DISTORTION-CONSTRAINT COMPRESSION OF THREE-DIMENSIONAL CLSM IMAGES USING IMAGE PYRAMID AND VECTOR QUANTIZATION

BY

YEGANG TAO

UNIVERSITY
of
GLASGOW

DISSERTATION SUBMITTED IN PART FULFILMENT
OF THE REQUIREMENTS FOR THE AWARD OF THE
DEGREE OF DOCTOR OF PHILOSOPHY IN COMPUTING SCIENCE
IN THE UNIVERSITY OF GLASGOW

2005-11-07

Contents

List of Figures

List of Tables

Abstract

The confocal microscopy imaging techniques, which allow optical sectioning, have been successfully exploited in biomedical studies. Biomedical scientists can benefit from more realistic visualization and much more accurate diagnosis by processing and analysing on a three-dimensional image data. The lack of efficient image compression standards makes such large volumetric image data slow to transfer over limited bandwidth networks. It also imposes large storage space requirements and high cost in archiving and maintenance.

Conventional two-dimensional image coders do not take into account inter-frame correlations in three-dimensional image data. The standard multi-frame coders, like video coders, although they have good performance in capturing motion information, are not efficiently designed for coding multiple frames representing a stack of optical planes of a real object. Therefore a real three-dimensional image compression approach should be investigated.

Moreover the reconstructed image quality is a very important concern in compressing medical images, because it could be directly related to the diagnosis accuracy. Most of the state-of-the-arts methods are based on transform coding, for instance JPEG is based on discrete-cosine-transform (DCT) and JPEG2000 is based on

discrete-wavelet-transform (DWT). However in DCT and DWT methods, the control of the reconstructed image quality is inconvenient, involving considerable costs in computation, since they are fundamentally rate-parameterized methods rather than distortion-parameterized methods. Therefore it is very desirable to develop a transform-based distortion-parameterized compression method, which is expected to have high coding performance and also able to conveniently and accurately control the final distortion according to the user specified quality requirement.

This thesis describes our work in developing a distortion-constraint three-dimensional image compression approach, using vector quantization techniques combined with image pyramid structures. We are expecting our method to have:

1. High coding performance in compressing three-dimensional microscopic image data, compared to the state-of-the-art three-dimensional image coders and other standardized two-dimensional image coders and video coders.
2. Distortion-control capability, which is a very desirable feature in medical image compression applications, is superior to the rate-parameterized methods in achieving a user specified quality requirement.

The result is a three-dimensional image compression method, which has outstanding compression performance, measured objectively, for volumetric microscopic images. The distortion-constraint feature, by which users can expect to achieve a target image quality rather than the compressed file size, offers more flexible control of the reconstructed image quality than its rate-constraint counterparts in medical image applications. Additionally, it effectively reduces the artifacts presented in other approaches at low bit rates and also attenuates noise in the pre-compressed images. Furthermore, its advantages in progressive transmission and fast decoding make it suitable for bandwidth limited tele-communications and web-based image browsing applications.

Chapter 1 Introduction

Hospitals and clinical environments are rapidly moving toward an environment comprising of digitisation, processing, storage, and transmission of medical images [Wong, 1995]. The trend in healthcare information technology is increasingly multimedia-oriented. The basic motivation is to represent medical images in a digital form to support image transfer and archiving, and the manipulation of visual diagnostic information in new and useful ways, such as image enhancement and volume rendering. The significance of digital images in medical imaging has been widely recognized since 1970's, where the appearance of the computerized tomography (CT) pushed the growth of medical imaging to a high gear. From 1980's many new imaging modalities have been developed and used in a variety of applications, such as magnetic resonance imaging (MRI) and ultra-sonography (US) specified for large tissues imaging and confocal laser-scanning microscopy (CLSM) specified for tiny tissue sample imaging. Moreover the rapid growth of computer hardware significantly improves the capacity of computation and storage and the maturity of image processing techniques can help people manipulate, understand and analyse medical images in a more efficient way. All of these give digital imaging a great impact on modern medicine.

One significant contribution of digital medical imaging over the conventional film-based imaging is the possibility of three-dimensional imaging functionality. In this technology a series of individual scans are performed in one experiment, which

capture the volumetric information of a real three-dimensional object from various depths and angles. After that image processing methods will combine these scans using a computer and then build up a three-dimensional image of the real object. This functionality assists people greatly in performing a more realistic and accurate diagnosis in three dimensions rather than in two dimensions using only one image. Therefore three-dimensional imaging has been received considerable attentions in medical applications, such as MRI and CLSM.

1.1 Three-dimensional Medical Image Compression

As opposed to the real world or analog images, such as photos, pictures, paintings and so on, digital images are digitised by sampling and discretizing the amplitude of the continuous signals from real world images. This process is sometimes referred as analog-to-digital (A/D) conversion. A digital image can be express as a regular grid of points with discrete values, while in the analog image the information is continuous. A digital image is a dense intensity function, $I(x,y)$, defined on a two-dimensional grid having N rows and M columns, where (x, y), $x \in Z_{[1,M]}$, $y \in Z_{[1,N]}$ is the coordinates of the image. The image segment represented by the coordinates (x,y) is called a picture element, or a *pixel*. In gray-scale mode, $I(x,y)$ gives a single value of the pixel representing its intensity; in colour mode, $I(x,y)$ gives a group of values representing the combination of intensities from all the colour channels of this pixel. The number of bits representing the intensity value of each pixel is determined by the capability of data range of the imaging device. For instance if the intensity is quantized into an integer ranging from 0 to 255, we can use an 8-bit integer for each intensity value and the pixel depth is 8 bit-per-pixel (8bpp). When the data range becomes larger we may use more bits, such as 12 bits or even 16 bits, for each intensity value, and the pixel depth is 12bpp and 16bpp accordingly. Therefore in the conventional representation a digital image requires $M \times N \times k$ bits for each colour channel, where M, N are defined by the image size and the k denotes the pixel depth.

The archiving and transmission of large-volume image data is one of the critical issues in medical imaging applications. This is because in medical imaging the image size could be very large, for instance 256×256 and even 2048×2048 pixels per image

in CLSM, and the pixel depth could be as high as 16bpp per channel. The situation could be more critical in three-dimensional imaging. For instance it is common to produce over 100 mega bytes (MB) in one examination in CLSM to build up a three-dimensional image of a tissue. Therefore even with the rapid development of the computer technology the increment of image dimensions has outstripped the growth of storage media and network bandwidth. This fact encourages the intensive studies on image compression techniques and the requirements for the compression are now more important than they were previously. Due to the significant improvement of the compression efficiency, the areas of applications for compression of medical images have proliferated. They include medical image archiving, wireless emergency medical services, remote surgery and medicine, progressive browsing of databases and others.

Briefly speaking, image compression maps an original image, described by a regular grid of discrete values, into a bit stream suitable for communication over or storage in a digital medium. The number of bits required to represent the coded image should be smaller than that required for the original image so that one can use less storage space or communication time. There are two basic types of compression. *Lossless* compression refers to algorithms which allow the original pixel intensities to be perfectly recovered from the compressed representation. *Lossy* compression algorithms do not allow that and they will introduce degradation on the reconstructed image. Although using lossy approaches in medical image compression may raise legal concerns in diagnosis, they have been given considerable attention. This is because these techniques can produce much higher compression rates than the lossless ones. Moreover, compared to the lossless approaches, they support more flexible control of the compression tasks, such as users can choose the target compression rate, which can be eventually used in a variety of image compression applications. These properties are very important in three-dimensional medical image compression due to the enormous image data size. Therefore a majority part of studies in three-dimensional medical image compression are focused on lossy approaches.

Varieties of three-dimensional medical image compressors have been proposed and investigated [Wilhelm, 1990; Ho, 1994; Lee, 1993; Abousleman, 1995; Vlaicu, 1995; Lee, 1997; Schelkens, 2001]. The initial attempts were frame-by-frame schemes using

a two-dimensional image compressor on a sequence of images. However by considering the physical process of the three-dimensional imaging, people have then tuned to exploit the inter-frame correlations to improve the compression efficiency, where approaches resembling a video compressor have been investigated. Till recent years methods based on three-dimensional transforms have received increased attentions. These real three-dimensional methods offer excellent coding efficiency which is significantly superior to the former two schemes. They treat the whole sequence as a 3D volume data and capturing the inter-pixel correlations, no matter whether they are arranged in the same frame or across frames, within one procedure.

We have noticed, however, that in these state-of-the-arts methods, such as JPEG2000 and 3D-Wavelet, although users can conveniently control the bit rate or compression ratio, the image qualities, or distortions, are not explicitly controllable, whereas in medical image applications the latter are more important than the former. This is because in medical applications, compared to the compression rate, the degradation in image quality is more directly related to the diagnosis accuracy when such accuracy relies on the review and the analysis of the reconstructed images. Moreover due to the difficulties in modelling the distribution of an image signal, it is impossible to predict the outcome distortion of a reconstructed image from its corresponding compression rate. For instance letting a compressor work at a particular rate, which produces acceptable image quality on an image containing large stationary-distributed regions, may lead to unacceptable quality due to artefacts and loss of information on another image with lots of non-stationary and strong local structures, such as edges. In fact users have to manually balance the compression rate and image quality without explicit constraints from the encoder and therefore the compression must be performed multiple times to achieve target image qualities. Therefore these methods are basically *rate-parameterized*, where the compressors are designed to suit a compression task specified by the target compression rate. If they are forced to work at a particular quality the compression process would be very inconvenient and repetitive, which significantly increase the costs in computation.

We attack the problem of distortion-control compression by considering a closed-looped image pyramid encoder. As a good example of the multi-resolution analysis

technique, the image pyramid has been received considerable attention in a variety of applications in image processing, such as image compression, image enhancement, image registration and so on. In image compression, although open-looped image pyramid encoding was proposed more than twenty years ago, it is not as successful as other approaches, such as methods based on discrete-cosine transform (DCT) and discrete-wavelet transform (DWT), because of its lower compaction of the image signal. More recently studies on closed-looped image pyramid encoding have shown a significant improvement in the compression efficiency over its open-looped counterpart [Aiazzi, 1997] and its coding performance is comparable with those in DCT and DWT approaches [Cockshott, 2003; Tao, 2004].

However in the literature we noticed that little attention have been paid to the distortion control functionality of the closed-looped image pyramid. In our work we have investigated the effect of the encoding-error feedback from the view of distortion control performance rather than the coding performance. We have found that closed-looped image pyramid encoding structure inherently supports convenience control of the reconstructed image quality. Actually in this structure the distortion between the original image and its reconstruction is strongly upper-bounded by the encoding of the bottom-level difference pyramid, and is relatively independent of the encoding of other levels, which makes the control of the image quality more convenient and straightforward. It is superior to DCT and DWT approaches, because in these approaches the final distortion depends on the encoding of all the coefficients and the relations between the distortion in the transformed domain and that in the original image domain is obscure.

Moreover *vector quantization* (VQ) techniques are considered in our method. According to Shannon's information theory VQ is superior to scalar quantization (SQ) in coding performance, especially at low rate. We designed a hybrid VQ technique, referred as cross-band classified vector quantization (CBCVQ), which is particularly adapted to the closed-looped image pyramid structure. We choose a classified vector quantizer for each level to exploit the non-stationary distributions of the intra-band coefficients; on the other hand we capture the cross-band correlations between coefficients from different levels by sharing their classification maps.

In addition, in our approach the entire compression system can be viewed as the corporation of several sub systems. They operate either in a dependent form, such as the encoding between two levels in the pyramid structure; or in an independent form, such as the sub-vector quantizers in the CBCVQ. Each sub system has its own performance characteristic and all of them contribute to the overall performance of the whole system. This will raise a question: How can we control the performance of each sub system to maximize the overall performance? In terms of rate-distortion performance this question can be interpreted as how we control the rate (or distortion) of each sub system such that the overall rate-distortion performance can be optimal. We solve this problem by investigating the rate-distortion performance of each sub system and their relations and proposed two bit allocation algorithms. The purpose of these two algorithms is adaptively and automatically controlling the rate (or distortion) of each sub system according to its rate-distortion characteristic to optimize the overall compression performance.

1.2 Overview

In this thesis we are going to propose a high-performance three-dimensional image data compressor. It should be able to conveniently and accurately control the final distortion according to a user specified quality requirement, which is expected to be a *distortion-parameterized* approach. Moreover we expect this compressor to have good coding performance in three-dimensional image compression, which should be comparable to the state-of-the-art approaches in the literature. This thesis will describe how we achieve these two goals and it will be organized as follows.

Chapter 2 introduces the basic theories in confocal laser-scanning microscopic imaging and describe the characteristics of its image signal, because the design of an efficient compression model strongly depends on the properties of the signal. The discussion includes image acquisition techniques, data dynamic range, intra- and cross- frame resolutions, and noise properties. Each of them has a significant role impacting the performance of a compression system.

Chapter 3 discusses several issues in compressing medical images. The significance of a distortion-parameterized approach over a rate-parameterized one will be described. Moreover we review the concerns in measuring the quality of a reconstructed medical image after decompression, including three types of quality measures and three types of artefacts introduced on the reconstructed image. These will help us evaluate the coding performance of our approach and then compare to other medical image compression algorithms.

Chapter 4 reviews the efforts in compressing three-dimensional medical images. The review starts from the standard image compressors, such as JPEG, JPEG2000 and MPEG, and then proceeds to the state-of-the-art three-dimensional approaches based on 3D-DCT and 3D-DWT. The advantages and the deficiencies of each method are discussed, which eventually conduct us to the expected features of our method when coding three-dimensional CLSM images.

From Chapter 5 to Chapter 8 we describe our new distortion-constraint three-dimensional image compression system. Chapter 5 presents an overview of this technique, where we investigate the distortion control property in the closed-looped image pyramid structure and briefly describe how we incorporate vector quantization techniques in this structure to achieve the distortion control functionality and also a high coding performance.

Chapter 6 describes the implementations of a three-dimensional image pyramid transform, where we primarily focus on the design of two band filters – one for subband decomposition and the other for subband composition. We also investigated the desired properties of three-dimensional image pyramid transform for image coding. We discuss the characteristics of the distribution of the pyramid coefficients, which can be exploited by the following vector quantization techniques to improve the coding performance.

Chapter 7 presents a novel vector quantizer – the cross-band classified vector quantizer to capture the inter-pixel dependencies from both intra-band and inter-band according to the distributional characteristics of the pyramid coefficients discussed in

Chapter 6. In this approach we choose a classified vector quantizer for each pyramid level to exploit the non-stationary distributions of the intra-band coefficients; on the other hand we capture the cross-band correlations between coefficients from different levels by sharing their classification maps.

Chapter 8 investigates the optimization algorithms of our compression system. We propose two bit allocation algorithms. They can adaptively and automatically control the parameters of each sub system and eventually optimize the overall compression performance. We also investigate the computational complexity problem of the system, which is one of the issues to be considered in practice. We focus on the complexity improvement in vector quantization techniques because they have been recognized as the most computation-cost part in our system.

Chapter 9 performs a comparison study on the coding performance of our approach compared to those of other standard and the state-of-the-art approaches. We objectively evaluate the following three types of performance. They are: distortion control performance, performance of different multi-frame compression schemes in compressing three-dimensional image data and the distortion-rate performance of a real three-dimensional compression algorithm. The testing data are all volumetric image data sets collected from CLSM devices, which have been described in Chapter 2.

Chapter 10 includes our concluding remarks of this thesis. Possible research orientations in the future are presented, complimenting the perspectives on individual topics made at the end of each section.

Chapter 2 Characteristics of CLSM Imaging

The design of an efficient compression model depends how well it can exploit the properties of the signal. Therefore we will discuss the characteristics of microscopic images obtained from confocal laser scanning microscopy (CLSM)[1] devices, which basically includes image acquisition techniques, data dynamic range, intra- and cross-frame resolutions, and noise properties. Each of them has an important role, which can affect the performance of a compression system.

