國 立 交 通 大 學

電子工程學系 電子研究所碩士班

碩 士 論 文

對於抑制高斯雜訊、脈衝雜訊與壓縮缺陷之研究

A Study on Reduction of Gaussian Noise, Impulse Noise and Compression Artifacts

研 究 生：張瑞男

指導教授：王聖智 博士

中 華 民 國 九 十 八 年 七 月

對於抑制高斯雜訊、脈衝雜訊與壓縮缺陷之研究

A Study on Reduction of Gaussian Noise, Impulse Noise and Compression Artifacts

研 究 生：張瑞男　　Student：Jui-Nan Chang

指導教授：王聖智博士　　Advisor：Dr. Sheng-Jyh Wang

國 立 交 通 大 學

電子工程學系 電子研究所碩士班

碩 士 論 文

A Thesis

Submitted to Department of Electronics Engineering & Institute of Electronics

College of Electrical and Computer Engineering

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of Master

in

Electronics Engineering

July 2009

Hsinchu, Taiwan, Republic of China

中華民國九十八年七月

對於抑制高斯雜訊、脈衝雜訊與壓縮缺陷之研究

研究生：張瑞男　　指導教授：王聖智 博士

國立交通大學

電子工程學系 電子研究所碩士班

摘要

在現今許多電子消費產品中，高畫質數位影像成為主流的資訊來源，對於影像品質要求也愈來愈高。因此，處理影像中的雜訊便成為首要解決的課題。本論文針對常見的高斯雜訊、脈衝雜訊以及壓縮缺陷，設計幾個簡單的處理方法。首先我們對於未知雜訊類型之影像作壓縮缺陷與脈衝雜訊之偵測，再使用我們所提出的演算法抑制雜訊以提升影像品質。為了符合實際應用需求，對於影像中之壓縮缺陷，我們所設計之處理方法不需任何原始影像壓縮的編碼資訊，僅處理解壓縮後的影像資料，而且演算法不需太多花費計算時間與硬體資源，以適用於彩色影像與視訊的處理。

A Study on Reduction of Gaussian Noise, Impulse Noise and Compression Artifacts

Student : Jui-Nan Chang Advisor : Dr. Sheng-Jyh Wang

Department of Electronics Engineering, Institute of Electronics

National Chiao Tung University

Abstract

In many of today's consumer electronics products, high-definition digital images have become a main source of information. As the demand of high quality of images keeps increasing, the reduction of image noise becomes a primary topic. In this thesis, we design several approaches for the reduction of Gaussian noise, impulse noise, and compression artifacts. First, images with unknown type of noise would be processed for the detection of compression artifact and impulse noise. The proposed methods are then used to enhance image quality. For practical applications, the proposed algorithms do not require any coding information, and can be directly applied over decompressed data. The proposed methods require low computational loads and hardware resources, and can be efficiently applied to various kinds of color images and videos.

誌謝

在這幾年的學習生涯，首先要非常感謝我的指導教授王聖智老師，從大學部開始導師生聚會以及修課，到碩士班結束期間，從老師身上學習到不僅僅在於研究領域相關的知識與方法，還有許多受用一生的學習態度與待人接物的道理，也是大家心中的模範，在此向老師至上最高的感謝之意。非常感謝慈澄學長、敬群學長與禎宇學長對於研究相關的經驗分享、討論與平日的照顧，託敬群學長之邀約讓我有機會到國外參與大型研討會，也特別感謝慈澄學長對於我研究方面的指導與協助，讓我能夠順利完成論文。很感謝一同在實驗室打拚的維辰、庭瑋與文中，還有瑋國、周節、育瑋學弟們諸多的幫忙，並協助處理實驗室各類事務，也讓我在這兩年實驗室生活中度過許多歡樂的時光。感謝已畢業的奕安、晴駿與博凱等學長姐在我碩一時的照顧，感謝你們畢業之後願意抽空回實驗室與大家分享在業界與軍旅生活的點點滴滴。最後，感謝我的家人與朋友對我的支持、體諒與鼓勵，在此將我的論文獻給所有曾給予我協助的師長、同學、朋友與家人。

Contents

List of Figures

List of Tables

Chapter 1. Introduction

Noise in digital images can be easily created during image acquisition and transmission. Nowadays, noise reduction has become a typical problem in image processing. Additive Gaussian noise and impulse noise can adequately represent most corrupting image noise. Many noise-removal methods depend on a specific type of noise. For example, classical linear filters, such as mean filter and Gaussian filter, can suppress Gaussian noise. For impulse noise, rank statistics and order information are usually used. These nonlinear methods have nice performance in noise reduction, but require higher computational cost. In this thesis, we aim to design algorithms that are simple but can still effectively eliminate noise for color images and videos.

On the other hand, with the wide-spreading usage of compressed images and videos, compression artifacts have become another noise problem for low bit-rate images and video streams. Based on modern coding standards, from the early JPEG to the latest H.264/AVC, these codecs usually apply non-overlapping block discrete cosine transform (BDCT). The quantization error or loss of high frequency components of DCT coefficients will result in blocking and ringing artifacts for highly compressed images and videos. The compression artifacts can be dealt with at the encoding end as well as the decoding end. Many pre-processing or BDCT-based methods require the information of the codec and the uncompressed data. However, for general digital storage, such as DVD videos and internet streams, the only information that is available is the decompressed data. Hence, for practical applications, we aim to develop algorithms that operate over decompressed images and videos to suppress compression artifacts.

The rest of the thesis is organized as follows. In Chapter 2, we introduce the noise models and several existing noise reduction methods. In Chapter 3, we discuss the purposed de-noising methods. Experiment results are showed in Chapter 4. Finally, we make the conclusions in Chapter 5.

Chapter 2. Backgrounds

In this chapter, related concepts and papers about noise reduction will be discussed. Firstly, we will introduce the noise models in Section 2.1. Several reduction methods for Gaussian noise, impulse noise and Compression artifacts will be introduced in Section 2.2, 2.3 and 2.4, respectively.

2.1 Noise Models

2.1.1 Additive Gaussian Noise

There are different sources of noise in a digital image. In the image acquisition of CCD sensors, dark current noise is generated due to the thermally excited electrons at the sensor side. The level of dark noise is proportional to the exposure time and is highly dependent on sensor temperature. On the other hand, shot noise, which is characterized by a Poisson distribution, is also observed due to the quantum uncertainty in photoelectron generation. Moreover, there are some other types of noise during image acquisition. According to the central limit theorem, the overall noise effect can be generally modeled as a Gaussian distribution. It is characterized by adding to each image pixel a random value with a zero-mean Gaussian distribution. Figure 2-1(b) shows an example of degraded image with additive Gaussian noise.

(a) (b)

Figure 2-1 (a) The pure image and (b) the degraded image with additive Gaussian noise

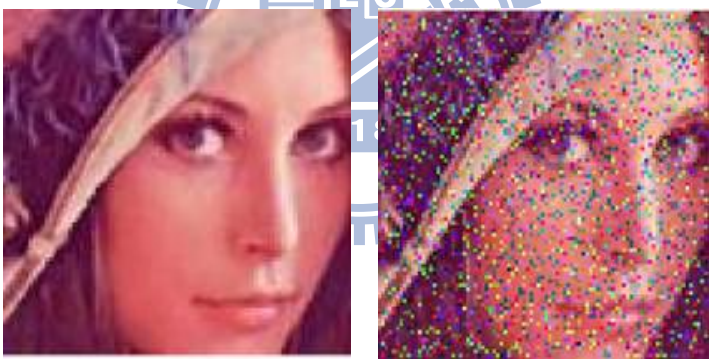
Here, we use the standard notation to model the degradation process. For example, $u(i,j)$ represents the intensity value of an image u at the pixel location (i,j). For the case of additive Gaussian noise, the noisy image u is related to the original image u^o by

$$u(i,j) = u^o(i,j) + n(i,j), \qquad where \;\; n(i,j) \sim N(0,\sigma^2) \qquad Eq.\ 2\text{-}1$$

In Eq.2-1, each noise value n is drawn from a zero-mean Gaussian distribution. The varianceσ^2 of this distribution determines the power of the corrupting noise. The zero-mean property allows removing such noise by locally averaging the pixel values.

2.1.2 Impulse Noise

Impulse noise is caused by errors in noisy sensors, the data transmission process in the communication channel, or by errors during the data capture from the digital camera. Figure 2-2 (b) shows an example of degraded image with impulse noise.



