-MRF, provides an efficient progressive transform-domain solution. It can reduce up to 90% of the computational complexity compared with spatial-domain MRF (SD-MRF) methods while still achieving comparable visual quality improvements. Then they discuss a hybrid framework, referred to as TSD-MRF, which exploits the advantages of both TD-MRF and SD-MRF.

Bilateral filter [1] is also a popular methods used for image compression artifacts reduction. Lin et al. [29] came up with a fast algorithm which alleviates the said artifacts in the DCT domain. It decomposed a row or column image vector to a gradually changed signal and a fast variation signal, which correspond to low-frequency (LF) and high-frequency (HF) DCT sub-bands respectively. Blocking artifacts between adjacent LF blocks are suppressed by smoothing LF components and discarding invalid HF ones, and ringing artifacts inside HF vectors are reduced by a simplified bilateral filter. With such a process, edges are preserved while blockiness and ringing alleviated. Experimental results confirm the robustness and computational efficiency of the proposed method.

### 2.3 Fast Bilateral Filter

Paris and Durand [30] analyzed accuracy in terms of bandwidth and sampling, and derive criteria for downsampling in space and intensity to accelerate the bilateral filter by extending an earlier work on high dynamic range images. Their method approximates the bilateral by filtering subsampled copies of the image with discrete intensity kernels, and recombining the results using linear interpolation. In other words, this method treats the intensity image as a 3D surface, applies Gaussian smoothing to binary and intensity modulated surface, and divides them to determine the filtered intensity values at the original surface location. It becomes faster as the size increases due to the greater subsampling of the surface. The exact output is dependent on the phase of the subsampling grid and the discretization leads to further loss of precision particularly on high dynamic range images.

Another fastest bilateral filter implementation whose computation cost converges to O(logn) (n is the total number of the pixels) was developed by Weiss [31] using a hierarchy of partial distributed histograms using a tier-based approach. Even though complexity has been lowered, simplicity has been lost due to filter size and optimal histogram count specific implementation requirements. This method is limited to rectangular spatial kernels and box filters. Another concern is the imperfect frequency response of their spatial box filter.

Porikli [32] described a constant time bilateral filtering method. He constructed an integral histogram and used the integral histogram to find the bilateral convolution response of a rectangular box filter with uniform domain kernel, where the intensity differences can be weighted with any arbitrary range function. The integral histogram enables computation of histograms of all possible kernels in a given image. It takes advantage of the spatial

positioning of data points in a Cartesian coordinate system, and propagates an aggregated function starting from an origin point and traversing through the remaining points along a scan-line. Histograms of image windows can be computed easily by using the integral histogram values at the corner points of those windows without reconstructing a separate histogram for every single one of them. For more generic Gaussian and polynomial range functions on arbitrary domain kernels, he applied Taylor series expansion of the corresponding norms. This second method can use “any” spatial kernel for bilateral filtering without increasing the complexity. It is shown that such bilateral filters can be expressed in terms of spatial linear filters applied on original image powers.

# 3. IMAGE DENOISING METHODS

## 3.1 Multi Resolution Bilateral Filter

### 3.1.1 Parameter Selection of Bilateral Filter

There are two parameters that control the behavior of the bilateral filter [1]. Referring to (1.1), $\sigma_d$ and $\sigma_r$ characterize the spatial and intensity domain behaviors, respectively. In case of image denoising applications, the question of selecting optimal parameter values has not been answered from a theoretical perspective; to the best of our knowledge, there is no empirical study on this issue either. In this section, I provide an empirical study of optimal parameter values as a function of noise variance.

To understand the relationship among $\sigma_d$, $\sigma_r$ and the noise standard deviation $\sigma_n$, the following experiments were done. Zero-mean white Gaussian noise was added to some standard test images and the bilateral filter was applied for different values of the parameters $\sigma_d$ and $\sigma_r$. The experiment was repeated for different noise variances and the mean squared error (MSE) values were recorded. Typical MSE contour plots are given in Figure 3.1-3.4. Examining these plots, it can be seen that the optimal $\sigma_d$ value is relatively insensitive to noise variance compared to the optimal $\sigma_n$ value. It looks the best range for the $\sigma_d$ value is between 1.5 and 2.0; on the other hand, the optimal $\sigma_r$ value changes significantly as the noise standard deviation $\sigma_n$ changes. This is an expected result. Because if $\sigma_r$ is smaller than $\sigma_n$, noisy data could remain isolated and untouched as in the case of salt-and-pepper noise problem of the bilateral filter.

To see the relationship between $\sigma_n$ and the optimal $\sigma_r$, we set $\sigma_d$ to some constant values, and plotted the optimal $\sigma_r$ values as a function of $\sigma_n$. Figure 3.5, Figure 3.6 and

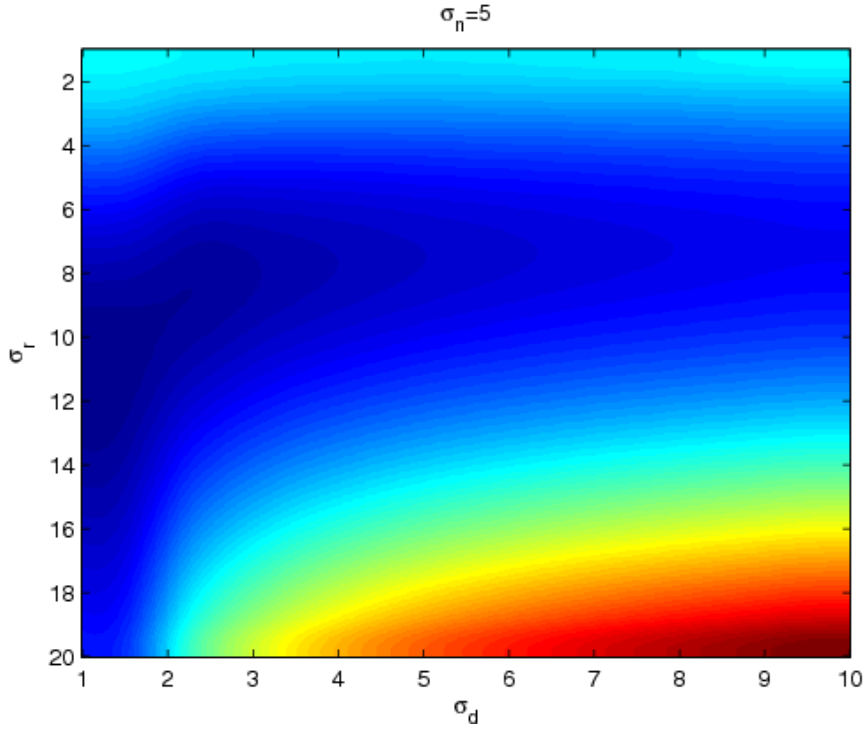



Figure 3.1 The contour plots of the MSE values between the original image and the denoised image for different values sigma_d and sigma_r. The noise standard deviation sigma_n=5. The results are averaged from 60 color and gray-scale test images.

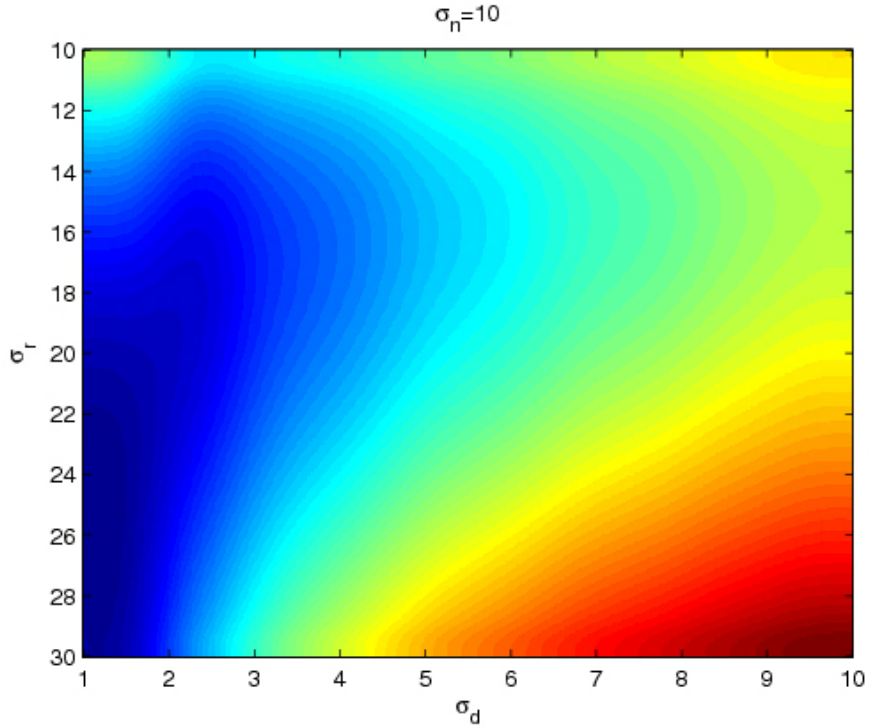



Figure 3.2 The contour plots of the MSE values between the original image and the denoised image for different values sigma_d and sigma_r. The noise standard deviation sigma_n =10. The results are averaged from 60 color and gray-scale test images.

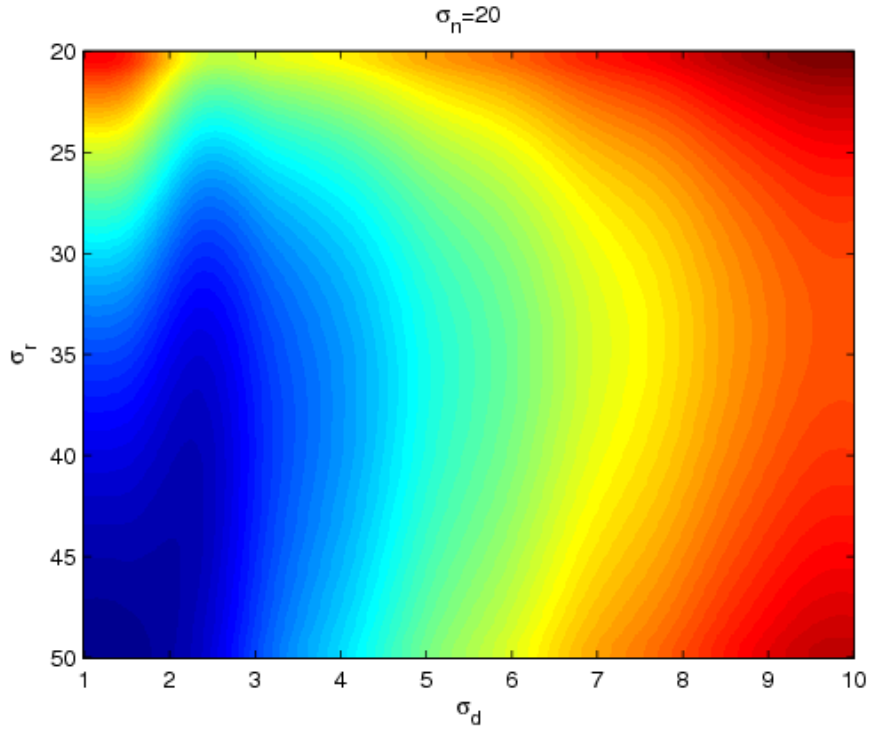



Figure 3.3 The contour plots of the MSE values between the original image and the denoised image for different values sigma_d and sigma_r. The noise standard deviation sigma_n =20. The results are averaged from 60 color and gray-scale test images.

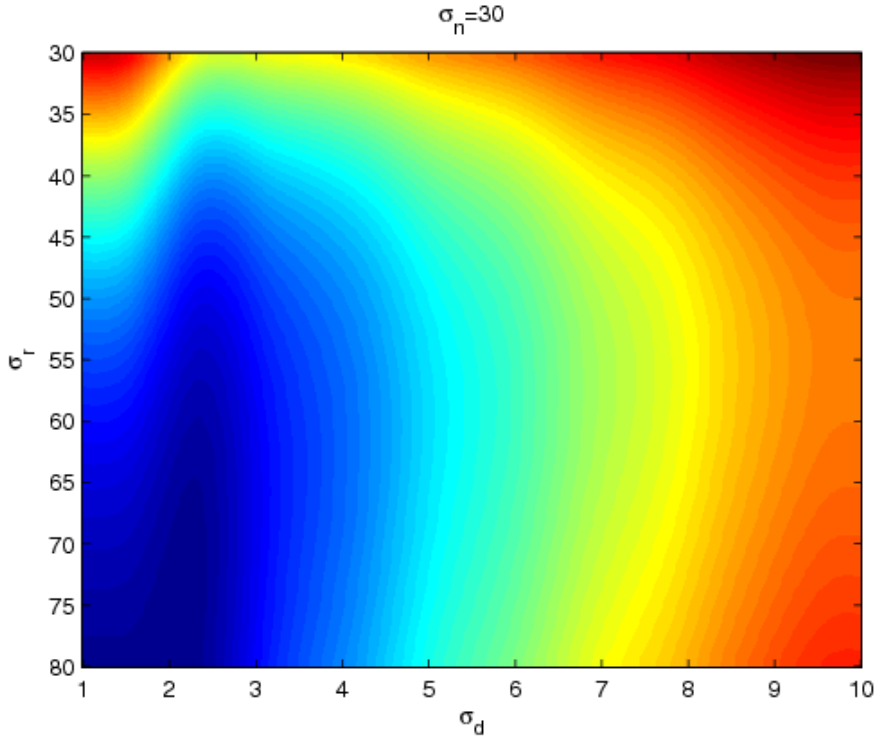



Figure 3.4 The contour plots of the MSE values between the original image and the denoised image for different values sigma_d and sigma_r. The noise standard deviation sigma_n =30. The results are averaged from 60 color and gray-scale test images.

