國立交通大學

資訊科學研究所

碩士論文

利用區塊間冗餘資料提高 JPEG 2000 影像壓縮效能之研究

A Study of Using Inter-Block Redundancy to Improve JPEG 2000 Compression Performance

研 究 生：王聖博

指導教授：薛元澤　教授

中華民國 九十三 年六月

利用區塊間冗餘資料提高 JPEG 2000 影像壓縮效能之研究

A Study of Using Inter-Block Redundancy to Improve JPEG 2000 Compression Performance

研究生：王聖博　　Student：Sent-Po Wang

指導教授：薛元澤　　Advisor：Yuang-Cheh Hsueh

國立交通大學
資訊科學研究所
碩士論文

A Thesis
Submitted to Institute of Computer and Information Science
College of Electrical Engineering and Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Computer and Information Science

June 2004

Hsinchu, Taiwan, Republic of China

中華民國九十三年六月

利用區塊間冗餘資料提高 JPEG 2000 影像壓縮效能之研究

學生：王聖博　　　　　　　　　　　　　　　　　指導教授：薛元澤

國立交通大學資訊科學學系（研究所）碩士班

摘要

近年來，隨著數位手機的普及，影像檔案也必須越小越好以達到更好的傳輸或儲存效率。在此應用環境之下，如何將影像的儲存空間做進一步的壓縮就成為一個日漸重要的課題。本研究所提出的方法利用區塊間的冗餘資料對影像做進一步的壓縮。實作上，在壓縮端以快速的方法判斷出適合被捨棄的區塊，而在解壓縮端利用影像區塊填補的技術，完整地補回被捨棄的區塊。藉此我們便可以達到進一步壓縮影像檔案的目的。實驗顯示，在低倍率的壓縮時我們的方法確實可以節省許多儲存空間，同時維持了良好的影像品質，而在高倍率的壓縮，我們的方法又能保持比純 JPEG 2000 更清析銳利的影像品質。

A Study of Using Inter-Block Redundancy to Improve JPEG 2000 Compression Performance

student：Sent-Po Wang　　Advisors：Dr.Yuang-cheh Hsueh

Institute of Computer and Information Science
National Chiao Tung University

ABSTRACT

In recent year, with the mobile phone with integrated digital camera becoming more and more popular, the size of image file is required smaller to achieve better storage or transmission efficiency. In order to fit such kind of application environments, how to compress the image file to a smaller size has become a more and more important topic. The method proposed in this paper utilizes the inter-block redundancy to further compress an image by a fast algorithm to drop redundant blocks in the encoder side and to utilize our block filling method to fill them back in the decoder. In this way, we can achieve the goal of further compressing an image file. Experimental results indicate that in high compression ratio the proposed method can indeed improve the compression ratio and meanwhile maintain a proper visual quality and on the other hand in low compression ratio the method is capable to maintain a finer image quality compared to pure JPEG 2000 standard.

誌謝

我在這裡要感謝我的指導教授 薛元澤教授，兩年來對我孜孜不倦的教誨，教導我研究學問的方法及待人處世道理，讓我畢生受益無窮，以及我的口試委員 張隆紋教授與 莊仁輝教授，二位老師不吝指教，改正我的缺點，讓這篇論文更加完善。

我還要感謝詹森仁學長、林瑞盛學長、林柏青學長、吳昭賢學長及張尹彬學長，給予我論文研究及生涯規劃等的各種建議，感謝石永靖同學、蔡盛同同學在這兩年內與我共同努力，互相砥礪。另外，我要感謝莊逢軒學弟、何昌憲學弟、王蕙綾學妹及高薇婷學妹陪我度過這段快樂的實驗室生活。

僅將此論文獻給我親愛的的家人與朋友，我的父母及妹妹，和我的女友 婉雯，感謝他們在這段期間給我的關心、支持與鼓勵，祝福他們永遠健康快樂。

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

1.1 Motivation

In recent year, with the mobile phone with integrated digital camera becoming more and more popular, the size of image file is required smaller to achieve better storage or transmission efficiency. In order to fit this kind of application environments, how to compress the image file to a smaller size has become a more and more important topic.

1.2 Introduction

The goal of this thesis is to propose a scheme imbedded in the latest JPEG 2000 standard architecture which utilizes block filling methods to remove inter-block redundancy in an image in order to improve the compression ratio and maintains a sharp visual quality in fine textural regions.

1.2.1 Inter-Block Redundancy

Like the famous inter-pixel redundancy, the inter-block redundancy is defined the redundancy between blocks. If one block is highly correlated with its neighbors, this block is defined as “redundant” and can be safely dropped.

While removing inter-block redundancy to compress an image, we first drop redundant blocks by a determining function. And then when decompressing, try to perfectly fill-in the dropped blocks by referring to its neighbors. As a result, the compression ratio and reconstruction quality highly depend on the performances of block dropping and filling functions.

1.2.2 Introduction to Our Scheme

Embedded in the JPEG 2000 standard, our method is based on wavelet transform and subband coding technique. Two important parts of our method are block dropping and block filling. In order to safely remove inter-block redundancy as more as possible, two matters are concerned by the image dropping method: the similarity of the target block to its neighbors and whether or not the target block contains unique objects that cannot be properly reconstructed.

Our filling method is to fill-in the dropped blocks with particular method in each subband of the image. Each method is just suitable for the specific subband image. It is proved by the experiments that our scheme is capable to precisely choose proper blocks to drop and fill them back with sharp image quality when comparing to pure JPEG 2000 standard.

1.2.3 Advantages of Our Scheme

As we can see, when we compress an image file with the JPEG standard which is commonly used in various domains, we first segment the image into blocks with the size of 8x8 pixels, and then we try to remove in-block redundant information to compress the image to a smaller file size. Therefore, when we need to further compress an image file to an extremely small size, we may suffer a serious distortion over the whole image.

On the other hand, in JPEG 2000 standard, the multiresolution process is adopted. The image is decomposed into subbands and coded respectively. The JPEG 2000 standard reaches a high compression ratio to about 100:1 and still maintains an acceptable image quality. But when the tradeoff between compression ratio and image quality is considered, at extremely low bit-rate, the whole image is also going to suffer a universal blurring.

In low bit-rate compression, though our method is not capable to support great improvement in the bit rate, it is able to protect the fineness of the whole image On the other hand, when compressing an image in relatively high bit-rate, our method is capable to effectively further compress the target image to a lower bit-rate and at the mean time maintains a great image quality.

1.3 Previous Researches

This paper is inspired by [1], in which Rane et al. proposed an image filling-in method utilized in the wireless transmission and image compression field. They also indicated that the image filling method is capable of being utilized in image compression based on JPEG standard and proposed a set of block dropping rules for textural and structural blocks respectively.

After that, in [2] Bertalmio et al. proposed an image decomposition method as the preprocessing and fill-in blocks in textural channel and structural channel respectively.

The structure recovery method used in [1] and [2] is proposed in [3], Bertalmio et al. used a time varying estimation of the isophotes field to solve the image inpainting problem.

Li in [4] indicated that the filling order plays an important role on the quality of the filling-in result and only one filling order is not enough. He also proposed a fuzzy interpolation method.

Grgic et al. presented the reason why we choose wavelet to compress an image in [5], and in [6], Marcellin et al. made a brief introduction on JPEG 2000 and a comparison with JPEG.

1.4 Organization of this Thesis

In this thesis, an image filling method utilized to improve the JPEG 2000 standard is presented. The remainder of this thesis is organized as follows. In chapter 2, we will briefly introduce the concept of image compression, wavelet transformation and multiresolution process, subband coding and the JPEG 2000 image compression standard. And then also in chapter 2 we will survey the previous research of image filling method.

In chapter 3 we will present the location of our work in the architecture of JPEG 2000 image compression system and introduce our scheme, including the reason why we apply image decomposition process, our block dropping method and our filling-in method used on each subband.

In chapter 4, we will describe the implementation of our method in detail, the filling-in result of our system and the comparison to pure JPEG 2000 standard. In chapter 5, the conclusion and future work will be stated.

CHAPTER 2

Background

2.1 Review of Digital Image Compression

In this section, we briefly describe the fundamental concepts of image compression including three redundancies and the tradeoff between the compression ratio and the quality of the reconstructed image.

2.1.1 Three Redundancies

Image compression addresses the problem of representing an image with less data. As a result, data redundancy becomes the main issue of digital image compression. In general, three data redundancies are utilized to compress an image file: coding redundancy, interpixel redundancy and psychovisual redundancy.

In utilizing coding redundancy, we use various encoding methods, variable length coding for instance, to remove redundant data in direct coding each pixel. The performance of variable length coding method depends on the distribution of the value on the histogram.

Talking about interpixel redundancy, we may utilize predictive coding or transform coding to collect the correlation between pixels in detail subbands, which facilitate the truncating or quantizing of the inter-pixel redundancy.

The psychovisual redundancy exists because human perception of the information in an image normally does not involve quantitative analysis of every pixel value in the image. The psychovisual redundancy is truncated by the quantizer, which truncates the tiny details which cannot be noticed by the human vision.

In the method proposed in this paper, we mainly utilize the inter-block redundancy briefly described in chapter 1 to further lower the bit rate of an image.

2.1.2 Image Compression Tradeoff

In lossy image compression scheme, the image compression algorithm should achieve a tradeoff between compression and image quality [7]. Obviously, if we need to compress an image file with an extremely high compression ratio, we need to remove some less important information from the original image. This makes the reconstructed image not precisely equal to the original image, but the “approximation” of it.

Therefore, we can conclude the aim of image compression is to achieve as high compression ratio as possible at a specific image quality, or as high image quality as possible at a specific compression ratio.

2.2 Image Compression Measurements

Three important measurements used to judge if an image compression method is good or not are compression ratio, visual quality and time efficiency. However there is a tradeoff between them, once we need to reach a high image quality, the cost of space must get high, however, on the contrast, once we need to compress an image into an extremely small size, we must suffer a serious distortion. Therefore, how to fit the different requirements and reach a balanced tradeoff becomes the main issue of image compression.

2.2.1 Visual Quality Measurements

The fidelity criteria are to count the difference between the original image and the compressed one. Two methods utilized in this paper are RMSE (root mean square error) and PSNR (peak signal-to-noise ratio). As implied in the name, the RMSE is the rooted value of the mean square error between two images. For example, the RMSE between the reconstructed images $\hat{f}(x,y)$ and the original image $f(x,y)$ is as follows, where both images are of size *M* x *N*.

$$RMSE = [\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}[\hat{f}(x,y) - f(x,y)]^2]^{1/2}$$

The basic idea of PSNR is to compute a single number that represents the quality of the reconstructed image. The reconstructed images with higher PSNR are judged as having better visual quality. PSNR in decibels (dB) is computed by the equation shown below.

$$PSNR = 20\log_{10}(\frac{255}{RMSE})$$

Typical PSNR values range between 20 and 40. However, each particular value is of no meaning, but the comparison between two values for different reconstructed images gives one measure of quality.

M. Ardito and M. Visca proposed that the PSNR is not adequate as a perceptually meaningful measure of picture quality, because the reconstruction errors in general do not have the character of signal-independent additive noise, and the seriousness of the impairments cannot be measured by a simple power measurement [8].

In [5], the Grgic et al. utilize a perception-based subjective evaluation quantified by the mean opinion (MOS) [9] and a perception-based objective evaluation quantified by the picture quality scale (PQS) [10] as the additional image quality measurements to PSNR. However, the two methodologies of image quality measurement involve the human objective testing, which is not properly to be applied to our work.

2.2.2 Compression Ratio Measurements

Compression ratio is a measurement of the space efficiency of a compression method. Let C_R denote the compression ratio. Then,

$$C_R = \frac{n_0}{n_1}$$

where n_0 denotes the number of bytes used to present the original image, and n_1 denotes the number of bytes used to present the compressed image. When C_R=1, it means the original image file contains no redundant data, therefore no storage space is saved by the compressor. On the contrary, the smaller C_R is, the more redundant data in the original image has been removed. In other words, the compressed image with high compression ratio is judged as having great spatial efficiency.

Another measurement called bit rate (bpp: bit per pixel) is also frequently used to present the space efficiency of an image compression method.

$$B_R = \frac{n_1}{M \times N}$$

2.3 Wavelet and Multiresolution Processing

The main purpose of multiresolution is to represent and analyze the image in various resolutions by reason of that some information or feature reveals only in particular resolution. As applied in image compression field, the multiresolution process is usually combined with transform coding, such as wavelet transform.

2.3.1 Image Pyramids

Image pyramid is a conceptually simple structure to represent an image in different resolutions. Due to its gradualness, it is suitable for use of network transformation of images. With this construction, the user is able to derive an image with gradually high resolution but with spatially growing, which helps to recognize the image when only small proportion of the whole image is received.

The image pyramid structure is illustrated in Fig. 2-1 [11].

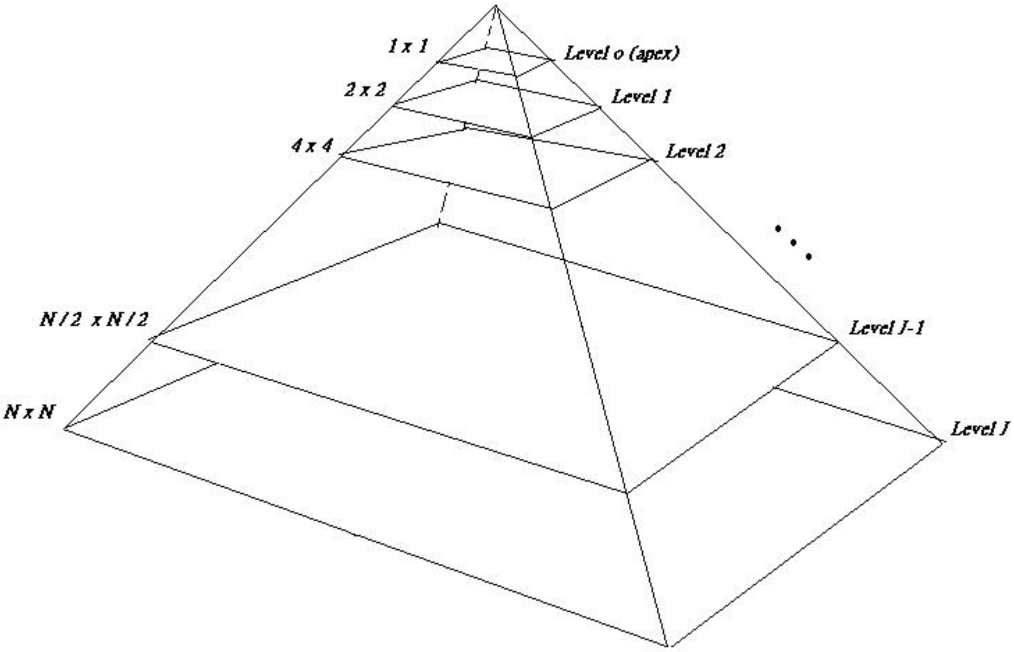


Fig. 2-1 Pyramidal image structure.

The system block diagram of image pyramid decomposition and synthesis is illustrated in Fig 2-2 [11]. The main concept is to calculate the difference between the approximation and the target image by prediction coding or transform coding method. And by applying the down-sampling process to the difference, we can obtain the result of image analysis. On the other hand, by applying up-sampling and interpolation to the subimages we can obtain the result of image synthesis process.