(a) (b)

Figure 2-2 (a) The pure image and (b) the degraded image with impulse noise

The impulse noise is modeled by replacing a portion of the original pixel values of the image with intensity values drawn from a certain distribution, usually a uniform distribution over the whole intensity range or a discrete distribution at specific intensity values. Throughout this thesis, we consider only the uniform distribution model, although the proposed method may also be used without modification for the discrete distribution model. Hence, for images corrupted with impulse noise, the noisy image is related to the original image u^o by

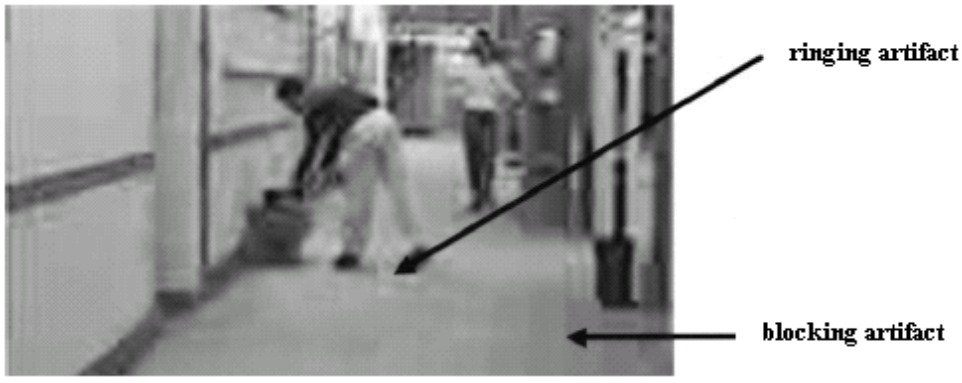
$$u(i,j)=\begin{cases} n(i,j), & \text{with probability } p \\ u^{o}(i,j), & \text{with probability } (1-p) \end{cases} \qquad Eq.\ 2\text{-}1$$

where n(i, j) has the uniform distribution over $[I_{min}, I_{max}]$.

2.1.3 Compression Artifacts

Artifacts in compressed images are common. For block DCT based compression, an image is first segmented into blocks, and then transformed, quantized and coded. Since each block is treated separately, coarse quantization steps could cause annoying exotic edges around block boundaries, and it's known as blocking artifacts. Furthermore, because of the quantization errors of different levels, annoying fluctuations may also be found on the regions near edges or corners. This phenomenon is called the ringing artifact. An example of the compression artifacts is showed in Figure 2-3.

(a)



(b)

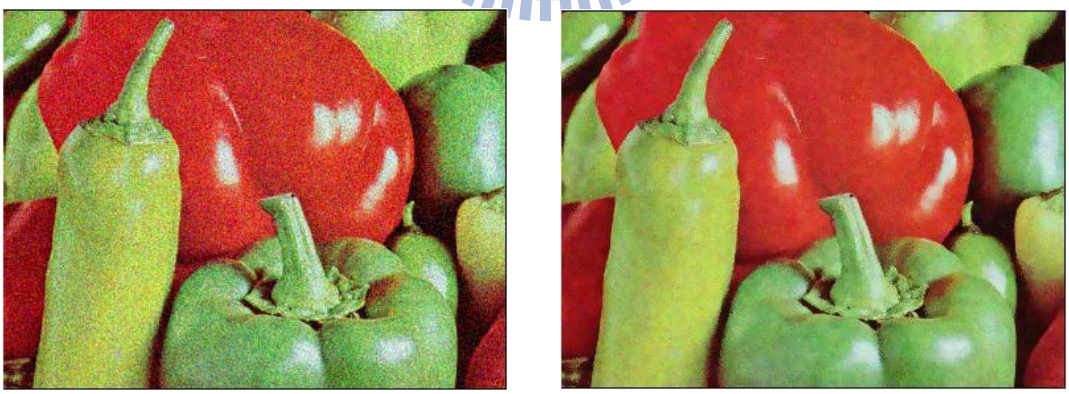
Figure 2-3 Compression artifacts: (a) original image (b) compressed image

2.2 Gaussian Noise Reduction Methods

Ideally, removing Gaussian noise would involve smoothing the different areas of an image without degrading either the sharpness of the signal edges or details. However, classical linear filters, such as the Arithmetic Mean Filter (AMF) or the Gaussian Filter, smooth noise but may also blur edges significantly. Usually, nonlinear methods are used to relief this problem. The anisotropic diffusion, proposed by Perona and Malik, is a well-known method described in [1]. In this technique, local image variation is measured at every point and its neighboring pixels and differential equations are involved. This diffusion method is inherently iterative. Efficiency and stability could be the important issues, depending on the adopted computational architecture.

2.2.1 Bilateral Filter

The bilateral filtering was originally represented by Tomasi and Manduchi [2]. They used Gaussian functions as the weighting functions to improve the filter's performance, analyzed the interaction between the weighting functions, and proposed metrics for color images. Figure 2-4 shows an example of Gaussian noise reduction by bilateral filter.



(a) (b)

Figure 2-4 An image (a) with additive Gaussian noise and (b) processed by bilateral filter

2.3 Impulse Noise Reduction Methods

Impulse noise is characterized by replacing a portion of an image's pixel values with random values, leaving the remainder unchanged. Such noise can be introduced due to transmission errors. The most noticeable and least acceptable pixels in the noisy image are those intensities that are very different from their neighbors.

The Gaussian noise removal methods mentioned above cannot adequately remove impulse noise because these methods interpret the impulse noise pixels as edges and need to be preserved. For this reason, a separate class of nonlinear filters has been developed specifically for the removal of impulse noise. Many of them are extensions of the median filter [3], or use rank statistics [4][5][6]. The common idea of these filters is to detect the impulse pixels and replace them with estimated values, while leaving the remaining pixels unchanged.

Impulse noise removal methods use many different techniques to determine whether a given pixel is an impulse. These approaches vary in complexity from relatively simple to highly complex. The simplest impulse detectors are based on two-state methods that attempt to definitively characterize each image pixel as either an impulse or an unaffected pixel [7]. The underlying goal of these two-state methods is to find pixels that are significant outliers when compared to their neighbors. More complex methods are naturally more successful for detecting impulses in general, but there is a tradeoff between performance and complexity. The most complicated methods require training procedure to make an optimal classification based on measures of pixels and their neighbors. Methods that require training are bound to be less controllable and more unpredictable than simpler methods. An additional concern with existing methods arises from the fact that when impulse noise appears in an image, a portion of the corrupted pixels will be replaced with intensities only slightly different from their original values. In this case, impulse noise is still visible in the processed image. To handle this issue, a universal noise removal method in [13] used rank statistic from local pixels to detect the impulse strength for each pixel. This detector is combined with the bilateral filter and can remove both Gaussian noise and impulse noise. More discussion and comparisons would be introduced later in this thesis.

2.4 Compression Artifacts Reduction Methods

There have been 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 a main focus in the field. These post-processing methods can be categorized into two types of approaches: spatial-domain approaches and transform-domain approaches, depending on which domain the image is processed on. There are also some methods that operate over both domains. In this section, we will introduce compression artifacts reduction methods in both domains.

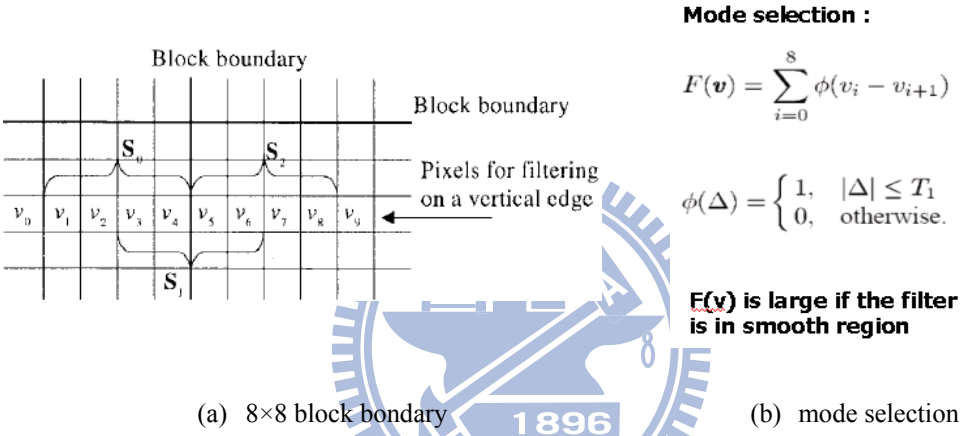
2.4.1 Reduction of Blocking Artifact

Basically, the blocking artifact represents the discontinuous pixels between 8×8 image blocks. Thus, a simple method in the spatial domain is to use a low-pass filter to eliminate the discontinuity. [8] utilized a Gaussian low-pass filter to remove blockiness. The result is showed in Figure 2-5.

(a) Original image (b) Deblocked by [8]

Figure 2-5 Deblocked result [8]

Blocking artifact may be effectively removed by low-pass filtering, but inevitably the over-smooth problem would be found in fine regions of images. In [9], the authors proposed a method that is based on human visual sensitivity in detecting blocking artifact on different kinds of regions, like smooth regions or fine-detail regions. Based on the human visual perception, the blocking artifact could be more easily found in smooth regions than fine-detail regions. The method in [9] separated images by smooth regions and fine regions. In order to avoid the over-smooth problem, the smoothing operation is implemented strongly over smooth regions while weakly over fine-detail regions. The details of the algorithm will be introduced in the following sections and illustrated in Figure 2-6:



(a) 8×8 block bondary (b) mode selection

Figure 2-6 Adaptive smoothing of blocking artifacts [9]

For the block boundary in Figure 2-6 (a), the method in [9] analyzed that whether the block boundary belongs to a smooth or fine-detail region by using a measure function $F(\boldsymbol{v})$. $F(\boldsymbol{v})$ was defined as the difference of intensity values between two neighbor pixels. If $F(\boldsymbol{v})$ was larger than a pre-defined threshold, the block boundary would be considered as over a smooth resign. In this case, the method in [9] implemented a stronger smoothing operator for de-blocking. On the other hand, if the block boundary was considered as on fine-detail regions, the method in [9] used a weak smoothing operator for de-blocking in the DCT domain. This adaptive process may well preserve the fine-detail features, as showed in Figure 2-7.