Figure 3.7 show these plots for 60 standard images. The $\sigma_r$ values as a function of the noise standard deviation $\sigma_n$ are plotted as the averaged data from 60 standard test images. The blue data points are the mean of optimal $\sigma_r$ values that produce smallest MSE for each $\sigma_n$ value. The blue vertical lines denote the standard deviation of the optimal $\sigma_r$ for the 60 different images. The least squares that fit to $(\sigma_r / \sigma_n)$ are also plotted as red lines and their slopes are written in each case: from the left to the right, the slope is 2.65, 2.11, and 1.85.

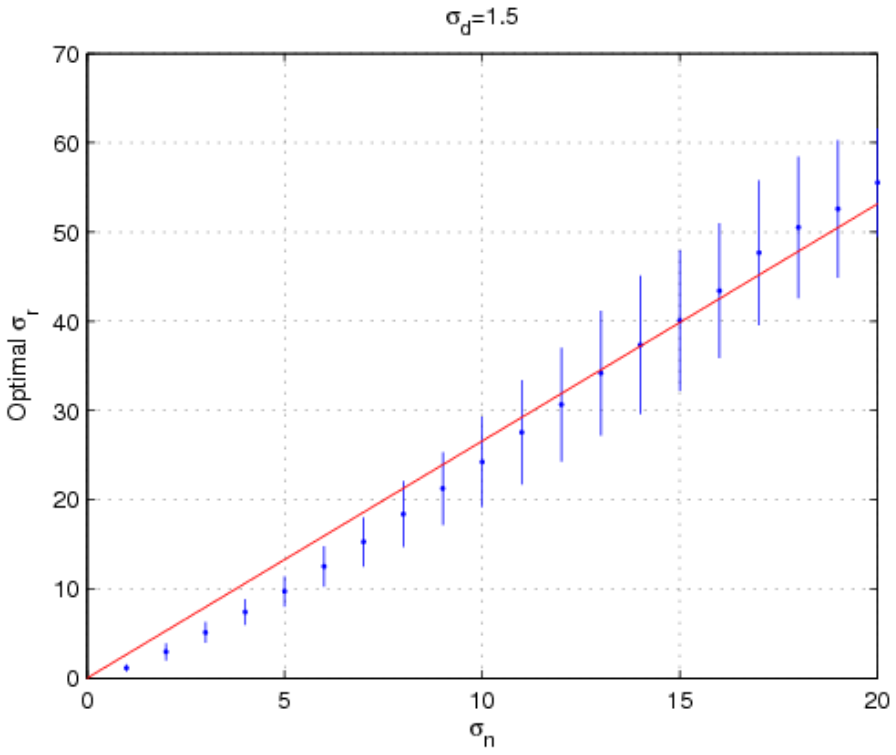



Figure 3.5 The optimal sigma_r vs. sigma_n. sigma_d=1.5. The blue points show the mean value of the optimal sigma_r, while the blue lines show the variance of the optimal sigma_r. The red line shows the averaged estimated slope of the optimal sigma_r vs. sigma_n

As seen in these plots, the $\sigma_r$ and $\sigma_n$ are linearly related to a large degree. The least squares fits to $(\sigma_r / \sigma_n)$ data are also plotted in the figure. Although there is no single value for $(\sigma_r / \sigma_n)$ that is optimal for all images and $\sigma_d$ values, we concluded that a value in the

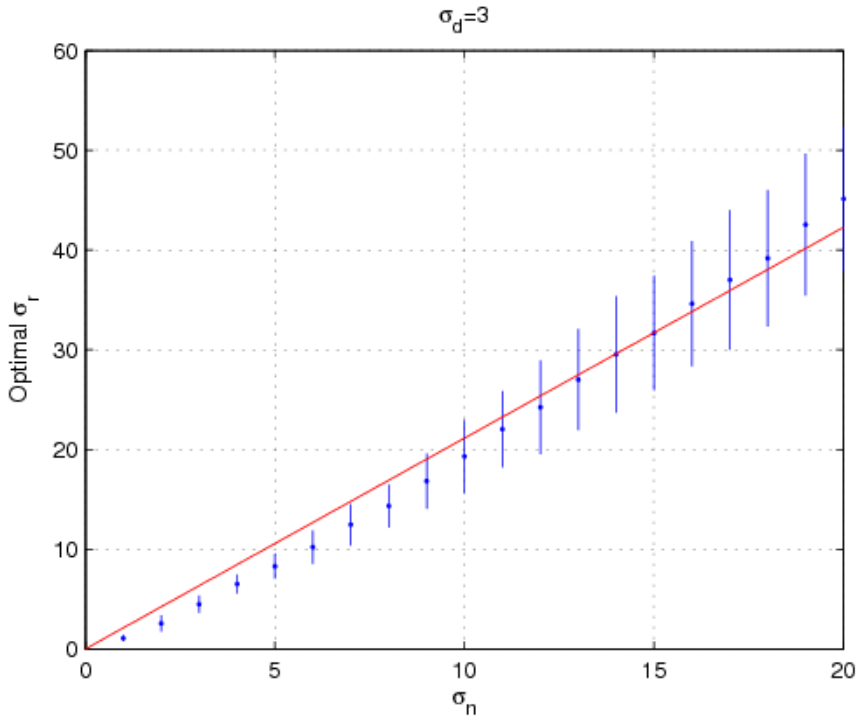



Figure 3.6 The optimal sigma_r vs. sigma_n. sigma_d=3. The blue points show the mean value of the optimal sigma_r, while the blue lines show the variance of the optimal sigma_r. The red line shows the averaged estimated slope of the optimal sigma_r vs. sigma_n

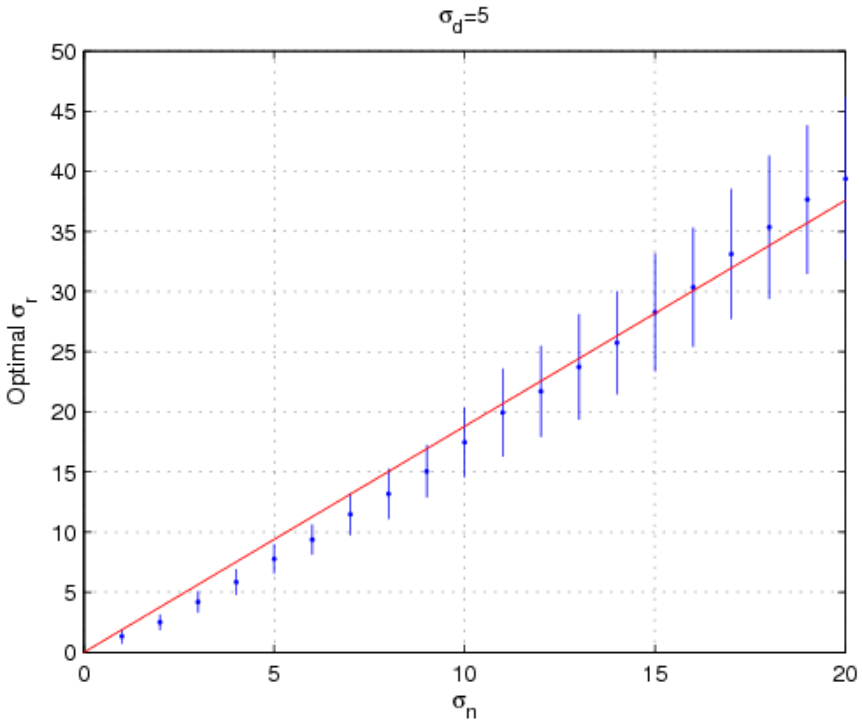



Figure 3.7 The optimal sigma_r vs. sigma_n. sigma_d=5. The blue points show the mean value of the optimal sigma_r, while the blue lines show the variance of the optimal sigma_r. The red line shows the averaged estimated slope of the optimal sigma_r vs. sigma_n

range 2-3 could be a good choice on average. We should note that we cannot expect to find universal optimal values for $\sigma_d$ and $\sigma_r$ as images may have a large variety of texture characteristics. However, these experiments at least tell us some guidelines in selecting these parameters.

### 3.1.2 Multi Resolution Bilateral Filter Framework

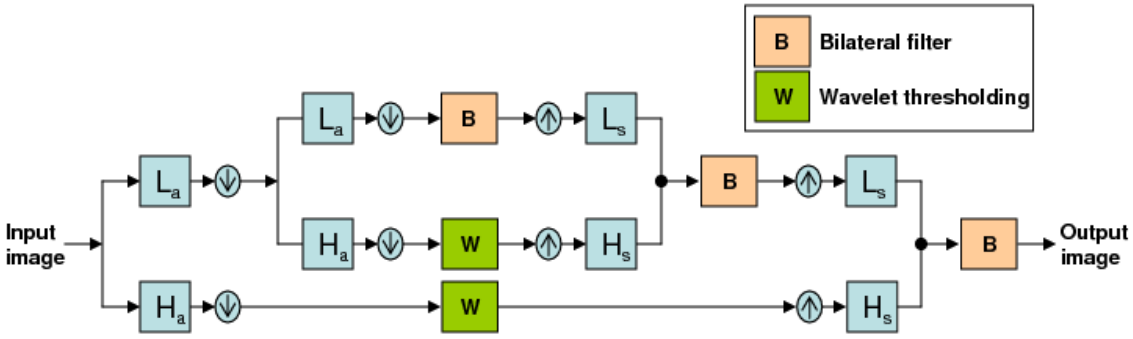



Figure 3.8 Framework of multi resolution bilateral filter.

As we have discussed in previously, image noise is not necessarily white and may have different spatial frequency (fine-grain and coarse-grain) characteristics. Multi resolution analysis has been proven to be an important tool for eliminating noise in signals; it is possible to distinguish between noise and image information better at one resolution level than another. Therefore, we decided to put the bilateral filter in a multi resolution framework: Referring to Figure 3.8, a signal is decomposed into its frequency sub-bands with wavelet decomposition. As the signal is reconstructed back, bilateral filtering is applied to the approximation sub-bands. Unlike the standard single-level bilateral filtering, this multi resolution bilateral filtering has the potential of eliminating low-frequency noise components. (This will become evident in our experiments with real data.) Bilateral filtering works in approximation sub-bands; in addition, it is possible to apply wavelet thresholding to the detail sub-bands,

where some noise components can be identified and removed effectively. This new image denoising framework combines bilateral filtering and wavelet thresholding. The experiment results will be given in the next chapter.

### 3.2 Fast Bilateral Filter

In Porikli's constant time bilateral filter [32], he applied Taylor expansion to the Gaussian spatial filter. Since for constant spatial filter, the response of bilateral filter can be written as the summation of the integral histogram, a bilateral filter can be interpreted as the weighted sum of the spatial filtered responses of the powers of the original image. So he used a box filter to compute the 2D spatial linear filter in constant time O(1) by using an integral image.

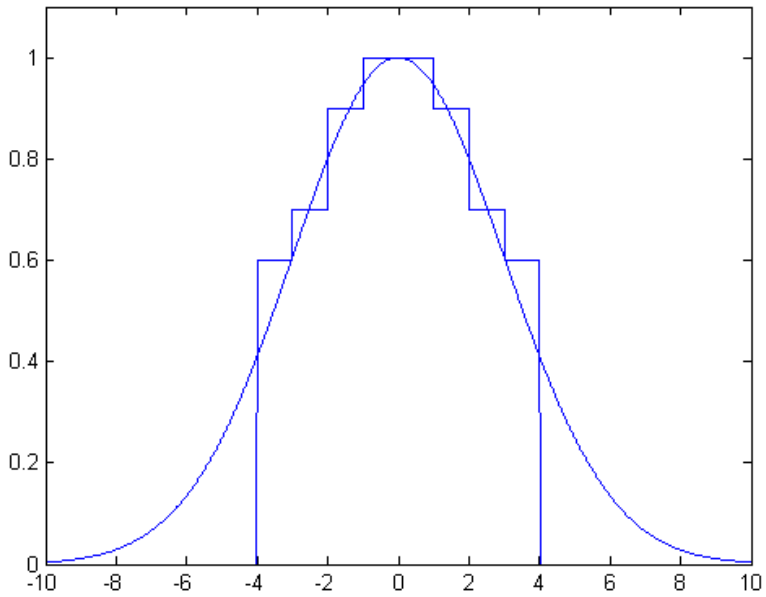



Figure 3.9 Multiple boxes filters and Gaussian filter

Based on the method provided by Porikli [32], we can find that he only use one box filter to approximate the Gaussian filter. So I extend one box filter to multiple box filters which can

be more precise and close to the Gaussian. The weight of each box depends on the area of each box. The summation of the area of every box should be equal to the area of the Gaussian. The multiple boxes filter is shown in Figure 3.9.