The principle of confocal imaging was first developed by Marvin Minsky and patented in 1957 [Minsky, 1957]. The major advantage of confocal microscopy over conventional wide-field microscopy is increasing contrast of microscope images, particularly in thick specimens. By restricting the observed volume, confocal technique keeps overlaying and nearby scatterers from contributing to the detected signal. To achieve this, the specimen must be point-wisely illuminated and the instrument must observe only one local point at a time, and then only signals satisfying confocal condition can be detected. One should notice that optical confocal microscopy we investigated is a subset of the extensive and well developed field of microscopy, which also include electron and acoustic microscopy, and other techniques contributing to this field. Moreover our interests are about microscopy in

[1] Sometimes also referred as LSCM – laser scanning confocal microscopy.

medical-purpose study and specimens are usually tissues from human and animals, which different from microscopy in, for instance viewing a semiconductor chip, where the object is totally opaque. The book by Pawley [Pawley, 1995] and the paper by Inoué [Inoué, 1994] give good references for biological confocal microscopy and general microscopy respectively.

2.1 Basic Concepts in Confocal Microscopy

2.1.1 Confocal vs. Conventional Wide-field Microscopy

The basic idea of confocal microscopy is the selection of light from one point and the rejection of light from all other points which leads to the very high contrast images. That increase in contrast happens even when the imaged point is buried deep in a thick sample and is surrounded by other bright points, which is not achievable in conventional wide-field microscopy. To demonstrate this, we begin with a conventional wide-field microscope shown in Fig. 2-1. This diagram illustrates the cells in a thick sample are illuminated simultaneously by a wide-field light source and imaged by objective lens. On the left side of the objective lens is the illumination part. Although the illumination is focused at each cell of sample, it lights up the entire sample, so the illumination of one cell will be influenced by those from other cells nearby. On the other side is referred as the collection part where an image of the thick sample has been formed. If we observed and recorded using charge-coupled device (CCD) or retina at the position of the image plane, the image of one cell in the sample will not come from this cell only, but will be affected by all the cells with a weighted illumination intensity function relative to the position of this cell, and eventually if we suppose the sample consists of several planes it will record a superposition of these planes. Therefore one has difficulty discriminating two close cells from intra- and inter- planes.

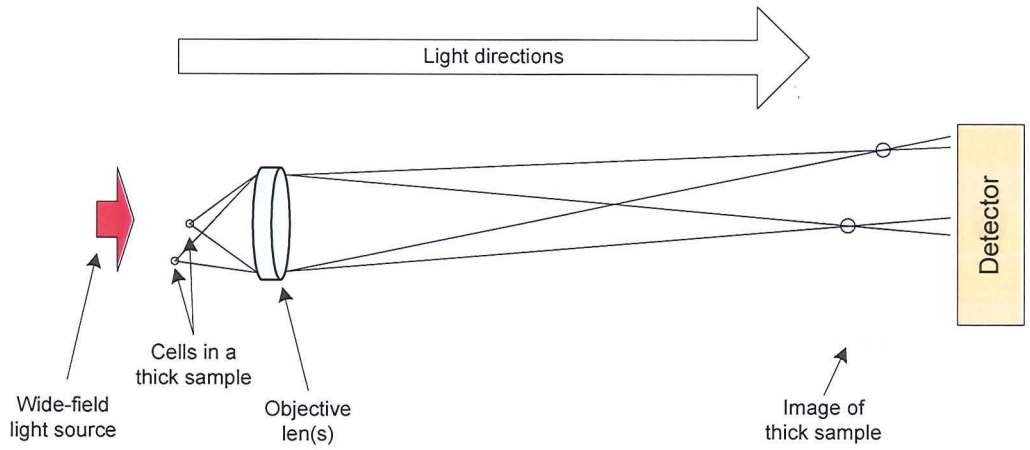


Figure 2-1: A simple view of conventional wide-field microscope.

An alternative to the illumination part of Fig. 2-1 is shown in Fig. 2-2. Instead of a broad light source, we use a single point source of light and focus it on one cell in the thick sample. That focused light illuminate one single point inside the sample at one time, thus restricting the illumination volume around this point. On the collection side, the image of the sample is very bright where the sample was brightly illuminated and dimmer where it was weakly illuminated. So the contrast of the recorded image is improved compared to its wide-field version.

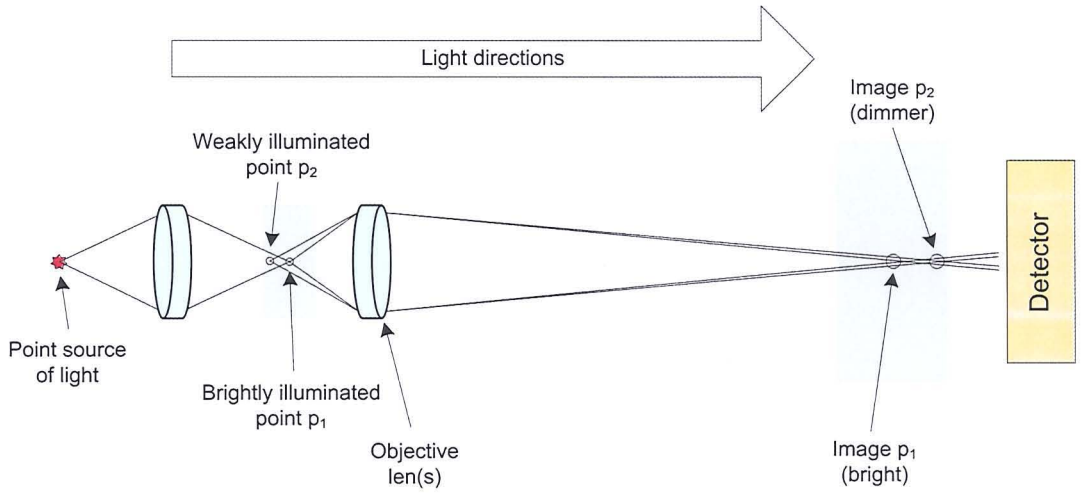


Figure 2-2: An alternative to broad light source with a point light source instead.

Although the image of the point light source brightly illuminates p_1 in Fig. 2-2, it also illuminates other points at least weakly, like p_2. By putting a pin hole at the image plane of p_1, we can effectively block the light from p_2 to the detector, which is illustrated in Fig. 2-3. The pinhole let through only the light from p_1, which is brightly

illuminated by focused point source. Therefore the detector behind the pinhole records the light intensity coming from p_1 and ignores the rest of points in the sample like p_2. Suppose we follow the reverse direction of light, the image of pinhole on the left side of the objective lens will be exactly at point p_1, so both the image of pinhole and the image of point light source are focused on p_1, or we can say they are confocal with each other [Webb, 1996].

So we have considered the features we need to look at a point inside a thick sample. By moving the focused spot over a specimen, we can build an image of thin layer within the specimen without cutting it into slices. This property is usually referred as *optical sectioning* which is one of the most significant features of confocal microscopy.

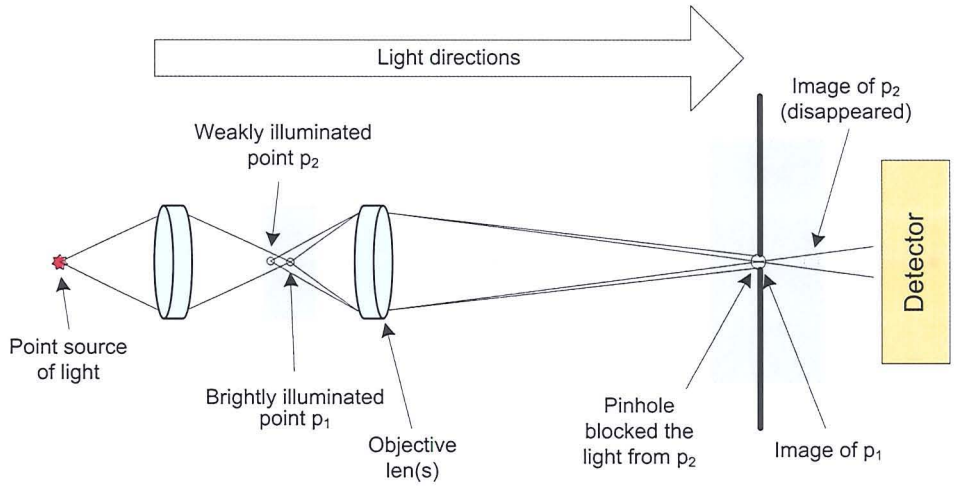


Figure 2-3: A pinhole at image plane of p_1 passes all the light from p_1 while blocks most of the light from p_2.

2.1.2 Resolution and Contrast

Resolution and contrast often work together. They are one of the most important measures and descriptors of any imaging systems. By resolution, we mean the ability of the imaging system to reproduce the contrast of objects of various sizes, or to what extent two objects are separated so they can be described and measured and 'resolved'. The term contrast refers to the differences in intensity within an object or between the object and the surrounding background [Kenneth, 1996].

2.1.2.1 Point-spread Function

Fig. 2-4 shows an optical system consisting of a simple lens. A point source p_0 at the origin of the focal plane produces a spot image at the origin at the image plane. Since we assume the source is a perfect point, no interference occurs. By taking a close look, we have observed a diffraction pattern due to the finite aperture of the objective lens, which is displayed on the right side of the figure. Such pattern is usually called point-spread function (PSF) in optical terms [Born, 1980].

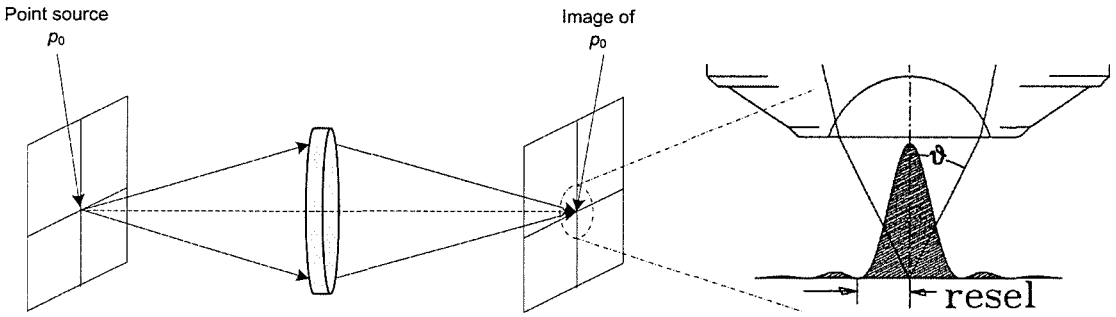


Figure 2-4: The image of a point source and its diffraction pattern at focal plane. The resolution element (resel) is the radius of the first dark fringe in the diffraction pattern.

The shape of the cone in Fig. 2-4 is only determined by *numerical aperture* (NA) of the lens.

$$\mathrm{NA} = n\sin\theta \qquad (2.1)$$

where n is the index of refraction of the medium and θ is the half angle of the cone of light converging to an illuminated spot or diverging from one. NA is the most important descriptor of lens. It describes the angular behaviour of the light cone, and governs the imaging.

The central portion with the first dark fringe is usually referred as *Airy disc*, and its radius fundamentally measures the resolution, it is therefore named as *resolution element* or *resel*. The size of *resel* is determined by NA of the lens and also λ of the illumination source,

$$r_{resel} = 0.61\lambda / \mathrm{NA} = 0.61\lambda / n\sin\theta \qquad (2.2),$$

where λ is the wavelength of the illumination light.

The PSF is a function in three dimensions, but usually lenses and their apertures are symmetric in the planes parallel to the focal plane, so, as shown in Fig. 2-5, only two

parameters are used to describe the three-dimensional PSF(ρ, ζ) [Webb, 1996]. ρ is the in-planer distance to the optical axis z; ζ is the distance between the focal plane and the out-of-focus planes which are parallel with each other. Fig. 2-5(a) is PSF at $\zeta = 0$, that is PSF (ρ, 0) at the focal plane and is actually a two dimensional function by ignoring φ because the distribution of PSF on the focal plane is isotropic. For fixed ζ and $\zeta = c \neq 0$, PSF (ρ, c) is a defocused version of PSF (ρ, 0), it is still a pattern of rings centered at the origin, but energy will be more scattered and not highly concentrated at the origin as that on focal plane. More precisely, an example of the distribution on PSF in three dimensions is illustrated in Fig. 2-5(b).

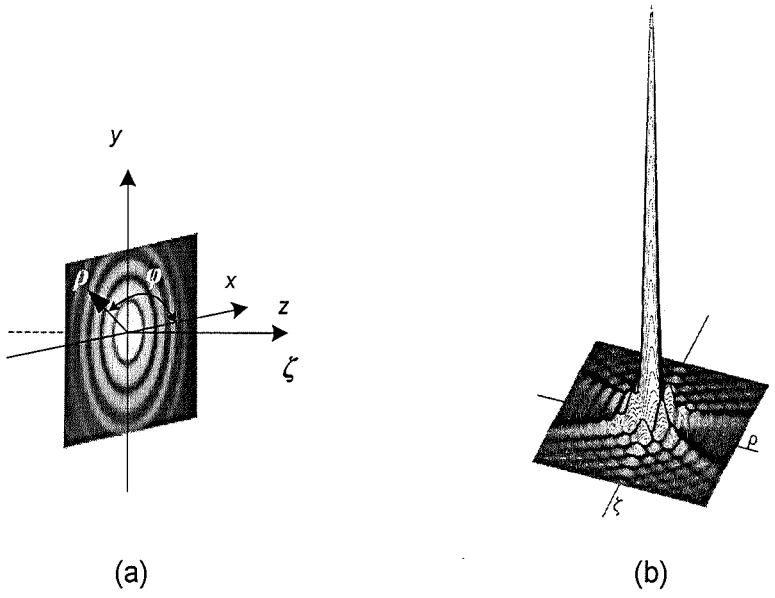


Figure 2-5: (a) PSF is a function in three dimentions *with respect to* (ρ, φ, ζ) and here is an example of PSF (ρ, φ, 0); (b) we can ignore φ and express the point-spread function as PSF (ρ, ζ) in three dimensions.

2.1.2.2 Resolution and Contrast in Conventional Microscopy

The PSF is a measure of resolution because two self-luminous points viewed by a microscope appear separate only if they are far enough apart for their PSF to be distinct. Both self-luminous points display diffraction patterns and they will be distinct only if they are far enough to display a reasonable dip between the two peaks. In Fig. 2-6, we illustrate three different presentations with distance d0, d1 and d2. Rayleigh [Castleman, 1995] gave a criterion of the distance as 'resolved', which is dRayleigh = rresel and the dip is about 26%. Ideally, the right figure can still be

resolved with much dimmed contrast, but in the presence of noise, such dips will be corrupted by unavoidable noise and they will not be distinguished in real situations.

From equation (2.2), the resolution in microscopy is only determined by wavelength of illumination light and NA, the most significant descriptor of lens. Therefore, an imaging system with large NA and small λ produces a smaller size of resel than that with small NA and large λ, and therefore a higher resolution.

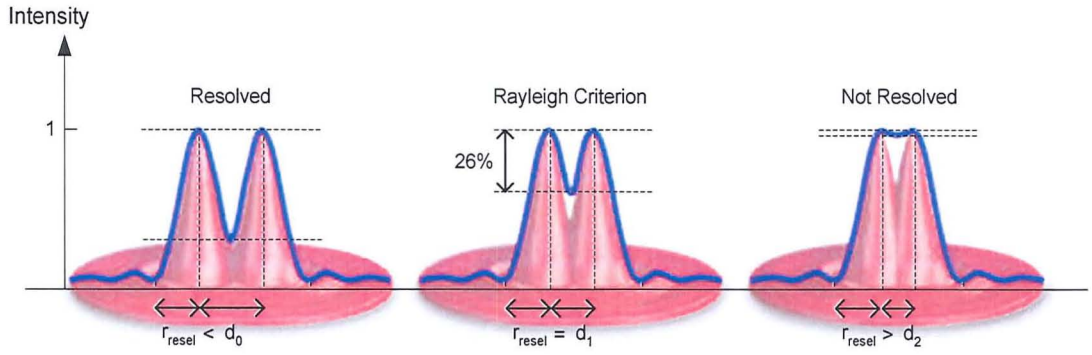


Figure 2-6: Rayleigh criterion for two self-luminous points can be resolved.