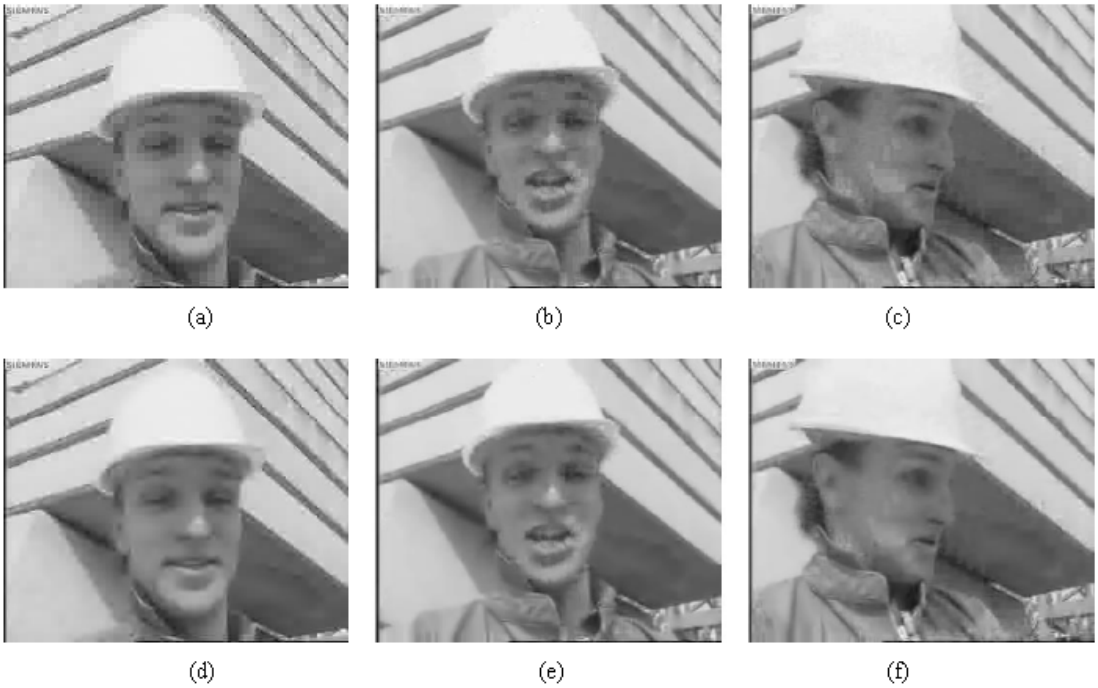


Figure 2-7 The results of [9].
(a) (b) (c) are the original frames. (d) (e) (f) are the de-blocked frames.

2.4.2 Reduction of Ringing Artifact

For DCT-based coding, Ringing artifacts are caused by the Gibbs phenomenon and appear around the edge regions of the image. Up to now, many filter-based de-noising methods have been proposed to reduce this kind of artifacts. To reduce ringing artifacts, the method in [10] proposed to first detect near strong edges the areas with ringing artifacts, and then applied linear or nonlinear isotropic filters to reduce the ringing artifacts. Non-linear filters help preserving edges of the images by exploiting the spatial order of the surrounding pixels, together with the rank order. The method in [11] used local variance of pixel values to find edge pixels. Because the ringing artifacts occur along the edges, the pixels whose local variance is larger than a pre-defined threshold are supposed to be more likely to possess ringing artifacts. For these pixels, the method in [11] utilized the fuzzy filter to smooth the ringing artifacts. This fuzzy filter is basically a edge-preserving filter that can avoid over-smoothing. Figure 2-8 shows the results of the fuzzy filtering [11].

(a) original image

(b) processed image

Figure 2-8 The ringing artifact reduction by [11]

One drawback of fuzzy filter is that the signal is converted to a vector before filtering. The relative positions of the pixels are ignored in the conversion.

On the other hand, there are some other methods that require domain transformation and iterative optimization. Even though these methods have good performance in removing compression artifacts, the required domain transformation and iterative optimization make them too complex for practical applications. In this thesis, we will focus on spatial-domain operations and aim to develop a simple algorithm that does not require complicated computations.

Chapter 3. Proposed Methods

3.1 Proposed Architecture

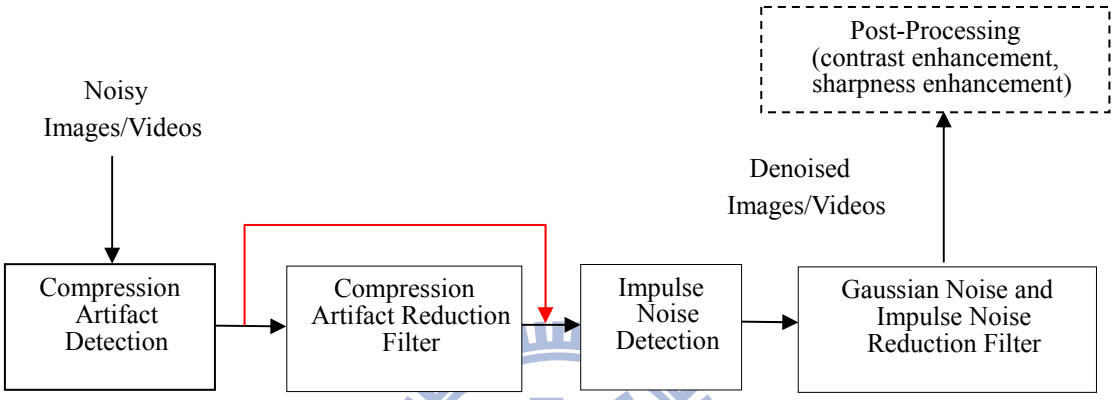


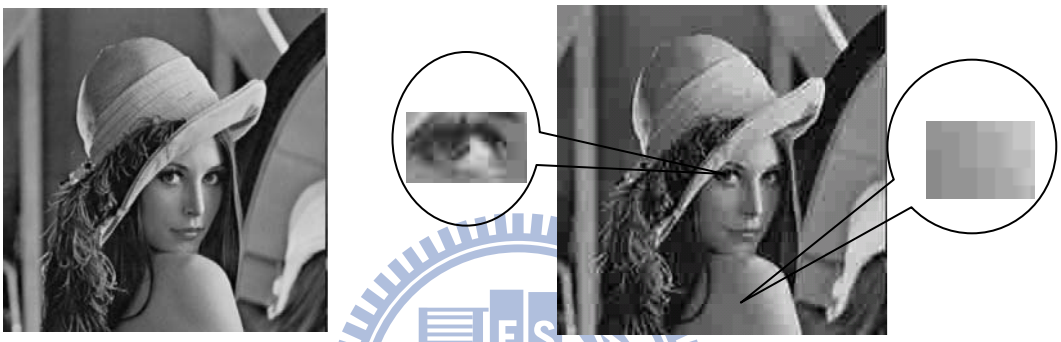
Figure 3-1 Proposed architecture for noise reduction

For convenience, in this section, our proposal for noise reduction is introduced first. Based on the architecture illustrated in Figure 3-1, the process flow of the proposed methods can be explained as follows. For noisy images and video frames that may contain three different types of noise (Gaussian noise, impulse noise, or compression artifacts), the results of compression artifact detection identify the location of compression artifacts in the images/videos. If the detection results indicate strong compression artifacts, the proposed compression artifact reduction filter is utilized. Otherwise, the compression artifact reduction process is skipped. The next module is impulse noise detection, which detects the possible positions of impulse noise in the images/videos. The proposed noise reduction filter is then used to suppress Gaussian noise and impulse noise. An optional module can be further used to enhance the contrast or sharpness of the denoised images.

3.2 Reduction of Compression Artifacts

3.2.1 Detection of Compression Artifacts

For 8x8 block-based compression codecs, compression artifacts may exist due to low-bitrate transmission and coarse quantization. Blocking artifacts appear as discontinuous intensity chance near block boundaries. An example for blocking artifact shows on Figure 3-2.



(a) original image (b) compressed image with blocking artifact

Figure 3-2 An example of blocking artifact

Compared to the original image in Figure 3-2 (a), the blocking artifact can be obviously found on the smooth regions of compressed image, like Lena's shoulder in Figure 3-2 (b). Relatively, the blocking artifact in fine regions would not be easily observed, like Lena's eye in Figure 3-2 (b). In human visual perception, the visibility of blockiness is highly related to the local variation of the image. Hence, we may check the local deviation of each 3×3 block by calculating the standard deviation $S(\mathbf{x})$ within the local neighborhood $\Omega_{\mathbf{x}}$, as expressed in Eq. 3-1.

$$S(\mathbf{x}) = \sqrt{\frac{\sum_{\mathbf{y}\in\Omega_{\mathbf{x}}} \mathbf{y}^2 - n\bar{\mathbf{y}}^2}{n-1}}, \text{ where } \Omega_{\mathbf{x}}(N) := \{\mathbf{x} + (i,j) : -N \le i,j \le N\} \qquad \textit{Eq. 3-1}$$

In Eq. 3-1, n represents the total number of pixels within the neighborhood. For a 3 by 3 window, we have $n = 9$. Figure 3-3 shows an example of local deviation distribution.