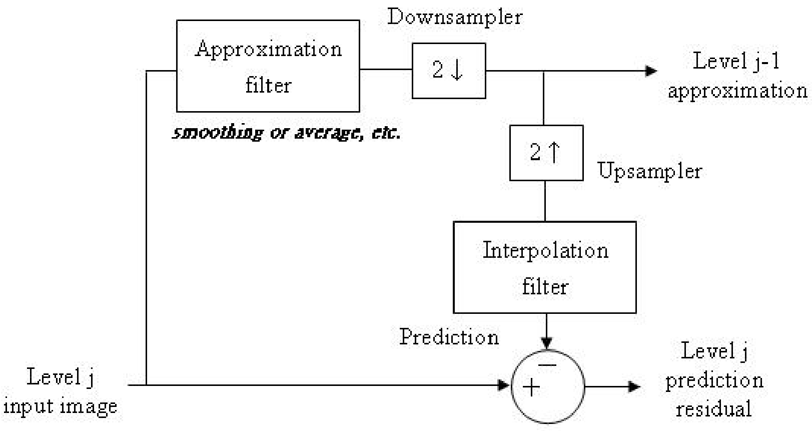


Fig. 2-2 System block diagram for creating an image pyramid.

The main advantage of multiresolution methodology utilized in image compression is that the values in differences e_1, e_2, …, e_{n-1}, e_n represent the correlation between pixels well. In a smooth image, the prediction residual will gather in a small region surrounding zero. As described in previous sections, such kind of value distribution facilitates the quantizer the variable-length symbol encoder to compress efficiently.

The efficiency of storage and concurrency of the reconstructed image depend on the approximation filter in the multiresolution decomposition system. There are various approximation filters to be utilized in the system, including neighborhood averaging for producing a mean pyramid, lowpass Gaussian filtering for producing a Gaussian pyramid, or no filtering for producing a subsampling pyramid.

2.3.2 Subband coding and Discrete Wavelet Transform

In 1987, wavelets were first utilized as a powerful new approach to signal processing and analysis called multiresolution theory (Mallat [1987]) [12]. In our works, multiresolution helps decomposing an image into subbands for us to fill-in dropped blocks in each subband with appropriate methods. Besides, in [5], Grgic et al. proposed that in recent years, much of the research activities in image coding have been focused on the DWT, which has become a standard tool in image compression applications because of their data reduction capability [13]-[15].

Compared to the block-wise DCT (discrete cosine transform), which is broadly used in standards for compression of still (e.g., JPEG [16]) and moving images (e.g., MPEG-1 [17], MPEG-2 [18]), DWT is to transform the entire image into frequency domain and offers adaptive spatial-frequency resolution (better spatial resolution at high frequencies and better frequency resolution at low frequencies) [5], that is well suited to the properties of a human visual system.

In multiresolution process, the image in each level is decomposed into a set of band-limited components, where each one is called a subband of the signal. If without quantized, the decomposition of an image is reversible, which means we can reconstruct the original image with the set of subbands.

However, in image compression field, since the coefficients in high frequency subbands usually gathers in narrow region beside zero, we can utilize the coding methods and the quantizer to raise the compression ratio.

Fig. 2-3 illustrates the subband image coding process.

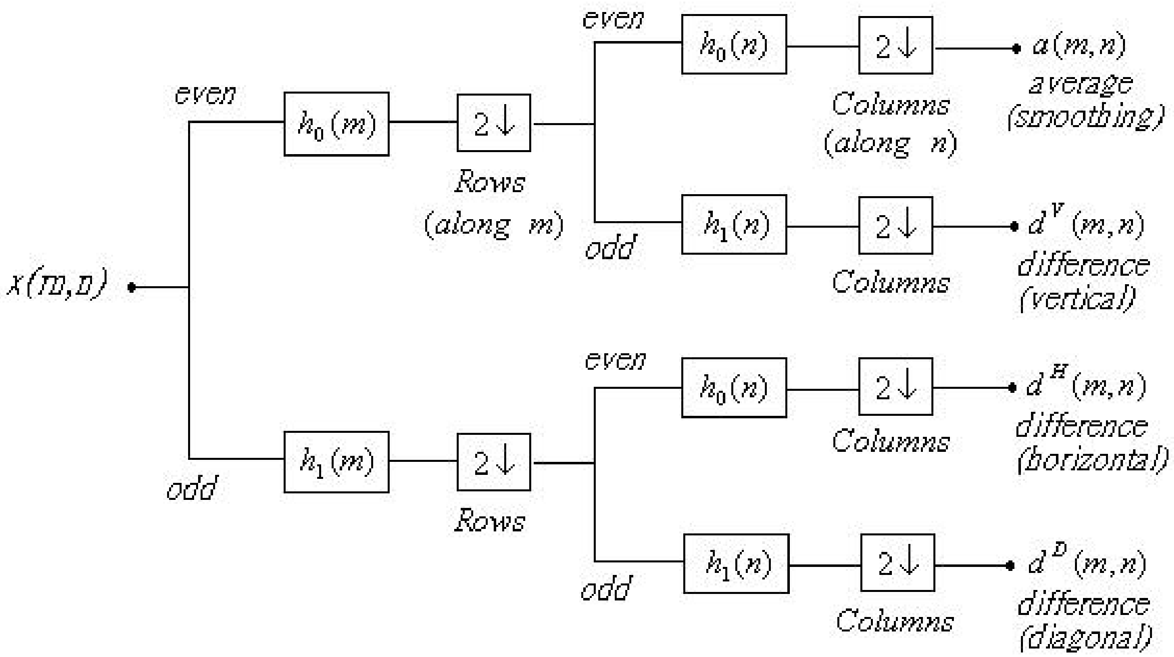


Fig. 2-3 Two-dimensional four-band filter band for subband image coding.

The discrete wavelet transform represents a signal in terms of shifts and dilations of a low-pass scaling function $\phi(t)$ and a bandpass wavelet function $\psi(t)$. The result of wavelet transform is a set of wavelet coefficients, which measure the contribution of the wavelets at these locations and scales [5]. The wavelet representation is efficient because images are often modeled as a set of locally smooth regions separated by edges.

2.3.3 Wavelet Transform in JPEG 2000 Standard

Two wavelet transform functions must be implemented in every compliant JPEG 2000 decompressor are Spline 5/3 transform and CDF 9/7 transform.

As described in [19], the remarkable simplicity of Spline 3/5 recommends it for practical image compression applications. The analysis and synthesis prototype sequence of Spline 5/3 transform are given by

$$\begin{pmatrix} m_0(z) \\ \tilde{m}_0(z) \end{pmatrix} = \begin{pmatrix} \frac{1}{4}z^{-1} + \frac{1}{2} + \frac{1}{4}z \\ -\frac{1}{8}z^{-2} + \frac{1}{4}z^{-1} + \frac{3}{4} + \frac{1}{4}z - \frac{1}{8}z^{2} \end{pmatrix}.$$

On the other hand, the CDF 9/7 transform has been found to yield optimal or near optimal performance in image compression application and has enjoyed widespread popularity in the image compression community. In practical, we apply the following coefficients as the wavelet transform basis.

	Analysis	
i	$h_0(i)$	$h_1(i)$
-4	0.02674875741080976	
-3	-0.01686411844287495	0.09127176311424948
-2	-0.07822326652898785	-0.05754352622849957
-1	0.2668641184428723	-0.5912717631142470
0	0.6029490182363579	1.115087052456994
1	0.2668641184428723	-0.5912717631142470
2	-0.07822326652898785	-0.05754352622849957
3	-0.01686411844287495	0.09127176311424948
4	0.02674875741080976	

Table 2-1 (a) Analysis coefficients in CDF wavelet transform.

Synthesis		
i	$g_0(i)$	$g_1(i)$
-4		0.02674875741080976
-3	0.09127176311424948	-0.01686411844287495
-2	-0.05754352622849957	-0.07822326652898785
-1	-0.5912717631142470	0.2668641184428723
0	1.115087052456994	0.6029490182363579
1	-0.5912717631142470	0.2668641184428723
2	-0.05754352622849957	-0.07822326652898785
3	0.09127176311424948	-0.01686411844287495
4		0.02674875741080976

(b) Synthesis coefficients in CDF wavelet transform.

2.4 JPEG 2000 Image Compression Standard

JPEG 2000 (Joint Photographic Experts Group) is the newest image compression standard so far which provides an excellent solution to the serious degradation problem in JPEG. However, JPEG 2000 is far more complex and also higher cost on memory consumption than JPEG. JPEG 2000 relies on the discrete wavelet transform, and EBCOT coding of wavelet coefficients in the blocks, which provide excellent improvement in the compression ratio.

2.4.1 JPEG 2000 Architecture

JPEG 2000 is composed of three main components, including wavelet transform, scalar quantization and EBCOT, as shown below in Fig. 2-4.

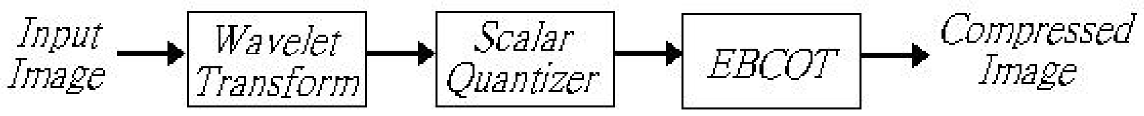


Fig. 2-4 JPEG 2000 block diagram.

Where the wavelet transform as introduced in the previous section, is to decompose the spatial correlation between pixels of the input image. And then the scalar quantizer quantizes the coefficients in each subband image.

EBCOT (Embedded Block Coding with Optimal Truncation) is similar in philosophy to the EZW and SPIHT algorithms. However, in EBCOT, each block resides entirely within a subband, and coded independent of other blocks, such scheme is just suitable for our method.

As described in [20], the adoption of independent embedded block coding for JPEG 2000 has significant benefits. The blocks may be coded and decoded parallel and in any order, which permits a degree of spatial random access to the image content, and facilitate the efficient implementation of geometric manipulation such as rotation; also it is appealing for applications requiring a degree of error resilient, since errors are confined spatially while the embedding lends itself to differential protection strategies. More notably, this method allows the users to arbitrarily select the contributions made by each code-block to each quality layer, which permits improvements in rate distortion performance which more than compensate for the cost of restarting the coding process at each block.

2.4.2 JPEG 2000 Features

JPEG 2000 has a long list of features, a subset of which are list below [6]:

- state-of- the-art low bit-rate compression performance.
- Progressive transmission by quality, resolution, component, or spatial locality.

- Lossy and lossless compression (with lossless decompression available naturally through all types of progression).
- Random (spatial) access to the bit stream.
- Pan and zoom (with decompression of only a subset of the compressed data)
- Compressed domain processing (e.g., rotation and cropping).
- Region of interest coding by progression.
- Limited memory implementations.

2.5 Block Filling

The block filling methods are widely used in various fields such as wireless communication, reverting deterioration of photographs and removal of undesired object from photographs, etc.; however, in our work we utilized the block filling to help raise the compression ratio in image compression field.

There are number of different methods in the literature related to block filling but mainly classified as three main groups. The first one mainly deals with the films; the second one is related to texture synthesis, the third one is related to structure reconstruction. In this section we introduce some of the methods in block filling domain.

2.5.1 Block Filling in Particular Applications

Different block filling methods are developed in different fields of applications. Take the method proposed in [21] for example, this method is designed for face reconstruction only, which is based on a database of face image. When reconstructing a mouth on a face, it first obtains the texture of a damaged face, and then reconstructs a full shape from the given incomplete shape which excludes damaged regions.

After that, it reconstructs a full texture from the obtained texture damaged by virtual objects and synthesizes a facial image by the reconstructed texture and the reconstructed shape. Finally it replaces the pixels in the damaged regions by the reconstructed ones and combines the original and the reconstructed image in the border regions outside of the damaged region using a weight mask according to the distance from the border.

This method requires a database of template images in the same kind with the damaged one, which limits the adaptation of this method in other fields of application. However, it still gives us inspiration with the linear combination method in texture synthesis field.

2.5.2 Restoration of Film

Kokaram et al. [22] proposed a method utilize motion estimation and autoregressive models to fill blocks in film. The basic idea is to refer to the neighbor frames, and obtain the information base on the motion estimation and fill them into the damaged block as shown in Fig. 2-5. However, this method cannot be applied to a still image.

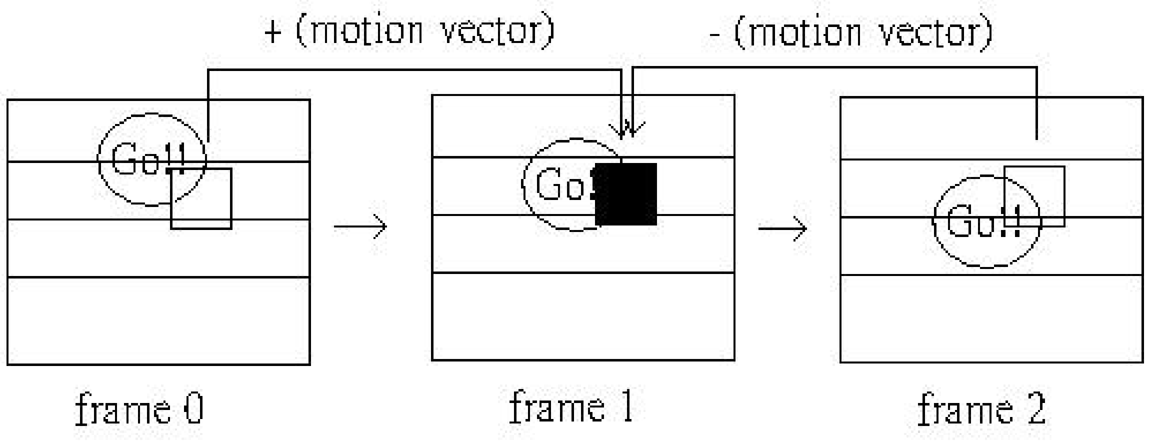


Fig. 2-5 Illustration pf film reconstructing.

2.5.3 Structure Recovery

In [1], Rane et al. refers to the method proposed by Marcelo Bertalmio et al. [3] in structure recovery. Briefly speaking, this algorithm is to progressively “shrink” the gap by prolonging the lines arriving at the gap boundary inward in a smooth way.

This prolongation is achieved by numerically solving the following partial differential equation below iteratively. Where ∇ stands for gradient, Δ stands for Laplacian and $\nabla^{\perp}$ stands for orthogonal-gradient (isophote direction).

$$\frac{\partial I}{\partial t} = \nabla(\Delta I) \bullet \nabla^{\perp} I$$

This method allows to simultaneously fill-in numerous regions containing different structures and surrounding backgrounds. However, this method is designed as a time variant algorithm; it result in excellent filling results when proceeded several thousands of time iteratively, however, it causes serious cost of processing time.

Beside of the image inpainting algorithm introduced above, Lee et al. [4] proposed some fuzzy interpolation algorithms to solve the block filling problem. The algorithms as illustrated below in Fig.2-6 extract the local self-similarity properties with a window scanning in an image and use these properties to interpolate pixels in the current window simultaneously.

This method results in an excellent interpolation result, but in our work, the blocks are densely dropped over the input image, which makes this method hard to obtain sufficient information about the neighboring blocks.