In Fig. 2-6 the contrast is the dip we need to resolve two points. It can be characterised by the intensity relative to the bright peak and the dim dip:

$$C = (I_b - I_d)/(I_b + I_d) \qquad (2.3)$$

where I_b is the intensity of bright peak and I_d is the intensity of the dim dip. It has been pointed that in real situations it is noise not really contrast determines the visibility [Webb, 1996]. For instance, one can easily tell black letters written on a white paper by daylight, but not in a dark environment even the contrast of the object is never changed. This is because the so-called photon noise [Castleman, 1995], where the random arrival times of the photons is a Poisson process and the variation is related to the square root of the number of photons excited. Therefore the more photons accumulated the smaller ratio of noise to signal (refer to Section 2.3.1) and the more easily the threshold can be set to distinguish the bright objects from the dimmer ones. This is why we can finally read the paper after we stay in the dark for a while and our pupils are adjusted to accept more light. In wide-field illumination, the object is not highly point-wise illuminated, therefore the photon noise could corrupt

the visible contrast even if two illuminated points satisfy Rayleigh criterion. Worse, when a point with less brightness is obscured by the interference brought from neighbouring much brighter points, it will not be resolved at all.

2.1.2.3 Resolution and Contrast in Confocal Microscopy

In confocal microscopy, we have two point-spread functions: PSF_s for the illumination source and PSF_p for the pinhole. Because the images of pinhole and light source are confocal (see Fig. 2-3), the point-spread function of this confocal system PSF_{conf} is simply the product of two PSFs. In the arrangement like Fig. 2-11, both illumination and observation use the same optics with point-spread function PSF_o.

$$PSF_{conf}(\varsigma,\rho) = PSF_s(\varsigma,\rho) \times PSF_p(\varsigma,\rho) = PSF_o^2(\varsigma,\rho) \qquad (2.4)$$

By self-multiplying PSF_o in Fig. 2-5(b), we have a much more peaked version in Fig. 2-7.

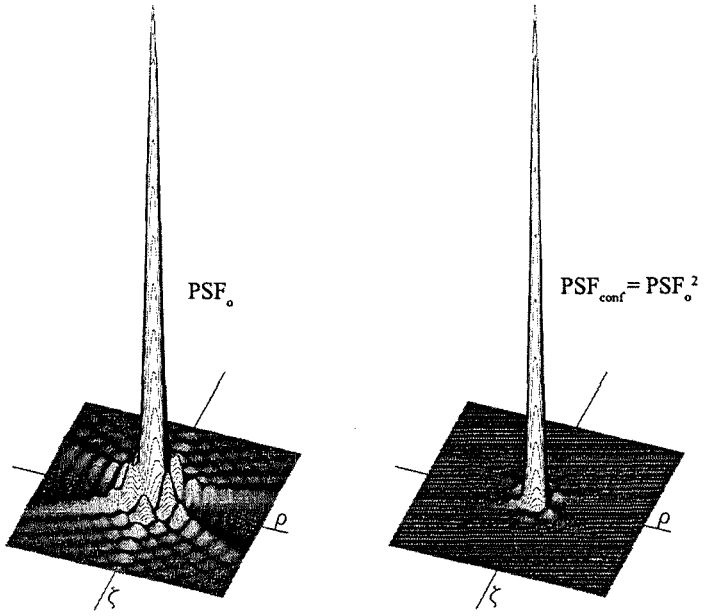


Figure 2-7: the PSFconf is the self-product of PSFo, which shows a highly peaked signal at origin with much reduced diffraction effects.

Following Rayleigh criterion as 26% dip between adjacent peaks, the lateral and axial resolutions for the confocal case are

$$\Delta r_{conf} = 0.72 r_{resel} = 0.44\lambda / \mathrm{NA} \qquad (2.5),$$

and

$$\Delta z_{conf} = 1.5 n\lambda / \mathrm{NA}^2 \qquad (2.6).$$

It is worthy noting that what confocal scheme achieved is significant contrast enhancement rather than improvement of resolution when compared to equation (2.2). In fact, as we noticed in Fig. 2-7, the diffraction effects in confocal case are substantially suppressed. The unwanted 'overlapping' in two closed points of very different intensity, which means in the wide field the intensity of a dimmer object will be overlapped by the diffraction pattern from its neighbouring much brighter object, can be greatly reduced (Fig. 2-8).

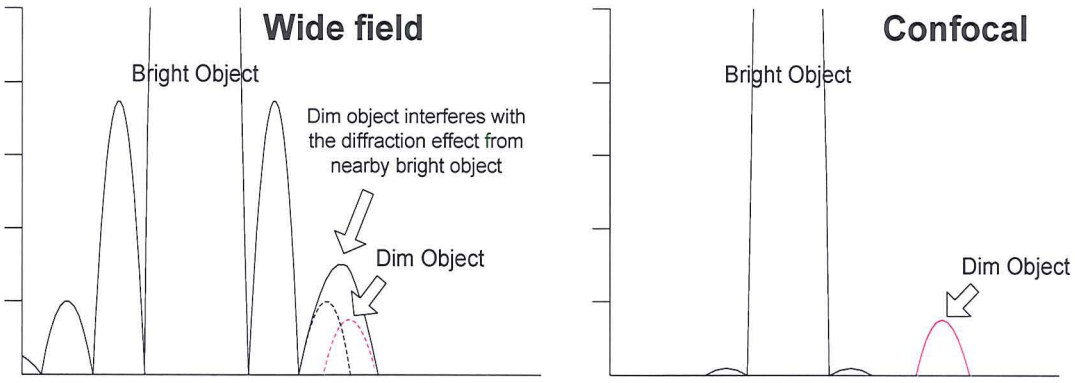


Figure 2-8: Two objects with very different intensity which are apart enough to resolve. In wide field (left), the dimmer object is obscured by the diffraction pattern from the brighter object and cannot be observed clearly; but in confocal case (right), the contrast of the dim object can be faithfully recorded and measured.

'Scanning' is one of most significant features in confocal microscopy, where one tiny volume within an object is sequentially illuminated and observed at one time by the image of a point light source. No (or negligible amount) energy from the light source reaches the object outside this volume, so no energy re-emitted from the object reaches the detector, and therefore the contrast is not spoiled by unwanted background noise from neighbouring points on the focal and out-of-focal planes.

The contrast enhancement, both in focal plane and along axial direction, is the heart of confocal microscopy. Therefore, at the focal plane, we can achieve a much clearer version than the image obtained by conventional wide field microscopy where out-of-focus light would obscure details and spoil the contrast. Moreover along the optical axis a thick specimen can be imaged as a series of optical sections at different depth. This is achieved by suppressing the signal from out-of-focus planes in confocal case,

and subsequently three-dimensional visualization and modelling can help people with more accurate analyses and measurements of the real object.

2.2 Critical Aspects of Fluorescence CLSM

We have described the basic concepts of confocal microscopy with optical physics theories. To convert the theory into a practical technique, several issues need to be studied. Fig. 2-9 illustrates a practical confocal microscopic imaging system.

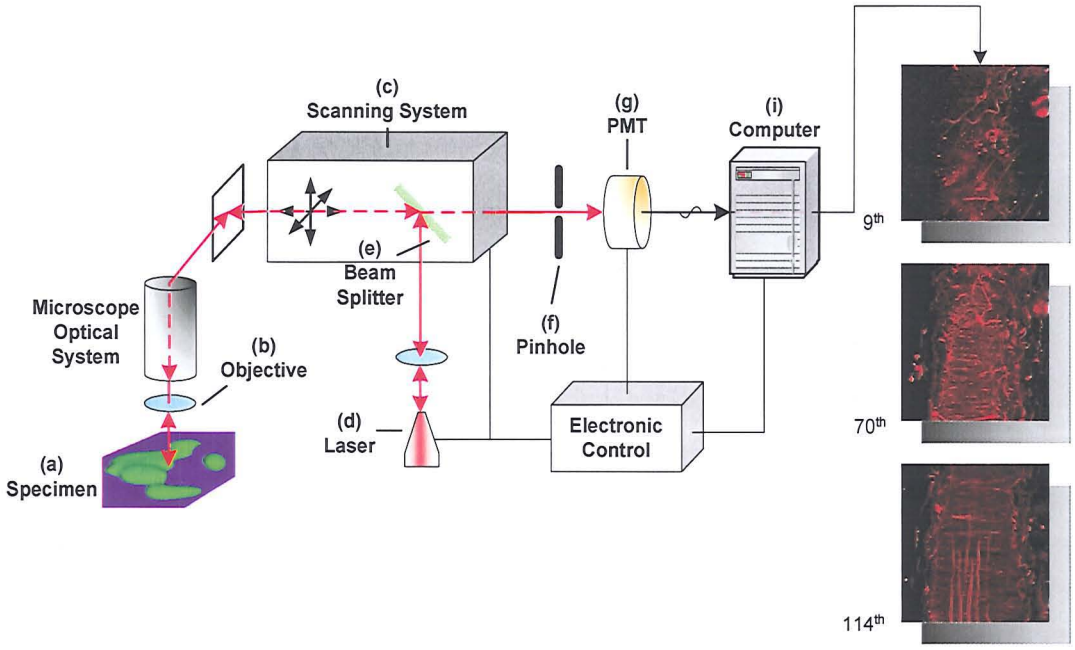


Figure 2-9: A generic confocal microscopic imaging system.

2.2.1 Specimen Preparation

Fluorescent imaging, which has been developed over many years for use with the conventional wide-field microscope, has been enhanced in confocal microscope. Due to the optical sectioning achieved by the confocal approach, fluorophores are stained within the thick specimen as compared to the epifluorescent in conventional microscope (Fig. 2-9(a)). The penetration of a focal spot into specimen is greatly influenced by the specimen properties. CLSMs usually have approximately 100~150 μm penetration, and sometimes a preparation of thick sections from specimen using a microtome is need if sufficient laser penetration cannot be achieved.

Apart from contrast enhancement and optical sectioning, another benefit of the confocal microscopy is that multiple images can be collected simultaneously. Since they are well registered with each other, these images, which are originally collected and displayed as pixel intensities in different colour channel to distinguish different parts in the specimen, can be merged into one multiple colour image to visualize the distributions, interconnections and behaviours of these parts.

2.2.2 Illumination

Anything that can emit photons onto the object can be used as the illumination source; most microscopes, however, use lasers (Fig. 2-9(d)). Laser light is highly coherent, which is not preferred in conventional microscopes since it will cause unwanted interference effects and degrade the images, we can treat it as incoherent because the specimen is point-wisely illuminated by a moving focus. Therefore when one spot is illuminated no other spots are illuminated, and interference cannot occur. A consequence of this is that there could be no speckle effects in imaging with a scanning laser and speckle is not a problem in confocal microscopy.

We can benefit from lasers since they have high radiance and are monochromatic. High radiance means a large amount of energy (or photons) is concentrated in a small volume, for instance a small laser putting 1 mW into its diffraction limited beam will deliver that full mW to the focal spot [Webb, 1996 (pp. 449)]. When viewing objects with fluorescent dyes, it becomes particularly important. Since the emission of photons from fluorophores is a stochastic process, the recorded image quality is primarily dependent on the number of photons accumulated. With the normal scan speed a point in specimen is illuminated within a very short period, therefore the fluorophores can easily emit enough photons to produce good quality images if we use high radiance laser light with moderate power; otherwise one has to choose a much more powerful light source which is expensive and might do harm to the specimen.

A monochromatic light source is perfect for fluorescent imaging, because in fluorescence imaging the observation is always by light of a longer wavelength than that of illumination, and one can use a beam splitter – a dichroic mirror (Fig. 2-9(e)),

which reflects light of short wavelengths and transmits light of longer wavelengths, to make the microscope epitaxial, which means that illumination light comes from the observation side. This arrangement is parsimonious by using one objective lens twice and also the image of the fluorescent objects will not be affected since any illumination light can be rejected.

2.2.3 Scanning objects

The simplest way to get one image of a planar section of the object is to leave the object fixed and move the stage, which is usually referred as stage scanning. This has advantages in good field of view and all the lenses work on axis. However the primary drawback to this design is that it is slow and cannot be made faster, for instance to obtain a 512 by 512 image, five seconds is typical [Webb, 1996 (pp. 447)]. Moreover it is not a happy environment for many samples due to the large acceleration in the forth-and-back movement of stage. For these reasons it is sensible to move the laser beams rather than the object.

Most recently developed confocal microscopes are moving laser-beam scanners (Fig. 2-9(c)). By subtly controlling the angles of the beams, the focal spot will scan the specimen at the focal plane. Moreover three-dimensional scanning can be accomplished by a computer-controlled stepping motor to coordinate step-by-step changes of the focal plane along z-axis with sequential image acquisition at each step. CLSMs can easily scan a single optical section within half second, but because several replicates are usually repeatedly scanned and averaged to improve the signal-to-noise ratio it takes approximately one second to acquire an acceptable section image.

Studies of living cells are enhanced by the improvement of imaging made possible with CLSM, but are still substantially more difficult than imaging fixed specimens. This is because more factors should be considered to make the living cell tolerate to the conditions involved. To balance the trade-off between spatial resolution and temporal resolution, people have to use a wider pin-hole to collect more photons to reduce the photon errors with the penalty of some loss of spatial resolution, also use acousto-optical devices or disk scanning techniques which are faster than

galvanometer-driven point scanning. These faster designs make the CLSM system work near video frame rate at full screen resolution.

2.2.4 Detector

Detectors (Fig. 2-9(g)) in confocal microscopy are mostly photomultiplier tubes (PMTs). The PMT has the advantage of good sensitivity to the visible light source, where the detected signal is proportional to the quantum efficiency which in most confocal instruments drops within the range from about 15% in the blue end to 4% in the red end of spectrum. For these spectrum ranges, compared to other photon detectors, such as CCD, photomultiplier tubes have least dark noise which is the noise coming from the detector itself and cannot be avoided, especially in the cases when very few photons are available to be detected.

The PMT voltage determines the amplitude of the output signal, and finally the pixel intensity value after analog-to-digital (A/D) conversion using a computer (Fig. 2-9(i)). Two parameters of PMT – gain and offset, are used to control the output image. The gain is the factor of PMT voltage to the quantized pixel intensity value, for instance an increase of 30 volts corresponds to one increment of pixel intensity, which determines the contrast of the digitized image. PMT offset is the black level of the image and determines the brightness. It should be set so that the background is black. One can do this by turning up the offset during scanning until the background develops some brightness and then slowly turn the offset down until the brightness fades to black. By actively adjusting the offset and gain, one can collect a high contrast image of the real object.

A/D converter usually outputs an 8-bit datum for each pixel value and for some equipment the pixel depth can reach up to 12-bit per pixel. The 12-bpp outputs are especially used for higher image quality since the larger the dynamic range of data the higher the signal-to-noise ratio. The digitized image can have one colour channel by using only one PMT; or multiple channels by merging together more single-channel images collected from each PMT equipped with different spectrum filter.

2.2.5 Multi-photon Laser Scanning Microscopy

Multi-photon laser scanning microscopy (MPLSM) is an improvement of CLSM, which is primarily featured for its sharper contrast and deeper penetration than CLSM [Pawley, 1995]. The basic difference of configuration between MPLSM and CLSM is the illumination source. In MPLSM solid state lasers are used, whose intrinsic property is that they give off photons in pulses, e.g. 10 ns interval for a titanium sapphire laser, rather than a steady stream of photons in CLSM. Also, the solid state lasers give light off at a longer wave length than the gas lasers use in CLSM.

The basic idea behind the MPLSM is that because of the low energy of light when a probe absorbs a photon from the first pulse of light the electron is not elevated completely to an excited state. Rather it is elevated to an intermediate or "pseudo" excited state. From this state it will fall back to the ground state (s_0) without giving off any fluorescence. If, by chance, the probe is to absorb a second photon, coming from a subsequent pulse of the laser, then the electron can be elevated to the true excited state. Consequently, florescent light will be given off as it returns to the ground state. In practice three or four sequential events are often required, referred to as multi-photon events and here comes the name multi-photon laser scanning microscopy.

What does MPLSM benefit from such multi-photon events is that the region excited by MPLSM is much smaller than that excited by CLSM, which results in a much more peaked point-spread function. As we discussed in Section 2.1.2.3 this peaked shape means much reduced energy from scatters and therefore gives greatly improved contrast in all the three dimensions. The reason the fluorescence is so restricted in MPLSM is because the probability of a multi-photon event is extremely low and occurs only within a very small region centered at the focal point. For instance the possibility of a two-photon event is 10^{-15} times as that of one-photon event, which is the situation in wide-field and confocal microscopy. A three-photon event is even rarer. So, in MPLSM only within the regions, where the laser light is most intense i.e. at the focal point of the laser beam, do we find multi-photon events. An example of two-photon event was illustrated in Fig. 2-10. In fact, the region of illumination is so restricted around the focal plane that no pin hole is required in MPLSM.