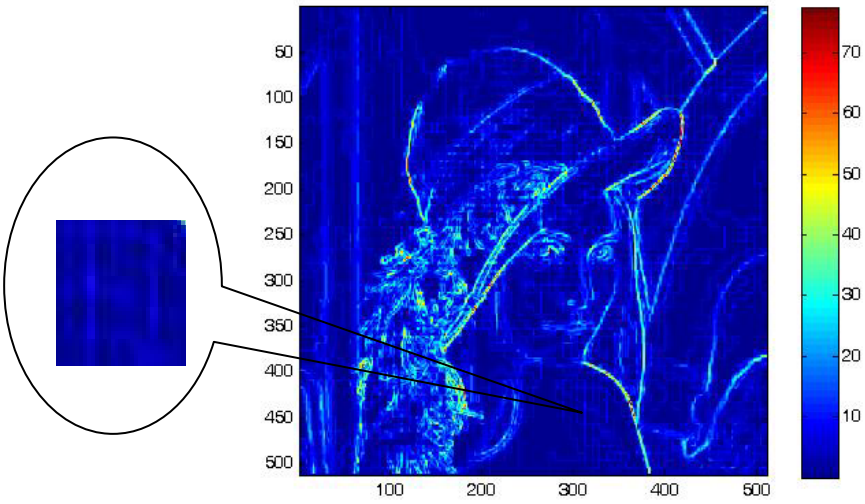


Figure 3-3 The distribution of local deviation of compressed image

In Figure 3-2, we can observe some blocky patterns in the local deviation distribution. To extract these patterns from the local deviation distribution, we design a proper range of deviation value, as expressed in the following equation

$$S_{BD}\left(S(\mathbf{x})\right)=\begin{cases}1, & if\ t_1 \le S(\mathbf{x}) \le t_2\\ 0, & otherwise\end{cases} \qquad \textit{Eq. 3-2}$$

The result of blocking artifact detection is showed Figure 3-4.

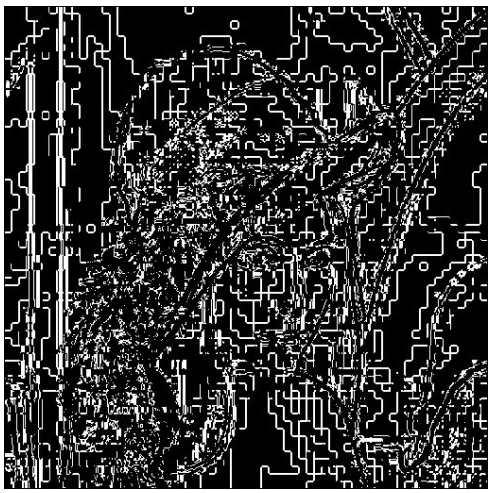


Figure 3-4 The result of blocking artifact detection

The two parameters t_1 and t_2 are related to the quality factor and the variation level of the image. If the quality factor is smaller than 15%, the blockiness could be stronger

and more observable in flat regions. For this case, in our experiments, we choose a larger value of t_1 to be within the range [2, 4] for detection. On the contrary, for the quality factor larger than 15%, the we choose t_1 to be within the range of [1.2, 2]. The range of t_1 <1.2 represents regions with small deviation. These variations would be almost unnoticeable to human vision. On the other hand, the value of t_2 is related to the variation of the signal features. In our experiments, a simple way to choose t_2 is choose the average of the local deviation $S(\mathbf{x})$. For most images, the t_2 value is typically within the range of [5, 12].

With modern coding standards, the compression artifacts usually occur at the boundaries between 8×8 blocks. Many traditional deblocking methods focus on these boundaries to suppress blockiness. However, because video coding also includes motion estimation and motion compensation, the compression artifacts may also occur at many other locations in an image frame. Furthermore, during image processing, the images with compression artifacts may sometimes be down-sampled or up-sampled. Hence, the location of compression artifacts may not necessarily occur at the boundaries between 8×8 blocks. Since our method doesn't restrict itself to the boundaries between 8×8 blocks, it can handle the compression artifacts at any location.

3.2.2 Reduction of Compression Artifacts

In compression artifacts removal, the blockiness can be eliminated by several local smoothing methods. An example of reduction of compression artifacts by using bilateral filter [2] is showed in Figure 3-5.

(a) blocking artifact (b) deblocked well but over smooth (c) fine region preserved but poorly deblcoked

Figure 3-5 The result of blocking artifact reduction by bilateral filter

As shown in Figure 3-5 (b), as we apply a strong smoothing operation over the image, most blocky regions are smoothed but fine-detail regions are also overly smoothed. In comparison, if we choose a weaker smoothing, as shown in Figure 3-5 (c), fine-detail regions are preserved but the blackness cannot be effectively suppressed. In this thesis, a simple method that destroys the blockiness by adding in random noise, rather than local smoothing, is proposed. This method is to eliminate the perceptual blockiness in human eyes. Here, we add in random noise of adaptive strength based on the level of blockiness. After the insertion of random noise, the blocky phenomenon can be successfully suppressed, as showed in Figure 3-6.

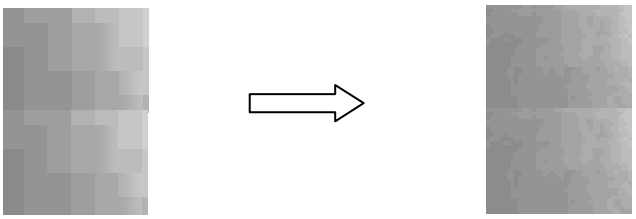


Figure 3-6 The result of blocking artifact reduction by adaptive insertion of random noise

In Figure 3-6, the blocky edges are obviously destroyed by the noise. However, another problem is that the insertion of noise makes the processed image look dustier. Hence, we use the bilateral filter to further smooth these regions and the whole algorithm can be described in the following pseudo code.

Table 3-1 The pseudo code of deblocking algorithm

```
for all x in images
   if   S_BD(x) = 1
        all  y ∈ Ω_x   replaced by U[Min(y),Max(y)]
   else
    u(x) = u^o(x)
end
```

In Table 3-1, the detected location **x** and neighboring pixels y in the corresponding window Ω_x would be replaced by uniform distribution noise **U**. The range of noise U is determined by the maximal and minimal values of the neighboring pixels.

In fact, a similar method by adding noise to destroy false contours can be found in the Improved Gray-scaled (IGS) Quantization method [12]. For lossy data compression,

false contouring may occur in smooth regions after quantization. By adding to each pixel a proper level of pseudorandom number, false contouring and artificial edges would not be observed by human visual perception.

Even though the blocking artifacts can be suppressed by the proposed method, the smoothing step is still required to suppress the perceptually dusty effect caused by the injected noise. Here, we use a modified bilateral filter to smooth the dusty regions. Figure 3-7 (c) shows the smoothing result of the modified bilateral filter. The modified bilateral filter will be introduced in the next section. With this smoothing step, ringing artifacts can also be suppressed. This is because the regions with ringing artifact are usually adjacent to the regions with blocking artifacts.

(a) original image with blocking artifact (b) inserted noise on detected regions

(c) smoothed result of (b) by modified bilateral filter

(d) over smoothed result of (a) by bilateral filter

Figure 3-7 Result of blocking artifact reduction

3.3 Reduction of Gaussian Noise and Impulse Noise

n this chapter, we'll discuss the purposed method for the reduction of Gaussian noise and impulse noise. The goal of the algorithm is to eliminate these two types of noise for color images and videos. Practically, in order to reduce the hardware storage and computation time, we demand the following requirements in the development of the noise reduction algorithm.

1. Wish to design a filter that can smooth noisy regions while preserving edges and fine-detail regions.
2. The algorithm is simple, non-iterative, and doesn't need complicated computations.

3.3.1 Bilateral Filter

The bilateral filter, as described in [2], applies a nonlinear filter to u to remove Gaussian noise while remaining the edge information. Each pixel is replaced by a weighted average of the intensities in a (2N+1) × (2N+1) neighborhood. The weighting function is designed to smooth regions with similar intensity while keeping edges intact.