## 4. EXPERIMENTAL RESULTS FOR IMAGE DENOISING

I have conducted some experiments to see the performance of the proposed framework quantitatively and visually. To do a quantitative comparison, I simulated noisy images by adding white Gaussian noise with various standard deviations to some standard test images. These noisy images were then denoised using several algorithms and the PSNR results were calculated. For visual comparisons, real noisy images were used.

### 4.1 PSNR Comparison for Gray-Scale Images

For each test image, three noisy versions were created by adding white Gaussian noise with standard deviations 10, 20, and 30. These images were denoised using four methods. The first method is the BayesShrink wavelet thresholding algorithm [10]. Five decomposition levels were used; the noise variance is estimated using the robust median estimator [9]. The second method is the bilateral filter [1]. Based on our experiments discussed in the previous sections, we chose the following parameters for the bilateral filter: $\sigma_d = 1.8, \sigma_r = 2 \times \sigma_n$, and the window size is $11 \times 11$. The third method is the sequential application of BayesShrink[10] and bilateral filter[1]. The reason this method was included is to see the combined effect of BayesShrink[10] and bilateral filter[1] and compare it with the proposed method. The fourth method is the proposed method. For the proposed method, DB8 filters in Matlab were used for one-level decomposition.

For the bilateral filtering part of the proposed method, we set the parameters as follows: $\sigma_d = 1.8$, the window size is $11 \times 11$, and $\sigma_r = 1.0 \times \sigma_n$ at each level. In case of the original bilateral filter, $\sigma_r = 2 \times \sigma_n$ was a better choice. However, for the proposed method this lead to a smaller PSNR value on average. The reason is the double application of the bilateral filter

Table 4.1 PSNR comparison among different methods under different noisy condition. The numbers are obtained by averaging the results of six runs.

| Image | $\sigma_n$ | Bayes[10] | Bilateral[1] | [1]after[14] | OWT[14] | Proposed |
|---|---|---|---|---|---|---|
| Barbara | 10 | 31.25 | 31.37 | 30.92 | 32.18 | 31.79 |
| 512×512 | 20 | 27.32 | 27.02 | 27.16 | 27.98 | 27.74 |
| | 30 | 25.34 | 24.69 | 25.23 | 25.83 | 25.61 |
| Boats | 10 | 31.94 | 32.08 | 31.93 | 32.69 | 32.48 |
| 512×512 | 20 | 28.69 | 28.90 | 28.80 | 29.52 | 29.50 |
| | 30 | 27.13 | 27.50 | 27.34 | 27.89 | 27.77 |
| Goldhill | 10 | 31.94 | 32.08 | 31.93 | 32.69 | 32.48 |
| 512×512 | 20 | 28.69 | 28.90 | 28.80 | 29.52 | 29.50 |
| | 30 | 27.13 | 27.50 | 27.34 | 27.89 | 27.77 |
| Peppers | 10 | 31.49 | 32.98 | 31.89 | 33.18 | 33.45 |
| 256×256 | 20 | 27.85 | 29.07 | 28.01 | 29.33 | 30.20 |
| | 30 | 25.73 | 27.02 | 26.07 | 27.13 | 28.18 |
| House | 10 | 33.07 | 33.77 | 33.09 | 34.29 | 34.62 |
| 256×256 | 20 | 29.83 | 29.63 | 29.79 | 30.93 | 31.37 |
| | 30 | 27.12 | 28.11 | 28.10 | 28.98 | 29.24 |
| Lena | 10 | 33.38 | 33.65 | 33.39 | 34.45 | 34.48 |
| 512×512 | 20 | 30.27 | 30.33 | 30.29 | 31.33 | 31.28 |
| | 30 | 28.60 | 28.54 | 28.62 | 29.55 | 29.33 |
| Average | | 29.24 | 29.54 | 29.31 | 30.29 | 30.34 |

in the proposed method. When $\sigma_r$ was large, texture in the image was smoothed to produce low PSNR values. After some experimentation, $\sigma_r = 1.0 \times \sigma_n$ turned out to be better in terms of PSNR values. Here, we should note that a higher PSNR does not necessarily correspond to a better visual quality. We will discuss this shortly. For the wavelet thresholding part of the proposed method, the BayesShrink method [10] was used; and the noise variance was estimated again with the robust median estimator technique. To eliminate the border effects, images were mirror-extended before the application of the bilateral filter and cropped to the original size at the end.

Computation time for the multi resolution bilateral filter is only 5% more than the original bilateral filter when it comes to one level wavelet decomposition. Because the wavelet thresholding is a really fast algorithm, it would not cost too much time to carry on the proposed method.

The PSNR results are given in Table 4.1. As seen, the proposed method is 0.8dB better than the original bilateral filter and 1.1dB better than the BayesShrink method on average. The sequential application of BayesShrink[10] and bilateral filter[1] is only slightly better than BayesShrink and worse than bilateral filter. Therefore, we conclude that the improvement of the proposed method is not due to the combined effect of BayesShrink and bilateral filter, but due to the multi resolution application of the bilateral filter.

In order to achieve the best performance for the real random noise, such as the fixed pattern noise induced by the high ISO speed, we test the proposed method on the standard test images blurred by the spatially varying random noise. The PSNR results compared with [1] is shown in Table 4.2, which prove that the application of our method can be effective in

the real noise denoising. Here we should notice that, since the noise is spatially varying, the automatic parameter estimator derived before cannot be used here. However, based on the experimental results, we still have to realize that the $\sigma_r$ influences the performance most and $\sigma_d$ should be a properly small value.

Table 4.2 PSNR comparison among various methods on real noisy images. The numbers are obtained by averaging the results of six runs.

| Image | BayesShrink[10] | Bilateral[1] | OWT[14] | Proposed |
|---|---|---|---|---|
| Barbara 512×512 | 31.96 | 32.12 | 32.26 | 32.63 |
| Lena 512×512 | 32.94 | 33.17 | 32.87 | 34.01 |
| Goldhill 512×512 | 32.06 | 32.40 | 32.28 | 32.76 |
| Boat 512×512 | 32.05 | 32.41 | 32.37 | 32.93 |
| House 512×512 | 32.45 | 34.05 | 32.39 | 34.52 |
| Peppers 512×512 | 31.76 | 33.42 | 32.2 | 33.58 |
| Average | 32.20 | 32.93 | 32.40 | 33.41 |

**4.2 Visual Comparison for Real Noisy Images**

PSNR comparisons for image denoising tell only a part of the story: First, it is well known that the PSNR is not a very good measure of visual quality; second, the white Gaussian noise assumption is not always accurate for real images. As a result, experiments with real data and visual inspections are necessary to evaluate the real performance of image denoising algorithms. In case of gray-scale images, we add the spatially varying random noise on the standard test image 'lena' ( $512\times512$ ). The results are shown in Figure 4.1, which compares the proposed method with the bilateral filter [1] and the BayesShrink method [10].

In case of color images, there is also the issue of what color space to use. To achieve good PSNR performance, the RGB space could be a good choice; however, for visual performance, it is a better idea to perform denoising in the perceptually uniform CIE-L*a*b* color space. Because the human visual system is more sensitive to color noise compared to luminance noise, stronger noise filtering could be applied to the color channels a and b compared to the luminance channel L. We test the image using the proposed method in both of the RGB channels and L*a*b* channels in Figure 4.2. From the results we can clearly find that the denoising in RGB channels still preserve quite perceptible color noises. One thing that should also be observed is not to over-smooth the luminance channel to avoid unnatural ``plastic" looking images. If we apply the multi resolution filter on L*a*b* together, it can eliminate the color noises but also over-blur the luminance channel that lost the details. Humans are better at detecting differences in luminance levels as opposed to color levels. Therefore, in our experiments, the proposed method and the standard bilateral filter were applied to each channel separately in the CIE-L*a*b* space for color images.

In Figure 4.2, we compare the standard bilateral filter and the proposed method. The bilateral filter was applied for various values of $\sigma_d$ and $\sigma_r$. As seen in the figure, the chroma noise was not eliminated effectively in any case. (We have also tested the iterative application of the bilateral filter. The results were not good either, and were not included in the figure.) Two results obtained by the proposed method are given: In the first case, the number of decomposition levels for the luminance channel is one; and in the second case it is two. In both cases, the number of decomposition levels for the chrominance channels is four. The resulting images in both cases are free of chroma noise to a great extent. Increasing the

number of decomposition levels for the luminance channel produces a smoother image as seen in the second case.

In Figures 4.3 and 4.4, results of the BLS-GSM method [13], the bilateral filter [1], and the proposed method are presented for real images provided at the website of the first author of [13]. The BLS-GSM method is considered as one of best denoising algorithms in terms of the PSNR results. However, the proposed method is apparently producing more visually pleasing results than the BLS-GSM method in case of real data. In Figure 4.3, neither the BLS-GSM method nor the bilateral filter was able to eliminate the chroma noise. In Figure 4.4, noise was not completely eliminated by the BLS-GSM method. The result of the bilateral filter is less noisy but overly smoothed. The result of the proposed method can be considered as the best visual one among three.

Also, note that in all these real image experiments, $\sigma_n$ values were estimated from the data, and the same $\sigma_d$ and $(\sigma_r / \sigma_n)$ values were used for the proposed method. That is, once the parameters were decided, there was no need to re-adjust them for another image.

**4.3 CIE-L*a*b* Distance Comparison for Color Images**

It is well known that the PSNR is not a good representative of visual quality. Other quality measures have been proposed to evaluate the performance of image restoration algorithms. In case of color images, Euclidean distance in the perceptually uniform CIE-L*a*b* color space gives a better sense of visual quality than the PSNR.

One of the most widely used perceptual color fidelity metric is the S-CIELAB, given as part of the CIE L*a*b* standard color space specification [33]. To measure perceptual difference between two lights using this metric, the spectral power distribution of the two


Figure 4.1 a) Input image with a PSNR 30.02, (b) The BayesShrink method[10], (c) The bilateral filter [1] with sigma_d=1.8 and sigma_r=5sigma_n, (d)The proposed method with the number of decomposition levels is 1. For the proposed method, sigma_d=1.8 and sigma_r=3sigma_n at each level. The wavelet filters are db8 in MATLAB.


Figure 4.2 Real noisy image comparisons. (a) Input image, (b) The bilateral filter [1] with sigma_d=1.8 and sigma_r=3sigma_n, (c) The bilateral filter with sigma_d=1.8 and sigma_r=20sigma_n, (d) The bilateral filter with sigma_d=5 and sigma_r=20sigma_n, (e) The proposed method with the decomposition levels is (1,4,4) for the (L,a,b) channels, respectively, (f) The proposed method with the decomposition levels (2,4,4). For the proposed method, sigma_d=1.8 and sigma_r=3sigma_n at each level. The wavelet filters are DB8 in MATLAB.