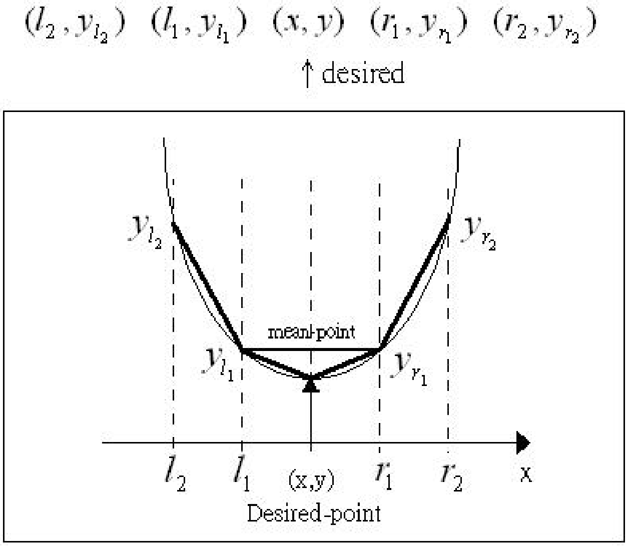


Fig. 2-6 The fuzzy Interpolation theoretical considerations.

2.5.4 Texture Synthesis

Rane et al. [1] proposed an algorithm to solve the texture synthesis problem. In this method, the lost region is filled-in pixel by pixel in a raster fashion with the texture from its neighbors.

As illustrated in Fig. 2-7 [1], when filling the pixel $p(i, j)$, the algorithm first defines a 3x3 template I_t next to $p(i, j)$, and looks for a $\hat{I}_t$ in the available neighboring blocks such that a given distance $d(I_t, \hat{I}_t)$ is minimized, where $d(I_t, \hat{I}_t)$ is defined as the normalized sum of squared differences (SSD) metric. Once the nearest template is found, we copy the pixel in the correspondent position to our pixel to be filled-in.

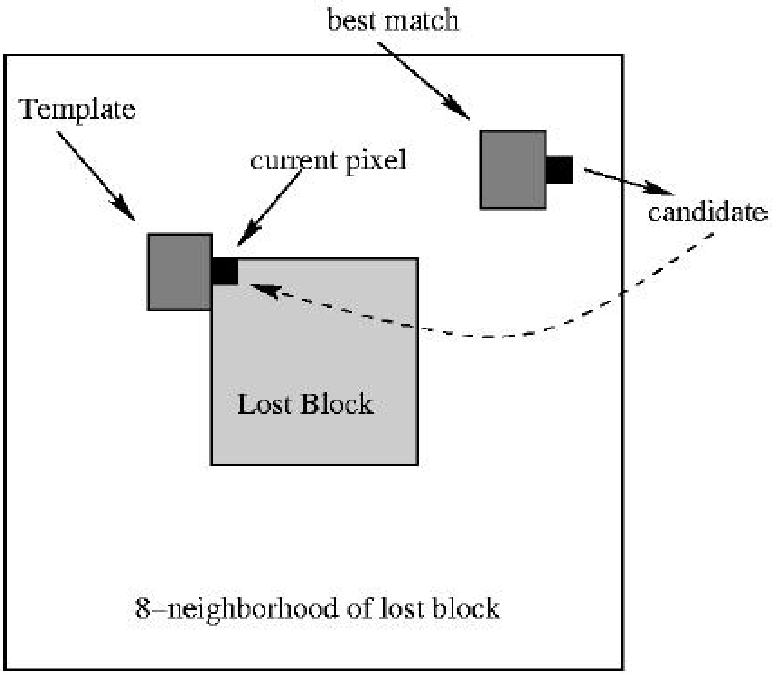


Fig. 2-7 Texture synthesis procedure.

Otherwise, Lee et al. [23] proved that the order of pixels to be filling-in in a lost block is an influential factor and only one space filling curve is not enough. They used five space filling curves in the experiments, including Peano curve, zig-zag curves, and raster scan curves below as shown in Fig. 2-8 [23].

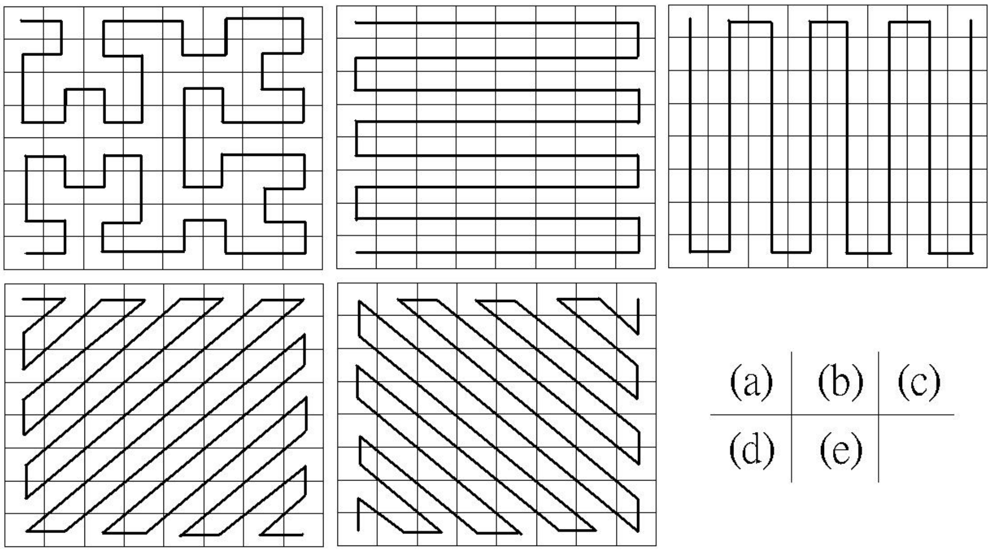


Fig. 2-8 The .8x8 window of space filling curve scan (a) Peano curve; (b) raster-1 curve; (c) raster-2 curve; (d) zig-zag-1 curve and (e) zig-zag-2 curve.

2.5.5 Block Filling and Image Decomposition

However, when filling-in a block contents structural object such as a line in it, the texture synthesis algorithm would be confused and hard to recover it. Also, the structure recovery algorithm is devised for inpainting in structured regions only, not for reproducing large textured areas.

Therefore, there is a need for combining this method with the texture synthesis algorithm. In [1] Rane et al. spatially separated an image into structural blocks and textural blocks and processed them in particular methods respectively. However, in most natural images the lighting effect plays an important role of the appearance of whole image. As a result, blocks are usually not able to be clearly judged as “structural” or “textural”.

For this reason, Rane et al. proposed another scheme in [2] to decompose an image into two layers of structure and texture in particular as shown in Fig. 2-9 [2] and then process each part with difference methods. Both are introduced in the previous sections.

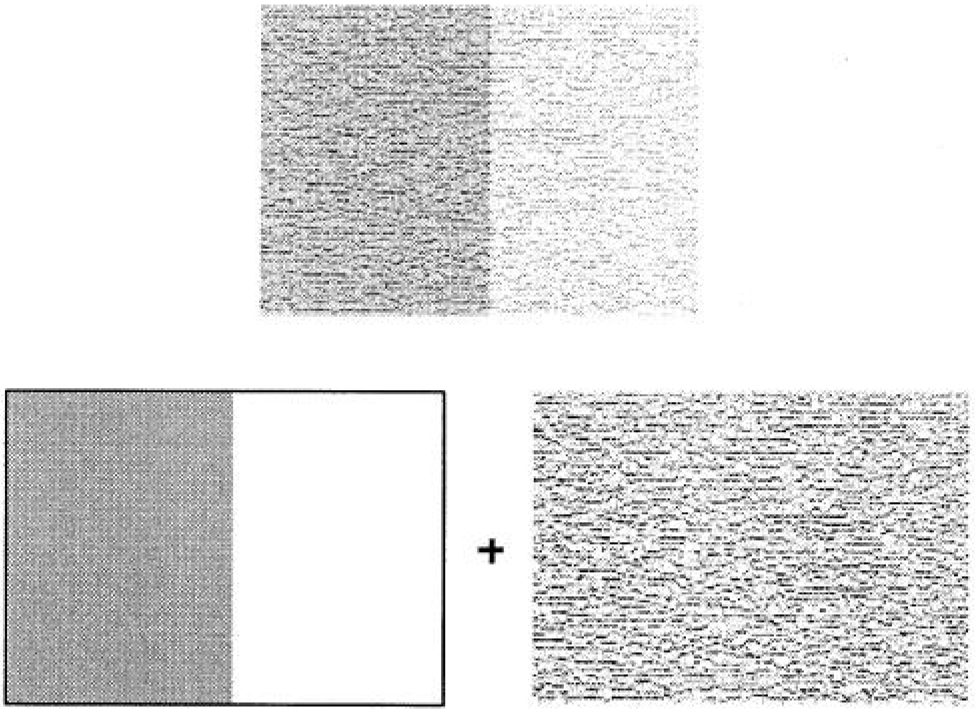


Fig. 2-9 Illustration of desired image decomposition. The top image is decomposed into a cartoon type (left) of image plus an oscillations one (right).

CHAPTER 3

Proposed Method

3.1 System Architecture

The method proposed in this paper is embedded in the JPEG 2000 image compression standard. The locations of our method in the encoder and decoder sides are illustrated in Fig. 3-1 and Fig. 3-2 respectively.

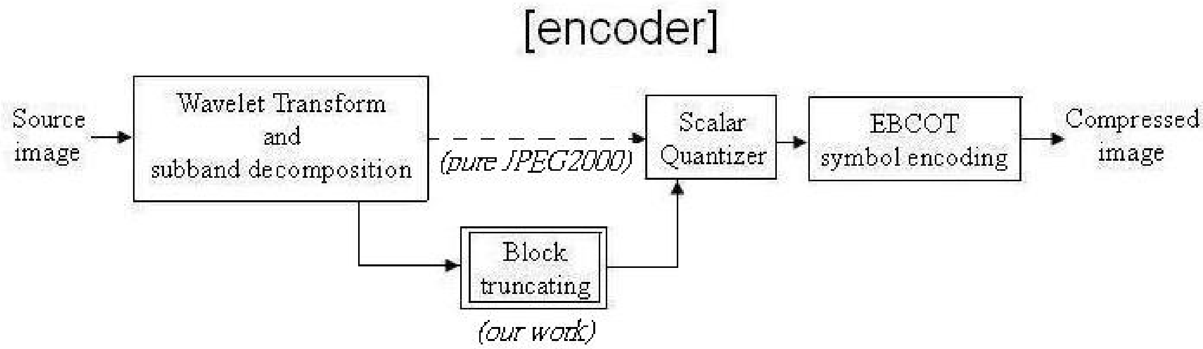


Fig. 3-1 The architecture of the Image encoder of our work.

When compressing, the source image is first decomposed into subbands with a three-level wavelet transform, then our method chooses a number of blocks which can be easily reconstructed by our method and drop them. After that, the rest of the blocks in each subband are quantized by a scalar quantizer and finally being encoded by the EBCOT symbol encoder.

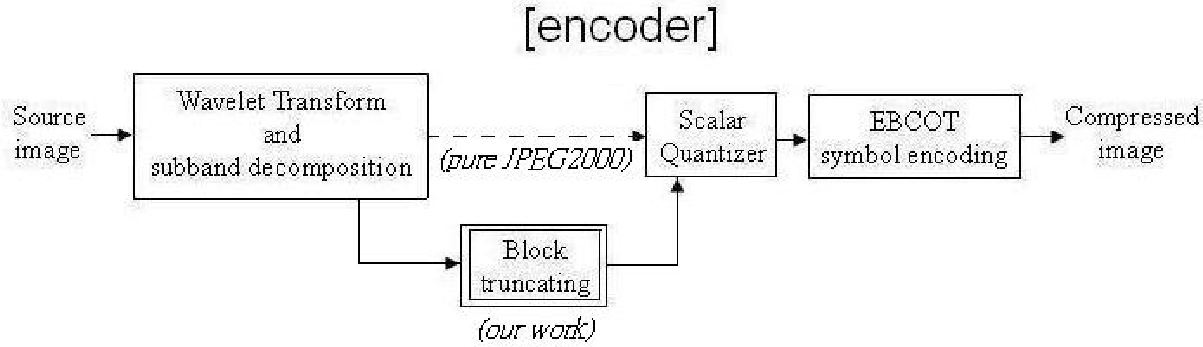


Fig. 3-2 The architecture of the Image decoder of our work.

When decompressing, the compressed image is first decoded by the EBCOT decoder, and then, our block filling method is used to recover the dropped blocks in each subband. Finally, the subimages are inversely transferred by the wavelet transform and combined into the decompressed image.

However, in order to make use of the characteristics of subband coding, more than one method are used to recover the dropped blocks in the decoder side. On the other hand, two methods are considered to choose blocks to drop in the encoder side. Both dropping and filling functions are introduced in the following sections in this chapter.

3.2 Wavelet Image Decomposition

In general, a natural image is composed of color regions, structural edges, tiny details, lighting effects and regular fine textures. Each component has specific characteristics. Utilizing the characteristics of each component, we must first try to decompose the image into components. In reaching the goal, we utilize wavelet image decomposition to separate the target image into subbands and process the subimage on each subband with the specific method respectively.

In our work, an image is decomposed into subbands with a 3-level 2-D discrete wavelet transform filter. As shown in Fig. 3-3.

Fig.3-4 is an example of this work, and Fig.3-4 (a) illustrates the original image and Fig.3-4 (b) illustrates the image pyramid of the source image decomposed by the 3-level multiresolution subband decomposition process.

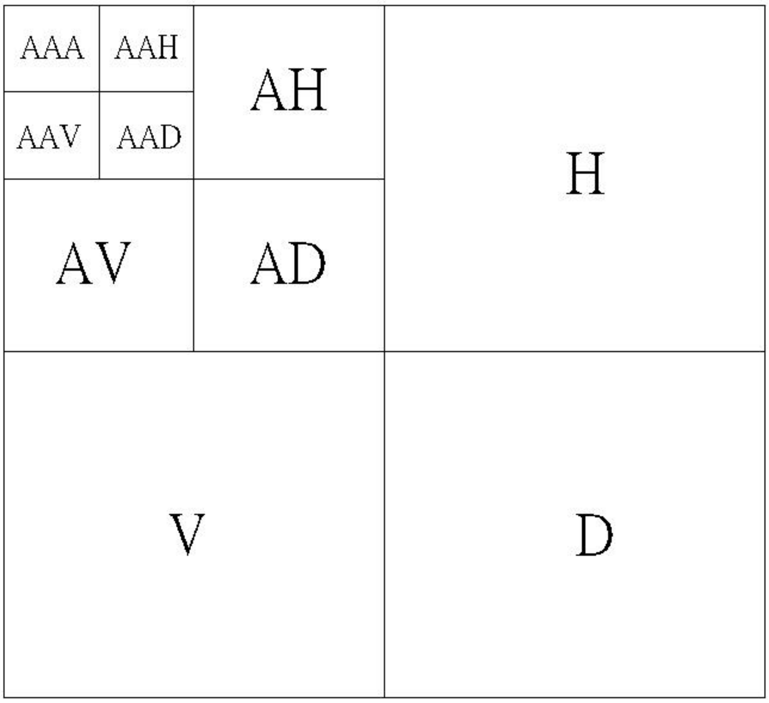


Fig. 3-3 Subband decomposition illustration.