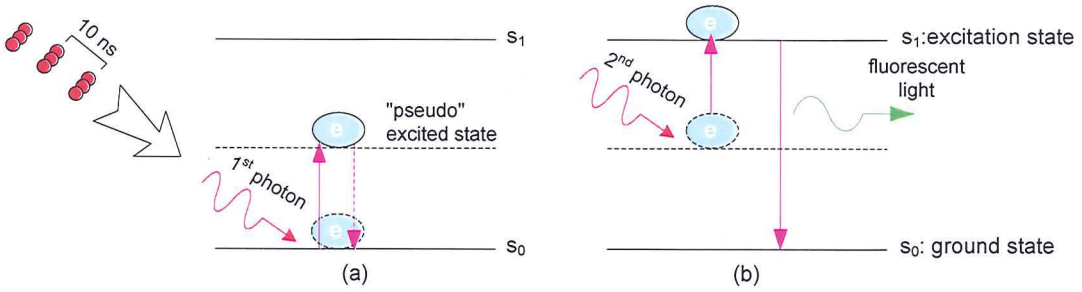


Figure 2-10: An example of two-photon event. (a) An electron is elevated by the first photon to a "pseudo" state without giving off any fluorescence. (b) The same electron is consecutively elevated from the pseudo state to the excitation state by the second photon before it fall back to the ground state and fluorescent light will be given off.

There are several advantages of MPLSM over CLSM. First of all, MPLSM is more sensitive in contrast than CLSM because all the light generated to make an image is sent directly to the photon multiplier tube, which is different from CLSM where a pinhole is placed in front of the detector. In CLSM there is considerable loss of signal in the optics required to direct the light to the pin hole. Therefore MPLSM gives a sharper image than CLSM because of the concentration of focal spot light and the lack of extraneous light, and thus the higher signal-to-noise ratio. Moreover the longer wave length used in MPLSM has an advantage because it is inherently less damaging to biological material than shorter wave lengths. Short-wave length laser light used in CLSM is known to produce free radicals which can damage biological material. In addition, the longer wave length light is more penetrating. It has been estimated that MPLSM penetrates five times deeper into biological material (up to 500μm versus 100μm for CLSM). This point is an important consideration when one requires more penetration in tissues.

In fact in many cases, at least in the vascular remodelling studies, what people primarily benefit from MPLSM over CLSM is its more penetration in observing tissues rather than its contrast enhancement [Daly, 2002]. This is because it is possible to achieve the same contrast level in CLSM as that in MPLSM by carefully configuring the imaging system and the use of deconvolution methods to improve the image quality [Arribas, 1999; Pawley, 1995]. This can be seen from examples of data sets in the following sections in this chapter, captured by CLSM and MPLSM

respectively. Moreover it is more expensive for both of the equipments in MPLSM and the preparation in imaging. Therefore MPLSM is used where deep penetration is required; otherwise CLSM is used in most cases.

In the consideration of image compression, we did not regard images by MPLSM having different characteristics from those by CLSM. In fact MPLSM can be viewed as a special case of CLSM, where "confocal" is achieved from consecutive individual photons in time domain rather than two PSFs in CSLM in spatial domain. Therefore both are considered as CLSM techniques of which what we want to emphasize is their contrast enhancement property in three dimensions. Without particular indication, we will generally use 'CLSM' in the following discussion, which could be either CLSM or MPLSM.

2.3 Vascular Remodelling by CLSM

Vascular remodelling can be considered a physiological process in the sense of constantly changing the vasculature in response to different activities and environmental stimuli. It is related not only to the gross structural differences such as an increase or reduction in lumen and wall dimensions, but also modifications at the cellular level, such as cell number, distribution, phenotype, and orientation [Arribas, 1999]. The study of vascular remodelling has gained a great deal of attention in recent years. This is because it can contribute to a wide range of studies, such as the pathophysiology of circulatory disorders, angiogenesis (elaboration of vessels from pre-existing ones) and vasculogenesis (de novo formation of vessels) in the embryo, adaptations to flow during pregnancy or exercises, and so on. Traditional methods, either physiological or histological, have their pitfalls, which can only analyse the vascular structure on vascular mechanics or on isolated cells. Confocal microscopy techniques, however, allows for the simultaneous study of vascular structure and function, which could give people information on what cellular events take place during vessel vasomotion. Superior to conventional fluorescence microscopy, CLSM produces optical sections throughout translucent tissue, commonly to a maximum depth of about 200 μm (≈400 μm in MPLSM), without the need for cutting thin slices. They eliminate blur and flare from out-of-focus planes in an object and axial

resolution is improved greatly. Therefore novel applications for the study of blood vessels by identifying cell type, distribution and viability have been developed [Arribas, 1999; Daly, 2002].

2.3.1 Four Groups of Data Sets

Four groups of CLSM data sets have been considered in the evaluation of compression applications. They are captured by the VASCAN-2000 project [VASCAN, 2000] for vascular imaging and *vascular remodelling* studies described above. Each group has typical image properties that are different from those of images from other groups, and therefore they were chosen as prototypes of CLSM images for the purpose of evaluating compression performance. To avoid repeated description we will label and describe them in this section and their labels will be used as references when they appear in the following sections. Readers can refer to the corresponding data set in this section whenever they would like to know more detailed information.

All the four groups of data sets are captured with Fluorescent-CLSM, either optical laser-scanning confocal microscopy or multi-photon laser-scanning microscopy. Specimens are tissues from human gluteal arteries and rat mesenteric arteries. They are fixed on the stage such that the imaging plane is parallel to the direction of blood flow. Specimens are stained with single type of fluorophore (single-labelled) and therefore images have one colour channel and generated as 256 gray levels for each pixel, or 8 bits per-pixel. The imaging properties for each group of data set are listed in Table 2-1 and the detailed information are described in the following sub-sections.

Table 2-1: Imaging properties for each group of data set.

Group labels[1]	Description	Image acquisition specifications	Image size (pixel)	Pixel size (μm/pixel)
BxCGP	rat mesenteric artery	ImT[2]: Confocal Obj[3]: x40	X= 256 Y= 256 Z(frames) ≈ 180	X= 0.98 Y= 0.98 Z= 0.35
C5	rat mesenteric artery	ImT: Two-photon Obj: x20	X= 256 Y= 256 Z= 147	X= 1.0 Y= 1.0 Z= 1.0
Gx_HG70	human gluteal artery	ImT: Confocal Obj: x40	X= 512 Y= 512 Z ≈ 100	X= 0.2 Y= 0.2 Z= 0.5
Hum_Glt_x	human gluteal artery	ImT: Two-photon Obj: x20	X= 512 Y= 512 Z ≈ 85	X= 0.6 Y= 0.6 Z= 1.0

[1]: symbol 'x' in the label is a number or a character to identify different data sets which are of the same group.

[2]: ImT: imaging technique used for capturing - optical laser-scanning confocal microscopy or multi-photon laser-scanning microscopy.

[3]: Obj: objective magnification factor.

2.3.1.1 BxCGP Group

Data sets in BxCGP are obtained from rat mesenteric arteries by optical laser-scanning confocal microscopes. A x40 oil-immersion objective (Nikon, NA=1) was used. The tissue was stained with fluorescent BODIPY-CGP12177 (a fluorescent beta-adrenoceptor antagonist drug), which attaches to beta-adrenergic receptors. Therefore, it shows the distribution of CGP binding sites, where one can see the drug binding to cells and nerves. Each image was digitized as 256x256 pixels, and the pixel dimensions are 0.98μm/pixel in planar and 0.35μm/pixel along the optical axis. An example of B3CGP data set from this group is illustrated in Fig. 2-11.

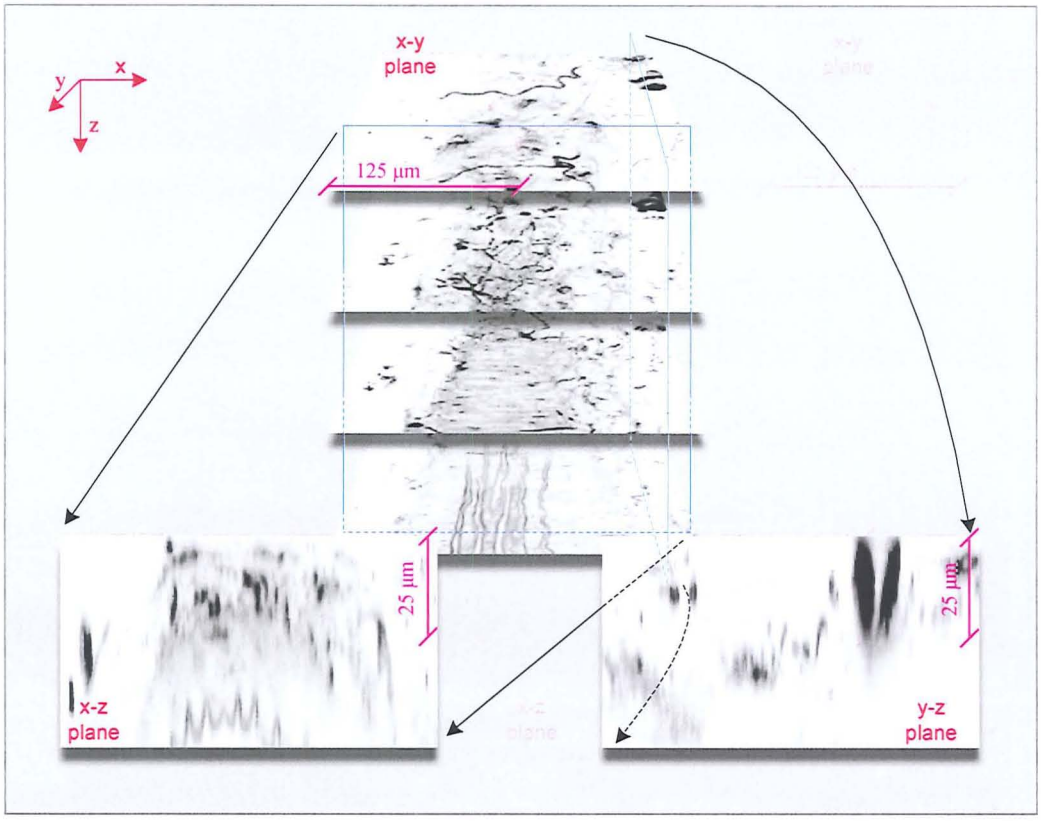


Figure 2-11: B3CGP (256×256×147) data set from B×CGP group. Pixel intensities are inversed for visualization purpose. Original images were collected in x-y plane while x-z and y-z images were obtained by stacking original images and projected to x-y plane and y-z plane, respectively.

This data set captures a portion of the vascular wall of rat mesenteric arteries with volume size about 250×250×50µm^3. It begins with the outermost side of the wall where one can see adventitial cells and nerves (frame 1~50); in the middle one can see the distributions of smooth muscle cells (frame 50~83) and in the innermost side of the wall endothelial cells are visible (frame 83~147).

2.3.1.2 C5 Group

Data sets in this group are also obtained from rat mesenteric arteries, but a multi-photon laser-scanning microscope was used for deeper penetration. A ×20 oil-immersion objective (Nikon, NA=1) was used. The tissue was stained with propidium iodide dyes, which intercalate with DNA of cells having a permeabilized membrane. Since DNA exists only within nuclei, it shows the distribution of three types of cells: adventitial cells, smooth muscle cells and endothelial cells. Each image was digitized as 256×256 pixels, and the pixel dimensions are 0.98µm/pixel in planar

and 0.35μm/pixel along the optical axis. An example C5 data set from this group is illustrated in Fig. 2-12.

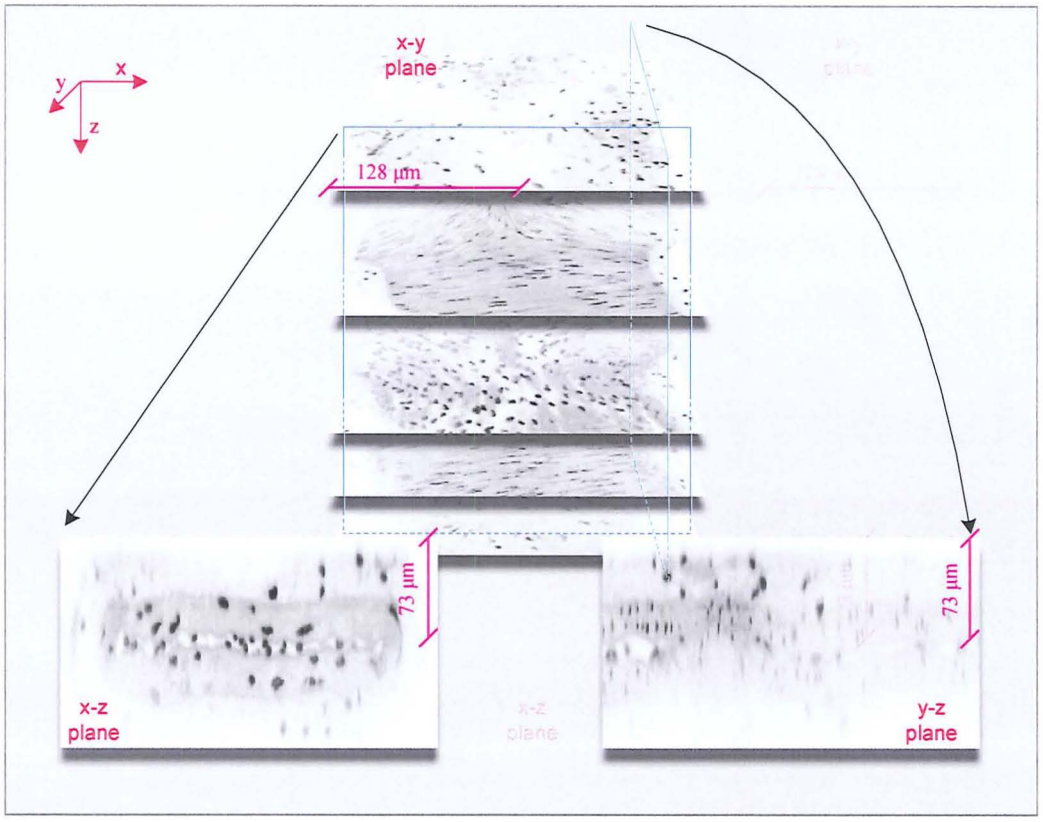


Figure 2-12: C5 (256×256×147) data set from C5 group. We follow the same visualization methods as used in Fig. 2-11.

This data set captures a joint portion of rat mesenteric arteries and describes the distributions of three types of cells within the vascular wall. The image volume size is about 250×250×150μm^3, which benefit from the deep penetration of MPLSM. The entire transverse section of blood vessel can be clearly observed from x-z planes, where round-shaped adventitial cells are located at the outermost side of the wall (frame 1~37, 114~144); the next inside are long and thin smooth muscle cells (frame 37~65, 84~114) and in the centre part of vascular wall are oval-shaped endothelial cells (frame 65~84).

2.3.1.3 G×_HG70 Group

This group of data sets are collected for studying the characteristics of the different cells in the wall of a blood vessel. These characteristics may include shape, length, breadth, area, volume, etc. An optical confocal microscope equipped with a ×40 oil-

immersion objective was used to capture a small portion of vascular wall of human gluteal arteries. The tissue was stained with propidium iodide dyes to show the shape of nuclei of cells. Each image was digitized as 512×512 pixels, and the pixel dimensions are 0.6μm/pixel in planar and 0.5μm/pixel along the optical axis. An example G27_HG70 data set from this group is illustrated in Fig. 2-13.