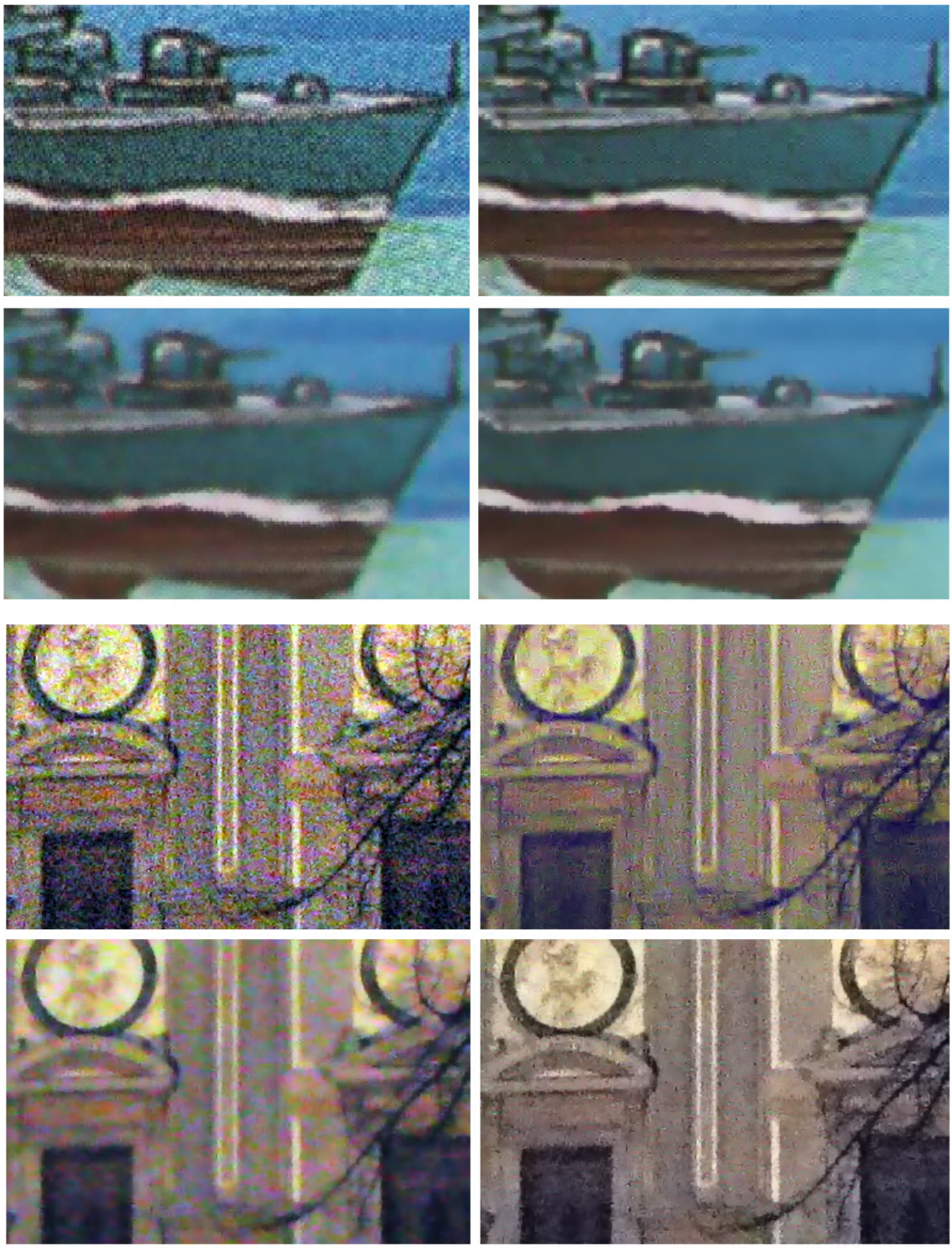


Figure 4.3 Up-left: Input image, Up-right: the BLS-GSM result [13], Bottom-left: bilateral filter [1] result, Bottom-right: Result of the proposed method. For the bilateral filter, sigma_d=1.8 and sigma_r=10sigma_n and the window size is $11\times11$. For the proposed method, sigma_d=1.8 and sigma_r=3sigma_n at each level, the window size is $11\times11$, and the number of decomposition levels is (1,4,4) for the (L,a,b) channels, respectively.

lights are first converted to XYZ representations, which reflect (within a linear transformation) the spectral power sensitivities of the three cones on the human retina. Then, the XYZ values are transformed into an L*a*b* space, in which equal distance is supposed to correspond to equal perceptual difference (a "perceptually uniform" space). Then, the perceptual difference between the two targets can be calculated by taking the Euclidean distance of the two in this L*a*b* space. The difference is expressed in "Delta E" units. One Delta E unit represents approximately the threshold detection level of the color difference. The larger S-CIELAB metric is, the lower quality of the restored color image is.

We compare the proposed algorithm with the original bilateral filter [1] and two wavelet thresholding algorithms: the BayesShrink [10] and the SURE Shrink [9]. In the first experiment, we added white Gaussian noise with variance 25 to the standard color images Baboon, Peppers, Boat and Goldhill. The denoising algorithms are applied to each color channel separately. The performance of the algorithms are measured using the Euclidean distance in the perceptually uniform CIEL*a*b* space [33]. The results are shown in Table 4.3. From Table 4.3, we can find that proposed multi-level bilateral filter has the least color difference which means that it is most suitable for the human visual system.

Table 4.3 Comparison of several methods in terms of the Euclidean distance in the CIEL*a*b* space [33]. For the proposed method, the images are decomposed by one level using the DB4 filters of MATLAB.

| Image | BayesShrink | SURE Shrink | Bilateral | Proposed |
|---|---|---|---|---|
| Baboon 512*512 | 11.9730 | 13.2110 | 7.9895 | 7.6034 |
| Peppers 512*512 | 8.4389 | 9.1117 | 5.1905 | 2.6680 |
| Boat 512*512 | 7.4325 | 7.8173 | 6.7754 | 3.5880 |
| Goldhill 512*512 | 8.6210 | 9.4387 | 7.608 | 5.1807 |

### 4.4 PSNR of Fast Bilateral Filter

In this section, I have conducted some experiments to see the performance of the proposed multiple box fast bilateral filter. To do a quantitative comparison, I simulated the standard test images with different quantization levels. The PSNR results were calculated.

I computed the PSNR for the fast bilateral filter to the original filter. Figure 4.4 shows the test results with different $\sigma_d$. I apply these fast bilateral filters on a standard test image "Barbara.jpg" blurred by the white random Gaussian noise with the standard deviation of the input noise as 30. For the original bilateral filter, I use same $\sigma_r$ and $\sigma_d$ as the fast bilateral filter.

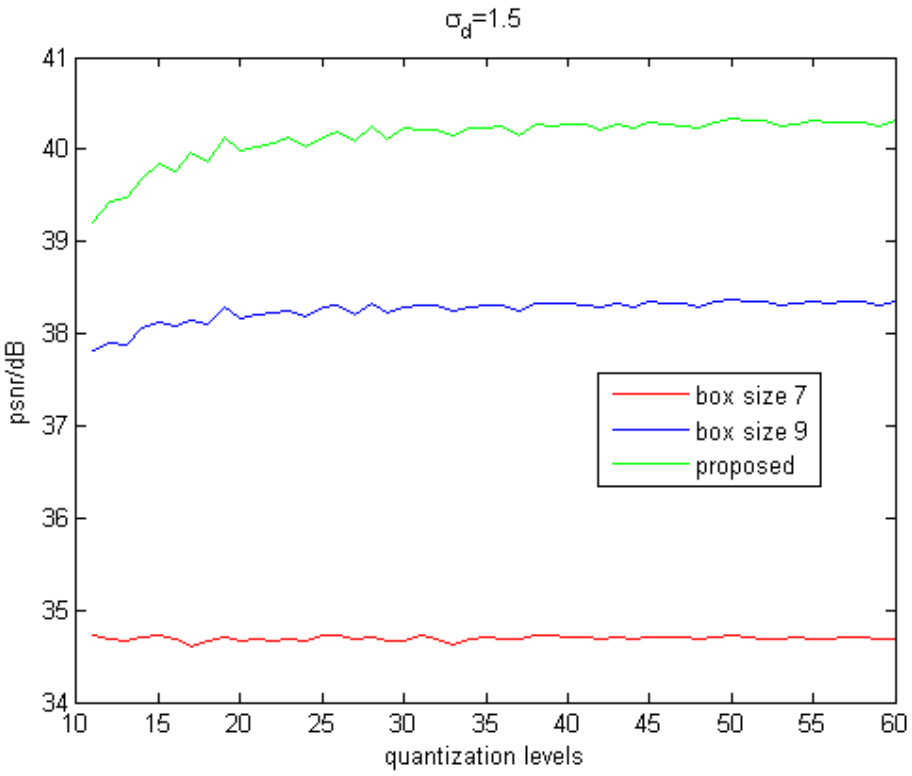



Figure 4.4 PSNR comparisons between single box fast bilateral [32] and proposed multi box fast bilateral filters. sigma_d=1.5, sigma_r=20, and the standard deviation of the simulation noise is sigma_n=30. The green line is the proposed method, while the red and blue are Porikli's method [32].

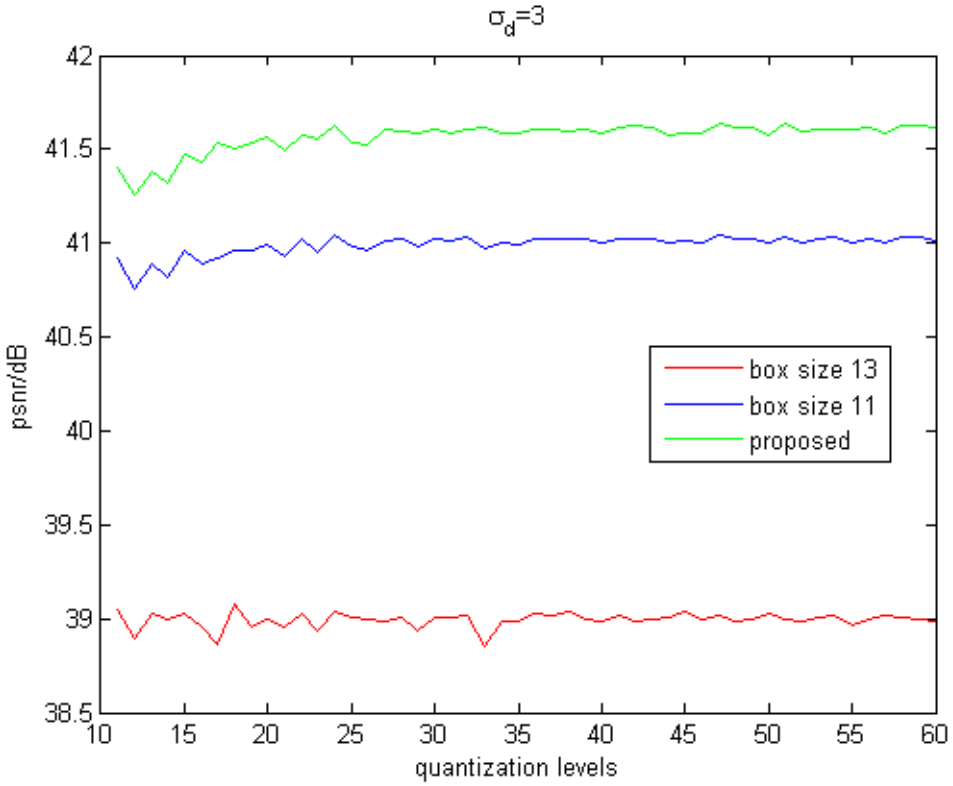



Figure 4.5 PSNR comparisons between single box fast bilateral [32] and proposed multi box fast bilateral filters. sigma_d=3 and sigma_r=20, and the standard deviation of the simulation noise is sigma_n=30. The green line is the proposed method, while the red and blue are Porikli's method [32].

Given the same parameters of original bilateral filter, the proposed multiple boxes fast bilateral filter can improve 0.5dB-1dB of PSNR than the single box fast bilateral filter. The computation time only increase 20%, which is still greatly smaller than the original bilateral filter. It proves that the proposed method has the better approximation of the original bilateral filter and would not cost much more computation time.

# 5. BLOCKING ARTIFACTS REDUCTION

## 5.1 Parameter Analysis

There are two parameters that control the behavior of the bilateral filter. Referring to (1.1), $\sigma_d$ and $\sigma_r$ characterizes the spatial and intensity domain behaviors, respectively. In case of compression artifact reduction, these parameters should be chosen carefully.

Figure 5.1 illustrates this on a one dimensional signal. The first subplot in that figure shows an edge signal; the edge discontinuity is 10. The second subplot displays the outputs of the bilateral filter for different values of $\sigma_r$. When the $\sigma_r$ value is less than the discontinuity amount, the filter is basically useless against eliminating the discontinuity. When $\sigma_r$ is larger than the discontinuity amount, the discontinuity can be eliminated. At the same time, the extent of the smoothing can be controlled by the $\sigma_d$ value. The larger the $\sigma_d$ value is, the wider the extent of smoothing can be. On the other hand, if $\sigma_r$ value is less than the discontinuity amount, elimination of the discontinuity is impossible no matter the value of $\sigma_d$. Therefore, we need to measure the discontinuity amount along the block boundaries and adapt the value of $\sigma_r$ accordingly. We also would like to avoid over-smoothing texture regions by adapting the $\sigma_d$ value on the texture regions. For a smooth region, the value of the $\sigma_d$ can be large; otherwise, it should be small.

As we will show shortly, the non-adaptive application of bilateral filter creates some problems: if strong parameters are chosen to eliminate blockiness, it over-blurs the texture details; if weaker parameters are chosen, the blocking artifacts are not completely removed. To address these issues, we present an adaptive bilateral filtering framework, whose block diagram is given in Figure 5.2.