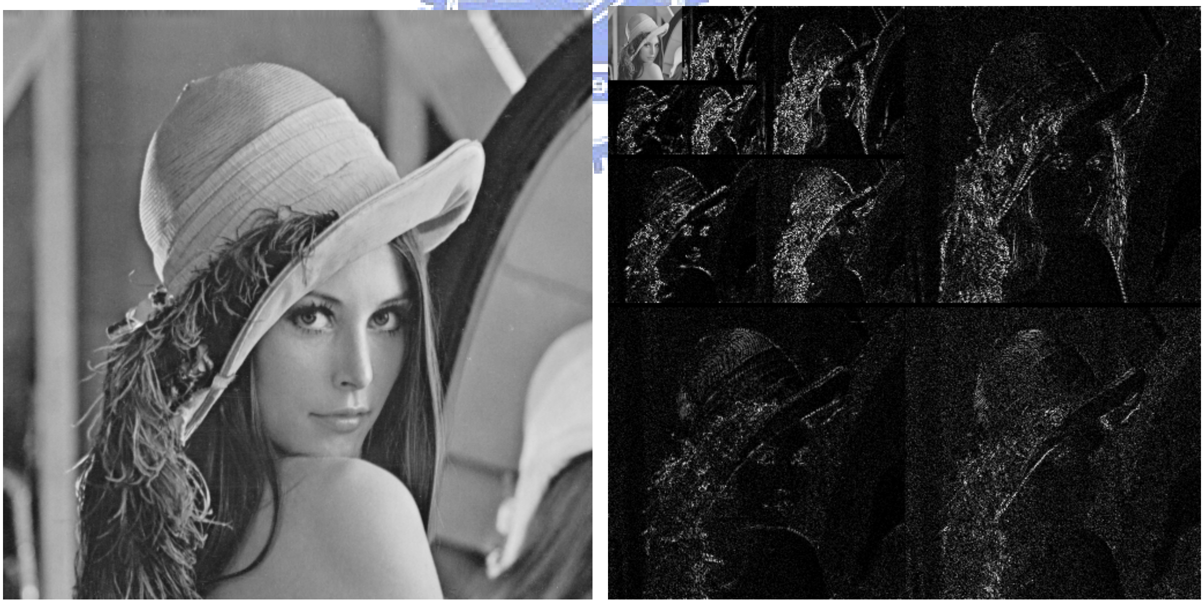


Fig. 3-4 (a) Lena image. (b) Lena image subband decomposition.

Note that in Fig. 3-4 (b), besides the average subband, the values in vertical, horizontal and the orthogonal detail channels are bounded in a narrow range beside zero. The implementation of this work is introduced in the previous chapter, and the method applied on each subband is introduced in section 3.4.

As shown in Fig. 3-5, in implementing the image decomposition process, we first apply edge padding, and then apply row-wise convolution to odd columns with wavelet basis h_1 and even columns with h_0, of which coefficients are as introduced in chapter 2. Edge padding process is to copy the edge information outside the border to fit the width of the wavelet transform basis mask in order to maintain the continuity on the edge, and the processing detail is shown in Fig. 3-6.

After that, a down-sampling is applied to separate the image into two subimages; one subimage contains the odd columns, the other contains the even columns. Same routine is also applied to column-wise convolution and row-wise down-sampling.

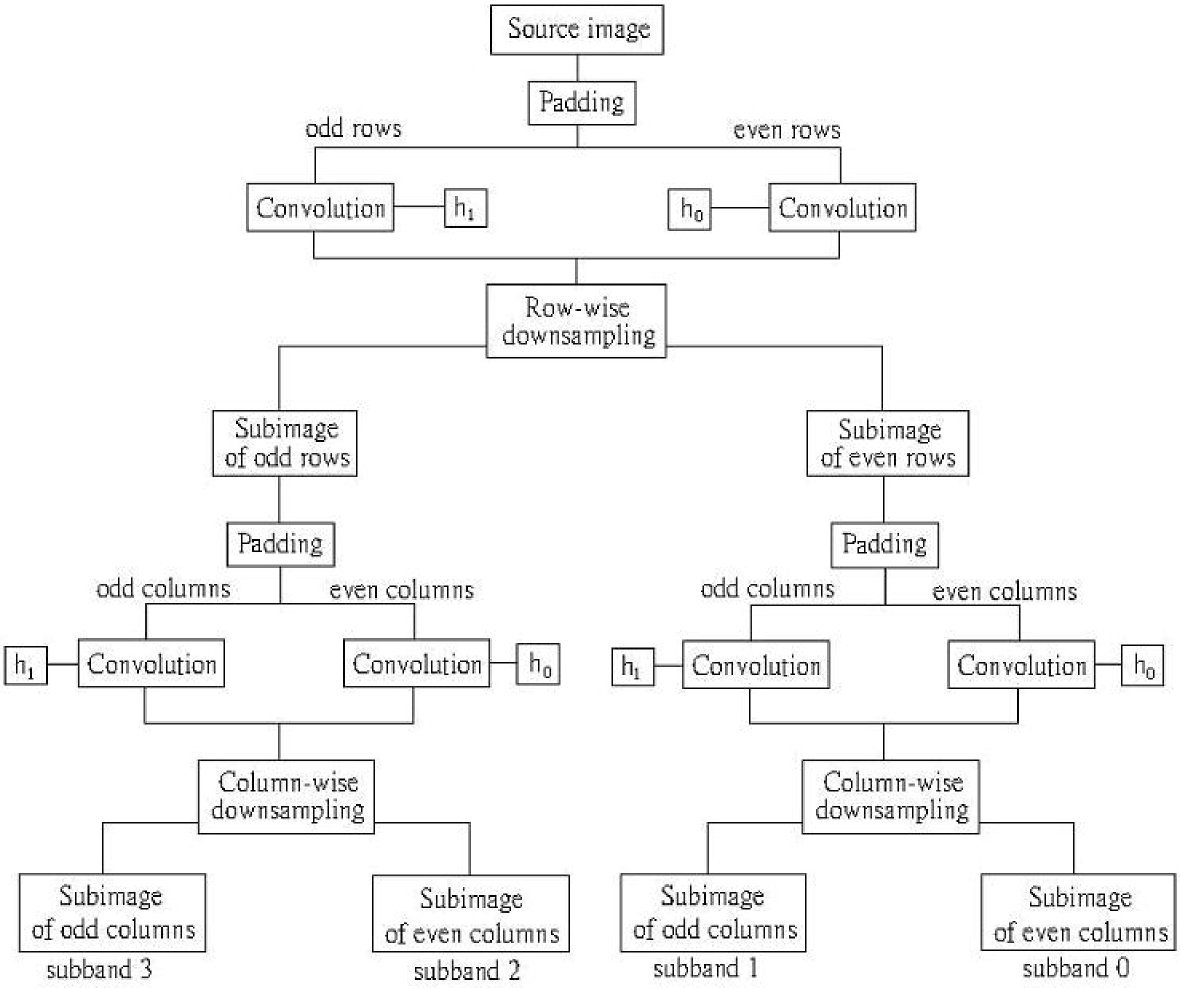
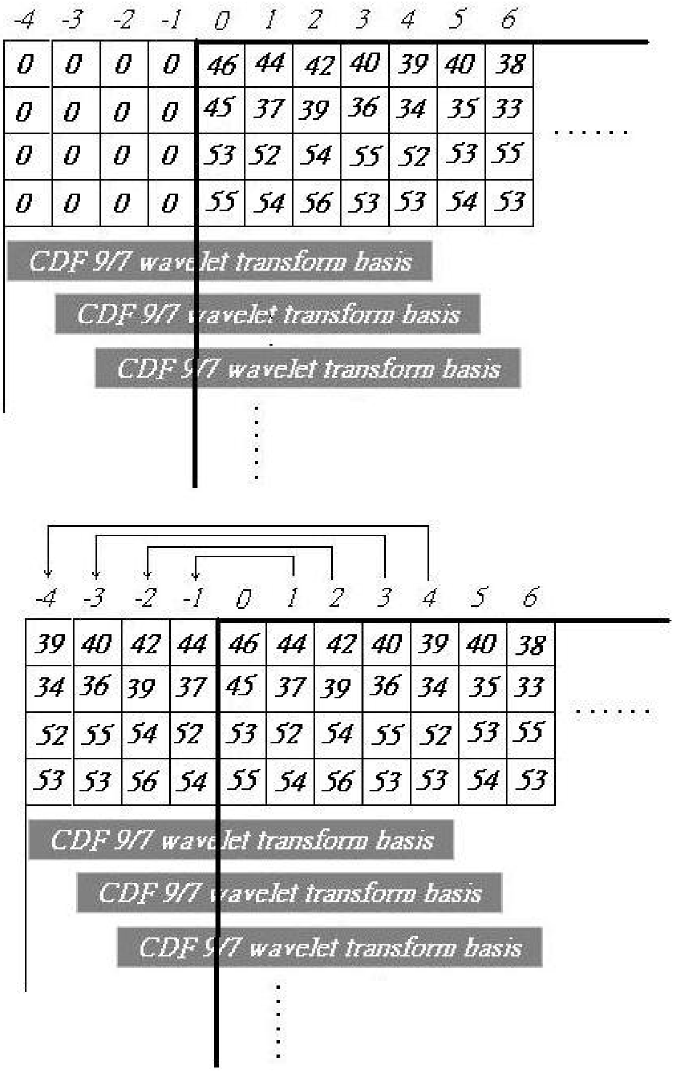


Fig. 3-5 Implementation of wavelet image decomposition.



-4	-3	-2	-1	0	1	2	3	4	5	6
0	0	0	0	46	44	42	40	39	40	38
0	0	0	0	45	37	39	36	34	35	33
0	0	0	0	53	52	54	55	52	53	55
0	0	0	0	55	54	56	53	53	54	53

-4	-3	-2	-1	0	1	2	3	4	5	6
39	40	42	44	46	44	42	40	39	40	38
34	36	39	37	45	37	39	36	34	35	33
52	55	54	52	53	52	54	55	52	53	55
53	53	56	54	55	54	56	53	53	54	53

CDF 9/7 wavelet transform basis
CDF 9/7 wavelet transform basis
CDF 9/7 wavelet transform basis
CDF 9/7 wavelet transform basis
CDF 9/7 wavelet transform basis
CDF 9/7 wavelet transform basis

Fig. 3-6 (a) The original image. (b) The image after padding.

After the two steps we obtain four subbands of the target image, so called single level image subband decomposition. After three-level wavelet decomposition, we obtain 10 different subimages.

In image combination, the procedure is shown in Fig 3-7; we first apply column-wise processes including zero padding upsampling, edge padding and convolution to each subband. Then, we add subband 00 and subband 01 to form the subband 0, and add subband 10 and 11 to form subband 1. After that, we apply row-wise processes to subband 0 and subband 1 and add them to form the reconstructed image. Note that, the zero padding process here is to fill-in the odd rows (columns) with zeros in even subbands and fill-in the even rows (columns) in with zeros in even subbands.

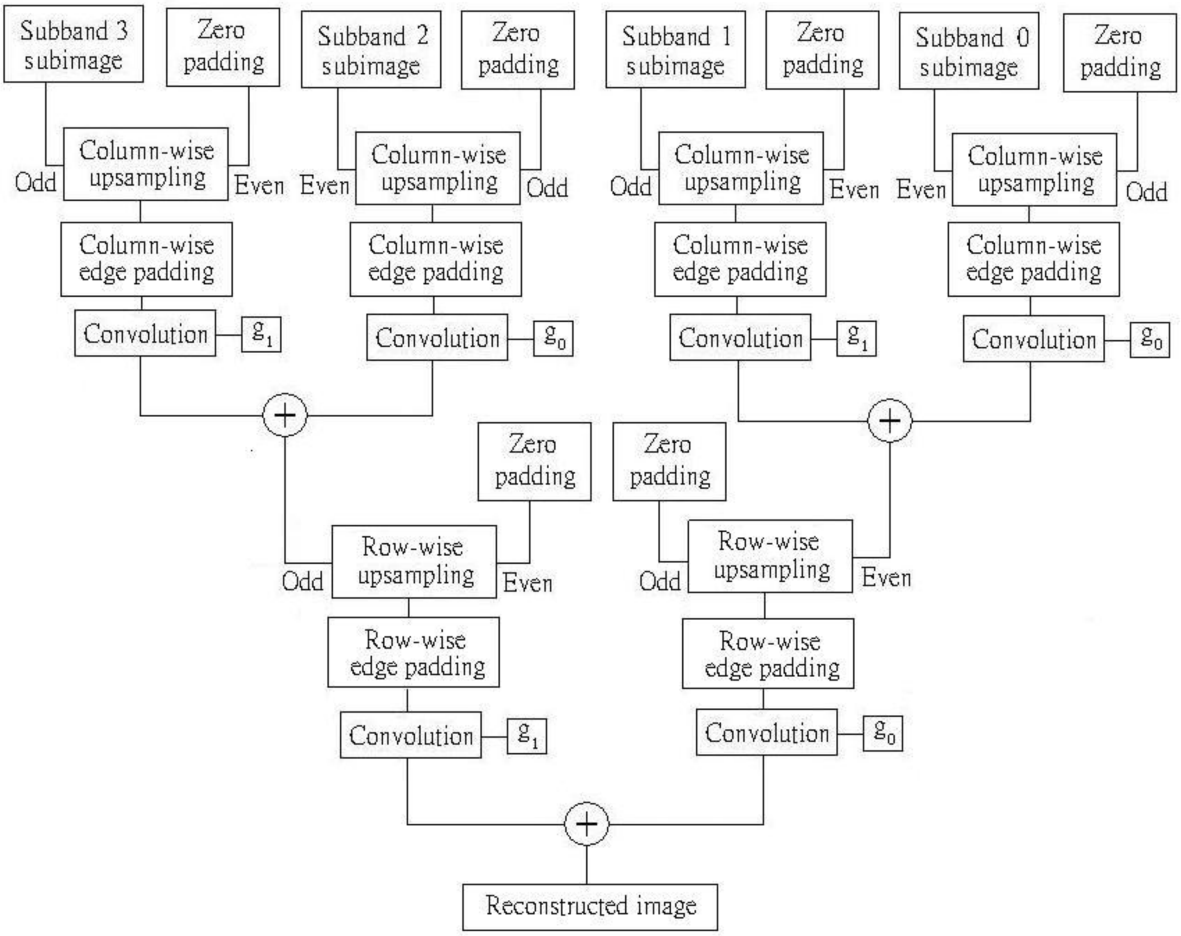


Fig. 3-7 Implementation of wavelet image combination.

In our method, we adopt CDF 9/7 filter, same as applied in JPEG 2000 lossy compression standard, as described in chapter 2.

3.3 Block Dropping

Some considerations are mostly discussed when dropping blocks, including the visual quality of the reconstructed image, time efficiency of the block dropping procedure, and the amount of dropped blocks. Among these considerations, the recovered image quality and the amount of dropped blocks form a tradeoff in between. Besides, the time efficiency problem is also an important problem in order to be applied in various fields of mobile application environments.

In this section several methods for choosing redundant blocks are introduced. First, the most intuitive one, we one by one drop each block and try to fill them back, once the recovered block reaches the image quality standard, it is judged “safely dropped” by the system, otherwise, it is judged “stayed in the file”. This method is surely able to make precise decisions of choosing most proper blocks to be dropped. However, the great cost of processing time makes it unsuitable for a lot of mobile applications.