More precisely, let **x** be the location of the pixel under consideration, and let

$$\Omega = \Omega_{\mathbf{x}}(N)$$

$$\Omega_{\mathbf{x}}(N) := \{\mathbf{x} + (i,j) : -N \le i,j \le N\} \qquad \textit{Eq. 3-3}$$

be the pixels in the (2N+1) × (2N+1) neighborhood of **x**. The weight of each $\mathbf{y} \in \Omega$ with respect to **x** is the product of two components, one spatial weight and one photometric weight:

$$w(x,y) = w_S(x,y)w_P(x,y) \qquad \textit{Eq. 3-4}$$

where

$$w_S(\mathbf{x},\mathbf{y}) = e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{2\sigma_S{}^2}} \qquad \textit{Eq. 3-5}$$

and

$$w_P(\mathbf{x},\mathbf{y}) = e^{-\frac{|u_\mathbf{x}-u_\mathbf{y}|^2}{2\sigma_P{}^2}} \qquad \textit{Eq. 3-6}$$

The $w_P(\mathbf{x},\mathbf{y})$ is further normalized and the smoothed pixel $\tilde{u}_x$ can be described as

$$\tilde{u}_\mathbf{x} = \frac{\sum_{\mathbf{y}\in\Omega} w(\mathbf{x},\mathbf{y})u_\mathbf{y}}{\sum_{\mathbf{y}\in\Omega} w(\mathbf{x},\mathbf{y})} \qquad \textit{Eq. 3-7}$$

The w_S weighting function decreases as the spatial distance in the image between **x** and **y** increases; while the w_P weighting function decreases as the intensity difference between the color vectors increases. The spatial component decreases the influence of the distant pixels and reduces the blurring effect; while the photometric component reduces the influence of those pixels that are perceptually different with respect to the one under processing. In this way, only perceptually similar areas of pixels are averaged and thus the sharpness of edges can be preserved.

The parameters σ_S and σ_P are used to adjust the influence of the spatial and the photometric weightings, respectively. They can be considered as rough thresholds for identifying pixels sufficiently close or similar to the central pixel. Note that when σ_P approaches infinity, the bilateral filter becomes a Gaussian filter. On the other hand, when both σ_P and σ_S approach infinity, the bilateral filter behaves like the mean filter.

Bilateral filter has been proved to have excellent performance for Gaussian noise removal. The algorithm is adequately simple and can be used for color images with low computation cost. However, for strong destructive noise, such as impulse noise, the bilateral filter performs rather poorly. In the following sections, we will make some modification over the bilateral filter to improve the reduction of impulse-like noise.

3.3.2 Detection of Impulse Noise

Here, we design a simple statistical measure to detect impulse noise. Let x be the location of a pixel and $\Omega_x(N)$ the neighborhood of x, as defined in Eq. 3-1. Considering the case N = 1, we can get

$$\Omega_{\mathbf{x}}^{0} = \Omega_{\mathbf{x}}(1) / \{\mathbf{x}\} \qquad \textit{Eq. 3-8}$$

which represents the neighbor pixels of **x**. For each point $\mathbf{y} \in \Omega_{\mathbf{x}}^{0}$, we define the absolute difference in intensity of the pixels between **x** and **y**; i.e.,

$$d_{\mathrm{x,y}} = \left| u_{\mathrm{x}} - u_{\mathrm{y}} \right| \qquad \textit{Eq. 3-9}$$

$u_{\mathrm{x}}, u_{\mathrm{y}} \in [0, 255]$ for 8-bit pixel value

We define the total absolute difference (TAD) as

$$TAD(\mathbf{x}) = \sum_{y \in \Omega_{\mathbf{x}}^{0}} d_{\mathbf{x,y}} \qquad \textit{Eq. 3-10}$$

The TAD statistic provides us a measure of intensity similarity between the center pixel and its neighboring pixels. Figure 3-8 shows examples from the Lena image, in which we compare the neighborhood of an impulse noise pixel with the neighborhood of an edge pixel.

Figure 3-8 An impulse noise pixel (upper side) and a typical edge pixel (right side). TAD of impulse =863; TAD of edge pixel = 88

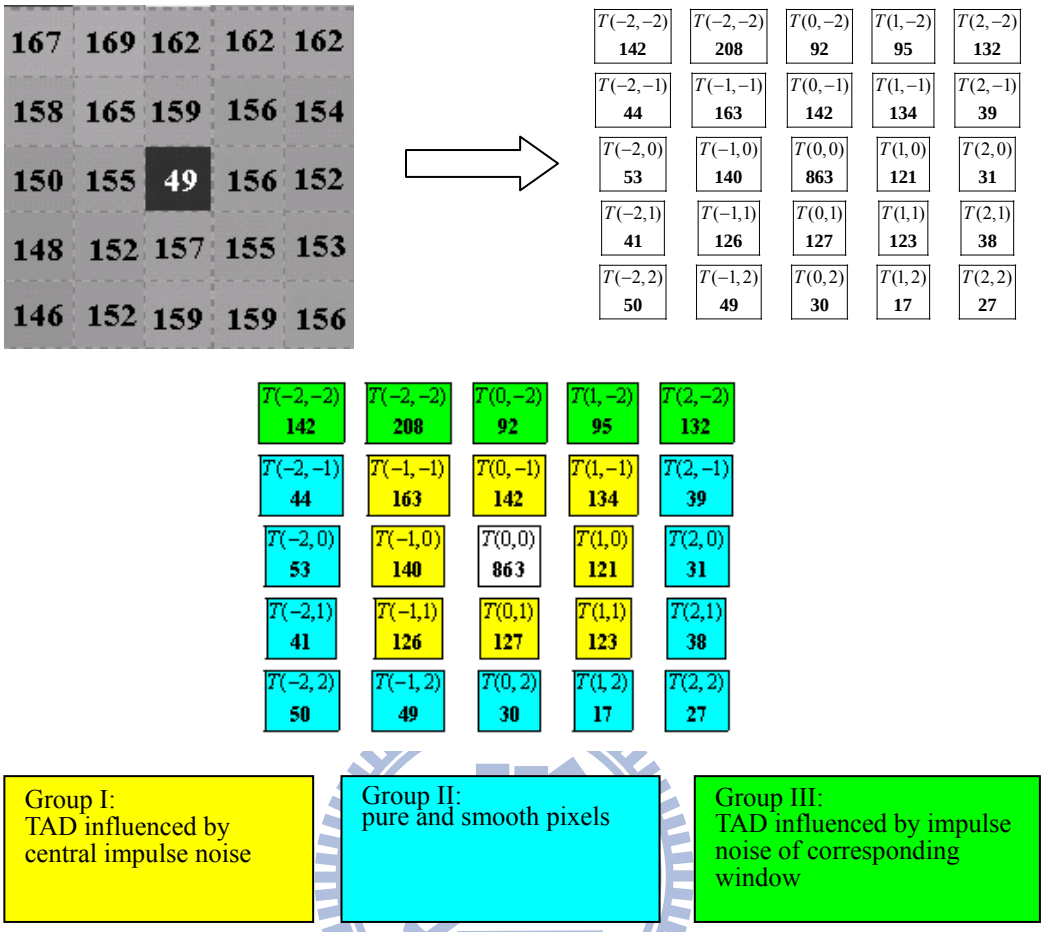


Figure 3-9 An 5×5 window with impulse noise on the center pixel, together with the corresponding TAD distribution

Figure 3-9 shows examples from the Lena image, with an impulse noise at the center pixel. We can find that the TAD value of the center pixel is obviously bigger than the TAD value of the pure edge pixel. To confirm that the TAD statistic is a good detector for impulse noise, we check the details of the TAD distribution in Figure 3-9. The center pixel T(0,0) is the only impulse noise in this 5×5 block, and the TAD of the 8 neighboring pixels (group I in Figure 3-9) are correlated with the center noisy pixel. The range of these values distribute over the range [120,165], while the range of outer pure pixels (group II in Figure 3-9) distribute in [15, 55]. Both groups have a tremendous difference with respect to the TAD value of impulse pixel. If we set a proper threshold value of TAD, the TAD difference between Group I and Group II would be negligible. Group III can be considered to have a similar distribution as Group I, because their TAD values are correlated with noisy pixels. Figure 3-10 shows the result of impulse noise detection in three color space.

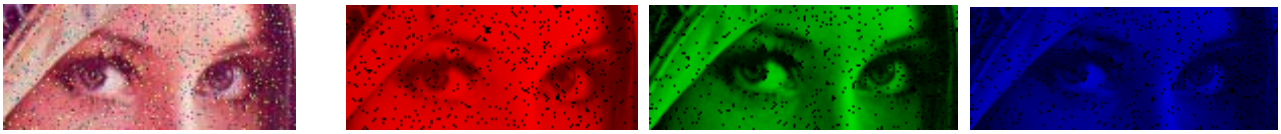


Figure 3-10 Impulse noise detection by TAD

3.3.3 Reduction of Impulse Noise

In this section, we utilize the TAD statistic and combine it with the original bilateral filter to develop our noise reduction algorithm. Here, we propose the third weighting function to indicate how likely an image pixel possesses an impulse noise. This weighting function can be defined as the "impulsive weighting function":

$$w_I(\mathbf{x}) = e^{-\frac{\mathrm{TAD}(\mathbf{x})^2}{2\sigma_I^2}} \qquad \textit{Eq. 3-11}$$

The parameter σ_I determines the penalty of high TAD values for impulse noise.

We combine the impulsive weighting function with the original bilateral filter to form a new weighting function, which includes spatial, photometric and impulsive properties. However, the direct combination of impulsive weighting function and photometric weighting function is not appropriate. This is because the impulsive weighting function works well to remove strong impulse outliers, while the photometric weighting function is used to smooth weak impulses. These two weighting functions need to be combined in a proper way to handle not only strong impulses but also weak impulses.