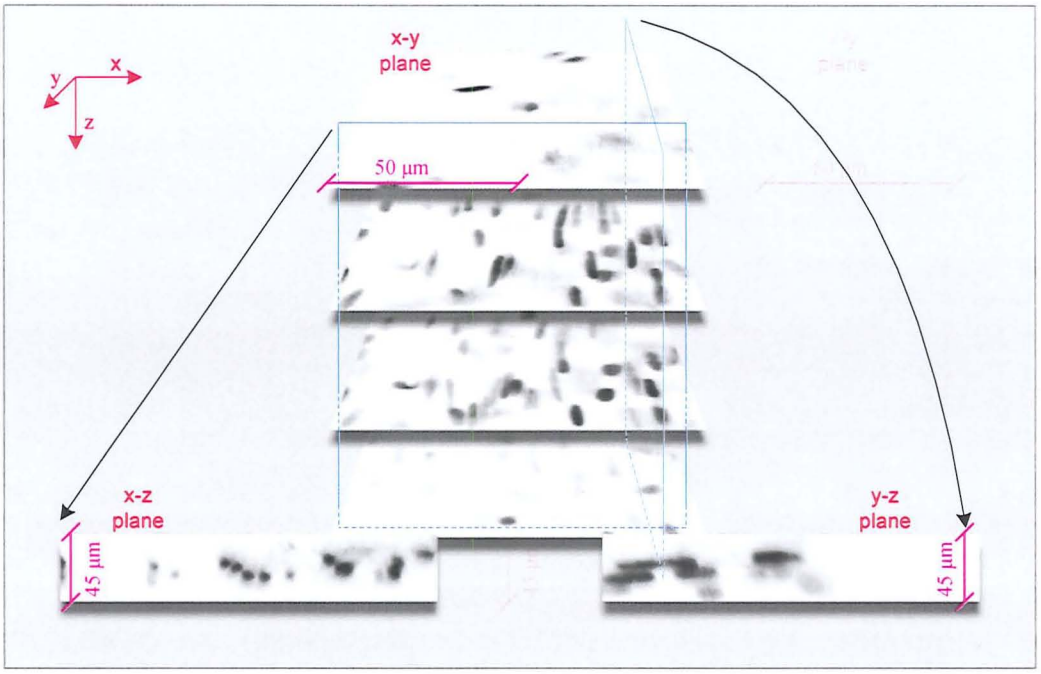


Figure 2-13: G27_HG70 (512×512×89) data set from G×_HG70 group. We follow the same visualization methods as used in Fig. 2-11.

The image volume size for G27_HG70 is about 100×100×45μm^3. Cells displayed on x-y plane in this data set are about 5 times than that from C5. Therefore compared to C5, one can perform more accurate measurements on cells by using this data set.

2.3.1.4 Hum_Glt_× Group

The last group of data sets considered are labelled as 'Hum_Glt_×', which, like G×_HG70, are also obtained from human gluteal arteries, but by a two-photon laser-scanning microscope. A ×20 oil-immersion objective (Nikon, NA=1) was used. The tissue was stained with propidium iodide dyes and captured as 512×512 pixel in x-y plane with pixel dimensions as 0.98μm/pixel in planar and 0.35μm/pixel along the optical axis.

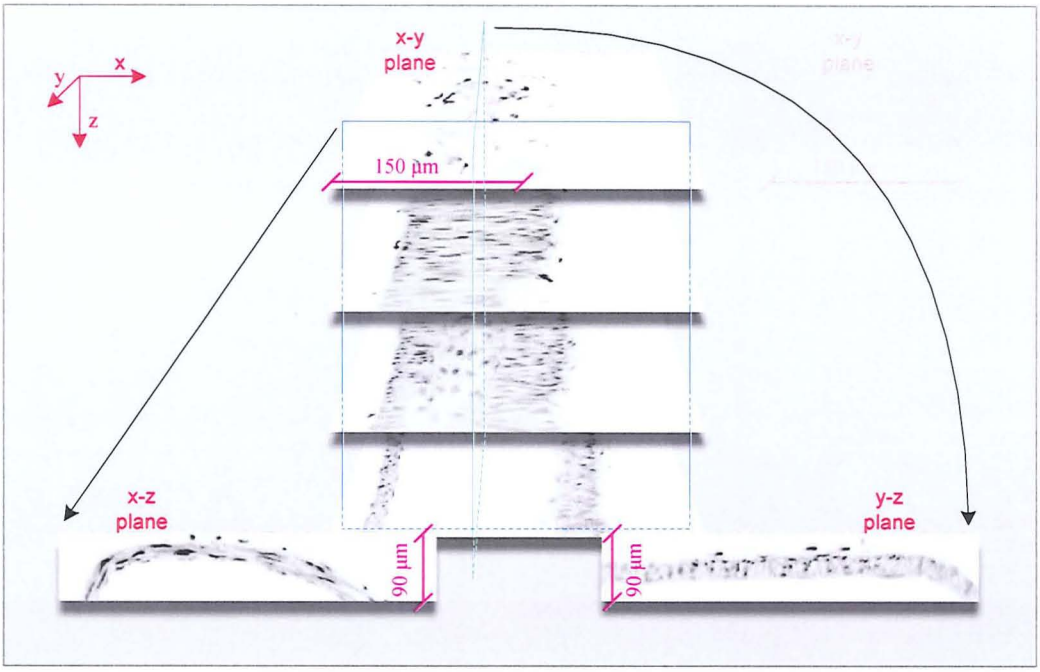


Figure 2-14: Hum_Glt_C (512×512×94) data set from Hum_Glt_× group. We follow the same visualization methods as used in Fig. 2-11.

An example data set labelled as Hum_Glt_C from this group is illustrated in Fig. 2-14. The image volume size for Hum_Glt_C is about 300×300×90µm^3. From x-z plane we can see that it captures the upper-half portion of the vascular wall from the surface to lumen. The cell sizes displayed on x-y plane in this data set are about 3 times than that from C5. Therefore this data set gives a clear description in the sense of vascular remodelling study, where both the distribution of cells and the shape characteristics of each individual cell can be measured.

2.4 Image Quality Analysis

The quality of image could be expressed as how well the interesting parts are presented on the image which might be disturbed by the useless and annoying signals, like noise. Quality of an imaging system is important for analysis. Moreover, in image coding applications the quality of the acquired source images will influence the efficiency of a coding system significantly because of the presence of noise, which however has no meaning to a microscopist. Thus the denoising feature of a coding system that can reduce the noisy signal is worthy of consideration.

2.4.1 Determinants of Fluorescent CLSM Image Quality

Although the 'confocal' scheme in CLSM has improved the contrast of an image by rejecting the out-of-focus signals using point illumination and pinhole, it also greatly reduces the number of photons reaching the photon detectors compared to that in the wide-field case and thus the photon noise mentioned in Section 2.1.2.2 will become more severe. There are several aspects that determine the image quality of fluorescent CLSM images.

(a) Imaging system configuration.

This includes the configuration of optical systems like objectives and pinhole to control the image resolution (see Section 2.1.2.3), the setup of gain and offset of PMT to control the brightness and contrast of the digitized image (see Section 2.2.4) and the preparation of specimen like fixation, environment temperature and fluorophores mounting. These aspects directly influence the output image quality, however when evaluate the performance of our compression system and assess the quality of images used in experiments we will not consider these issues because such errors can be reduced to minimum if they are well prepared during image acquisition and we always assume the defects of the images are from unavoidable noises that we will discuss in the following.

(b) Noises. There are two major types of noises in CLSM.

a. *Signal-dependent noise* is noise RELATED to the signal. It is the photon noise we mentioned in the beginning of this section and also referred as *shot noise*. Because of the quantum nature of light, every now and then a photon will arrive at the detector with a fixed probability P_{phn} and all the photons know nothing about each other, so the probability of arrival of one single photon is not influenced by the others. The mathematical description for this random process can be referred as "Poisson Process". In [Castleman, 1995] we noticed that the Poisson noise has the property such that it is in proportion to the square root of the number of photons emitted:

$$N_{noise} \propto \sqrt{N_{signal}} \qquad (2.7),$$

and the signal-to-noise ratio is:

$$SNR \equiv {N_{signal}}/{N_{noise}} \propto \sqrt{N_{signal}} \qquad (2.8).$$

It is therefore only related to the intensity of the signal; for instance a signal pixel intensity 64 would have a possible value ranging as [64-8, 64+8] because of the shot noise and for intensity 256 the range would be [256-16, 256+16]. Thus the variation of the expected value related to the square root of its intensity would be reduced as we increase the pixel intensity or the number of photons counted and the better the signal.

b. *Signal-independent noises* are NOT related to the signal. They can be *dark noise* from PMT which means the unexpected output in the absence of excitation; *stray light* from illumination source and environment; fluctuations in laser power. These noises are independent from signal, which differ from photon noise. For instance, compared to standard illumination level, if we collect the same amount of photons using a much dimmer illumination and much longer pixel dwelling time (slow scan), the photon noise is the same but the image quality will be severely defected by signal-independent noises like stray light.

Noise, even though unavoidable, can be reduced by increasing the dwelling time per pixel or the number of image replicates collected and then averaged. The former increase the photon count at each pixel and therefore will reduce the photon noise and increase the signal-to-noise ratio; and the latter will reduce both signal-dependent and signal-independent noises because the variation will fall and the average will finally approximate the true signal as the number of replicates increase.

2.4.2 Assessment of Image Quality

Image quality in an imaging system can be described as how adequately can the displayed image represent the objects of interest [Castleman, 1995]. It can be determined by many aspects, which in confocal microscopy imaging include optics configuration, illumination, detectors and so on. A high quality image, high

resolutions and good contrast for instance, may well illustrate the objects. Though because of the noise and limits, people cannot exactly describe a real object, its high quality image obtained from imaging system gives them a reasonably clear impression on the real object. This will finally lead them to know about the real object more adequately.

A lack of standard image quality assessment means there is no acknowledged routine that can quantify the image quality. As we discussed above, the quality which is determined by the imaging system configuration can be assessed by eye. However, in most cases the image quality is determined by noise (first type or second type mentioned above). An objective measure which is proportion of the signal to noise – the signal-to-noise ratio can quantify the quality of image but this is rarely used. This is because we have photon noise that depends on signal and usually is the major noise factor, and then it is difficult to distinguish noise from signal.

Recently, a new method proposed by Daly [Daly, 2002] based on comparison between image replicates can be used as a good candidate to assess the quality of noisy images. It derived from a reasonable assumption that if free from noise image replicates will be almost identical with same imaging system configuration. In real experiments the degree of mismatch is employed as the measure of image quality. This can be displayed as a two-dimensional histogram which maps the occurrence of the pairs of intensities, one from the pixel at the same position in each replicate. For identical images the histogram will show a single line at 45 degrees. A graphical display of this type could be built into image acquisition software and would provide immediate feedback to the operator about image quality.

The image quality described in this chapter may differ from that in image compression applications. In the latter the quality refers to the goodness of the image reconstructed at the decoder, or the similarity between the original and its reconstructed version. The original image can be regarded as 'real' and the difference to it can be quantitatively described by some objective mathematical measures, for instance the mean-square-error measure. However in an imaging application due to the presence of noise it is impossible to tell the real signal from the unpredictable

noise. Therefore the image quality assessments in an imaging application usually involve subjective measures, while in a compression application they are usually objective.

2.5 Three-dimensional CLSM Image Analysis

2.5.1 Mathematical Analysis

Extended from two-dimensional imaging, three-dimensional imaging in CLSM is basically a projection of a three-dimensional real object O onto a three-dimensional hyper-plane specified with t = *constant* in a four-dimensional linear space spanned by (x, y, z, t). Component t is constant since we have assumed that the object is static and its position and shape are not changing with respect to t in the period of imaging. Because of the finite resolution of optics in imaging system (actually there is no imaging system of infinite resolution) and the presence of noise, $O_{\hat{x}\hat{y}\hat{z}}$ is digitized: one - as pixels by sampling in spatial domain following the Nyquist sampling criterion; two – as a discrete value with finite range, e.g. integer value from 0 to 255, for each pixel by quantizing the amplitude of signal to represent the contrast and smooth the noise [Castleman, 1995]. The output of the CLSM imaging system is therefore a variable I specified by (x, y, z), which is the digital version of $O_{\hat{x}\hat{y}\hat{z}}$ describing and measuring the physical properties related to the object like shape and material. That is,

$$\begin{aligned} &I(x,y,z) = \{v \mid v,x,y,z \in N\} = \text{Digitized}\,(O(\hat{x},\hat{y},\hat{z})), \text{ where} \\ &O(\hat{x},\hat{y},\hat{z}) = \{\hat{v} \mid \hat{v},\hat{x},\hat{y},\hat{z} \in R\} \end{aligned} \qquad (2.9).$$

2.5.1.1 Spatial Resolution Analysis

To measure the object, it is convenient to assume $O_{\hat{x}\hat{y}\hat{z}}$ has equal spatial resolution in each dimension: $\Delta\hat{x} = \Delta\hat{y} = \Delta\hat{z} = \hat{c}$. The resolutions in output results from CLSM imaging system are however not equal in most cases. One reason is that the PSF, even though it can be view as symmetrical on the plane perpendicular to the optical axis, is not symmetrical in three dimensions (see Fig. 2-5(b)). Actually, because of the optical limits, we noticed that the axial resolution in CLSM is roughly three times as large as lateral resolution from equation (2.5) and (2.6). Another reason is that they have

different scanning techniques to collect intra-plane and inter-plane data (refer to Section 2.2.3). Intra-plane image acquisition is achieved by controlling the angle of the laser to move the focal spot on the focal plane. The inter-plane images are collected by a stepping motor to coordinate step-by-step changes of the focal plane along z-axis with sequential image acquisition at each step. Each acquisition is an individual process and in most cases $step_x = step_y \neq step_z$. Now we can conclude the spatial resolution in CLSM imaging system as:

$$\begin{cases} \Delta x = \hat{c} / f_x \\ \Delta y = \hat{c} / f_y \\ \Delta z = \hat{c} / f_z \end{cases} \qquad (2.10),$$

where f_x, f_y, f_z are factors of resolutions in the discrete image space to those in the original object space in each dimension. They are determined by the profiles from optical system and operator only, and no relations to the characteristics of object itself, and their units are for instance μm/pixel.

From above we notice that if mapped to the same measurement like a micrometer, the actual distance between two consecutive pixels from intra-plane (Δx and Δy) is different to that between two consecutive pixels from inter-plane (Δz). Image processing techniques like stacking and interpolation may be required to restore the original resolutions of the object from I_{xyz}, before we can apply visualization, analysis and modelling methods.

2.5.1.2 Statistical Analysis

As it will be discussed in Section 3.1, the coding efficiency is theoretically dependent on the distribution of source. Since the original signal $O_{\hat{x}\hat{y}\hat{z}}$ is not a random process and actually a well-organized infrastructure describing the distribution and characteristics of objects in the 'micro-world' like cells and nucleus [Arribas, 1999], we can hypothesis that the output signal I_{xyz}, which is obtained by CLSM imaging of $O_{\hat{x}\hat{y}\hat{z}}$, is not random either and the individual elements (pixels) in I_{xyz} are correlated even with the presence of noise in imaging process. Here we give an example to quantify the inter-pixel correlations of I_{xyz}.

Here an example is given on 'C5' data set, which is from 'C5' group (Section 2.3.1.2). To measure the inter-pixel correlations along each direction, we compute the occurrence of two adjacent pixels by remapping I_{xyz} into two dimensional vectors using three schemes. In the first scheme vector $v1$ is formed from $I_{2x-1,y,z}$ and its successive neighbouring pixel along x direction $I_{2x,y,z}$; in the second scheme vector $v2$ is formed from $I_{x,2y-1,z}$ and $I_{x,2y,z}$ along y direction and in the third scheme vector $v3$ is formed from $I_{x,y,2z-1}$ and $I_{x,y,2z}$ along z direction. The histograms of these three types of vectors, $v1$, $v2$ and $v3$, are illustrated in Fig. 2-15 from left to right, respectively. We noticed that all the three types of vectors have a strong pattern of distribution along the diagonal direction of vector space, which is directly describing the correlations between two adjacent pixels since if two variables x_1 and x_2 are 'perfectly' correlated (correlation coefficient = 1) all of their values would finally fall on the line specified by $x_2=a\times x_1+b$. We quantified the correlation coefficient for each type of vector and used *linear regression* to get (a, b) (the white line in each figure) which can be viewed as the first principle component (FPC) of the distribution. The correlation coefficient (cc) is used to quantify the bias of the distribution to the first principle component by computing the mean of the distance from a vector to its projection on the first principle component, which is illustrated as the breadth of the distribution along first principle component. Now we can conclude that I_{xyz}('C5') has strong inter-pixel correlations along all the directions. Moreover along z-axis the correlation is greater than that along the other two directions. This is because f_z is smaller than f_x and f_y, which means the distance between two pixels along z-axis is smaller than that along x- and y-axis. We therefore can hypothesis that a compression system considering the correlations along all the directions can achieve better rate-distortion performance than its two-dimensional counterpart which cannot exploit the correlations along z-axis. This will be eventually examined and verified in Chapter 9.