The second method is to utilize several characteristics of the block to make decision by judging if it is similar to its neighbors. The characteristics we utilize in this paper are the mean values of our 10 subbands of each block, doted by $n_i(x, y)$, where (*x*,*y*) is the coordinate of the blocks, and *i* is the subband number.

In implementation, we first analyze every blocks to obtain the information we need for determine if one block is properly to be removed. Then we count the number of fields which is similar to its neighbors of each block. Here, “similarity” between block (x, y) and its neighboring block (x_1, y_1) is defined as that if there exist more than three $d_i = (n^s_{x,y} - n^s_{x+i,y+j})^2$, $-1 \le i \le 1,$ $-1 < j < 1,$ $i \cdot j \ne 0,$ $0 \le s \le 9$ such that $d_i < T$ where T denotes a threshold where s denotes the subband number. If there exist more than two neighboring blocks similar to the target block, we define the target block as “safely dropped”.

Note that if a strict threshold is defined, a small amount of candidate blocks to be dropped are picked up, it lowers the degradation of the bit rate, but maintains a higher fidelity of the reconstructed image. The threshold T is decided by the user or the rate control algorithm of the JPEG 2000 standard.

3.4 Block Filling

After going through the 3-level subband decomposition, the image is decomposed into one analysis subband, three vertical detail subbands, three horizontal detail subbands and three orthogonal detail subbands. In the analysis subband, we utilize linear interpolation with direction detection and the K-means algorithm to fill dropped blocks and texture synthesis method proposed in [1] with different filling order and different fuzzy mask to fill dropped blocks in vertical, horizontal and orthogonal detail subbands.

The architecture of our block filling method is illustrated below in Fig. 3-8.

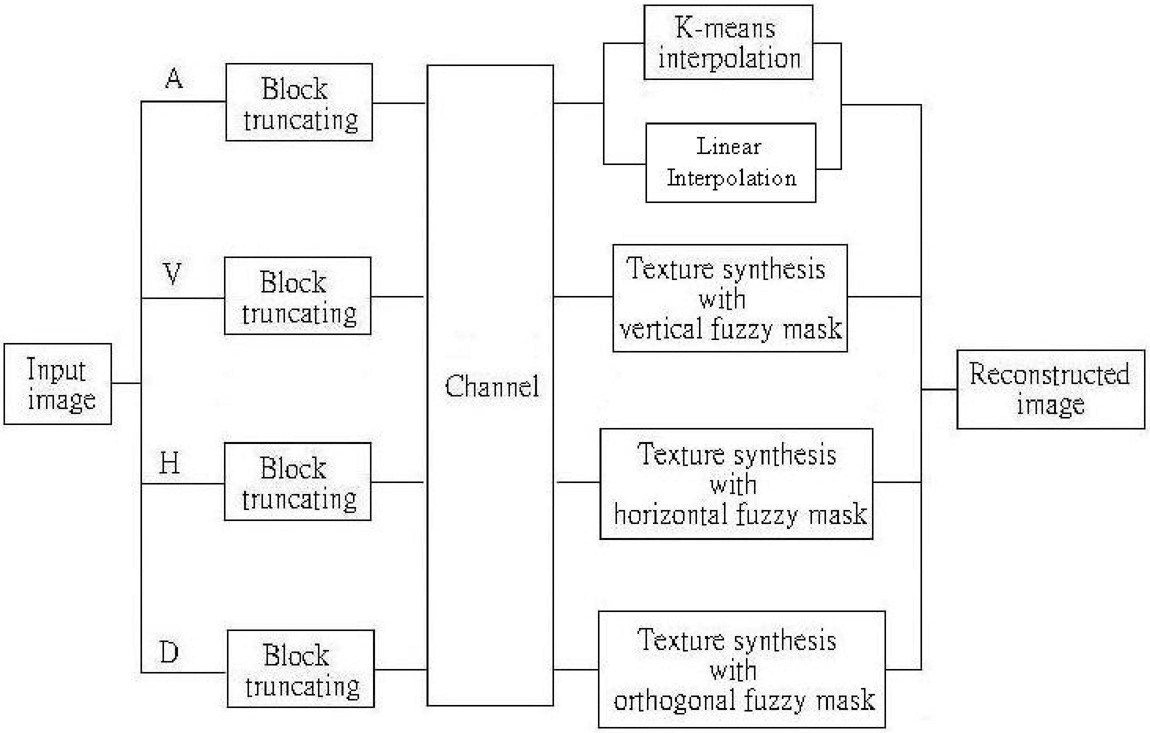


Fig. 3-8 The architecture of our block filling method.

3.4.1 Structure Recovery

As described in the previous section, we utilize two different methods to fill dropped blocks in the analysis subband. The analysis subband is conceptually understood as the mean gray value of a single 8x8 block in the original image. As a result, the block filling result in this subband determines the color of the whole block which has great influence on the quality of the reconstructed image.

In our structure recovery process, we first consider the linear interpolation method, which is the most intuitive method. If in some situations, the linear interpolation method is not able to perfectly fill up the lost block, we utilize the K-means algorithm to complete the interpolation work. However, if the region contains a narrow edge or small scale structures, the K-means algorithm may result in a blur, where linear interpolation with structure direction determination scheme can easily recover the edge.

3.4.1.1 Linear Interpolation

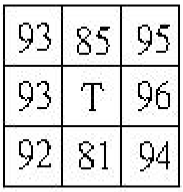
The decision of which method should be utilized is made by the equation below,

$$MRV = \frac{1}{5}[|f(i-1,j-1)-f(i,j-1)|+|f(i,j-1)-f(i+1,j-1)|+\frac{1}{2}|f(i-1,j)-f(i+1,j)|$$
$$+|f(i-1,j+1)-f(i,j+1)|+|f(i,j+1)-f(i+1,j+1)|]$$

$$MCV = \frac{1}{5}[|f(i-1,j-1)-f(i-1,j)|+|f(i-1,j)-f(i-1,j+1)|+\frac{1}{2}|f(i,j-1)-f(i,j+1)|$$
$$+|f(i+1,j-1)-f(i+1,j)|+|f(i+1,j)-f(i+1,j+1)|]$$

where MRV denotes the mean row-wise vector and the MCV denotes the mean column-wise vector. If the minimum of MRV and MCV is less than a threshold, the lost block is judged to be processed by the linear interpolation method; otherwise, it is judged to be processed by the K-means algorithm.

Consider an illustration in Fig 3-9; we first determine the direction of the structure going through the target pixel's neighborhood is column-wise by the equation above. Then we should fill-in the target pixel *T* with the average of its column neighbor, as in this illustration, we can obtain $\frac{1}{2}(85+81)=83$ as the result. However, if we apply K-means algorithm, the result will be $\frac{1}{6}(93+93+92+95+96+94)\approx 94$, which result in discontinuity on the middle line.



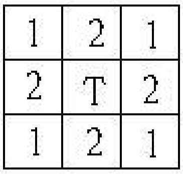
93	85	95
93	T	96
92	81	94

Fig. 3-9 Illustration of linear interpolation.

3.4.1.2 K-means Interpolation

On account of the large-scale continuity of the natural images and the strict requirements of candidate blocks in our block dropping function, we can confidently assume that there exist a number of blocks in each candidate block's neighborhood that are similar to the target block. Therefore, the K-means algorithm seems to be the best method to fill-in the blocks in the analysis subband.

Note that as the block dropping algorithm, in order to emphasize that the 4-neighbors of one block are with more importance than the corner neighbors, we give different weights to positions in the neighborhood as shown below in Fig. 3-10.



1	2	1
2	T	2
1	2	1

Fig. 3-10 Illustration of K-means interpolation.

3.4.2 Texture Synthesis with Particular Fuzzy Mask

As briefly described in chap 2, the texture synthesis algorithm proposed in [1] is to choose an available pixel in the target pixel's neighborhood whose reference mask has the closest distance to the reference mask of the target pixel. When filling a block with more than one pixel in it, the algorithm tends to fill them in a raster fashion.

However, in our work, we improved the texture synthesis method by utilizing different scanning order and different reference mask in each subband. Observing that the horizontal subbands are filled with horizontal detail signals, while vertical and orthogonal subbands filled with vertical and orthogonal detail signals respectively. Therefore, we designed different reference mask and filling order for each subband as shown in Fig. 3-11.

Note that the searching range has a great influence on the time efficiency of our texture synthesis methods; however, if we limit our sample space in an extremely small area, the search result may not have sufficient samples to recover the lost block. But if we allow our system to search in an extremely large area, not only the time efficiency is hurt but also wrong synthesis results may be produced by copying a pixel from the regions out of the neighborhood of the target block. Therefore, it is also an important issue to choose a proper size of the searching region for our texture synthesis algorithm.

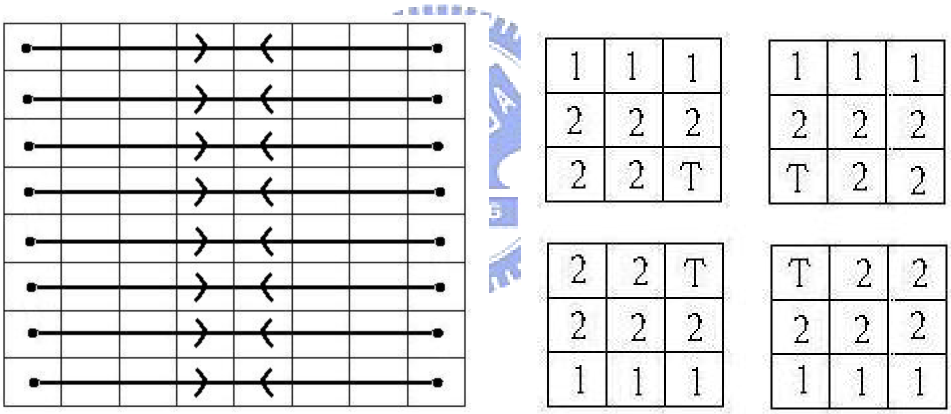
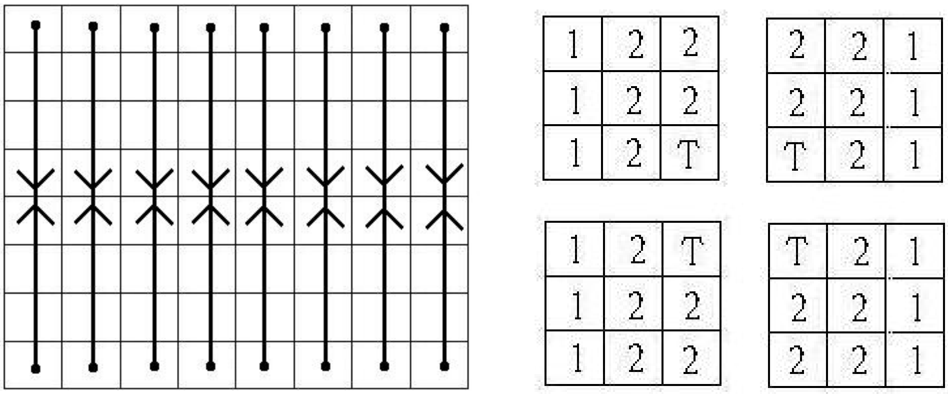
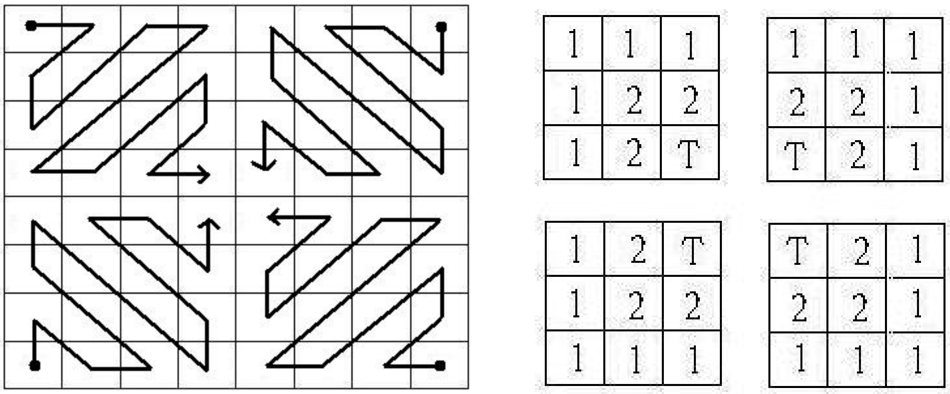


Fig. 3-11 (a) Block filling order and reference mask for vertical detail subband.



(b) Block filling order and reference mask for horizontal detail subband.



1	1	1
1	2	2
1	2	T

1	1	1
2	2	1
T	2	1

1	2	T
1	2	2
1	1	1

T	2	1
2	2	1
1	1	1

(c) Block filling order and reference mask for orthogonal detail subband.

Fig. 3-12 (a) and (b) illustrates the image processed by only K-means interpolation and processed by both K-means interpolation and texture synthesis algorithm respectively. Notice that in image Fig. 3-12 (a) the dropped blocks are filled up only with blurred color, but in Fig. 3-12 (b) the texture in the dropped blocks are reconstructed precisely.