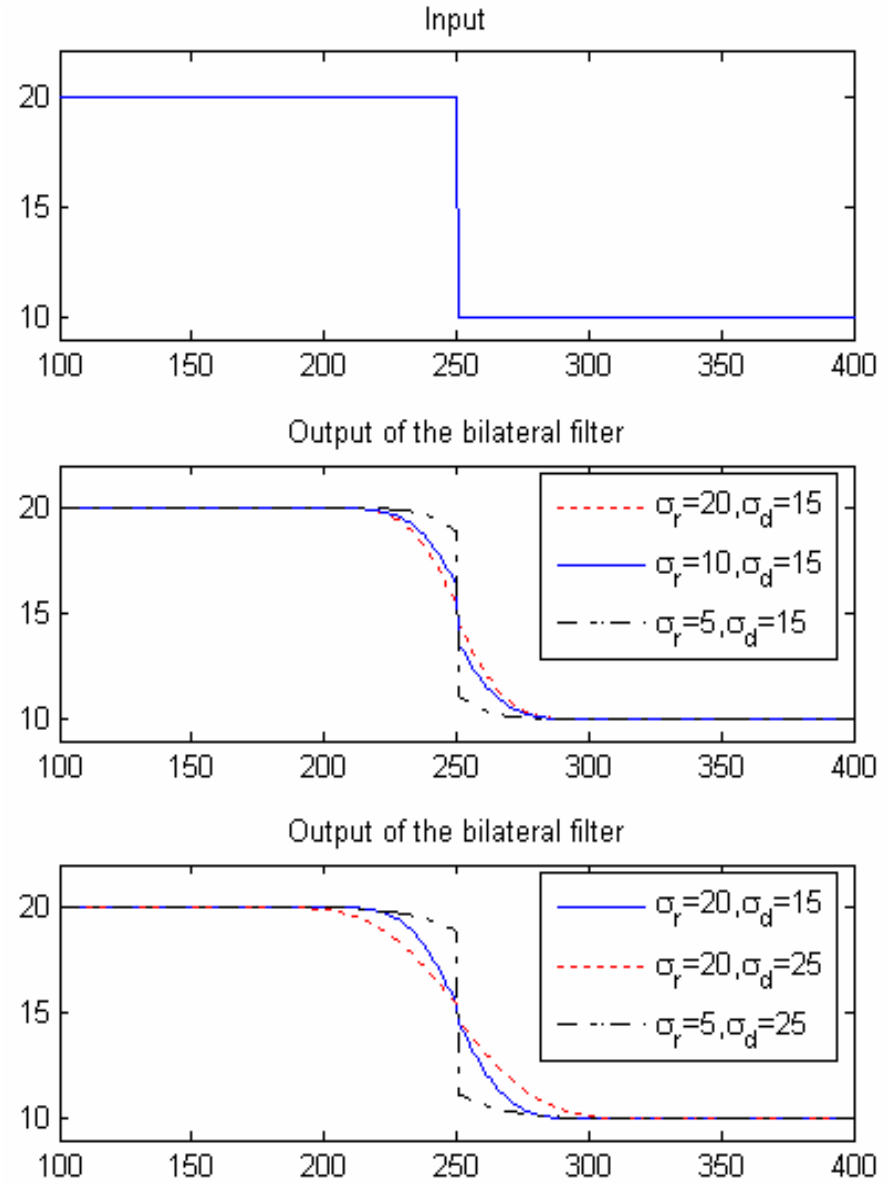



Figure 5.1 Effects of the values of the bilateral filter parameters sigma_r and sigma_d on a block discontinuity are illustrated. Input is a step signal with the step value=10. The middle image shows the performance of the bilateral filter using same sigma_d value and different sigma_r. The bottom image shows the performance of the bilateral filter using same sigma_r value and different sigma_d.

In the light of discussion of the previous section, we included two modules in the framework. One module detects the block discontinuities and adjusts the value of $\sigma_r$ accordingly; the other module detects smoothness of local regions and adjusts the value of $\sigma_d$ accordingly.

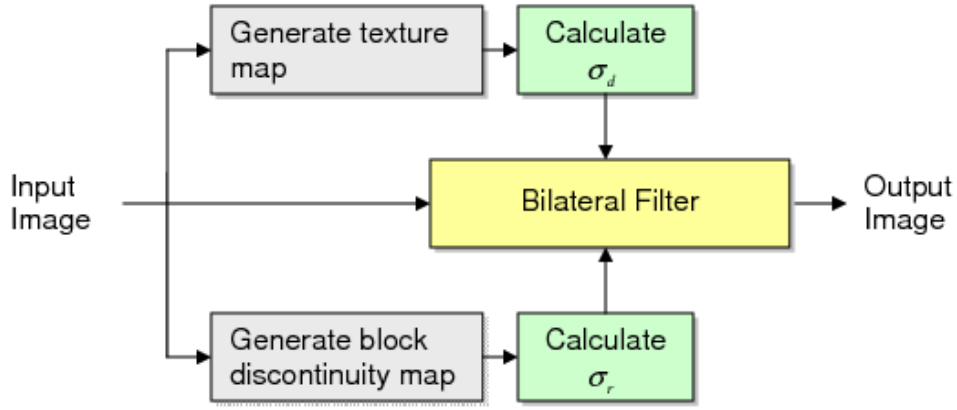



Figure 5.2 Framework of adaptive bilateral filter

## 5.2 Parameter Selection

To detect block discontinuities, the input image is filtered with $[-1,0,1]$ (for vertical boundaries) and with $[-1,0,1]^T$ (for horizontal boundaries) along the block boundaries, and then absolute values of the results are taken. The $\sigma_r$ value should be at least equal to these values to be effective. The discontinuities are detected along the block boundaries; however, if the bilateral filter is applied along the boundaries only, the blockiness cannot be eliminated. Consider a single block; if the bilateral filter is applied along the boundaries only, the discontinuity moves further inside the block. To eliminate the blockiness effectively, the bilateral filter should be applied to the entire block. Thus, the discontinuities along the boundaries should be diffused into the blocks. One approach is as follows: Given the input image with blocking artifacts as Figure 5.3, referring Figure 5.4, the center four pixels inside a block is set to zero; the horizontal and vertical discontinuities along the boundaries are kept except for the corner pixels, where the larger of the horizontal/vertical discontinuities is chosen; and then the rest of the pixels are interpolated linearly. This is repeated for all blocks to obtain the block discontinuity map $M_b(\mathbf{x})$. The block discontinuity map $M_b(\mathbf{x})$ for the

JPEG compressed “Lena”" image is shown Figure 5.5. Once $M_b(\mathbf{x})$ is calculated, the adaptive $\sigma_r(\mathbf{x})$ is calculated as

$$\sigma_r(\mathbf{x}) = max(\sigma_{r,min}, k_0 M_b(\mathbf{x})) \qquad (5.1)$$

Figure 5.3 Compressed input image Lena. The compressed bit-rate is 0.18 which means quantization quality=8 in the Matlab.

Where $\sigma_{r,min}$ is the minimum value of $\sigma_r(\mathbf{x})$, and $k_0$ is a scale factor. The reason we use such a minimum value is that we would like to apply a minimal filtering to the entire image; if this was not done, other compression artifacts, such as the mosquito artifact, could not be eliminated and some sort of spatial unevenness appear in the final image. Therefore, we need a controller to balance the power of the intensity filter of the standard bilateral filter. Only if the estimated intensity parameter is large enough, the $\sigma_r$ is considered as the parameter.

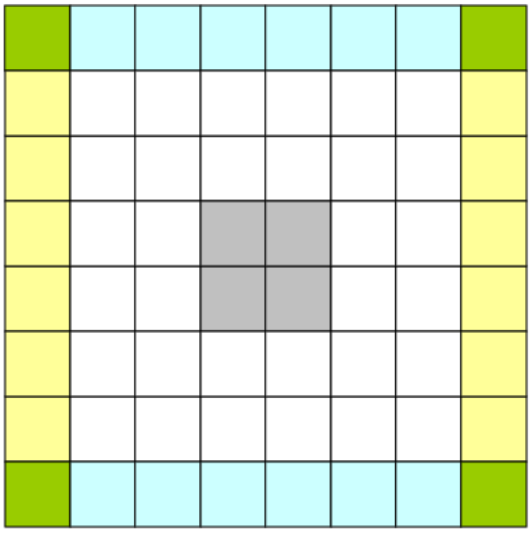


Figure 5.4 Interpolation of the block discontinuities at each block.

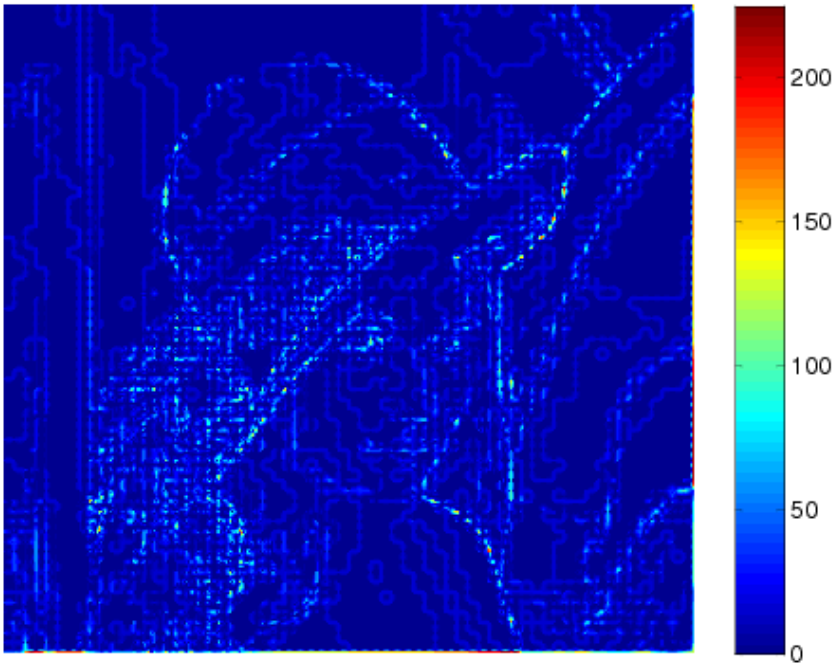



Figure 5.5 Block discontinuity map

To detect high-frequency texture regions, we compute the standard deviation of each block. The standard deviation is used as an indicator of texture and to adapt value of $\sigma_d$ to preserve the texture information. Figure 5.6 shows the standard deviation of each block for the compressed ``Lena'' image. Note that the edge regions

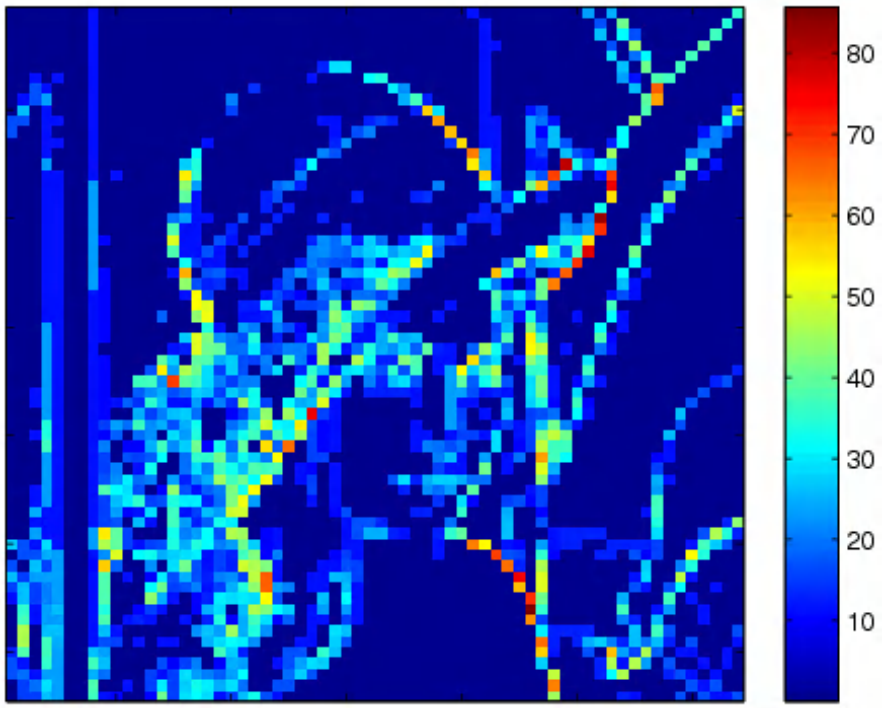



Figure 5.6 Local standard deviation for texture detection

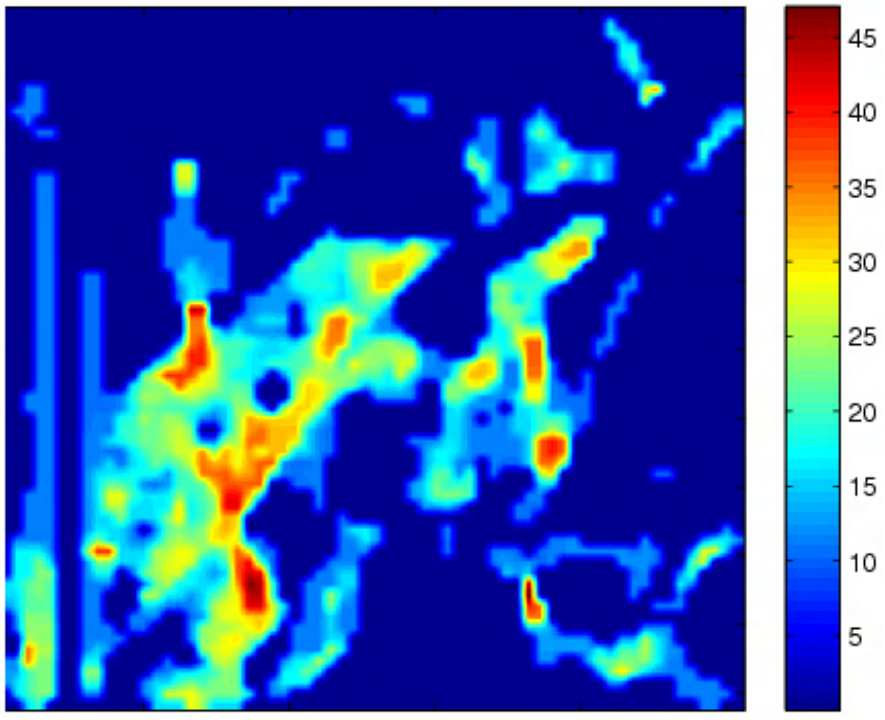



Figure 5.7 Median filtered local standard deviation

are highlighted in addition to the texture regions. However, we would like to apply strong bilateral to edge regions as well to eliminate ringing type of artifacts. One solution is to apply