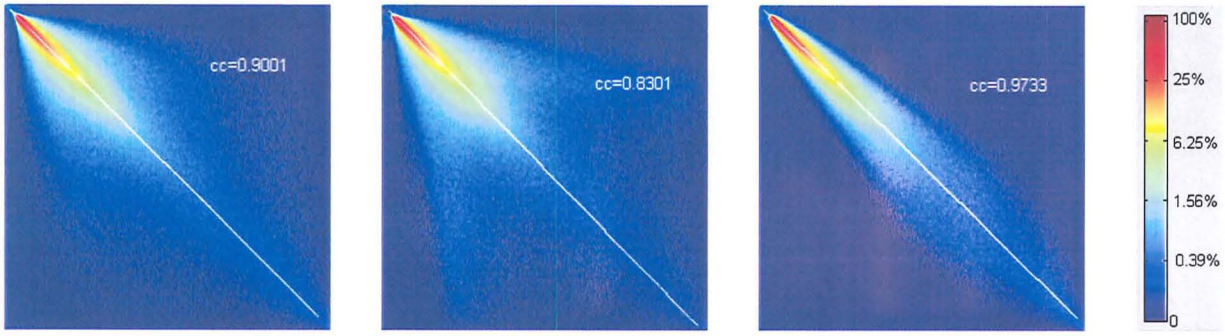


Figure 2-15: Two-dimensional histograms of the cooccurence of two neighbouring pixels from 'C5' data set, which are formed by sampling along (a) x-direction, (b) y-direction and (c) z-direction. The histograms are normalized by dividing the maximum value and displayed in logarithm scale.

2.5.2 Z-series vs. Temporal-series Imaging

Video imaging, which consists of techniques capturing moving objects, can produce a sequence of two-dimensional images. Its imaging theory is however fundamentally different from *z*-series imaging system like CLSM, where the object is digitized in a three-dimensional grid (pixels) and artificially presented as a sequence of two-dimensional images but inherently three-dimensional [Cockshott, 2003]. In video case it is a temporal-series imaging process, where in each frame the three-dimensional objects and their relative positions are projected on to a two-dimensional image plane like film of the camera according to perspective rules, and when finally reviewed along temporal axis, it captures the motion information of objects which is particularly a kind of planar motion normal to the camera axis.

Methods like motion compensation can well capture the motion information of objects between consecutive frames in a video sequence by assuming that the properties of objects, like shape and brightness, are not significantly changed [Watkinson, 2001]. In *z*-series imaging, however, there is no physical process of planar motion. Therefore we hypothesis that the compression strategy in z-series imaging applications will differ from that used in video applications. A comparison work verifying our statement will be presented in Chapter 9.

Chapter 3 Issues in Compressing Medical Images

Like the natural image, a medical image which is generated from a medical imaging system can be viewed basically as a multi-dimensional array and each pixel value is corresponding to the signal intensity accumulated in each sub volume whose size is determined by the resolution of the imaging system like microscopic imaging system. However medical images are collected for medical study or diagnostic purpose, where they are required for even higher resolutions, i.e. 4096×4096 in mammography, larger data range, i.e. 12 bits/pixel in computerized tomography and higher signal-to-noise ratio. Therefore distortion is a more important factor than rate in medical imaging applications.

Conventional image compression routines like JPEG, which performs well on natural images, may not meet the requirements on medical images for medical study purposes since they may suffer from quality degradation in decompressed images. Actually, to achieve higher compression ratio, people have to take the risk of introducing noises and artefacts in the decompressed image. Do people always have to choose lossless compression methods to avoid taking such risk? Or is there a solution that a lossy encoder can achieve higher compression ratio to meet the requirement of smaller storage and faster transmission while the decompressed image is still good enough and will not affect the diagnostic results? Issues regarding choosing approaches between those rate-parameterized and distortion-parameterized, lossy and lossless

methods will be discussed in this chapter. The quality measure of the decompressed images will be reviewed as well.

3.1 Distortion and Rate

The performance of medical image compression depends on two factors: bit rate or *rate* and image quality or *distortion*. Rate represents the average number of bits produced per pixel in compressed image. Sometimes it could be normalized by computing the ratio relative to the original bit rate. In this case the rate is usually referred as *compression factor* or *compression ratio*. For instance, suppose the original bit rate is 8 bit per pixel and after compression the bit rate is reduced to 0.5 bit per pixel, and then the compression ratio is 8/0.5=16. Image quality or distortion can be measured by the goodness of the resulting reproduction, which is generated by decoding process, in comparison with the original. Higher quality means smaller distortion, and the reconstructed images can show finer structural or functional information of body organs and support more reliable diagnostic outcomes. More specifically, in medical imaging the requirement of image quality varies according to different applications, such as diagnosis, recall, archival, or educational use, and a "good" quality compressed image might be one that is perceptually pleasing, contains no visible artefacts, and has no loss in interested content. As we will see in the next section, no single approach to quality measurement has gained universal acceptance. However some unbiased numerical measurement, especially mean-squared-error (MSE) measure and its variants have received more intensive consideration than others, due to their advantages such as simplicity in computation, effectiveness in quality measure and supportability in mathematics.

In information theory people [Elias, 1970; Gersho, 1979] have found that for a compression method applied on a specific image source, the relation between its rate R and distortion D is basically in an *exponential* form, which can be expressed as:

$$D = \delta_D(R) \equiv M \cdot \beta \cdot \sigma^2 \cdot 2^{-2R} \qquad (3.1).$$

where M is the term depending on the compression method, β is depending on the source distribution which is normalized by its variance and σ^2 is the variance of the source. The function $\delta_D(R)$ is called a *distortion-rate function*. It theoretically

describes the change of output distortion with respect to the change of input rate. Alternatively, one can regard the distortion as input and rate as output and describe the change of rate with respect to the distortion. In this case the operational function is called *rate-distortion function* $\delta_R(D)$, which is the inverse of $\delta_D(R)$:

$$R = \delta_R(D) \equiv \frac{1}{2}\log(\frac{M \cdot \beta \cdot \sigma^2}{D}) \qquad (3.2).$$

Although the distortion-rate function and rate-distortion function are theoretically interchangeable, in practice, it is difficult to design a compression method which can be parameterized as both rate and distortion. This is because it is impractical to give the distribution model β for a natural image, especially in the presence of noise. Actually a compressor is either rate-parameterized or distortion-parameterized. In the former a compressor is specified by a target rate and trying to make the actual rate as close to the target as possible; while in the latter a compressor is specified by a target distortion measure, such as MSE, and trying to make the actual distortion close to the target distortion. Moreover if a method is rate-parameterized and wants to achieve a target quality, it is impractical to achieve the target quality in only one compression attempt, but alternatively multiple attempts are needed, which would be very unadvisable and not flexible; and if a method is distortion-parameterized and wants to achieve a target rate the situation would be the same. Therefore users have to decide which factor is more important in a specific application and choose a suitable compression scheme accordingly [He, 2001].

Rate-parameterized methods are suitable for applications where rate is more important than distortion. For instance the file size is restricted by the storage capacity and network bandwidth. This is particularly useful in real-time applications like net-meeting and real-time video broadcasting. In these applications users are more concerned with the continuous of movement than the individual frame image quality. Moreover in real-time the network bandwidth is unstable, so a user would expect a rate-parameterized method to be adaptable to the change of bandwidth – when the bandwidth is getting small the compressor should reduce the target rate with the penalty of the degradation of image quality; while when the bandwidth is getting large the compressor can increase the rate and improve the image quality accordingly.

When the quality is the most important concern a distortion-parameterized method would be more advantageous [Chen, 1998; Blanchett, 1998; Miaou, 2004]. Various applications can be found in medical imaging and archiving, where usually the compression is an off-line process. As it will be discussed in the following sections, in medical environment, the quality plays a more important role, especially in the situation of lossy compression, because the diagnostic accuracy may be affected by using the compressed version compared to the original image. If the quality is decreased to the extent that the important characteristics of tissues on the original image have lost and been obscured, they will be unusable at all; otherwise it will be dangerous since it may mislead the diagnosis. Therefore the situation of compression in these applications is that it would be desirable if the compressor can reduce the image file size to facilitate the storage and transmission, but definitely it will not loss any important information at least will not affect the diagnostic accuracy.

3.2 Lossless or Lossy?

3.2.1 Lossless Compression Schemes

Lossless compression is also called *reversible* compression where these techniques allow exact reconstruction of the original image both the resolution and the pixel intensity value, which means there is no distortion. As illustrated in Fig. 2-15, image signal, from both medical and natural images, are correlated; and the basic idea behind a lossless compression method is that using an entropy coder to reduce the average bit length per pixel and usually preceded by prediction procedures to remove the inter-symbol redundancies.

There are many general-purpose data compression systems which basically work on some variation or combination of three well-known algorithms: Run-length coding, LZW coding and Huffman coding. All these algorithms look for the sequence of identical symbols or the repeated symbol sequence and replace them with a unique token which can be binary coded with shorter bit length. Since they do not take the inter-pixel correlations into account, none of these works well with real images. So that when a standard lossless compression scheme such as the LZW algorithm is applied to image signals, typically compression gains are very modest – at best the

average bit length per pixel will reduce by 5-10% compared to the first-order entropy of the original image data.

Due to the high fidelity requirement for some specific-purposed imaging applications, such as medical imaging for diagnosis and satellite imaging for earth study, lossless image compression has recently come under intense investigation. Moreover some of them, such as the LOw COmplexity lossless Compression of Images algorithm (LOCO-I) [Weinberger, 1996] have been accepted by Joint Photographic Experts Group in JPEG-LS as the new standard for lossless image compression [Weiberger, 2000]. We should notice that all these methods have outperformed general-purpose data compression systems.

Recent studies have shown that, with these complex lossless compression approaches, typical compression ratios of 2:1 are achievable for CT and aerial images [Weinberger, 1996; Wu, 1996] and even 3:1 or 4:1 are possible for volumetric MRI and CT images when the inter-frame dependencies have been considered [Kim, 1999; Bilgin, 1998]. However, lossless compression schemes still cannot meet the space and speed requirement in storage and tele-communication. For instance, 512 kilo-bits per second (kbps) is common in most of the service from the broadband providers in UK who are providing high-speed Internet connections. Transmission of a typical CT chest scan having 1024 by 1024 in size and 12 bits per pixel over such networks will cost approximately 20 seconds for the raw image data and still about 10 seconds for the lossless-compressed version. This problem would be more severe when we transmit a sequence of images which are obtained from imaging system having three-dimensional imaging functionalities like CLSM. A natural question to ask is "Is it possible to improve the compression ratio further while we can still get nearly perfect image quality?" To answer this question, we have to turn to lossy compression schemes and also issues in image quality measure.

3.2.2 Lossy Compression Schemes

Most of the image compression applications in use are "lossy", such as JPEG [Wallace, 1991]: part of the image information is discarded and fundamentally *irreversible*. The philosophy behind such approaches is that not all of the data in an

image are of same relevance to the impression they make on the human eye. It has been shown that the human visual perception system is sensitive to changes in luminance rather than the absolute luminance values themselves, and that perception is most sensitive to mid-frequencies and less sensitive to high frequencies in the image [Lee, 1992]. In a CLSM image, for example, there will always be noises caused by photons' random arrival to the detector and the random release of electrons in the photomultiplier even if we use methods like long pixel dwelling time and averaging replicates to reduce them [Webb, 1996]. Noises characterised as high frequency signals will superimpose onto the "real" image data which largely consists of low frequency signals and make originally uniform-distributed regions having a small amount of fluctuation. The eye will typically not even see these superimposed signals, since human visual systems are sensitive to structural information like edges and patterns not to the small pixel variations in uniform regions especially for those with low pixel intensities like background. It will therefore not be perturbed if they are missing. After proper image transformation, real image data are expected to be grouped into a small part of coefficients and a large part of coefficients representing high frequency signals such as noises can be discarded and quantized to achieve much larger compression ratio compared to lossless approaches, while the losses of image information are undetectable to the eye.

It is worth noting the development of a promising image compression approach for clinical use, the *progressive transmission* or *progressive coding*, which inherently supports lossy coding [Tzou, 1987; Wang, 1989; Xiong, 1999]. This methodology allows for example to prioritise the order of encoded bits to reconstruct a low-resolution and/or low-quality version of the requested image first, and then to progressively refine the resolution and the quality of the visualized data by transferring additional data. Progressiveness in resolution is referred as resolution scalability; and progressiveness in quality is referred as quality scalability. Such scalable image representation offers many advantages for medical image compression. It will be very suitable for medical telecommunication applications where doctors are accessing remotely stored images via small-bandwidth networks. They do not have to spend quite a long time to browse through those full-resolution images before locating the particular images of interest. The low-resolution image with high compression

ratio can serve as a guide for the initial selection process, saving time and bandwidth. Moreover, methods based on the wavelet transform, for example JPEG2000 [Christopoulos, 2000] and SPIHT [Said, 1996], have been proposed, which support for lossy-to-lossless coding functionality. They have the capability to start from lossy compression at a very high compression ratio and to progressively refine the data by sending detail information, eventually up to the stage where a lossless decompression is obtained.

The noise process results in a superimposition effect of unpredictable noisy signals onto the locally-predicable real signals, so it is impossible to separate them apart. This means in real images noise cannot clearly distinguished from the real signal, since by no means can one tell what the real signal is – random process is existing and unavoidable in an imaging system, thus cannot be thoroughly removed. The way people employed in most of lossy compression schemes to characterize noises is to regard them as high-frequency signals and reduce them by removing the high-frequency components after the decomposition in frequency domain like Fourier transform [Castleman, 1995]. The problem using this approach is that discarding high-frequency components may cause the loss of the high-frequency information of real signals, for instance the information around sharp edges. An edge will tend to decrease its contrast and eventually blurred and even cannot be identified from its surroundings. Since eye is sensitive to edges and such structural information is even more important than low variation regions in image-based medical studies where the losses may be detectable and such compression quality may not be accepted.

Compared to reversible compression approaches, the irreversible approaches can provide considerably higher compression ratios, however introducing unavoidable noise to the reconstructed image due to quantization. It is disputable to introduce 'lossy' in medical image compression, because there is no legal standard that exists for this area at present. Lacking a standard means that there is no objective clinical reference for the court to judge a mal-diagnosed case that involves medical imaging device, which incorporates lossy compression [Lemke, 1993; Wong, 1995]. Therefore lossless techniques have been widely used in medical images especially in diagnostic-purpose studies. However in recent years considerable attention has been given to the

evaluation of compressed image quality, where they were aimed to find out the impact of lossy compression upon tasks that involve detection [Cosman, 1993b; MacMahon, 1991; Sayre, 1992] and measurement [Cosman, 1998; Cockshott, 2003]. In their works, for instance in [Cosman, 1998], they concluded that in statistics, the subjective measurement accuracy at compression ratios up to 16:1 did not differ significantly from that using original images. Moreover their results showed that at some compression ratios, e.g. 6:1, the measurement accuracy is even better than that using original when measured based on independent gold standard (refer to section 3.3.1.3). This is because in a clinical environment, which involves subjective measurement and detection, the accuracy or the variance of error is primarily dependent on human observers not the compression techniques and the effects of lossy compression can be neglected.

3.2.3 Near-Lossless Compression

In some specific circumstances a reconstructed image can be viewed as lossless even if it was obtained from a lossy scheme and is different from the original. These schemes are referred as *near-lossless* compression [Cosman, 1994]. There are two typical applications which give this term different explanations – one is related to the subjective interests and the other is related to the statistical analysis.

In the first case an image is typically recognized as two parts – the region of interest and the region of non-interest. The criterion of the interest is application-dependent and usually determined by the user. For instance suppose one is only interested in a specific region of the image and not the whole image. If the compression method can perform high-ratio lossy compression on the non-interest regions while lossless compression on the interest regions, then the reconstructed image will still be acceptable, since it does not lose any useful information in the sense of interests. Therefore in such near-lossless compression, image quality is guaranteed by lossless compression in the interest regions and higher compression ratio is achieved by lossy compression in the non-interest regions.