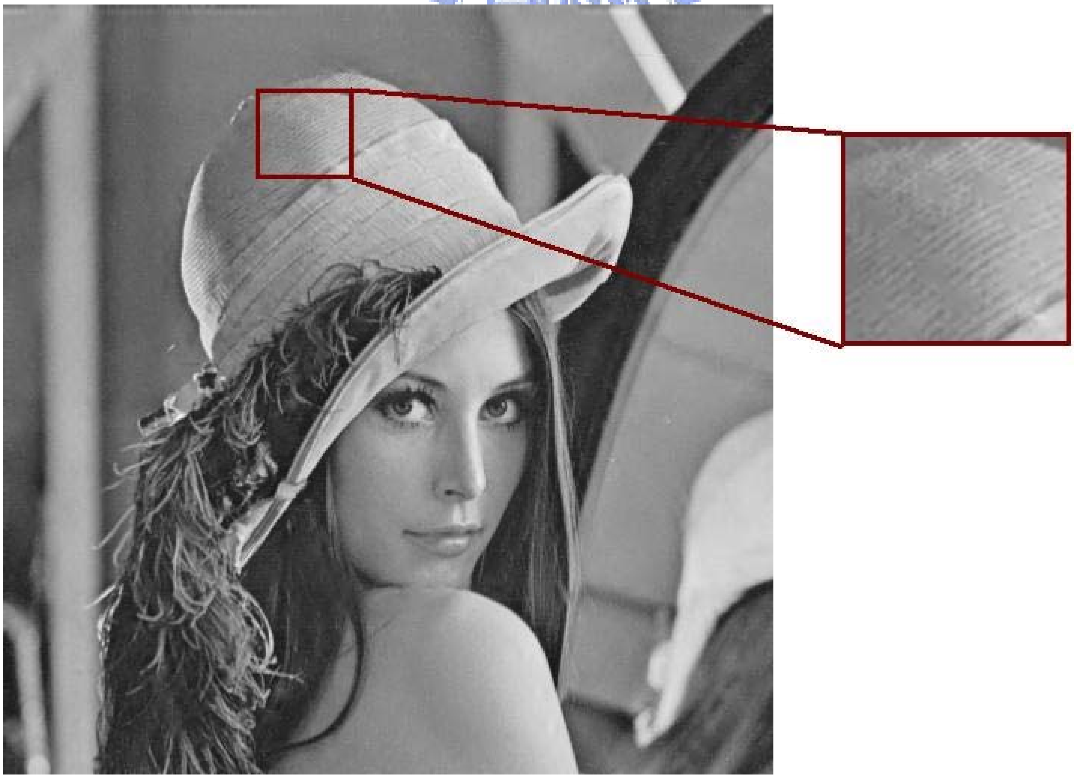
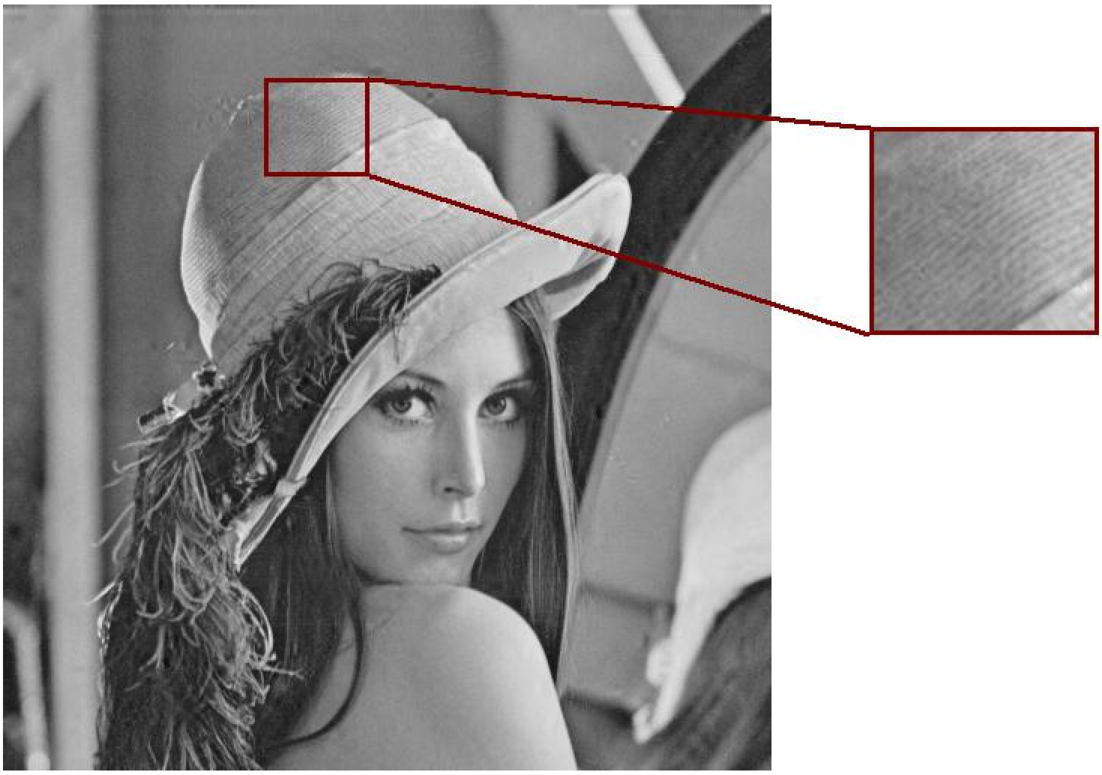


Fig. 3-12 (a) Lena image processed with only K-means interpolation algorithm.



(b) Lena image processed with K-means interpolation algorithm and texture synthesis algorithm.

CHAPTER 4

Experimental Results

4.1 Test images

In this chapter, we will present some experimental results obtained by applying the method proposed in this thesis.

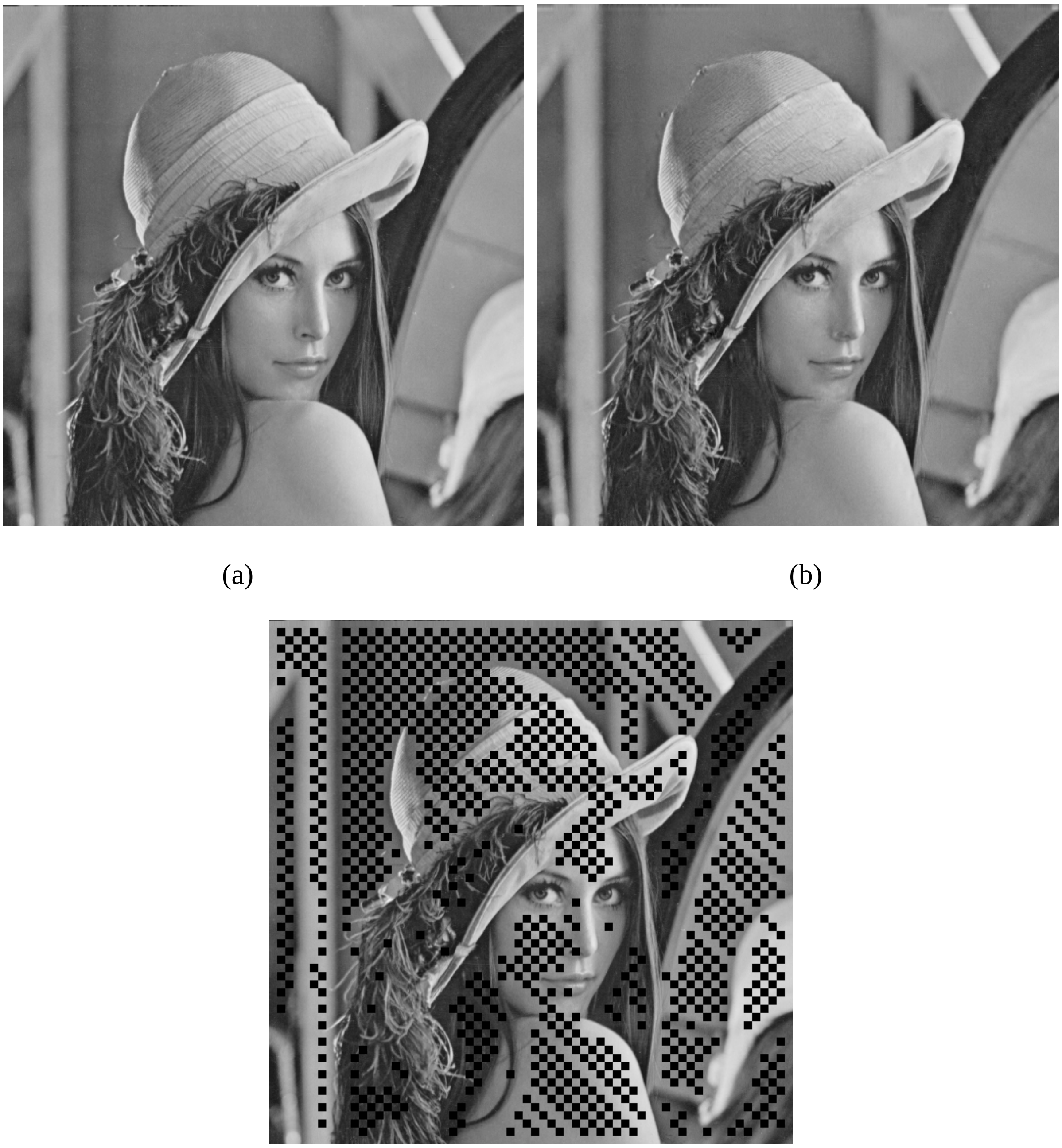
The test images we use for experiments are all common images including those with great portion of smooth regions such as Lena and also those with tiny details such as texture. The size of the test images are all of size 512x512 with 256 gray levels.

4.2 Block Dropping Results and Quality Performance

Obviously, the block dropping results and the quality of the block filling result have great influence on the performance of our method when applied in image compression. In this section we illustrate some block dropping and filling results to prove the capability of our method to be applied in image compression.

Fig. 4-1(a) is the original Lena image; Fig. 4-1(b) is the reconstructed image, and Fig. 4-1(c) illustrates the allocation of the dropped blocks. In this case 909 blocks are dropped, the PSNR of the reconstructed image is 36.85624, and the RMSE of the reconstructed image is 3.66208. However, if only dropped blocks are considered, the PSNR is 32.43088, and the RMSE is 6.09531.

Note that the texture synthesis method somehow lowers the signal to noise ratio, but it produces distinct texture details which improves the subjective visual quality.

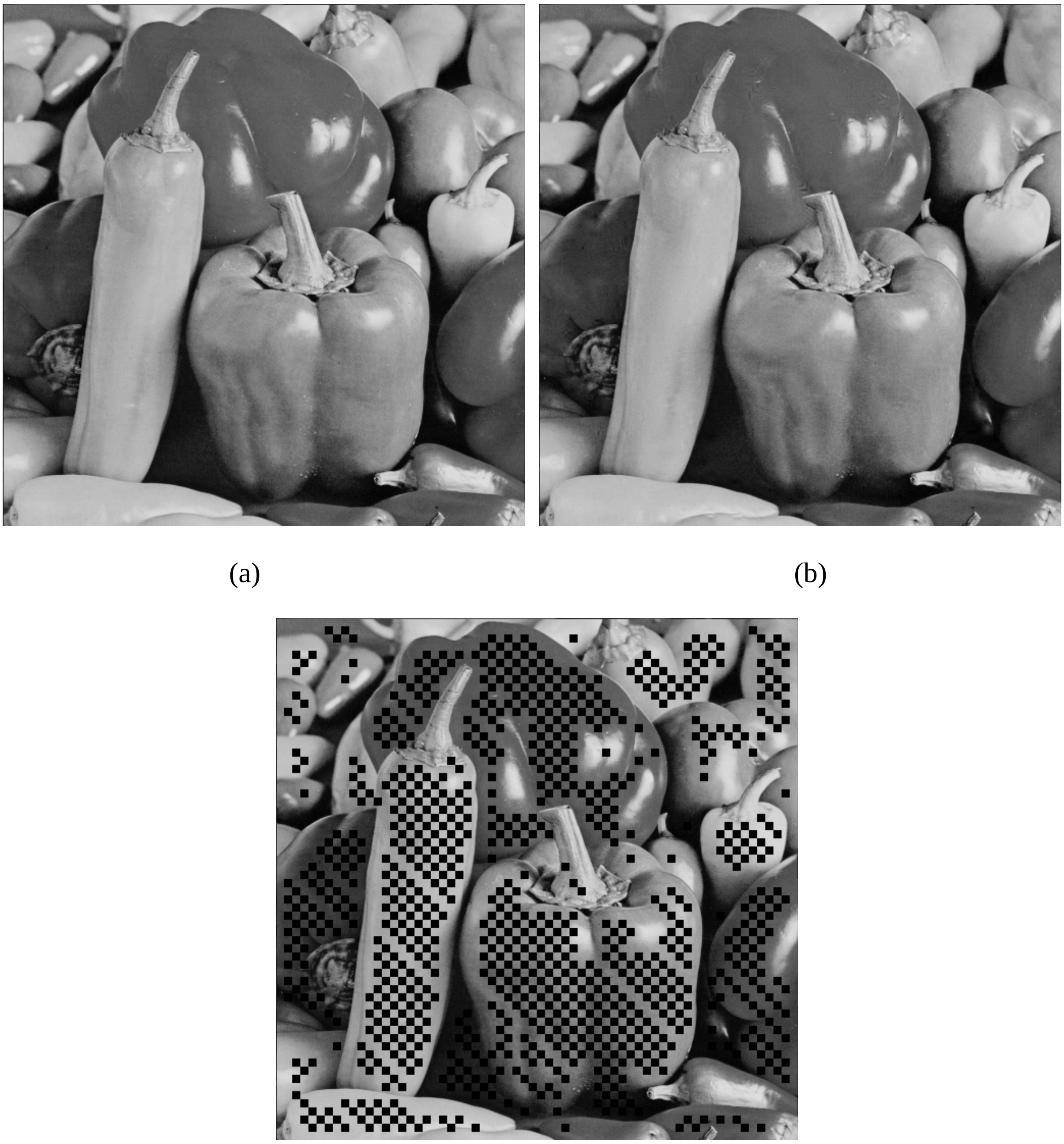


(a) (b)

(c)

Fig. 4-1 (a) The original Lena image (b) The reconstructed Lena image (c) Allocation of the dropped blocks.

In the second case: Pepper, we consider another smooth image. Total 882 blocks are dropped, the PSNR of the reconstructed image is 36.91718, and the RMSE of the reconstructed image is 3.63648. However, if only dropped blocks are considered, the PSNR is 32.07078, and the RMSE is 6.35332. Fig. 4-2(a) is the original image, Fig. 4-2(b) is the reconstructed image, and Fig. 4-2(c) illustrates the allocation of the dropped blocks.



(a)

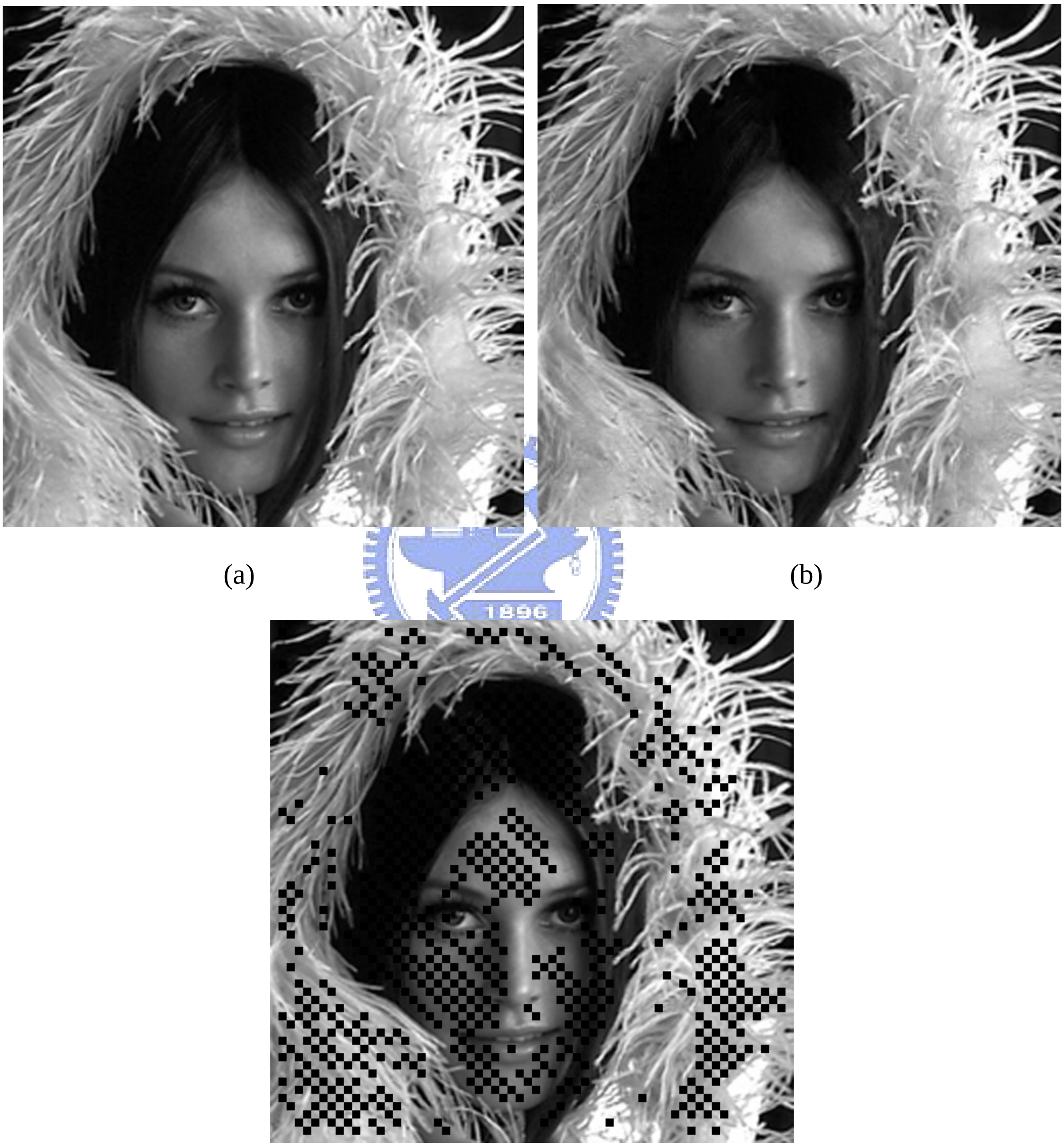
(b)

(c)

Fig. 4-2 (a) The original Pepper image (b) The reconstructed Lena image (c) Allocation of the dropped blocks.