a median filter to eliminate edge regions from the texture map. In our experiments, the standard deviation of each $8\times8$ block is calculated, a $3\times3$ median filter is applied, and then the resulting image is interpolated to obtain the texture map $M_t(\mathbf{x})$. (For the Lena image of Figure 5.3, the texture map is shown in Figure 5.7. The value of $\sigma_d$ should be inversely proportional to $M_t(\mathbf{x})$. One way of calculating $\sigma_d(\mathbf{x})$ is

$$\sigma_d(\mathbf{x}) = max(\sigma_{d,min}, \frac{k_1}{1+M_t(\mathbf{x})}), \qquad (5.2)$$

where $k_1$ is a constant parameter controlling the mapping from $M_t(\mathbf{x})$ to $\sigma_d(\mathbf{x})$, and $\sigma_{d,min}$ is minimum value of $\sigma_d(\mathbf{x})$. Such a minimum $\sigma_{d,min}$ is introduced again to do a minimum level of filtering to the entire image.

**5.3 Summary**

The adaptive parameter selection here is simply separating the filtering job into two categories: one for eliminating boundaries of the blocks, another for recovering the texture and details of the original image.

Using the framework provided before, we can easily get the parameters of bilateral filter adaptively. All the parameters are tested in the next chapter to show the performance of this framework.

# 6 EXPERIMENTAL RESULTS FOR COMPRESSION ARTIFACTS REDUCTION

In this section, I have conducted some experiments to see the performance of the proposed framework quantitatively and visually. To do a quantitative comparison, I simulated some standard test images under different compression rates. These compressed images were then deblocked using several algorithms which introduced in the previous literature review section. The PSNR and MSDSt results were calculated. For visual comparisons, standard images and snapshot from standard test videos are provided.

There have several measurements for the blocking artifacts. The most popular two are PSNR and MSDS (Mean Squared Difference of Slope) [34]. MSDS involves the intensity gradient (slope) of the pixels close to the boundary of two blocks. It is based on the empirical observation that quantization of the DCT coefficients of two neighboring blocks increases the MSDS between the neighboring pixels on their boundaries. Consider an $8\times8$ block of the input image and four blocks w, s, e, n horizontally adjacent to f. The MSDS is defined by

$$\epsilon_w = \sum_{m=0}^{7} (d_1(m) - d_2(m))^2 \tag{6.1}$$

$d_1(m)$ is the intensity slope across the boundary between the f and w blocks, defined by

$$d_1(m) = f(m,0) - w(m,7) \tag{6.2}$$

$d_2(m)$ is the average between the intensity slope of f and w blocks close to their boundaries, defined by

$$d_2(m) = \frac{w(m,7) - w(m,6)}{2} + \frac{f(m,1) - f(m,0)}{2} \tag{6.3}$$

Then, the MSDS which involves both horizontal and vertical adjacent blocks is given by

$$MSDS_1 = \epsilon_w + \epsilon_s + \epsilon_e + \epsilon_n \tag{6.4}$$

MSDSt is proposed in [35], which extends the definition of MSDS by involving the four

diagonally adjacent blocks. If nw is a block diagonally adjacent to f, then define

$$\epsilon_{nw} = (g_1(m) - g_2(m))^2, \tag{6.5}$$

$$g_1(m) = f(0,0) - w(7,7) \tag{6.6}$$

and

$$g_2(m) = \frac{nw(7,7) - nw(6,6)}{2} + \frac{f(1,1) - f(0,0)}{2} \tag{6.7}$$

If ne, ns, nw and nn are the four blocks diagonally adjacent to f; the MSDS involving only the diagonally adjacent blocks is

$$MSDS_2 = \epsilon_{nw} + \epsilon_{ns} + \epsilon_{ne} + \epsilon_{nn} \tag{6.8}$$

The total MSDSt considered for the intensity slopes of all the adjacent blocks is

$$MSDS_t = MSDS_1 + MSDS_2 \tag{6.9}$$

From the formulas above, we can find that the MSDSt is typically designed for the measurement of blocking artifacts. The large MSDSt means the restored image still has more blocking artifacts remained compared to the original test image.

We test this proposed adaptive bilateral filter for blocking artifacts reduction for some standard images, such as "Lena", "Cameraman", "Boat", "Airplane", "Mandrill", "Peppers" and "Goldhill". We compare our results in PSNR, with JPEG Coded, H.263, MPEG-4, POCS[25], Post-DCT Method in [26], blind DCT measurement Method in [27] and original bilateral filter[1]. We also compare our method in MSDSt with JPEG Coded, TSD-MRF [28] which is considered as the best in MSDSt among the previous methods and original bilateral filter. The parameters in Table 6.1 are $\sigma_r$=20, $\sigma_d$=3, window size is 6, $k_0 = 1.0$, $k_1 = 10.0$, Table 6.1 and Table 6.2 show the results with the parameters cited in the caption. From Table 4.4 and Table 4.5, we notice that the bilateral filter and the proposed method has the best

performance in PSNR, while the proposed method performs best in MSDSt.

Table 6.1 Comparison of the proposed method in PSNR with JPEG Coded, H.263, MPEG4, POCS[25], Method in[26], Method in [27] and original bilateral filter.

| Image | Bit-rate | JPEG | H.263 | MPEG4 | POCS | [25] | [26] | Bilateral | Proposed |
|---|---|---|---|---|---|---|---|---|---|
| Lena | 0.22 | 29.47 | 30.20 | 30.02 | 30.23 | 30.32 | 30.27 | 30.37 | 30.49 |
| Peppers | 0.22 | 29.21 | 30.02 | 30.04 | 29.85 | 29.95 | 29.93 | 30.59 | 30.59 |
| Goldhill | 0.23 | 27.90 | 28.50 | 28.31 | 28.46 | 28.51 | 28.40 | 28.38 | 28.53 |
| Mandrill | 0.30 | 22.05 | 22.35 | 22.15 | 22.44 | 22.49 | 22.39 | 22.46 | 22.26 |
| Airplane | 0.24 | 28.72 | 29.34 | 29.32 | 29.34 | 29.39 | 29.33 | 29.87 | 29.81 |
| Average | | 23.38 | 26.68 | 27.96 | 28.06 | 28.13 | 28.06 | 28.33 | 28.34 |

We also test the parameters in our proposed method. Figure 6.1 shows how $k_0$, which controls $\sigma_r$ along the blocking boundaries according to (5.1), influences the PSNR and MSDSt. Obviously, when the $k_0$ increases, the PSNR will be undermined while the MSDSt can be improved. Therefore, we have to choose a proper $k_0$ to make the balance between the PSNR and MSDSt. Figure 6.2 shows that under different bit-rate, using the proposed adaptive $\sigma_d$ for the texture detected. Both of the PSNR and MSDSt can be improved a lot. In the (5.2) we use the $k_1$ to control the intensity of the $\sigma_d$ along the blocking boundary. Figure 6.3 illustrates the relationship between $k_1$ and PSNR, MSDSt, which implies only the moderate $k_1$ can make both metric best.

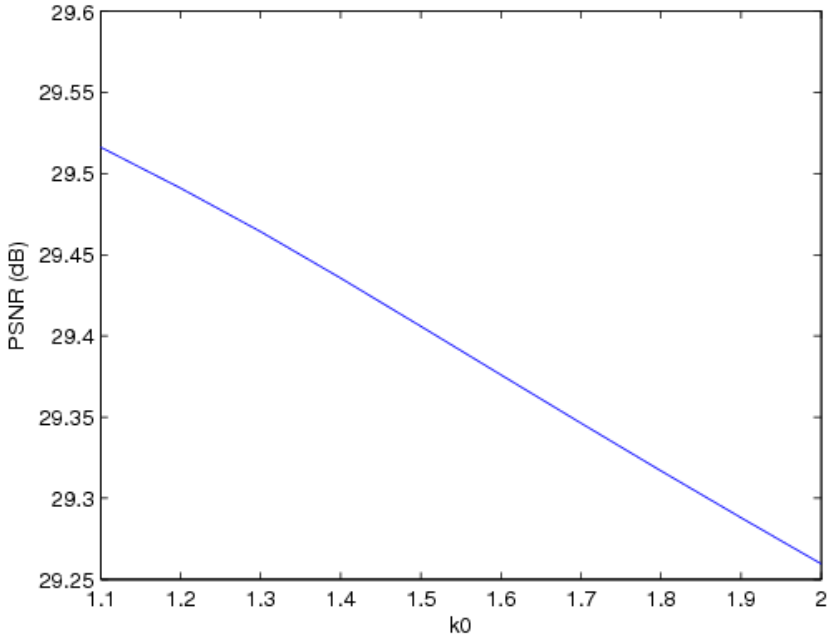
In order to differentiate the performance of bilateral filter and proposed method, we plot the comparison of bilateral filter using different parameter $\sigma_r$ with the proposed method.

Table 6.2 Comparison of the proposed method in MSDSt with JPEG Coded, TSD-MRF[28] and original bilateral filter.

| Image | Bit-rate | JPEG | TSD-MRF | Bilateral | Proposed |
|---|---|---|---|---|---|
| Lena | 0.20 | 6674 | 2229 | 3841 | 526 |
| | 0.30 | 4384 | 1641 | 2365 | 374 |
| Boat | 0.20 | 10947 | 3844 | 5744 | 879 |
| | 0.30 | 8695 | 3969 | 4095 | 747 |
| Cameraman | 0.20 | 5164 | 2904 | 4512 | 438 |
| | 0.30 | 4276 | 2554 | 2616 | 351 |
| Peppers | 0.20 | 6341 | 2212 | 5622 | 2157 |
| | 0.30 | 3524 | 1322 | 4525 | 2193 |
| Average | | 6250 | 2584 | 4163 | 958 |

From Figure 6.4, we can find that when $\sigma_r$ increases, the PSNR of the bilateral filter become worse, although the MSDSt decrease. It proves that intensity parameter of bilateral filter can control the blocking reduction as well as blur the details. Only the proposed method can reduce the blocking artifacts as well as saving the details with the best PSNR and MSDSt.

Visually, we can find the same conclusion from in Figure 6.5. It is captured from a region of the standard test image Lenna. From left to right: (a) The standard bilateral filter with $\sigma_r = 20, \sigma_d = 3$; (b) The standard bilateral filter with $\sigma_r = 50, \sigma_d = 3$; (c) The adaptive bilateral filter with $\sigma_{r,min} = 20$, $\sigma_{d,min} = 3$. Obviously, when using the large intensity parameter, the image will be blurred too much, but the blocks can be eliminated very well.