Based on the method proposed in [13], we introduce an adaptive function to determine the how much to use the impulsive weighting and photometric weighting. Considering a central pixel **x**, and the neighbor pixels $\mathbf{y} \in \Omega$, we define the "adaptive weighting" ***T*** of **y** with respect to **x** as

$$T(\mathbf{x},\mathbf{y}) = 1 - e^{-\frac{\left(\frac{\mathrm{TAD}(\mathbf{x})+\mathrm{TAD}(\mathbf{y})}{2}\right)^2}{2\sigma_T^2}} \qquad \textit{Eq. 3-12}$$

The range of adaptive weighting is [0, 1]. The σ_T parameter controls the shape of the function. If **x** or **y** has a large TAD value with respect to σ_T, then $T(\mathbf{x},\mathbf{y}) \approx 1$. On the other hand, if neither pixel has large TAD value, then $T(\mathbf{x},\mathbf{y}) \approx 0$. For the property of adaptive weighting function, we define the final weighting function as

$$w(\mathbf{x},\mathbf{y}) = w_S(\mathbf{x},\mathbf{y}) w_P(\mathbf{x},\mathbf{y})^{1-T(\mathbf{x},\mathbf{y})} w_I(\mathbf{x},\mathbf{y})^{T(\mathbf{x},\mathbf{y})} \qquad \textit{Eq. 3-13}$$

The restored pixel can be computed as expressed below:

$$\tilde{u}_{\mathbf{x}} = \frac{\sum_{\mathbf{y}\in\Omega} w(\mathbf{x},\mathbf{y}) u_{\mathbf{y}}}{\sum_{\mathbf{y}\in\Omega} w(\mathbf{x},\mathbf{y})} \qquad \textit{Eq. 3-14}$$

The modified bilateral filter can remove Gaussian noise and impulse with choosing proper parameters σ_S, σ_P, σ_I, and σ_T. For the impulse level p<20%, we can get good performance by 3×3 window size and without iterative calculation. The selection of parameters and experiment results will be introduced in the following sections.

3.3.4 Selection of Parameters

As indicated in [14], for Gaussian noise removal, the optimal σ_S value is relatively insensitive to the optimal σ_P, and σ_P can be approximately equal to 1.7 × σ_n, where σ_n represents the standard deviation of the Gaussian noise. In [15], Immerkaer provides a fast method to estimate the σ_n of Gaussian noise in an image. The estimator is given by

$$\hat{\sigma}_N = \sqrt{\frac{\pi}{2}} \frac{1}{6(W-2)(H-2)} \sum_{\mathbf{x}} |(u(\mathbf{x}) * L)| \qquad \textit{Eq. 3-15}$$

$$L = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

W and H represent the weight and height of the image, and L denotes the Laplacian filter. To suppress impulse noise, we take $\sigma_S = 5$. Experiment results have shown that the proper value of σ_I is around 2400, and $\sigma_J = 0.1$ is a good choice for removing both impulse noise and Gaussian noise. For compression artifact reduction , similar parameters are selected to smooth dusty regions. The experimental results will be showed in the following chapter.

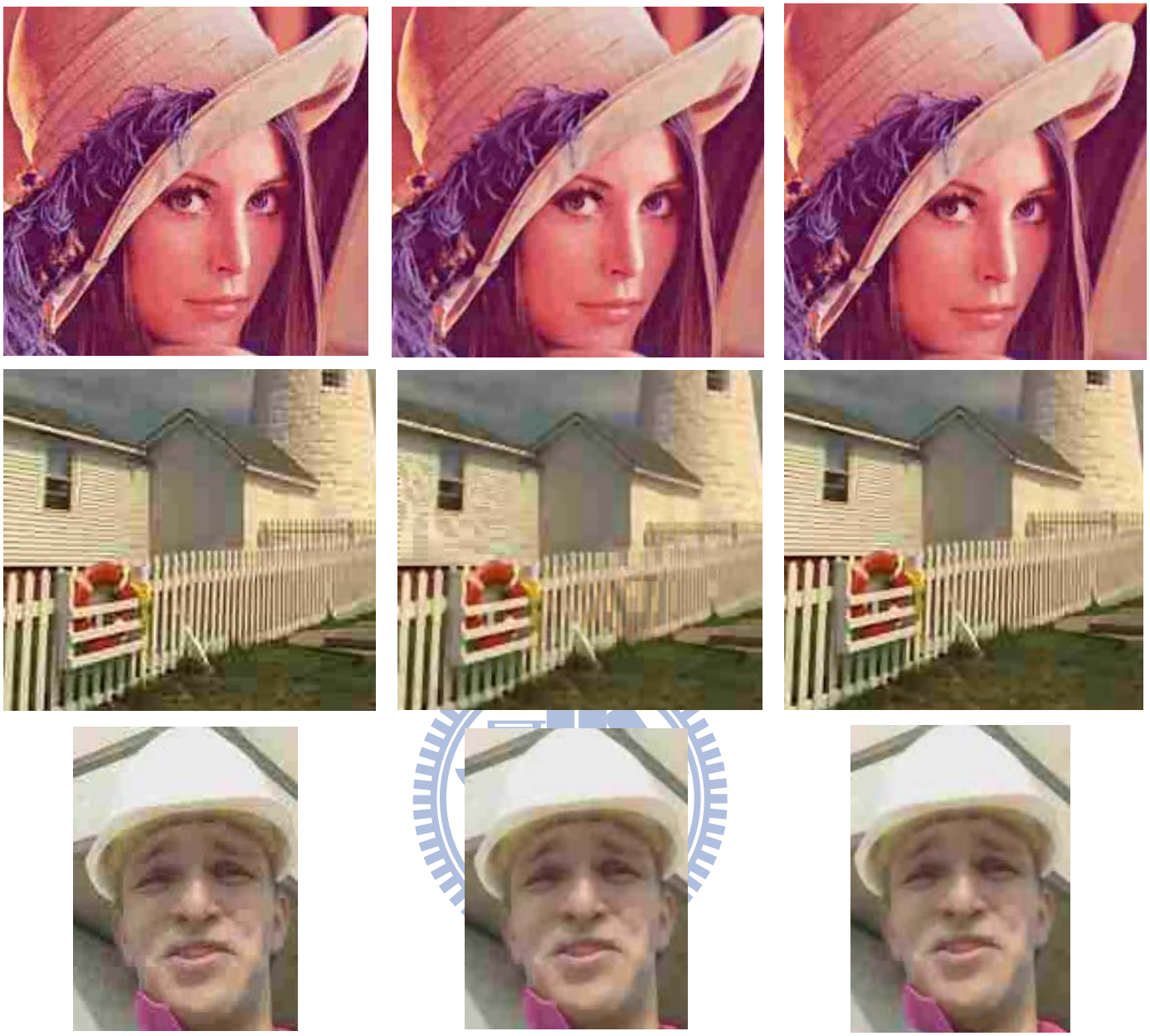
Chapter 4. Experimental Results

4.1 Reduction of Compression Artifacts

To evaluate the deblocking performance, we test several JPEG images with different quality factors ($QF \in [1,100]$), with the case $QF = 100$ indicating no quantization at all. We compared this method with the fuzzy filter proposed in [16] which is also a simple method with similar complexity. Both methods are implemented with a 3×3 filtering window. Figure 4-1 shows the experimental results and Table 4-1 lists the PSNR performance comparison. It can be seen the proposed method consistently outperforms the fuzzy filter for various images.

Table 4-1 Comparison of PSNR (in decibels) performance of deblocking algorithms applied on JPEG images (QF stands for quality factor)

Image (QF)	Input	Fuzzy Filter [16]	Proposed Method
Lena (20)	30.02 dB	30.33 dB	**30.42 dB**
Lena (15)	29.13 dB	29.50 dB	**29.56 dB**
Lena (10)	27.68 dB	28.06 dB	**28.08 dB**
Tower (20)	27.57 dB	25.07 dB	**27.45 dB**
Tower (15)	26.74 dB	24.86 dB	**26.95 dB**
Tower (10)	25.43 dB	24.12 dB	**25.73 dB**
Foreman (20)	30.49 dB	30.94 dB	**31.29 dB**
Foreman (15)	29.28 dB	29.82 dB	**30.05 dB**
Foreman (10)	27.79 dB	28.43 dB	**28.50 dB**



(a) (b) (c)

Figure 4-1 Compression artifact reduction examples (a) JPEG images with QF = 15, deblocking results for the (b) fuzzy filter [16] and (c) proposed compression artifact reduction filter

4.1.1 Blocking Strength

A measure of the blocking visual strength has been proposed in [17]. The blocking strength indicates the ratio of the absolute gradient over the average gradient calculated by adjacent pixels. Assume we have an image $\boldsymbol{I}$ with elements $\boldsymbol{Y}_{i,j}$, where i an j denote the pixel and line position, respectively. The normalized horizontal gradient $D_{H,norm}$ calculated over N adjacent pixels is defined as

$$D_{H,norm}(i,j) = \frac{\left|Y_{i+1,j} - Y_{i,j}\right|}{\frac{1}{2N}\sum_{n=-N\ldots N, n\neq 0}\left|Y_{i+n+1,j} - Y_{i+n,j}\right|} \qquad \textit{Eq. 4-1}$$

With $D_{H,norm}$, we further define

$$S_H(i) = \sum_{j=1}^{nl} D_{H,norm}(i,j) \qquad \textit{Eq. 4-2}$$

as the summation of $D_{H,norm}$ of all image lines. Figure 4-2 shows an example of S_H.