The implementation of this idea requires the compression method has different bit allocation schemes in spatial domain. For instance in JPEG2000 method at each band

the intra-band coefficients are decomposed into blocks and the quantization of the coefficients is performed on each block individually; therefore if a block is recognized as 'interest' more bits will be allocated to encode this block to ensure the lossless quality and otherwise fewer bits and a coarser quantization will be used.

In the second case the term 'lossless' can be explained in the sense of statistical analysis of noise; therefore it is suitable for the images obtained from real imaging systems, such as medical imaging and microscopic imaging, but not suitable for artificial images such as human-painted and computer-generated images, since they are free from noise and the signal-to-noise factor can be infinite. In the presence of imaging noise characterised by its variance $\delta^2_{noise,org}$, if the compression noise, computed as the variance between the reconstructed image and the original, $\delta^2_{noise,rct}$, is not more than the imaging noise in the original image ($\delta^2_{noise,rct} \leq \delta^2_{noise,org}$) we will not be able to distinguish the type of the noise on the decompressed image such that whether they come from the imaging system or from the lossy compression. In other words, the clinical results, such as the diagnostic accuracy and the measurement accuracy, will not be statistically affected due to the loss of information in compression. Therefore in this sense a lossy compression scheme can be viewed as lossless.

To achieve the target distortion the compression method should be able to work at a particular quality level, where the distortion should be effectively bounded by the specified target distortion. The near-lossless compression can be implemented as first computing the variance of noise $\delta^2_{noise,org}$ of an image, and then set the compressor the target distortion as $\delta^2_{noise,org}$ and the distortion constraint functionality of the compressor will ensure the variance of the output distortion no greater than the variance of noise.

In confocal microscopic images we have learned that the noise is Poisson-distributed, whose variation is proportion to the square root of the variation of the signal.

Therefore given an image, $I(x, y)$, having Y rows and X columns of pixels, its variance of noise $\delta^2_{noise,org}$ can be computed by:

$$\delta^2_{noise,org} = \kappa \cdot \frac{\sum_{y=1}^{Y}\sum_{x=1}^{X}\sqrt{I^2(x,y)}}{X \cdot Y} = \kappa \cdot \frac{\sum_{y=1}^{Y}\sum_{x=1}^{X} I(x,y)}{X \cdot Y} \quad (3.3),$$

where κ can be any factors in the procedure of image acquisition to reduce the noise. For instance if the image is the average of N replicates, κ will be $1/\sqrt{N}$ [Castleman, 1995].

3.3 Medical Image Quality Measures

The evaluation of reconstructed image quality can judge the quality of images obtained by a particular compression technique, but also for the quality judgement across various compression techniques. The evaluation of lossless techniques is normally a simple and straightforward task, where a number of standard criteria, such as compression ratio, are employed. A major problem in evaluating lossy techniques is the difficulty in describing the type and amount of degradation in reconstructed images. This is because each technique implements a different algorithm to discard irrelevant information on the image and thus may have different types of degradation, or artefacts, when associated with subjective evaluations.

3.3.1 Image Quality Measurements

For lossy-compressed images, no single approach to quality measurement has gained universal acceptance, but three general approaches have been widely used: *objective measure* and *subjective measure* and *post-processing validation* [Cosman, 1994; Wong, 1995].

3.3.1.1 Objective Measure

A. Numerical Measure

The numerical measures are designed to indicate the similarities between two images, say the original and its decompressed version, as a numerical value. They are all easily-computed, discrete and bivariate. They are computed by exploiting the

differences in statistical distributions of pixel values – one from the original image and one from the decompressed image. Eskicioglu [Eskicioglu, 1995] has described several numerical measures in image compression applications. All of them suffer from that they do not correlate well with subjective quality measures, for instance judged by human visual system (HVS). Recent research has incorporated the HVS models into objective measures, where they weight the distortions according to the different sensitivities of HVS to the patterns on the image. However equal-weighted measures without considering the HVS are widely used in image compression applications since the HVS is too complex to fully understand with present psychophysical means.

Among these measures the mean squared error (MSE), also called L2-norm, and its variants are the most common objective criterion in image compression applications. Suppose I is the original image and $\hat{I}$ is its decompressed version of the same size specified by w pixels in width and h pixels in height.

$$D = MSE^{(2)} = d(I,\hat{I}) = \frac{1}{w \cdot h}\sum_{i=1}^{h}\sum_{j=1}^{w}| I_{i,j} - \hat{I}_{i,j} |^2 \qquad (3.4).$$

In measuring decompressed quality of three-dimensional image data, *MSE*s is extended for the measurement of the whole sequence of $\#_f$ images not a single image, which is defined as the mean of $\{MSE_f\}$ for each image I_f in the stack.

$$D = MSE^{(3)} = \frac{1}{\#_f}\sum_{f=1}^{\#_f} MSE_f^{(2)} = \frac{1}{\#_f \cdot w \cdot h}\sum_{f=1}^{\#_f}\sum_{i=1}^{h}\sum_{j=1}^{w}| I_{f,i,j} - \hat{I}_{f,i,j} |^2 \qquad (3.5).$$

A common alternative normalization of MSE is peak signal-to-noise ratio (PSNR) when the distortion is normalized with the maximum variation of the signal. Suppose the input image is n-bpp, we have

$$PSNR = 10\log_{10}\frac{(2^n - 1)^2}{D} \qquad (3.6).$$

For a single two-dimensional image, D is referred to $MSE^{(2)}$ and in the three-dimensional cases, *PSNR* is the called *Mean-PSNR*, which is defined on the whole stack and D is $MSE^{(3)}$.

Square error distortion measure has several desirable properties. First of all, it is an easily computable distortion measure and it reflects perceptual quality of reconstructed image. In addition, it is tractable in analysis. Numerical theories and methods are proposed for the analysis and designing of systems which are optimal in the sense of minimizing mean squared error. Common faults of squared error are that a slight spatial shift of an image causes a large numerical distortion but no visual distortion and, conversely, a small average distortion can result in a damaging visual artefact if the error is concentrated in a perceptually sensitive region [Cosman, 1994]. Studies on human vision system conclude that human vision is sensitive to edges, insensitive to textures and other masking effects. The pioneering work of developing computable distortion measures incorporating knowledge of human vision has been proposed by [Nill, 1992; Daly, 1992].

B. Graphical Measure

The graphical measures are computed from numerical measures and indicate the degree of distortion using plots and graphs. For instance the histogram of the error image between the original and its reconstruction gives the probability distribution of the pixel value's difference between the original image and its decompressed version. Different from numerical measures, they can describe several features of distortion and present them in a more straightforward manner.

Hosaka plot, or h-plot, has been used to appropriately specify not only the amount but also the type of distortion in reconstructed images [Hosaka, 1986; Eskicioglu, 1993]. In this method images are firstly portioned into several classed based on local activity level. Distortions $\{D_i\}$ are then computed for each class i between the original and the reconstructed images. Finally $\{D_i\}$ is plotted in polar coordinates. The radian denotes the class i and radius denotes the amount of distortion D_i for this class. Therefore the shape of the plot gives the information about the type and the amount of distortion.

3.3.1.2 Subjective Measure

The subjective quality of reconstructed images can be measured in many ways and there is no standard for rating still images. All are based on the reviews and judgements from typical users like radiologists. This can be performed by score-based

rating, which usually referred as mean opinion scoring (MOS) approach [Ahmet, 1995]. For instance, a suitably randomised set of images can be presented to experts or typical users who rate them on a scale of 1 to 5. It can also be performed by rank-based rating, where the decompressed images, which are obtained from different compression methods at the same objective measure or compression rate, will be asked to compare with the original image and rank the quality from the best to worst. In the former the goal is to find what image quality would be accepted by a typical user, and the testing images will include both the decompressed images and the original. While in the latter, most efforts involve the comparisons of the similarity of the decompressed image to the original. Both are followed by subsequent statistical analysis to highlight the averages variability, and other trends in the data [Cosman, 1993b].

Moreover in the scoring approach there are two types of scoring processes: one is absolute and the other is relative [Ahmet, 1995]. In the absolute score each viewer compares the reconstructed image, encoded at different objective quality levels, with the original one to decide which quality level it belongs to and gives the score. It helps to investigate the consistency between subjective and objective measures. In the relative score a group of images, consisting of the original and the reconstructed images at various objective levels for a particular image, are simultaneously presented to the viewer in one examination. The viewers are asked to rank the images in this group, for instance from the best quality to the poorest quality. This approach can reduce the bias of the judgement from the original image.

One example of a subjective quality rating experiment is introduced in [Benoit-Cattin, 1997]. The evaluation has been done on 20 volumes of 128^3 voxels obtained from 3D X-ray scans of a mini pig's left ventricle, which are compressed using 3D wavelet decomposition and lattice vector quantization, at 6 different bit rates: 0.1, 0.13, 0.16, 0.2, 0.26 and 0.4 bpp. Two blind experienced senior radiologists were asked to evaluate "the quality of the image for diagnosis purposes" based on the relative scoring approach according to the sharpness of small vessels and quality of interested regions. In each session, a random sequence, consisting of the original and reconstructed images, is presented without the knowledge of the bit rates. [Benoit-

Cattin, 1997] concludes that 0.16 bpp rate seems to be a lower bound giving a usable image quality for diagnosis purposes.

By subjective rating one can on one hand determine a criterion of acceptable decompressed image quality or compression rate, which is directly related to those typical users. On the other hand he can make comparison between compression methods, which might exhibit different distortion artefacts but have same level of objective quality, say MSE. Eventually it would help users choose a suitable compression technique and tune up the parameters in a compression system to adapt for a particular application.

3.3.1.3 Consistency Test

Unlike natural images, in medical applications the primary concern is that the diagnostic accuracy of the lossy compressed images remains not less than that of the original images [Cosman, 1994]. They act as the central role in diagnosis of deficiencies of tissues and organs. Objective measures, like MSE may indicate diagnostic accuracy, but the accuracy must be demonstrated directly. Therefore a consistency test is necessary to statistically investigate the effect of compression approaches on results of the diagnostic tasks, including measurement of structures, detection of lesions and interpretation of texture. There is no general rule to design the task, since it is application-dependent. Usually it can involve human observer (subjective tasks) or without human observer (objective tasks).

In subjective tasks measurement and detection results are obtained from human experts, e.g. radiologists, and then subsequent statistical analysis is based on their subjective judgements. Since human observation process is biased, the experiment has to be designed to reduce this bias when they want to simulate the normal clinical practice. An example of such consistency test design was given in [Perlmutter, 1998], where they simulated a clinical experiment to determine the effects of lossy compression on the accuracy of measurement on blood vessels in the evaluation of aneurysms. Perlmutter *et al.* established two 'gold standards' for the 'correct vessel sizes' before they can quantify to what bit rate the testing images can be compressed without loss in measurement accuracy. The first gold standard was the *personal*

standard, which is derived for individual radiologists based on their own measurements of uncompressed images. It assumes the original image to be correct therefore it directly describes the trends in a judge's performance across different compression ratios, however it cannot judge the variability of the measurement on an original image and therefore can wrongly reject an accurate measurement. The second gold standard was *individual standard*, which is based on the consensus measurements taken by two radiologists on the original images and these two radiologists are different from the radiologists whose judgements are used to determine measurement accuracy. The independent gold standard method reduces the bias from judges, but the change of accuracy may be due to differences between judges not the compression method and therefore increase the risk of wrongly accepting an inaccurate measurement.

In objective tasks measurement and detection are primarily performed using automatic image processing routines, such as enhancement and segmentation. Then subsequent statistical analysis is applied on the objective results. Such methodology is often used in non-diagnostic-purpose studies. For instance in [Daly, 2002], the image data sets examined were obtained from CLSM devices and used for analysing the three-dimensional structures of blood vessels by measuring the shapes and distributions of cells. The corresponding consistency test in [Cockshott, 2003] was designed such that the same methods, which were used to segment the cells and analyse their shapes, were performed on the decompressed images reproduced from different compression approaches and then measurement results are compared to those of the original image. By statistical analysis, they can quantitatively describe the consistency of the measurement results using the decompressed images to those using the original.

3.3.2 Image Artefacts

All lossy compression methods suffer from image artefacts, which can be detected by human eyes when working at a low bit rate. Most of the lossy medical image compression approaches are designed either in spatial (temporal) domain like VQ and Quad-tree decomposition or in frequency domain like DCT and DWT [Wong, 1995]. None of these methods can avoid image artefacts; however artefacts produced by

these methods have different patterns and sensitivities to human eyes. Typically there are three patterns of image artefacts: *blurring*, *blocking* and *ringing*.

3.3.2.1 Blurring Artefact

Blurring effects will make the decompressed image look smooth – an image will appear to lose its contrast, and variations between neighbouring pixels will tend to be reduced and even disappear. Such pattern is caused either by averaging neighbouring pixel values in spatial domain, for instance using one vector (centroid) to represent a cluster of vectors in vector quantization methods, or by discarding high frequency coefficients after decomposition in the frequency domain, for instance discarding coefficients associated with the high-frequency components after DCT or DWT decomposition.

Blurring artefacts is not obvious in originally smooth areas where pixels have low variations between each other since human visual system is not sensitive to this pattern. However in Fig. 3-2(b), we noticed the contrast of the sharp edge is reduced and it makes the edge look blurred within its neighbouring pixels. Compared to originally smooth areas, the losses of information in edge areas are more distinct to eye and therefore more severe in evaluating the distortion by subjective rating and post-processing validation.

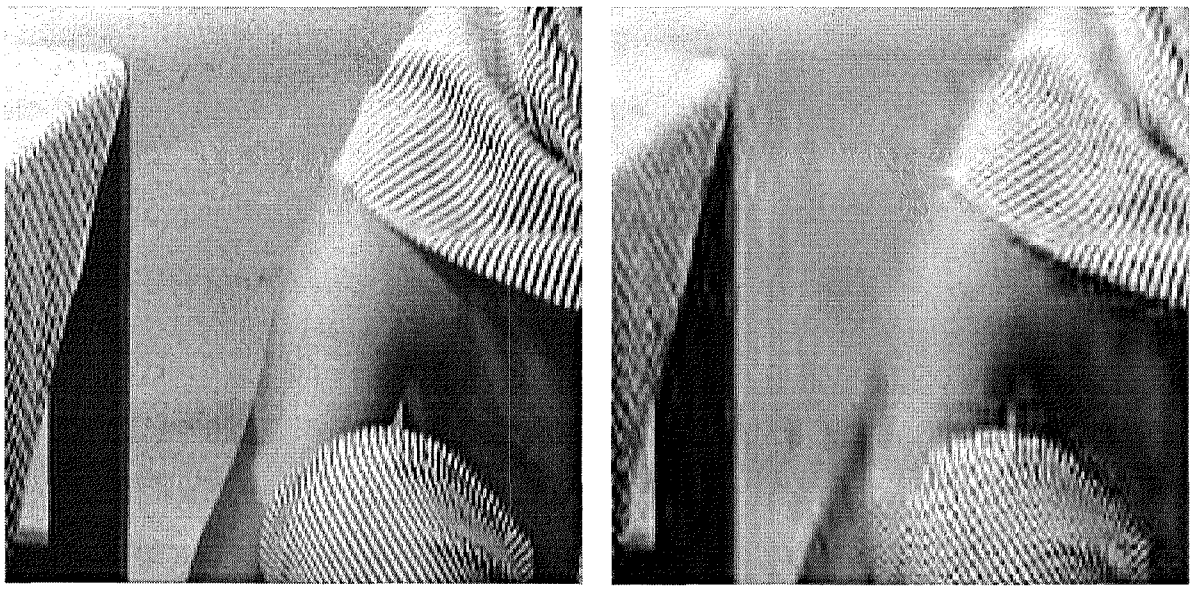
3.3.2.2 Blocking Artefact

A decompressed image will suffer from blocking artefacts when the image is decomposed into blocks and the compression routine is performed on each block individually. In JPEG [Wallace, 1991] method for instance, an image is first decomposed into 8-by-8 blocks, then DCT is performed on each block to reduce the overall computational complexity and finally coefficients located at the tail after zig-zag ordering are discarded according to the property of this block. Another example is the method used in [Cockshott, 1999], where they use vector quantization with very large vector size (32×32) to reduce the bit rate. Both will cause discontinuities between neighbouring blocks because the individual encoding process of non-overlapping blocks did not take into account the correlations across neighbouring blocks.