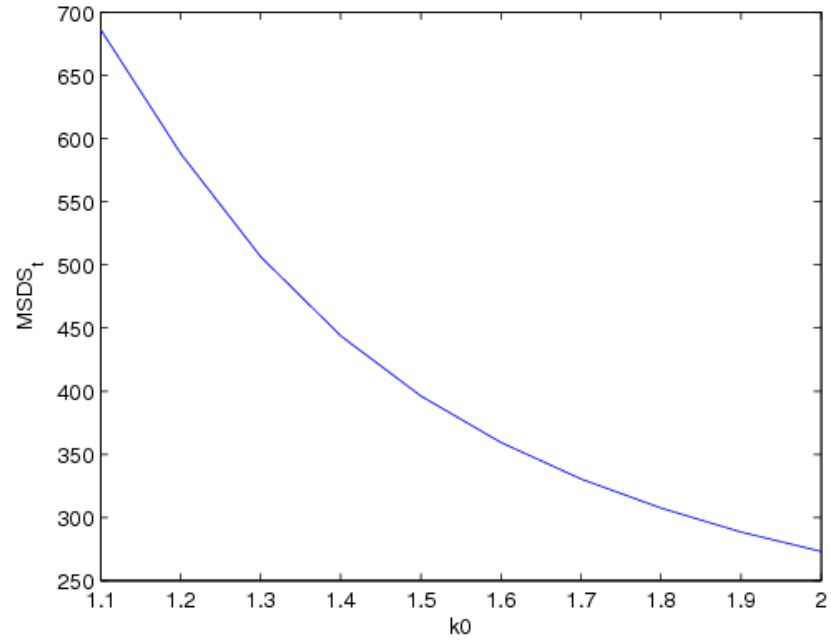




Figure 6.1 Relationship between the k0 and the PSNR, MSDSt for the Lena test image under bit-rate=0.18. The image size is 512 by 512. The k1=1.5.

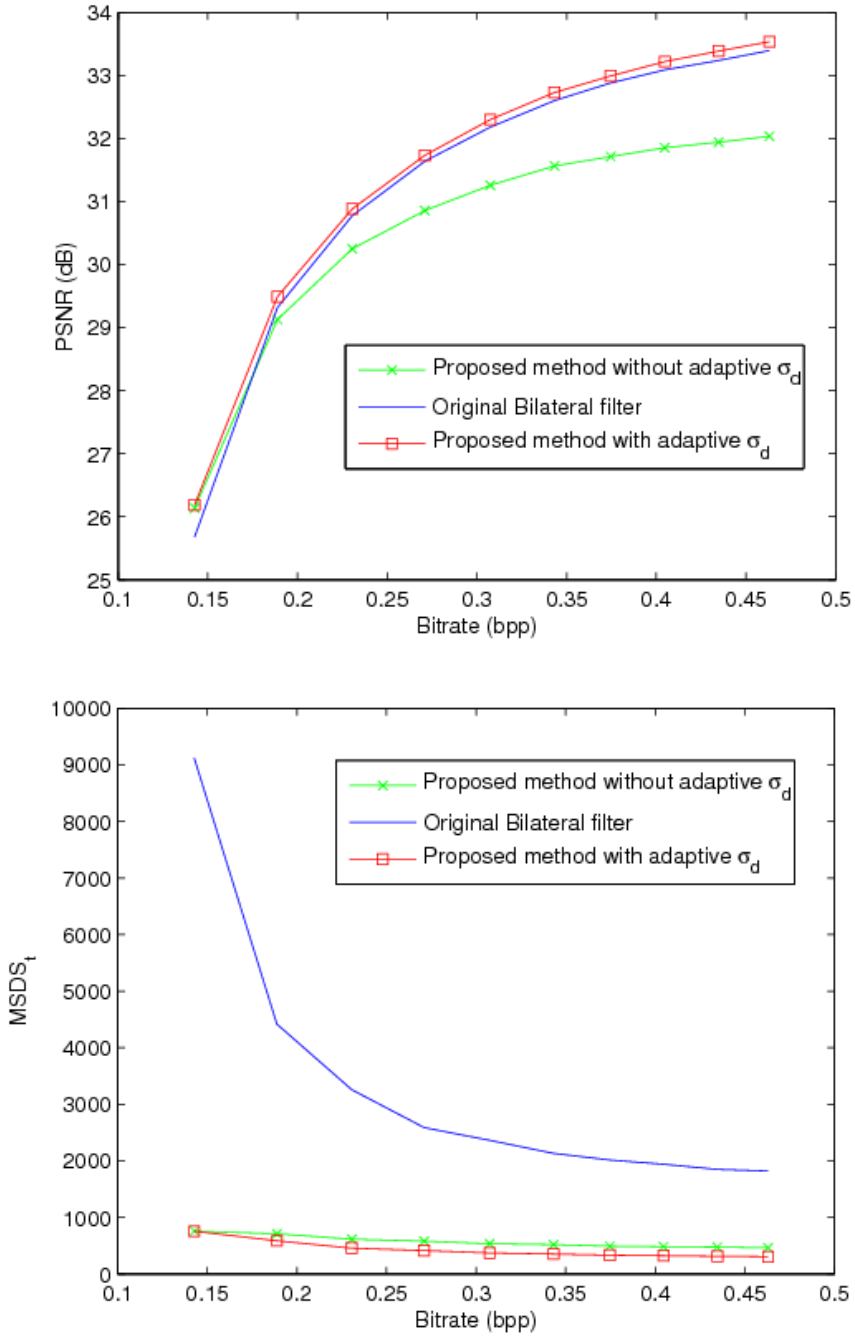



Figure 6.2 Comparison of original Bilateral with the proposed method with and without the adaptive sigma_d for the Lena test image under different bit-rate.

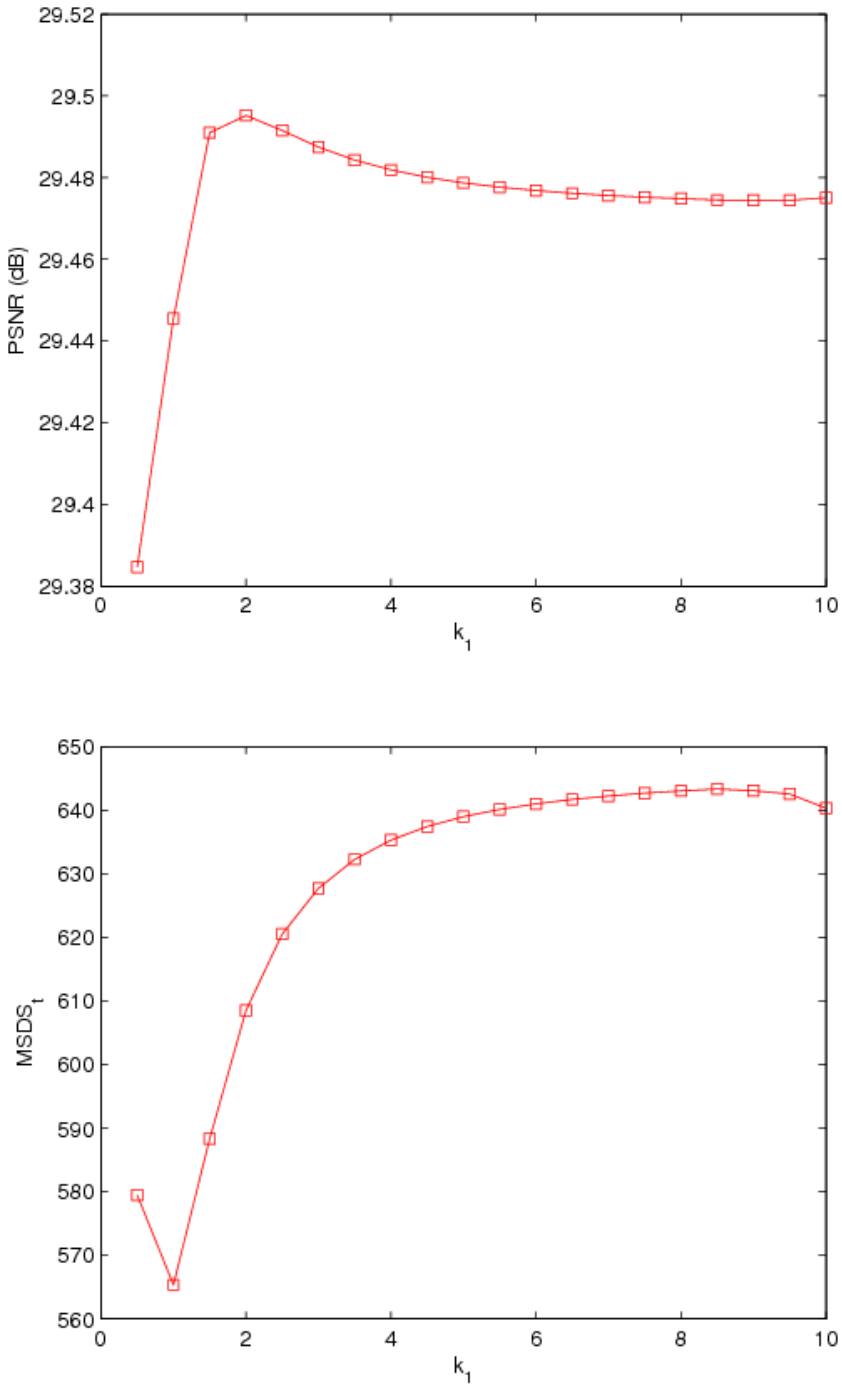
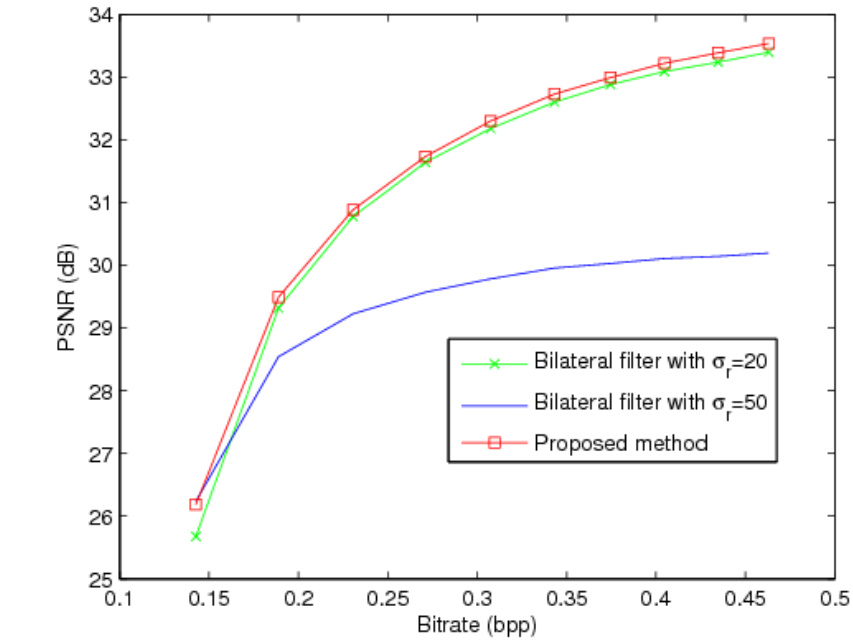



Figure 6.3 Relationship between k1 and the PSNR, MSDSt of the proposed method for the Lena test image under bit-rate=0.18. The image size is 512 by 512, k0=1.2



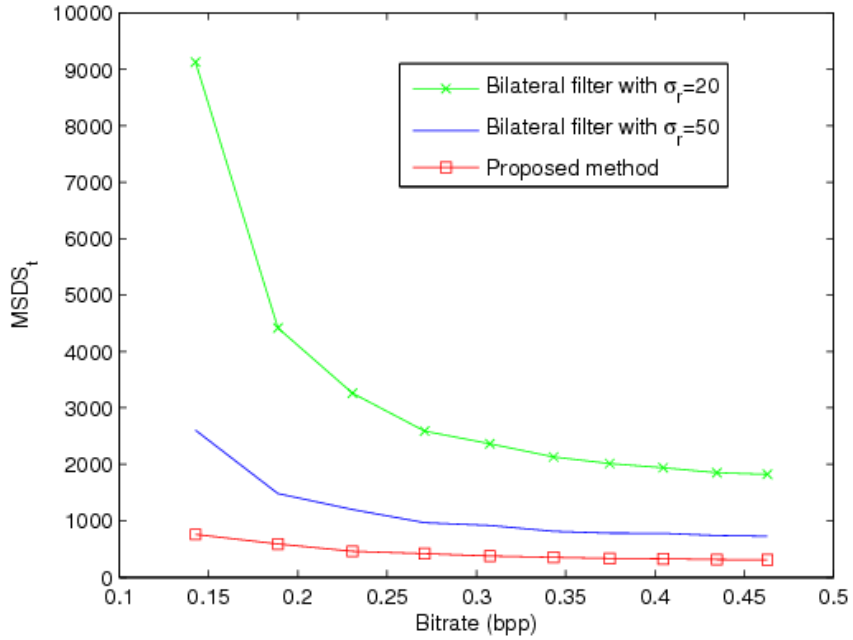




Figure 6.4 Comparison of original Bilateral filter with sigma_r of 20 and 50, and the proposed method for the Lena test image under different bit-rate.

Figure 6.5 Performance of Bilateral Filter. From left to the right: input compressed image, bilateral filtered image and adaptive bilateral filtered image.