After that, we apply this method to another test image: Lena2, which contains almost equally smooth regions and complicated texture regions. Fig. 4-3(a) is the original Lena2 image, Fig. 4-3(b) is the reconstructed image, and Fig. 4-3(c) illustrates the allocation of the dropped blocks

700 blocks are dropped in this case, while the PSNR of the reconstructed image is 35.17631 and the RMSE of the reconstructed image is 4.44349. When only dropped blocks are considered, the PSNR is 30.58620, and the RMSE is 7.53755.



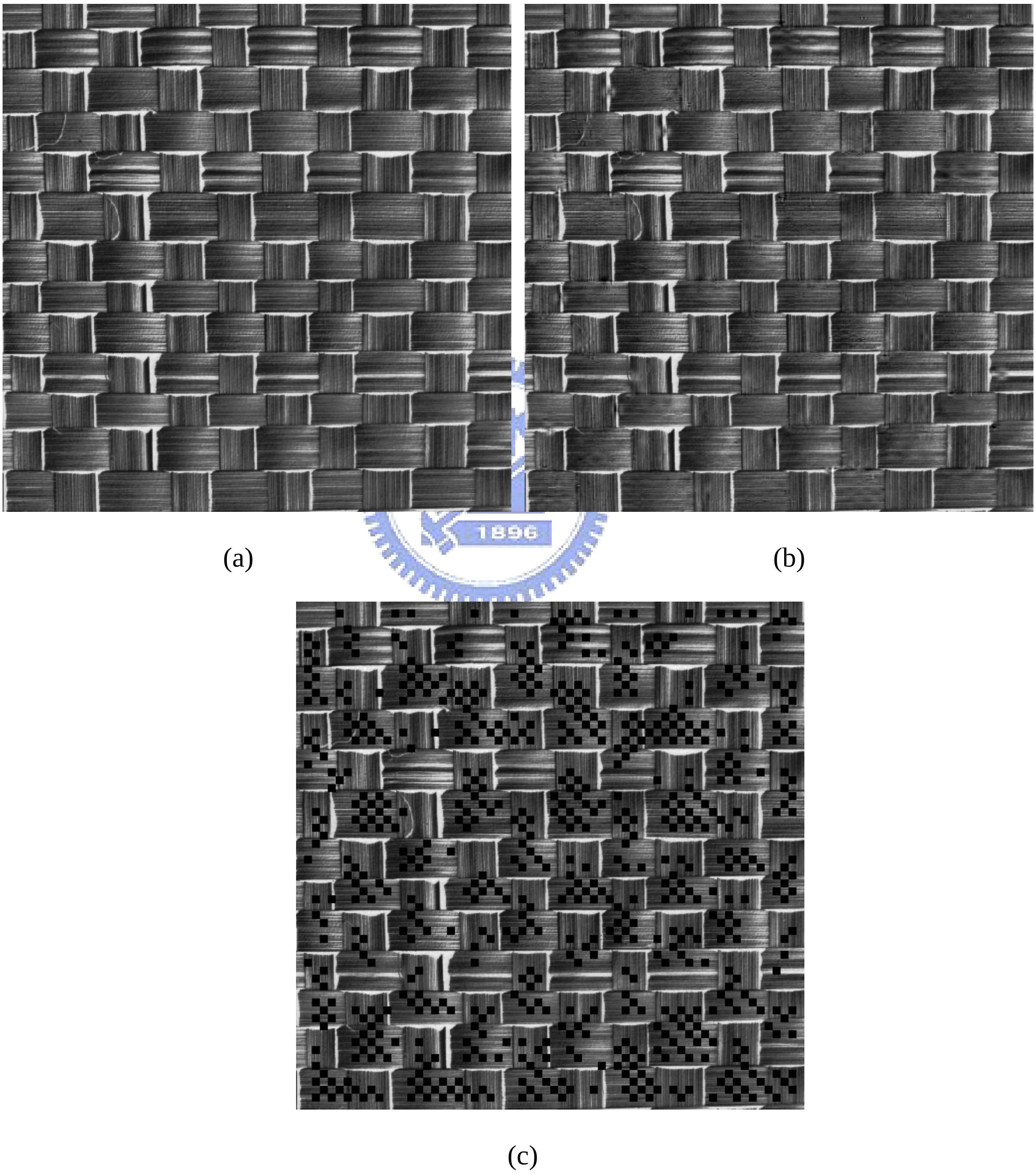
(a)

(b)

(c)

Fig. 4-3 (a) The original Lena2 image (b) The reconstructed Lena2 image (c) Allocation of the dropped blocks.

After testing three natural test images, we now illustrate one pure texture image as a radical case. In this case our method drops 641 blocks with totally PSNR of 30.99055 and RMSE of 7.19470. Consider only dropped blocks, PSNR is down to 25.11077 while RMSE is 14.15799. Fig. 4-4(a) is the original Texture image, Fig. 4-4(b) is the reconstructed image, and Fig. 4-4(c) illustrates the allocation of the dropped blocks



(a)

(b)

(c)

Fig. 4-4 (a) The original Texture image (b) The reconstructed Lena2 image (c) Allocation of the dropped blocks.

Note that the PSNR and RMSE measurements are not as good as the measurements of the three previous test images. It is because that when our texture synthesis algorithm produces tiny textures, it may also causes little mismatch of it. It may not be easily observed, however makes great influence on the image fidelity measurements. Besides, since the universally high differences between adjacent blocks, our block dropping algorithm is hard to make decisions to remove redundant blocks in the image.

4.3 Compare with JPEG 2000 Standard

Two topics are discussed when comparing JPEG 2000 system with our method to the pure JPEG 2000 standard scheme, the first is the improvement of the compression ratio, and the second one is the visual quality of the reconstructed image when our scheme is applied on the quantized images.

4.3.1 Improvement of the Compression Ratio

On account of the quality of the reconstructed image depends on the characteristics of an image, the improvement of the compression ratio by our scheme also differs from image to images. Fig. 4.5 illustrates the compression ratio of the pure JPEG 2000 standard scheme and our method with different quantization step-sizes.

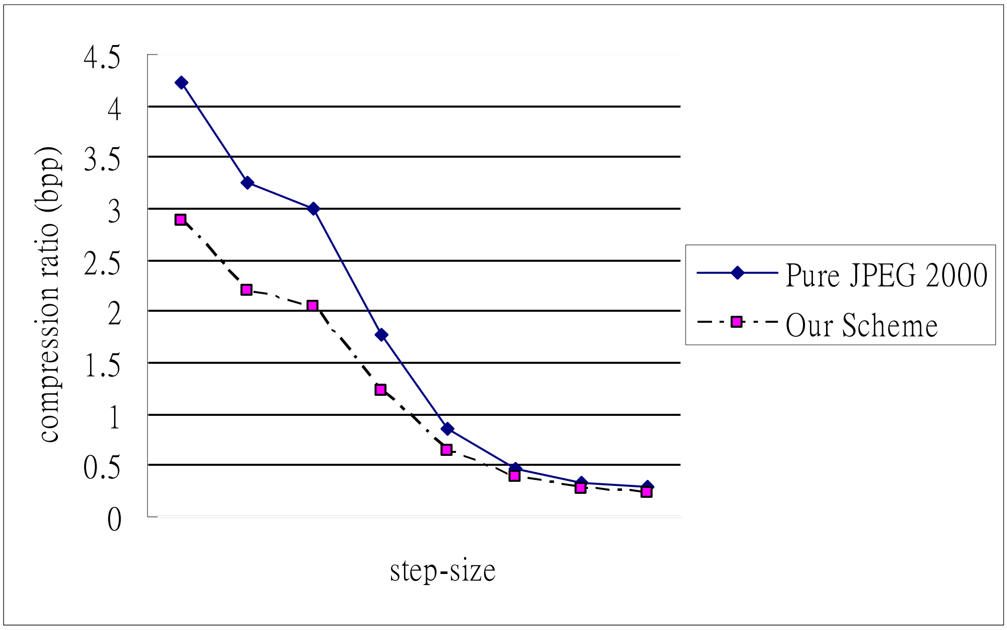
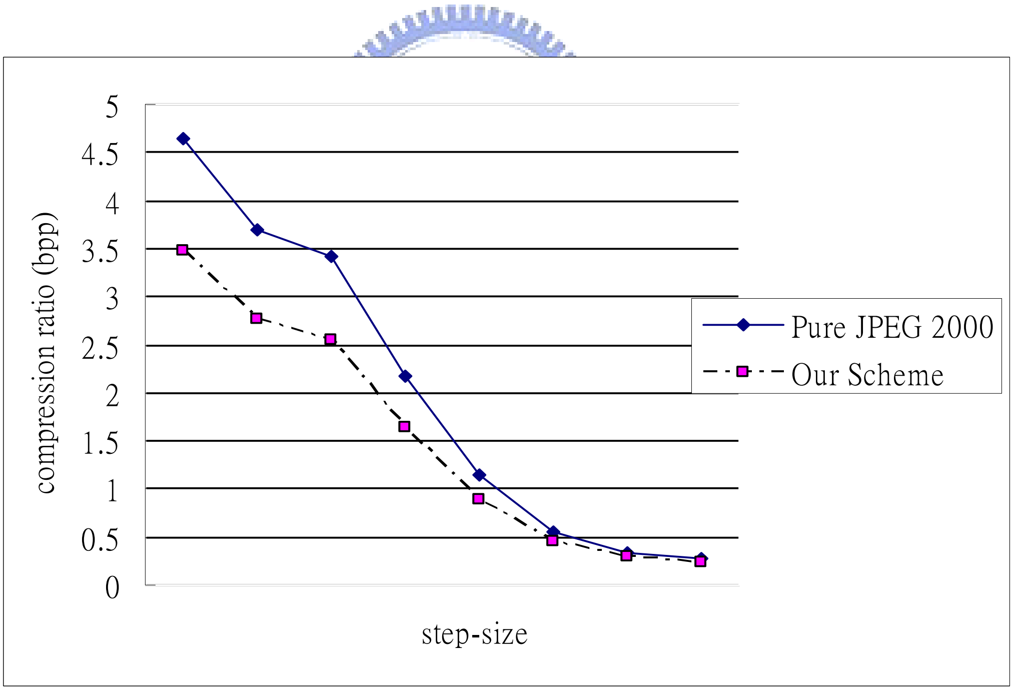
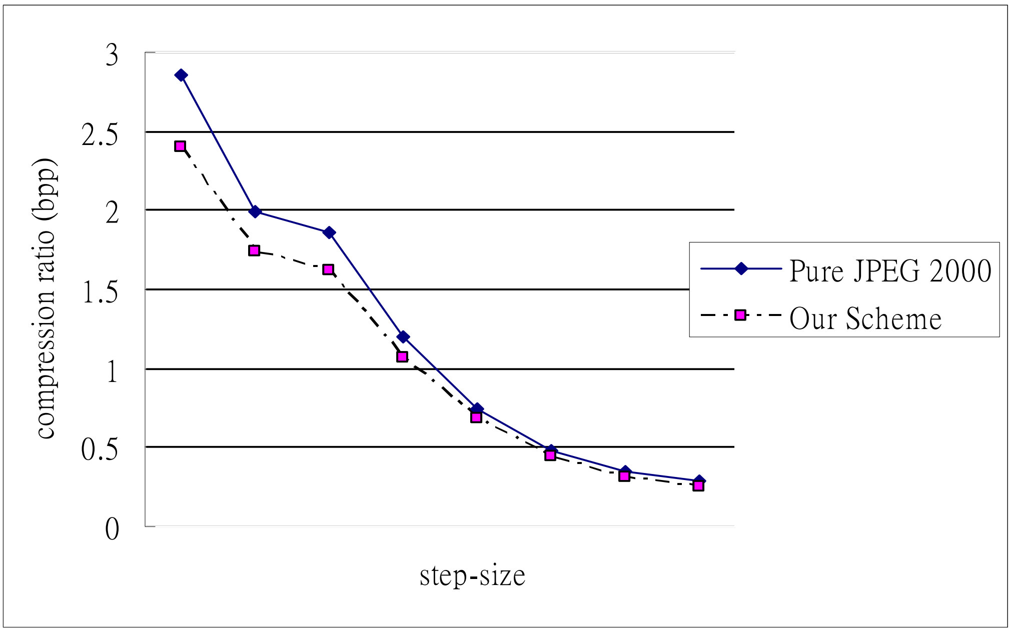


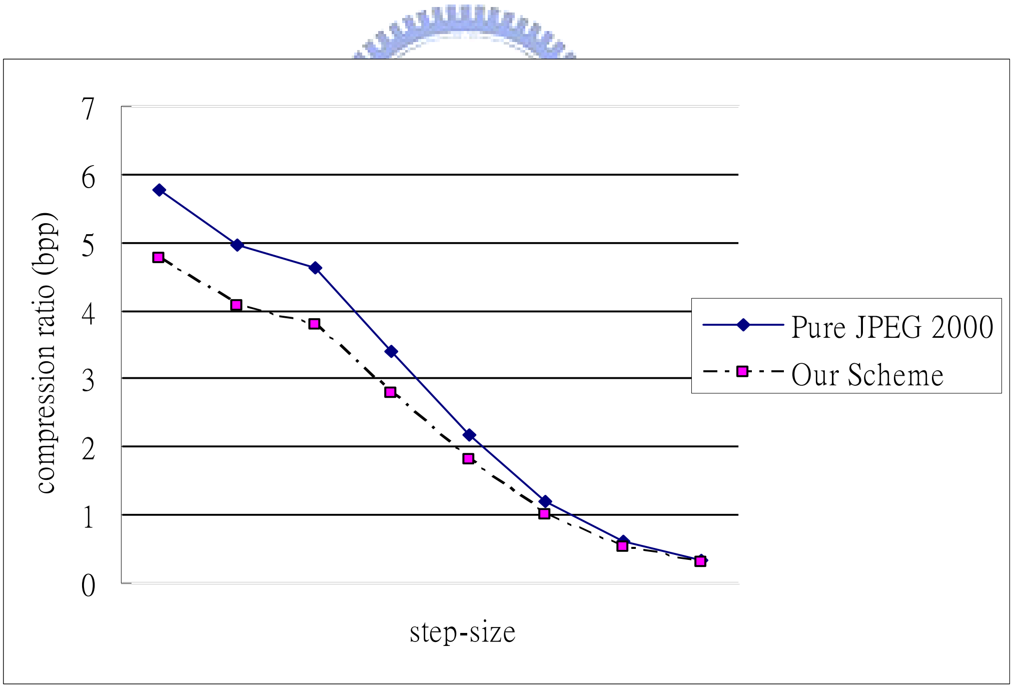
Fig. 4-5 (a) The comparison between pure JPEG 2000 standard and our scheme on Lena image.