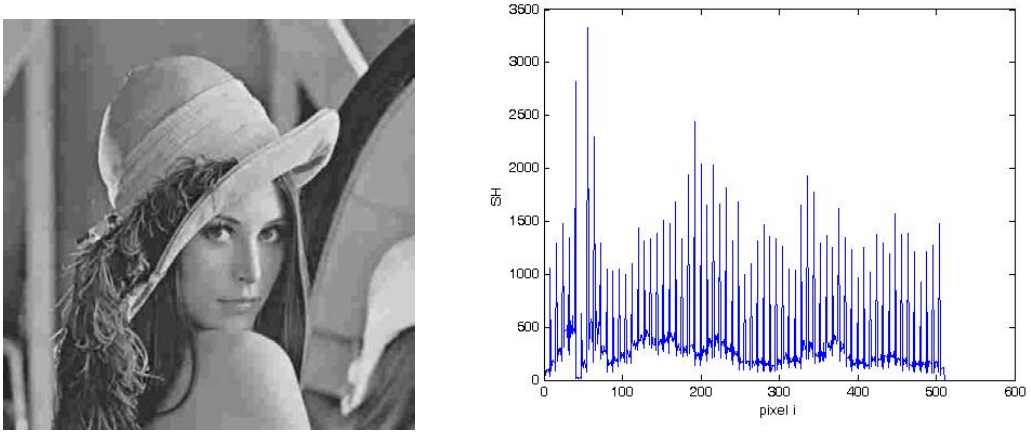


Figure 4-2 An impulse noise pixel (upper side) and a typical edge pixel (right side). TAD of impulse =863; TAD of edge pixel = 88

The periodic structure of the encoding grid is clearly revealed in the S_H, as shown in Figure 4-2. The blocking strength (BS) for the whole frame is then defined as

$$BS_H = \frac{\overline{S}_H(block)}{\overline{S}_H(non-block)}$$

$$BS_V = \frac{\overline{S}_V(block)}{\overline{S}_V(non-block)} \qquad \text{Eq. 4-3}$$

$$BS = \frac{1}{2}(BS_H + BS_V)$$

where $\overline{S}_H(block)$ and $\overline{S}_H(non-block)$ denote the average value of SH at the block edge and intermediate positions, respectively. Similar definition is utilized for vertical block strength ($\overline{S}_V(block)$ and $\overline{S}_V(non-block)$). Total block strength (BS) is the average of horizontal and vertical strength. Table 4-2 Comparison of blocking strength (BS) of deblocking algorithms applied on JPEG images (QF stands for quality factor) shows the block strength of the experimental results.

Table 4-2 Comparison of blocking strength (BS) of deblocking algorithms applied on JPEG images (QF stands for quality factor)

Image (QF)	Input	Proposed Method	Fuzzy Filter
Lena (20)	1.92	**1.48**	1.65
Lena (15)	2.46	**1.47**	1.62
Lena (10)	3.71	**1.51**	1.95
Tower (20)	2.06	**1.67**	1.94
Tower (15)	2.66	**1.57**	1.97
Tower (10)	4.15	**1.77**	2.49
Foreman (20)	3.38	**1.29**	1.46
Foreman (15)	4.29	**1.39**	1.54
Foreman (10)	5.85	**1.65**	1.88

4.1.2 Compression Artifact Reduction for Videos

The following images show compression artifact reduction for video frames. We can find that not only blockiness but also ringing artifact can be removed by our method.

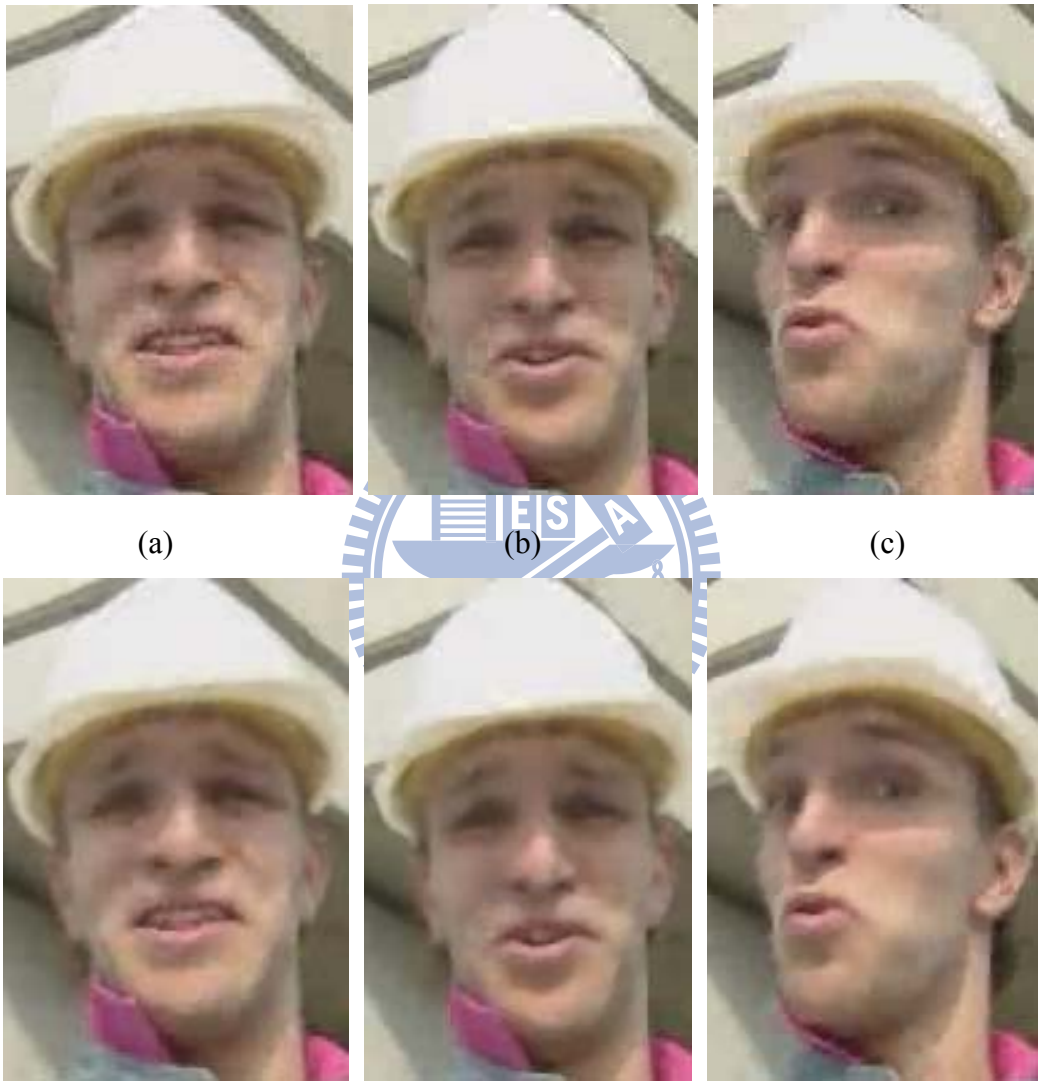


Figure 4-3 The compression artifact reduction results for video frames
(a)(b)(c) : original frames (d)(e)(f): deblocked frames

Figure 4-4 The compression artifact reduction results for video frames

(left : original frames right: deblocked frames)

Figure 4-5 The compression artifact reduction results for video frames
(left : original frames right: deblocked frames)

4.2 Reduction of Gaussian Noise and Impulse Noise

We have extensively tested the noise removal capabilities of the proposed method with respect to a few other filters. The following images show the experimental results for the removal of impulse noise, Gaussian noise and mixed noise patterns.

Figure 4-6 Comparing proposed method and median filter on images with Gaussian noise and mixed Gaussian and impulse noise

Table 4-3 Performance of applying different filters to images corrupted with Gaussian noise, impulse noise and mixed noise (PSNR in decibels)

Method	Lena Image			Pepper Image		
	p=20%	σ=10	p=20%, σ=10	p=20%	σ=10	p=20%, σ=10
3×3 median filter	30.68 dB	31.41 dB	28.83 dB	29.51 dB	30.33 dB	28.14 dB
5×5 median filter	29.34 dB	28.91 dB	28.53 dB	28.89 dB	29.58 dB	28.21 dB
3×3 ROAD Trilateral Filter [13]	32.98 dB	28.80 dB	28.87 dB	31.99 dB	28.98 dB	28.37 dB
Proposed Adaptive Bilateral Filter	31.08 dB	31.93 dB	29.70 dB	29.28 dB	30.27 dB	28.26 dB

From the results showed in Figure 4-6, we verified that the proposed adaptive bilateral filter may not only retain the ability to remove Gaussian noise but also can effectively remove mixed noise. Furthermore, another comparison between the proposed method and the ROAD trilateral filter [13] in listed in Table 4-3, where we can find that ROAD trilateral filter has better performance for impulse noise removal. In fact, the main difference between ROAD trilateral filter and the proposed method is their ROAD statistic and our TAD measure. The calculation of ROAD requires rank statistic, which spends more computations than our method. Table 4-4 shows the comparison of computation time between these two methods. Basically, our method has similar performance with respect to the ROAD trilateral filter, but with a lighter computational load.