Figure 6.6 Visual tests for adaptive parameters. (a) Original image. (b) Input compressed image. (c)The standard bilateral filter, (d) The adaptive bilateral filter

There is a test of proposed adaptive method is presented in Figure 6.6. $\sigma_r = 20$, $\sigma_d = 3$, $\sigma_{r,min} = 20$, $\sigma_{d,min} = 3$, $k_0 = 1.0, k_1 = 10.0$. It is obvious that for the texture part such as the shoulder of 'Lena', the proposed method can preserve the details smoothly.

A test of a region from ‘Lena’ under different rates is displayed in Figure 6.7.

Figure 6.7 Visual comparisons under different bit-rates. the first row shows the compressed image, the second row shows the original Bilateral filter, and the third row presents the proposed method. From left to right the compressed images’ bit-rate are 0.18, 0.22, 0.24, respectively.

In Figure 6.7, image size is $64\times 64$; The parameters are $k_0 = 1.2$, $k_1 = 10.0$, $\sigma_{r,min} = 20$, $\sigma_{d,min} = 1.5$, $\sigma_r = 20, \sigma_d = 3$. We can clearly find that the proposed adaptive method can eliminate the blocks effectively, compared to the residual blocking artifacts in the bilateral filter.

We also test for the video sequences, shown as Figure 6.8. One frame from our experiment with the video sequence "Foreman" is displayed.


Figure 6.8 Test for video sequence (a)Uncompressed frame; (b)Compressed frame; (c)Bilateral filter; (d)Proposed method.

Image size is $144\times137$; the input bitrate is 0.495. $k_0=1.2$, $k_1=10.0$, $\sigma_{r,min}=20$, $\sigma_{d,min}=1.5$, $\sigma_r=20$, $\sigma_d=3$. The results indicate that our method works best for saving the details in the texture part as well as removing the artifacts.

# 7 SUMMARIES AND CONCLUSIONS

## 7.1 Multi Resolution Bilateral Filter

In this section of the research, I have made an empirical study of the optimal parameter values for the bilateral filter in image denoising applications and present a multi resolution image denoising framework, which integrates bilateral filtering and wavelet thresholding. In this framework, I decompose an image into low- and high-frequency components, and apply bilateral filtering on the approximation sub-bands and wavelet thresholding on the detail sub-bands. We have found that the optimal $\sigma_r$ value of the bilateral filter is linearly related to the standard deviation of the noise. The optimal value of the $\sigma_d$ is relatively independent of the noise power. Based on these results, we estimate the noise standard deviation at each level of the sub-band decomposition and use a constant multiple of it for the $\sigma_r$ value for bilateral filtering.

As a result of these experiments in the previous chapter, we can reach the following conclusions:

(1) Clearly, with the multi-level bilateral filter, the strong noise is eliminated most effectively.

(2) For the gray-scale images, wavelet thresholding like BayesShrink [10] will still keep some random noises in the texture which can be clearly observed, while bilateral filter can remove the noises in the texture very well. However, original single level bilateral filter has a stronger intensity parameter $\sigma_r$ than the multi-level bilateral filter such that it will lose some details in the texture. With wavelet thresholding, multi-level bilateral filter can eliminate the noises in the high frequency components without obvious influence on the

texture.

(3) Wavelet thresholding like BayesShrink [10] and OWT SURE [14] is not effective for the real noisy images. The reason to explain this can be inferred that because wavelet thresholding is based on the robust median estimation, the real noise doesn't have the same property as the standard Gaussian random noise so that it could not be estimated correctly.

(4) Multi resolution bilateral filter can use any type of wavelet thresholding method, such as OWT SURE [14], as long as this method is effective. Compared with the proposed method, although the OWT SURE has good performance for the slight noisy image, but when it comes to the strong noisy image, using same parameters, the proposed method has the best output performance.

(5) When we transfer the color space into the CIEL*a*b* space, we can use the advantage of multi-level bilateral filter more effectively. Because human visual system is more sensitive to the red and green color noises, we can use higher level decomposition on the a and b channel. According to the observation, for those noises in the color channels a and b, like the red and green color noises on the girl's face and hair, only the multi-level bilateral filter can work with the most effect.

(6) We compare our algorithm with Portilla's BLS-GSM [13], which is considered as the most effective recent image denoising method. From Figure 4.3, we can find that BLS-GSM and single-level bilateral have some obvious color noises remained, while multi-level bilateral filter can avoid. Therefore, our proposed method has the best efficiency in the denoising works for real noisy image.

The key factor in the performance of the proposed method is the multi resolution

application of the bilateral filter. It helped eliminating the coarse-grain noise in images. The wavelet thresholding adds power the proposed method as salt-and-pepper type of noise components cannot be eliminated with the bilateral filter. We used a specific wavelet thresholding technique (i.e., the BayesShrink method); it is possible to improve the results further by using better detail-sub-band-denoising techniques or using redundant wavelet decomposition. These issues and the detailed analysis of parameter selection for the proposed framework are left as future work. We believe that the proposed framework will inspire further research towards understanding and eliminating noise in real images.

### 7.2 Compression Artifacts Reduction

In this section of research, a spatial adaptive method for the blocking artifact reduction is presented, which is manifested as an automatic detection for the texture and the discontinuity in the image so that we can apply different spatial and intensity parameters of the bilateral filter upon them. The value of the parameters is determined by the local mapping from the index assigned by the detection. From the experiment, the proposed method has the best performance in PSNR and MSDSt. At the same time, the visual quality of the results show that the proposed method can eliminate the blocking artifacts better and keep more texture details than the original bilateral filter.

In our preliminary experiments, the parameters were selected after some trial and error. As a next task, we will do a further analysis of these parameters. The preliminary results indicate that the adaptive method reduces the blockiness effectively while keeping the texture. Further improvement can be achieved by applying the smoothing process repeatedly. Another possible approach to improve the results is constraining the DCT coefficients of the resulting

image. The upper and lower bounds for each DCT coefficient are available at the decoder side. By iterating the processes of projecting the resulting image onto these bounds in the DCT domain and applying adaptive bilateral filtering in spatial domain, a better reconstruction can be achieved.

## REFERENCES


1. C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” Proc. Int. Conf. Computer Vision, 1998, pp. 839–846.

2. S. M. Smith and J. M. Brady, “Susan - a new approach to low level image processing,” Int. Journal of Computer Vision, vol. 23, pp.45–78, 1997.

3. L. Yaroslavsky, Digital Picture Processing - An Introduction, Springer Verlag, 1985.

4. R. Kimmel N. Sochen and R. Malladi, “Framework for low level vision,” IEEE Trans. Image Processing, Special Issue on PDE based Image Processing, vol. 7, no. 3, pp. 310–318, 1998.

5. R. Kimmel N. Sochen and A.M. Bruckstein, “Diffusions and confusions in signal and image processing,” Mathematical Imaging and Vision, vol. 14, no. 3, pp. 195–209, 2001.

6. R. Kimmel A. Spira and N. Sochen, “A short time beltrami kernel for smoothing images and manifolds,” IEEE Trans. Image Processing, vol. 16, no. 6, pp. 1628–1636, 2007.

7. M. Elad, “On the origin of the bilateral filter and ways to improve it,” IEEE Trans. Image Processing, vol. 11, no. 10, pp. 1141–1151, October 2002.

8. D. L. Donoho and I. M. Johnstone, “Ideal spatial adaptation by wavelet shrinkage,” Biometrika, vol. 81, no. 3, pp. 425–455, 1994.

9. D. L. Donoho, I. M. Johnstone, G. Kerkyacharian, and D. Picard, “Wavelet shrinkage: Asymptopian,” Journal of Royal Statistics Society, Series B, vol. 57, no. 2, pp. 301–369, 1995.

10. S. G. Chang, B. Yu, and M. Vetterli, “Adaptive wavelet thresholding for image denoising and compression,” IEEE Trans. Image Processing, vol. 9, no. 9, pp. 1532–1546, September 2000.

11. L. Sendur and I. W. Selesnick, “Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency,” IEEE Trans. Signal Processing, vol. 50, no. 11, pp. 2744–2756, November 2002.

12. A. Pizurica and W. Philips, “Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising,” IEEE Trans. Image Processing, vol. 15, no. 3, pp. 654–665, March 2006.

13. J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, “Image denoising using scale mixtures of gaussians in the wavelet domain,” IEEE Trans. Image Processing, vol. 12, no. 11, pp. 1338–1351, November 2003.

14. F. Luisier, T. Blu, and M. Unser, “A new sure approach to image denoising: Inter-scale orthonormal wavelet thresholding,” IEEE Trans. Image Processing, vol. 16, no. 3, pp. 593–606, March 2007.

15. A. Buades, B. Coll, and J. Morel, “Neighborhood filters and pde’s,” Technical Report 2005-04, CMLA.

16. P. Perona and J. Malik, “Scale-space and edge detection using anistropic diffusion,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 12, no. 7, pp. 629–639, July 1990.

17. M. Elad and M. Aharon, “Image denoising via learned dictionaries and sparse representation,” Proc. IEEE Computer Vision and Pattern Recognition, June 2006.

18. M. Elad J. Mairal and G. Sapiro, “Sparse representation for color image restoration,” IEEE Trans. Image Processing, vol. 17, no. 1, pp.53–69, January 2008.

19. M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” IEEE Trans. Image Processing, vol. 54, pp. 3736–3745, 2006.

20. H. Takeda, S. Farsiu and P. Milanfar, “Deblurring Using Regularized Locally Adaptive Kernel Regression,” IEEE Trans. Image Processing, vol.17, no. 4, pp. 550-563. April 2008.

21. C. Kervrann and J. Boulanger, “Optimal spatial adaptation for patch-based image denoising,” IEEE Trans. Image Processing, vol. 15, pp. 2866–2878, 2006.

22. K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising by sparse 3D transform-domain collaborative filtering,” IEEE Trans. Image Process., vol. 16, no. 8, pp. 2080-2095, August 2007.

23. J. G. Apostolopoulos and N. S. Jayant, “Postprocessing for very low bit-rate video compression,” IEEE Trans. Image Processing, vol. 8, no. 8, pp. 1125–1129, August 1999.

24. A. K. Katsaggelos, J. Mateos and R. Molina, “A bayesian approach for the estimation and transmission of regularization parameters for reducing blocking artifacts,” IEEE Trans. Image Processing, vol. 9, no. 7, pp. 1200–1215, July 2000.

25. A. W. C. Liew, C. Weerasinghe and H. Yan, “Artifact reduction in compressed images based on region homogeneity constraints using the projection onto convex sets algorithm,” IEEE Trans. Circuits and Systems for Video Technology, vol. 12, no. 10, pp. 891– 897, October 2002.

26. H. R. Wu, T. Chen and B. Qiu, “Adaptive postfiltering of transform coefficients for the

reduction of blocking artifacts," IEEE Trans. Circuits and Systems for Vide Technology, vol. 11, pp. 594–602, May 2001.

27. S. Liu and A. C. Bovik, "Efficient dct-domain blind measurement and reduction of blocking artifacts," IEEE Trans. Circuits and Systems for Vide Technology, vol. 12, no. 12, December 2002.

28. Z. Li and A. J. Delp, "Block artifact reduction using a transform-domain markov random field model," IEEE Trans. Circuits and Systems for Video Technology, vol. 15, no. 12, December 2005.

29. W. Lin, W. Zhang, C. Wang, P. Xue and S. Yu, "Fast edge-preserved postprocessing for compressed images," IEEE Trans. Circuits and Systems for Video Technology, vol. 16, no. 9, pp. 1142 – 1147, September 2006.

30. S. Paris, F. Durand, "A fast approximation of the bilateral filter using a signal processing approach", Proc. European Conference on Computer Vision, 2006

31. B. Weiss, "Fast median and bilateral filtering", Proc. SIGGRAPH, 2006.

32. F. Porikli, "Constant Time O (1) Bilateral Filtering", IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), July 2008.

33. X. Zhang and B. A.Wandell, "A spatial extension of CIELAB for digital color image reproduction", Society for Information Display Journal, 1997

34. S. Minami and A. Zakhor, "An optimization approach for removing blocking effects in transform coding,"IEEE Trans. Circuits and Systems for Video Technology, vol. 5, no. 2, April 1995.

35. D. Tzovaras, G. A. Triantafyllidis and M.G. Strintzis, "Blocking artifact detection and reduction in compressed data," IEEE Trans. Circuits and Systems for Video Technology, vol. 12, no. 10, October 2002.June 2008

## VITA

Ming Zhang is graduated from Beijing University of Posts and Telecommunications in 2006 with the bachelor degree in Information Engineering. He attended the Department of Electrical and Computer Engineering in Louisiana State University in September 2006. He joined the image processing lab under the directing from Dr. Bahadir Gunturk. His research interests are image denoising, blocking artifacts reduction and high dynamic range imaging. He has one paper published on IEEE Transaction on Image Processing and two conference papers during the time he worked in the lab.