(b) The comparison between pure JPEG 2000 standard and our scheme on Pepper image.



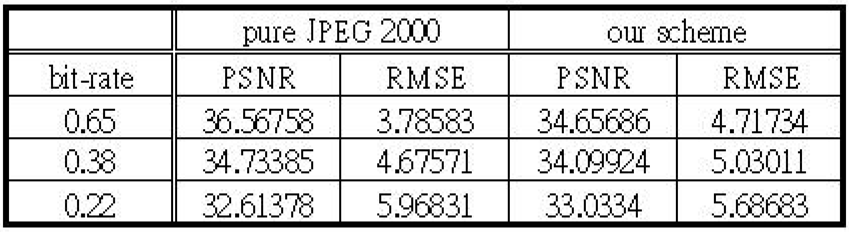
(c) The comparison between pure JPEG 2000 standard and our scheme on Lena2 image.



(d) The comparison between pure JPEG 2000 standard and our scheme on Texture image.

Note that in low bit rate of JPEG 2000 compression, our method is hardly to make improvement on compression ratio. It is because the number of zero coefficients goes extremely high in low bit rate, which means the block dropping work is no longer able to save a large amount of storage space.

Comparing our scheme to pure JPEG 2000 at the same bit-sate with Lena image, the result is illustrated in table 4-1. At high bit-rate, our scheme is “numerically” behind to pure JPEG 2000 standard; however, at low bit-rate, our scheme maintains more clear fine texture and a better PSNR. Fig. 4-6(a) is the result of pure JPEG 2000, and Fig. 4-6(b) is the result of our method in fine texture area at B_R=0.38 bpp.



	pure JPEG 2000		our scheme	
bit-rate	PSNR	RMSE	PSNR	RMSE
0.65	36.56758	3.78583	34.65686	4.71734
0.38	34.73385	4.67571	34.09924	5.03011
0.22	32.61378	5.96831	33.0334	5.68683

Table 4-1 Comparison of our scheme to JPEG 2000 at the same bit-rate.

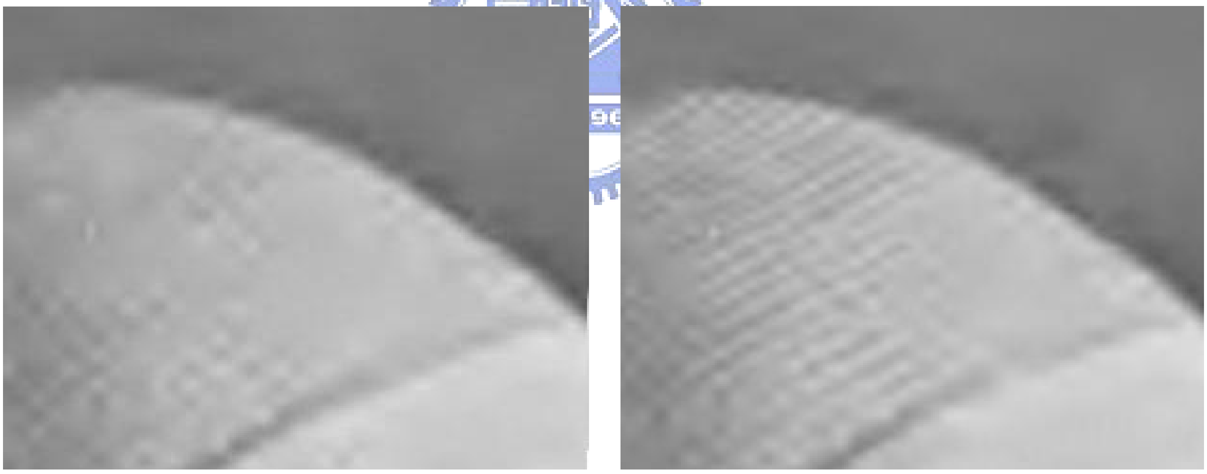


Fig. 4-6 lena image compressed at 0.38 bpp (a) Result of tiny texture region in pure JPEG 2000 compression. (b) in our scheme.

4.3.2 Application on Extremely Low Bit-Rate

In section 4.2 we have discussed the experimental results when our method is applied on uncompressed images, now we illustrate the performance of our filling method applied on low bit-rate compressed images. Fig 4-7(a) is the Lena image compressed with JPEG 2000 in bit rate=0.3bpp, and Fig 4-7(b) is the reconstructed image by our method at same bit rate. After the image is compressed, we have PSNR=33.91687, RMSE=5.13684, while our method is applied in the system, we have PSNR=33.12677 and RMSE=5.62602.

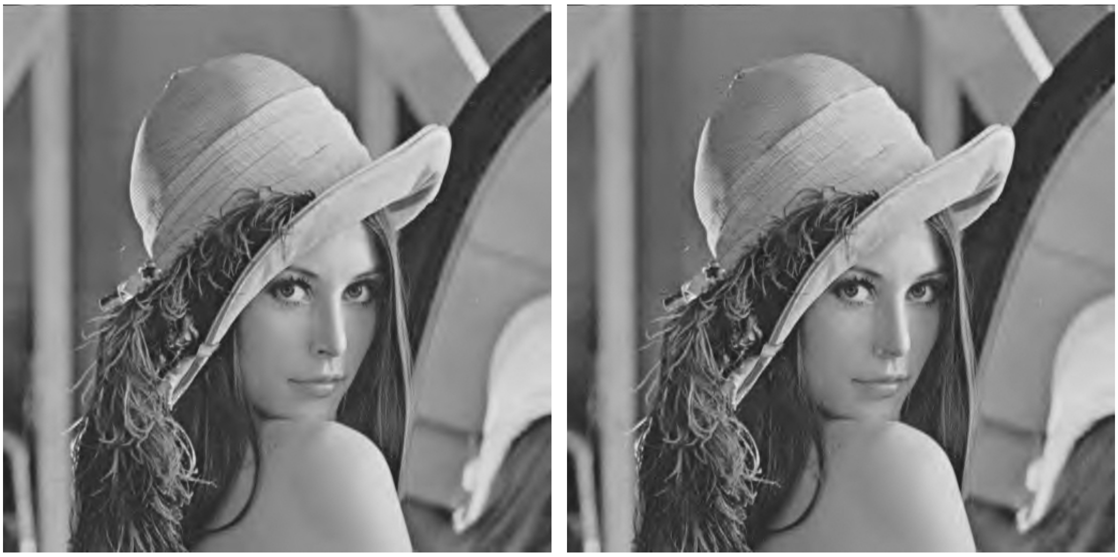


Fig. 4-7 (a) The compressed Lena image at bpp=0.3 (b) The reconstructed image.

Equally, Fig 4-8(a) is the Lena image compressed in bit rate=0.1bpp, and Fig 4-8(b) is the reconstructed image by our method. After the image is compressed, we have PSNR=29.50323, RMSE=8.53844, while our method is applied in the system, we have PSNR=29.14449 and RMSE=8.89848.

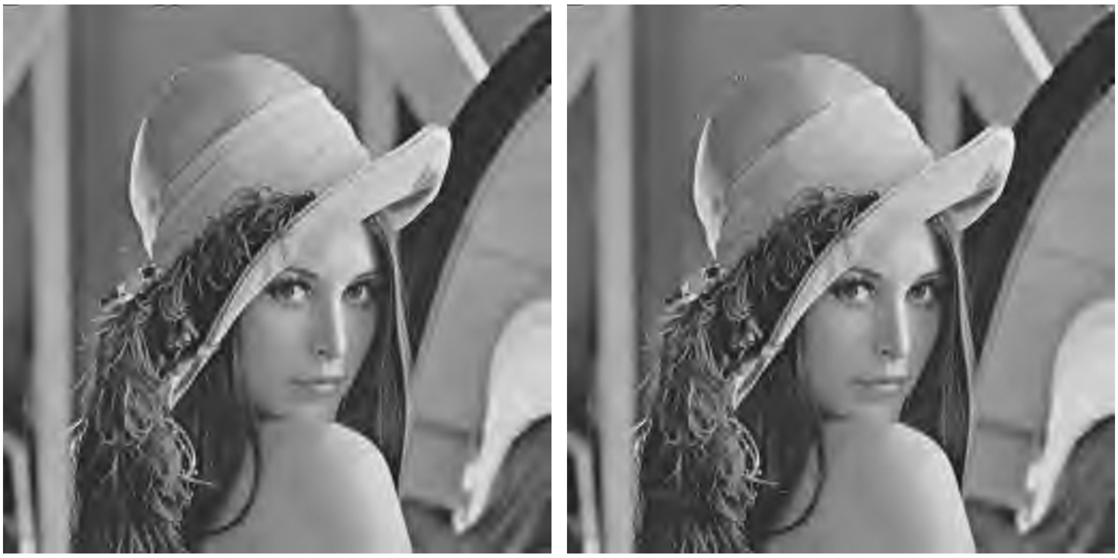


Fig. 4-8 (a) The compressed Lena image at 0.1bpp (b) The reconstructed image.

Note that our method is capable to properly reconstruct the highly compressed image with a hardly observed difference, and in tiny textures it is capable to produce fine texture if the fine details are not seriously destroyed by the quantizer.

CHAPTER 5

Conclusions and Future Work

5.1 Conclusions

In this study, we propose a new Scheme embedded in JPEG 2000 standard still image compression standard to further compress an image. We utilize the advantages of the methodologies in JPEG 2000 standard to remove inter-block redundancy by dropping redundant blocks by our block dropping algorithm in the encoder side and fill-in dropped blocks by various blocking methods in the decoder side.

As introduced in Chap 3, we utilize simple interpolation with direction determination and the K-means algorithm for the low frequency subband and the texture synthesis in high frequencies with different filling order and reference mask in each high frequency subband. Each method is just suitable for the subimage in each subband.

From the results as shown in Chapters 4, we can conclude that our method has some advantages as shown below:

1. In high bit-rate compression, our method is able to effectively improve the performance of compression ratio and at the mean time maintains an acceptable image quality.
2. In low bit-rate, our method is able to maintain a sharper quality in tiny detail regions than pure JPEG 2000.
3. Our method is embedded in the latest JPEG 2000 standard which provides an excellent performance on compression ratio by the symbol encoder.

5.2 Future Works

Future work can be directed to the following topics. First, the application of our work in

MPEG standards, by referring the neighboring frames, our block dropping method is able to remove much more redundant blocks. However, the time efficiency of our decoding algorithm has to fit the strict requirement of the MPEG standards. Second, when applying our method to the color space, the correlation between subimages in each color is able to be referred as to remove more redundant blocks. Finally, we can combine our method with the interpolation functions in the digital cameras used to raise the resolution of image quality when same number of CCD is used.

Reference

[1] Shantanu D. Rane, Guillermo Sapiro, and Marcelo Bertalmio, *“Structure and Texture Filling-In of Missing Image Blocks in Wireless Transmission and Compression Applications”* IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 12, NO. 3, MARCH 2003

[2] Marcelo Bertalmio, Luminita Vese, Guillermo Sapiro, and Stanley Osher, *“Simultaneous Structure and Texture Image Inpainting”* IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 12, NO. 8, AUGUST 2003

[3] Marcelo Bertalmio, Guillermo Sapiro, Vicent Caselles, Coloma Ballester *“Image Inpainting”* in Comput. Graph. (SIGGRAPH 2000), pp.417–424, July 2000.

[4] Yih-Gong Li *“Application of Fuzzy Set Theory in Image Processing”* Dissertation, Institute of Computer and Information Science College of Electrical Engineering and Information Science, National Chiao Tung University, July 1996.

[5] Sonja Grgic, Mislav Grgic, and Branka Zovko-Cihlar *“Performance Analysis of Image Compression Using Wavelets”* IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 48, NO. 3, JUNE 2001

[6] Michael W. Marcellin1, Michael J. Gormish2, Ali Bilgin1, Martin P. Boliek2 “An Overview of JPEG-2000” Proc. of IEEE Data Compression Conference, pp. 523-541, 2000.

[7] B. Zovko-Cihlar, S. Grgic, and D. Modric, *“Coding techniques in multimedia communications,”* in Proc. 2nd Int. Workshop Image and Signal Processing, IWISP’95, Budapest, Hungary, pp. 24–32, 1995.

[8] M. Ardito and M. Visca, *“Correlation between objective and subjective measurements for video compressed systems”*, *SMPTE J.*, pp. 768–773, Dec. 1996.

[9] J. Allnatt, “*Transmitted-Picture Assessment*.” New York: Wiley, 1983.

[10] M. Miyahara, K. Kotani, and V. R. Algazi. “*Objective picture quality scale (PQS) for image coding*.” Aug 1996.

[11] Rafael C. Gonzalez and Richard E. Woods “Digital Image Processing 2nd Edition” New Jersey: Prentice Hall, 2002.

[12] S. Mallat, *“A Compact Multiresolution Representation: The Wavelet Model,”* Proceedings IEEE Computer Society Workshop on Computer Vision, IEEE Computer Society Press, Washington, D.C., pp. 2-7, 1987

[13] A. S. Lewis and G. Knowles, *“Image compression using the 2-D wavelet transform,”* IEEE Trans. Image Processing, vol. 1, pp. 244–250, Apr. 1992.

[14] M. L. Hilton, *“Compressing still and moving images with wavelets,”* Multimedia Syst., vol. 2, no. 3, pp. 218–227, 1994.

[15] M. Antonini, M. Barland, P. Mathieu, and I. Daubechies, *“Image coding using the wavelet transform,”* IEEE Trans. Image Processing, vol. 1, pp. 205–220, Apr. 1992.

[16] “*Digital Compression and Coding of Continuous Tone Still Images”*, ISO/IEC IS 10918, 1991.

[17] “*Information Technology—Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mb/s: Video”*, ISO/IEC IS 11172, 1993.

[18] “*Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Video”*, ISO/IEC IS 13818, 1994.

[19] David S. Taubman, and Michael W. Marcellin, *“JPEG 2000 Image Compression Fundamentals, Standards and Practice”* Norwell Massachusetts, Kluwer Academic Publishers, 2002.

[20] David Taubman, Eric Ordentlich, Marcelo Weinberger, Gadiel Seroussi, Ikuro Ueno and Fumitaka Ono, *“Embedded Block Coding in JPEG 2000”*, IEEE, 2000

[21] Bon-Woo Hwang, Seong-Whan Lee, *“Reconstruction of Partially Damaged Face Images Based on a Morphable Face Model“*, IEEE TRANSACTIONS ON PATTERN

ANALYSIS AND MACHINE INTELLIGENCE, VOL. 25, NO. 3, MARCH 2003.

[22] A.C. Kokaram, R.D. Morris, W.J. Fitzgerald, P.J.W. Rayner. “*Interpolation of missing data in image sequences*”, IEEE Transactions on Image Processing 11(4), 1509-1519, 1995.

[23] Jia-Hong Lee, *“A Study on Texture Analysis Using Geometrical Feature Spectra”* Thesis, Institute of Computer and Information Science College of Electrical Engineering and Computer Science, National Chiao Tung University, May 1996.