Table 4-4 The average time cost of two methods with different window size (The implementation environment used Matlab platform with 2.10GHz AMD Athlon™ 64X2 Dual Core 4000+)

Method	512×512 Lena Image	512×512 Pepper Image
3×3 ROAD Trilateral Filter	4.98 sec	5.01 sec
3×3 Proposed Adaptive Bilateral Filter	**3.76 sec**	**3.87 sec**
5×5 ROAD Trilateral Filter	13.86 sec	13.90 sec
5×5 Proposed Adaptive Bilateral Filter	**9.80 sec**	**9.83 sec**

4.2.1 Reduction of Impulse Noise for Videos

The following images show impulse noise reduction for video frames. We can find that our proposed method has excellent performance.

Figure 4-7 The impulse noise reduction results for video frames (p =10% for RGB color channels)

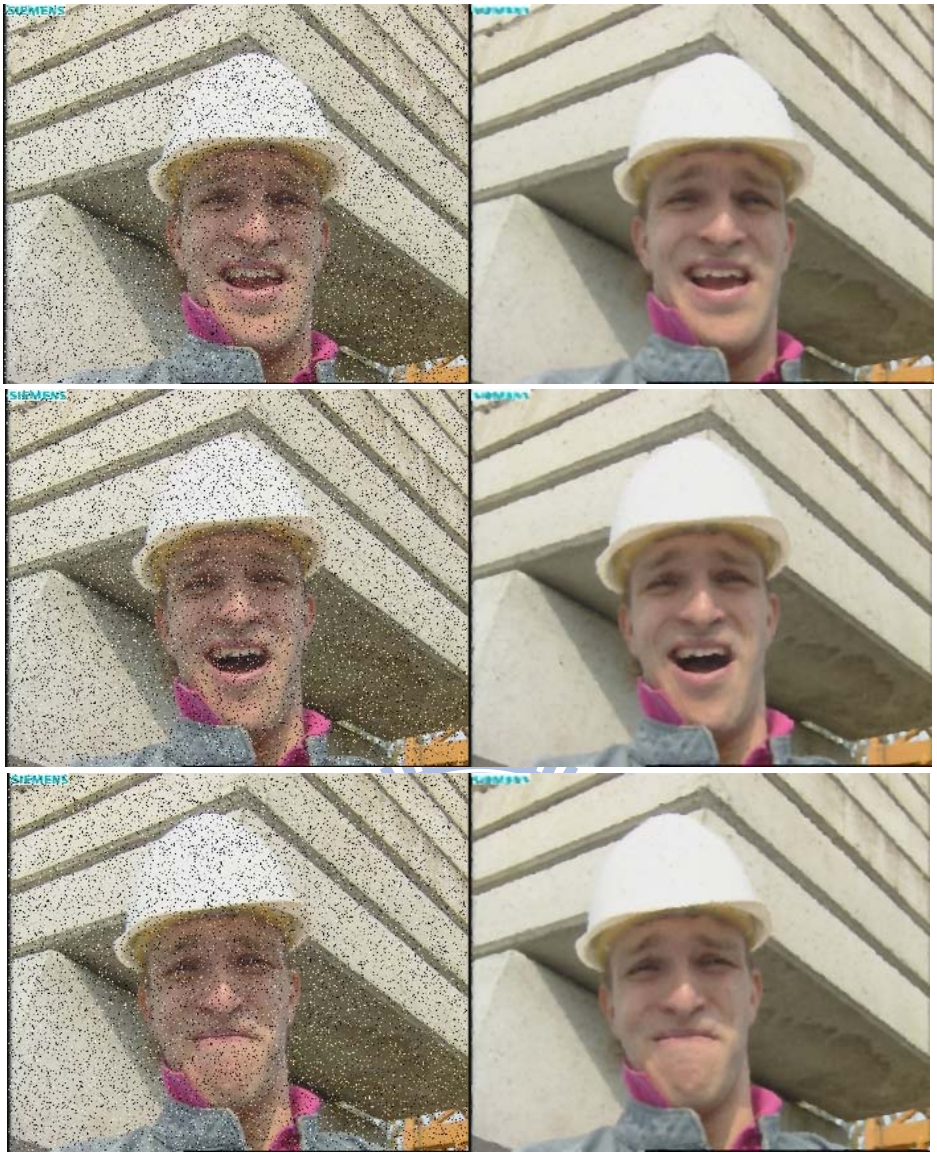


Figure 4-8 The impulse noise reduction results for video frames (p =15% for gray channel)

Chapter 5. Conclusions

In this thesis, we proposed several algorithms to remove Gaussian noise, impulse noise, and compression artifact. A simple compression artifact detector, which utilized local deviation and proper thresholds, can indicate the locations of perceptual blockiness. The blocking artifact is suppressed by the proposed noise-injection based deblocking filter. Furthermore, based on the bilateral filter, we add in the TAD measure to take into account the strength of impulse noise. The proposed modified bilateral filter has nice performance in smoothing Gaussian noise and impulse noise while may preserve fine-detail regions in the images. Without requiring the computation of rank statistic, the computational load of our method is lighter than the ROAD trilateral filter. The experiment results show that the proposed method can efficiently remove compression artifacts, Gaussian noise, and impulse noise.

Reference

[1] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *Pattern Analysis and Machine Intelligence, Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 12, pp. 629-639, 1990.

[2] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," *Sixth International Conference on Computer Vision (IEEE Cat.no.98CH36271),* pp. 839; 839-846; 846, 1998.

[3] H. -. Lin and A. N. Willson Jr., "Median filters with adaptive length," *IEEE Transactions on Circuits and Systems,* vol. 35, pp. 675-690, 1988.

[4] A. Bovik, T. Huang and D. Munson Jr., "A generalization of median filtering using linear combinations of order statistics," *IEEE Transactions on Acoustics, Speech and Signal Processing,* vol. 31, pp. 1342-1350, 1983.

[5] R. C. Hardie and K. E. Barner, "Rank conditioned rank selection filters for signal restoration," *IEEE Transactions on Image Processing,* vol. 3, pp. 192-206, 1994.

[6] Gouchol Pok, Jyh-Charn Liu and A. S. Nair, "Selective removal of impulse noise based on homogeneity level information," *IEEE Transactions on Image Processing,* vol. 12, pp. 85-92, 2003.

[7] T. Sun and Y. Neuvo, "Detail-preserving median based filters in image processing," *Pattern Recog. Lett.,* vol. 15, pp. 341-347, 4. 1994.

[8] H. Reeve III and Jae Lim, "Reduction of blocking effect in image coding," *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '83.,* vol. 8, pp. 1212-1215, 1983.

[9] Sung Deuk Kim, Jaeyoun Yi, Hyun Mun Kim and Jong Beom Ra, "A deblocking filter with two separate modes in block-based video coding," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 9, pp. 156-160, 1999.

[10] T. Jarske, P. Haavisto and I. Defee, "Post filtering methods for reducing blocking effects from coded images," *IEEE Transactions on Consumer Electronics,* vol. 40, pp. 521-526, 1994.

[11] Hao-Song Kong, Yao Nie, A. Vetro, Huifing Sun and K. E. Barner, "Coding artifacts reduction using edge map guided adaptive and fuzzy filtering," *2004. ICME '04. 2004 IEEE International Conference on Multimedia and Expo,* vol. 2, pp. 1135-1138 Vol.2, 2004.

[12] R.C. Gonzalez and R. E. Woods, *Digital image processing*. Reading, MA: Addison-Wesley, 2002.

[13] R. Garnett, T. Huegerich, C. Chui and W. J. He, "A universal noise removal

algorithm with an impulse detector," *IEEE Trans. Image Process.,* vol. 14, pp. 1747-1754, NOV. 2005.

[14] Ming Zhang and B. Gunturk, "A new image denoising method based on the bilateral filter," *2008. ICASSP 2008. IEEE International Conference on Acoustics, Speech and Signal Processing,* pp. 929-932, 2008.

[15] J. Immerkaer, "Fast noise variance estimation," *Comput. Vision Image Understanding,* vol. 64, pp. 300-302, SEP. 1996.

[16] Y. Nie and K. E. Barner, "Fuzzy transformation and its applications," *ICIP 2003. Proceedings. 2003 International Conference on Image Processing,* vol. 1; 1, pp. I-893-6 vol.1, 2003.

[17] L. Shao, J. Wang, I. Kirenko and G. de Haan, "Quality adaptive trained filters for compression artifacts removal," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing, Vols 1-12,* pp. 897-900, 2008.