Blocking artefact (see Fig. 3-2(c)) causes more severe distort to human visual system than other two artefacts in medical images [Pawley, 1996]. Moreover the post-processing results on decompressed images will be more likely affected [Cockshott, 2003]. Works in [Hu, 1997] and [Kim, 1992] were trying to reduce the discontinuities by post-smoothing and inter-vector prediction on the decompressed image, but the mosaic-like blocky distortion cannot be successfully removed in high-variation regions and more computational cost is introduced. Therefore large blocks, although having benefits in increasing compression ratio [Shannon, 1959], will suffer from artefacts and introduce extra operations in the decoder if block-removing process is considered, and therefore are not preferable in medical images.

3.3.2.3 Ringing Artefact

Discarding high frequency wavelet coefficients will cause ringing artefacts which are characterised as repeated high frequency patterns (see Fig. 3-1). This is because the high frequency coefficients are largely concentrated on finer scales (higher resolution layers) when the image is decomposed as a multi-resolution representation. Coefficients on finer scales contain the information of difference of the original image and the reconstructed image reproduced by coefficients from coarser scales. Omitting these coefficients will cause blurring effects in low-variation regions and oscillation patterns produced by wavelet coefficients from coarse scales, thus eventually leaving apparent ringing artefacts around sharp edges surrounded by smooth regions such as regions besides the leg of the table in Fig. 3-1; and also distort the regular pattern in the high-frequency regions such as the strip patterns on the scarf and trousers.



(a) (b)

Figure 3-1: (a) a region (190 × 190) clipped from Barbara (512 × 512, 8 bpp); (b) its decompressed version using JPEG2000 [Christopoulos, 2000] at the rate of 0.25 bpp with PSNR = 29.08 dB. Ringing effects are apparent around strong edges surrounded by smooth regions like the leg of the table and also in the regular-patterned high-frequency regions such as the strip patterns on the scarf and trousers.

Ringing effects are apparent when compressing artificial images like cartoons since they are featured by strong edges and smooth regions. In medical images like CLSM images, the severities could be alleviated because noise reduces the sharpness of contrast between edges and their surroundings, as illustrated in Fig. 3-2(d).

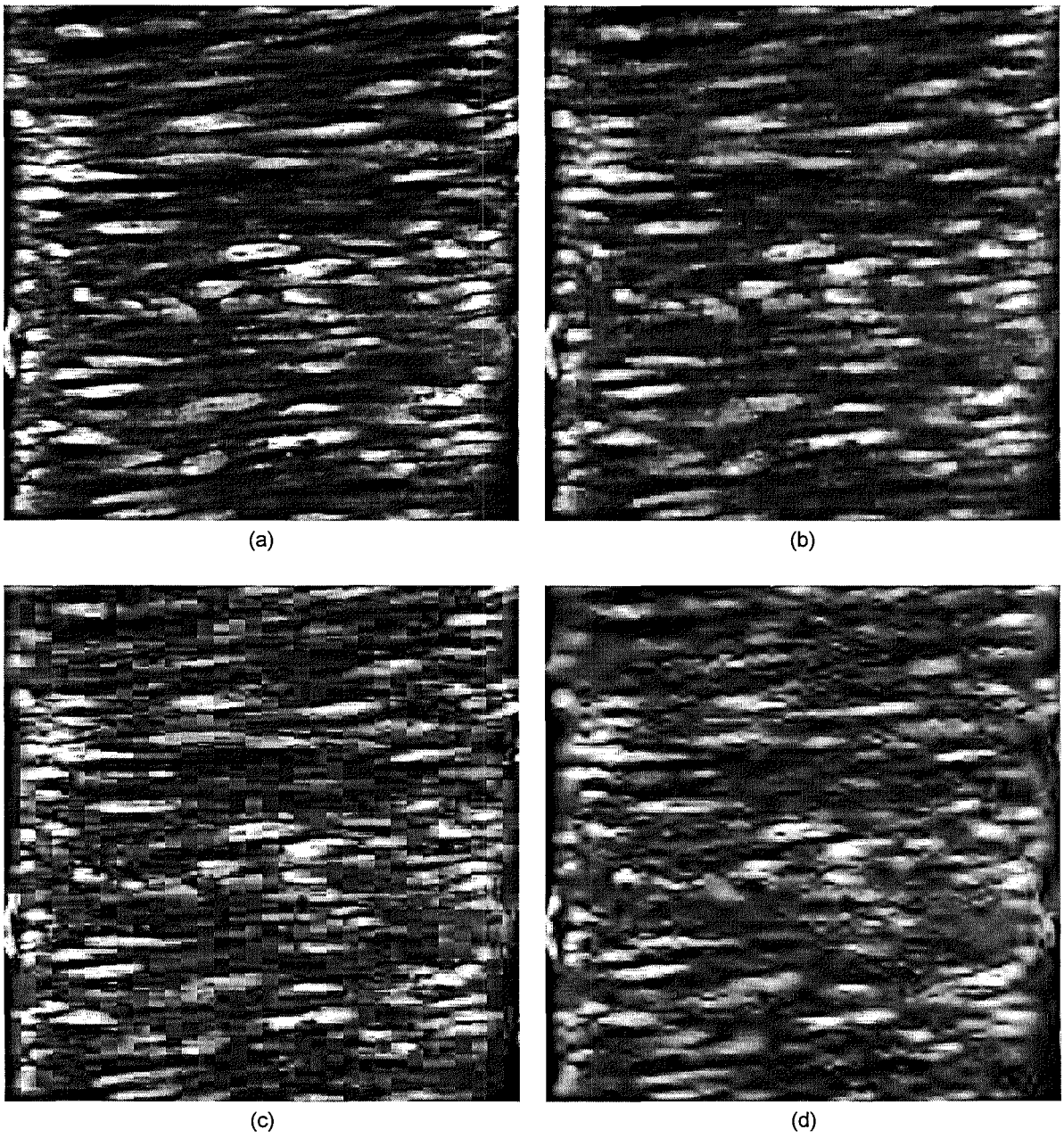


Figure 3-2: (a) A region (256 × 256) clipped from 32th frame of HUM_GLT data set (see Section 3.3.1.4); (b) Compressed using the proposed method 3D-CoMIC. Contrast is reduced by blurring effects; (c) Compressed using 3D-DCT [Schelkens, 2003]. Blocking effects of discontinuities across block boundary are visible. (d) Compressed using 3D-SPIHT [Kim, 1999]. The ringing effects recognized as high-frequency speckle patterns are distinct in smooth regions between nuclei. All the three methods are configured to produce the image quality as PSNR=30dB.

3.4 Conclusions

Trading-off image quality and/or implementation complexity against bit-rate introduces specific constraints. On one hand, currently, lossy techniques are not broadly accepted in clinical diagnostic practice. The reason is that lossy compression has raised new legal and regulatory question for manufacturers and users. The unwillingness to lose information which may affect a diagnosis is understandable. So far, there have not yet been appropriate policies and acceptable standards for the use of lossy compression on medical images. Therefore, lossy compression schemes are still under questions as whether they can ever be used in a medical context, despite encouraging results regarding the maintenance of the diagnostic accuracy.

However on the other hand the compression performance of lossless schemes is eventually bounded by noises characterised as a random process, therefore if the noise in an 8-bpp source image has 4 bits dynamic range no lossless method can achieve more than 2:1 compression ratio. Thus in the cases, where the quantity of medical images has outstripped the capabilities of storage media and network bandwidth even with the lossless compression, people would prefer lossy compression rather than lossless compression. This is because lossy schemes can offer people more flexible control of compression ratio and decompressed image quality and moreover their distortion-rate performance has improved enormously over the past decade [Taubman, 2001]. Possible areas of application for lossy compression of medical images have proliferated. They include wireless emergency medical services, battlefield and shipboard surgery and medicine, progressive browsing of databases, medical teaching archives and others. Moreover some efforts have been done to put image compression issues to the standardization in medical applications [Hamid, 1998]. Medical image compression today receives unusual considerations, with the emergence of a variety of new digital imaging modalities and the maturity of image compression techniques.

Chapter 4 Methods in Microscopic Image Compression

Since most of the three-dimensional techniques are extension of their two-dimensional versions, we begin with a review of planar (two-dimensional) compression techniques. These techniques like JPEG2000, even though performed very well and featured with many desirable characteristics in medical imaging applications, is not competent in compressing three-dimensional image data collected from three-dimensional imaging systems like CLSM. This is because the frame-by-frame schemes using basically two-dimensional compression methods have not taken into account the inter-frame correlations, which inherently exist in three-dimensional image data though in most cases they are artificially represented as a sequence of two-dimensional images.

Conventional multi-frame coders, such as video coders based on motion compensation techniques, are designed for exploiting the motion correlations of projected objects. Such motion correlations between two consecutive frames, however, are different from inter-frame correlations generated from three-dimensional imaging systems. We therefore can hypothesis that video coders are not well suited for three-dimensional microscopic images.

We then proceed to real three-dimensional compressors. We review several three-dimensional approaches based on two widely used techniques: Three-dimensional

Discrete Wavelet Transform (3D-DWT) and Three-dimensional Discrete Cosine Transform (3D-DCT). By reviewing their advantages and deficiencies, they conduct us to a better approach to compress three-dimensional microscopic images.

4.1 Two-dimensional Compression Techniques

4.1.1 2D-DCT

The discrete cosine transform (DCT) has been widely used in lossy image compression more than any other techniques since it is included in the Joint Photographic Expert Group (JPEG) and Moving Picture Experts Group (MPEG) compression standards for general-purpose still-image and video image applications [Wallace, 1991; Taubman, 2001; Watkinson, 2001]. There are many kinds of transform coding techniques [Castleman, 1995]. Among them Karhumen-Loeve transform (KLT) performs best in terms of decorrelation. The KLT constructs the set of basis functions that correspond to the normalized eigenvalues of the covariance matrix of the image and, thus provides maximum decorrelation and entropy reduction. However, the KLT is seldom used in image compression applications as the required inverse matrix is difficult for compute and a fast algorithm does not exist. Among other transform techniques, DCT has decorrelation properties very close to the KLT, and the complexity of the DCT and its inversion IDCT are very low. In two-dimensional images, the 2D-DCT and its inversion 2D-IDCT for an image block $f(x, y)$ with size $M \times N$ pixels are given by

$$F(u,v) = \frac{2}{\sqrt{MN}} C(u)C(v) \sum_{y=0}^{N-1} \sum_{x=0}^{M-1} f(x,y) \cos\frac{(2x+1)u\pi}{2M} \cos\frac{(2y+1)v\pi}{2N} \quad (4.1)$$

and

$$f(x,y) = \frac{2}{\sqrt{MN}} \sum_{v=0}^{N-1} \sum_{u=0}^{M-1} C(u)C(v)F(u,v) \cos\frac{(2x+1)u\pi}{2M} \cos\frac{(2y+1)v\pi}{2N} \quad (4.2)$$

where $C(w) = \begin{cases} 2^{-1/2}, & w = 0 \\ 1, & otherwise \end{cases}$.

The transformed coefficients $F(u, v)$ after DCT have the same size as original signal and the magnitude of $F(u, v)$ represents the intensity of signal response at the

frequencies of ($u/2M$, $v/2N$) along x, y directions respectively. Since high-frequency signal occurs only around edges and in most cases it takes only a small part in the whole image, $F(u, v)$ has such properties that the higher the value of u and v the more likely the coefficients to be zero or near zero and the lower the significance for improvement of reconstructed image quality, as illustrated in Fig. 4-1. To simplify the entropy coding, the DCT coefficients are organized in a zigzag order, beginning from upper left corner (u=0, v=0) and ending at bottom right (u=M-1, v=N-1). This zigzag ordering helps to facilitate better entropy coding, and the progressive transmission can be easily achieved using this ordering.

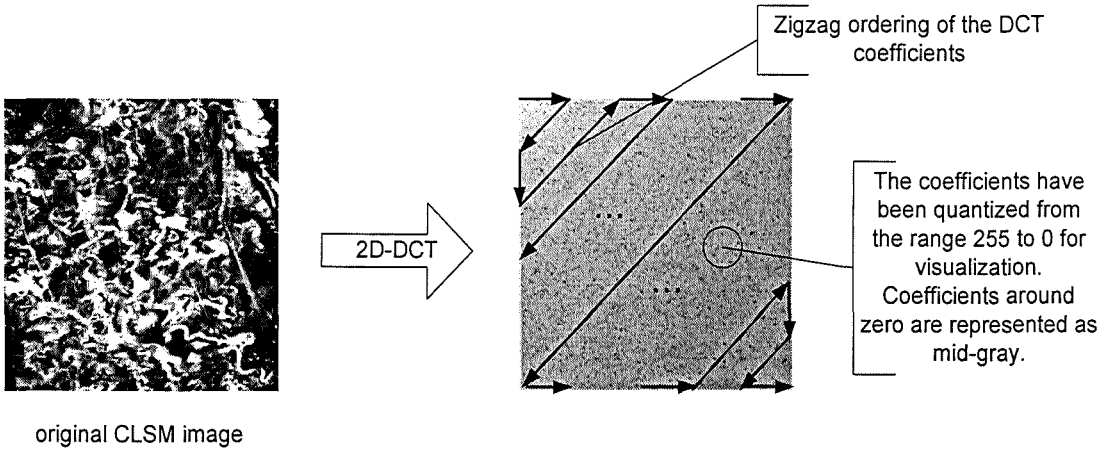


Figure 4-1: The distribution of 2D-DCT coefficients and zigzag ordering of coefficients for entropy coding.

4.1.1.1 JPEG

In JPEG standard [Wallace, 1991], the original image is divided into adjacent non-overlapped blocks, e.g., 8x8 block. The DCT and the zigzag ordering are applied on each block. Then each of 64 DCT coefficients is uniformly quantized in conjunction with a 64-element quantization table, of which the quantization step must be specified by the application (or user) as the input to the encoder. To avoid smooth and blocky artefacts, the quantization step is determined based on the psycho-visual experiments. The quantization tables obtained from the experimental works in [CCIR, 1982] have been used by JPEG standard.

JPEG produces acceptable image quality at moderate compression rate (10~16:1), and has been widely used in natural images. However two primary drawbacks make it a

poor candidate for medical images. Small block sizes have the advantage that the computational and memory requirements are very moderate and the quantization can be designed to adapt to the properties of each image block as well, but it may be too small to take the inter-block correlations into account and even worse blocking artefacts are introduced. Such block-based image compression techniques using DCT work well if an image distributes within a relatively narrow range of frequency, such as a CT image of human chest. However, microscopic images especially obtained from CLSM with contrast enhancement characteristics, are non-stationarily distributed and featured by strong structures, such as sharp edges. This means one has to be careful in quantizing the coefficients; otherwise block boundary will be apparent in the decoded image, particularly when edge enhancement must be applied after coding. It has been reported that for compression ratios greater than about 10:1 using JPEG, introduces noticeable blocky artefacts in reconstructed images, which is unacceptable for microscopic and medical image applications [Avinash, 1995; Wittenberg, 1993].

Additionally, the quantization is based on pre-defined quantization tables which are elaborately designed for optimized visual quality based on psycho-visual experiments on natural images. That means the quality control of JPEG cannot precisely adapt to every image coming from different types of imaging modalities. For instance a setup like 'moderate quality' that produces acceptable image quality on a natural image with large stationary distributions may lead to blocky artefacts and unacceptable image quality on a CLSM image with lots of non-stationary distributions. Here we give an example using standard JPEG to test on two images having different characteristics. In Fig. 4-2 the left image from 'C5' dataset has many details while the right image from 'G27HG70' dataset comprises large areas of low-variation regions. Two images were compressed using JPEG coder at the same quality level. The JPEG is implemented in Adobe® Photoshop®, in which the quality is scaled from 1 to 100. We chose the quality level as 5 for both images. However we noticed that although we made the same configuration the output images have different visual and numerical quality. The detailed image presents more blocking artefacts. Therefore although the JPEG is claimed to be quality scalable, the relation between the input

and the output quality is not clear and it is not convenient to working at a particular quality or rate requirement.

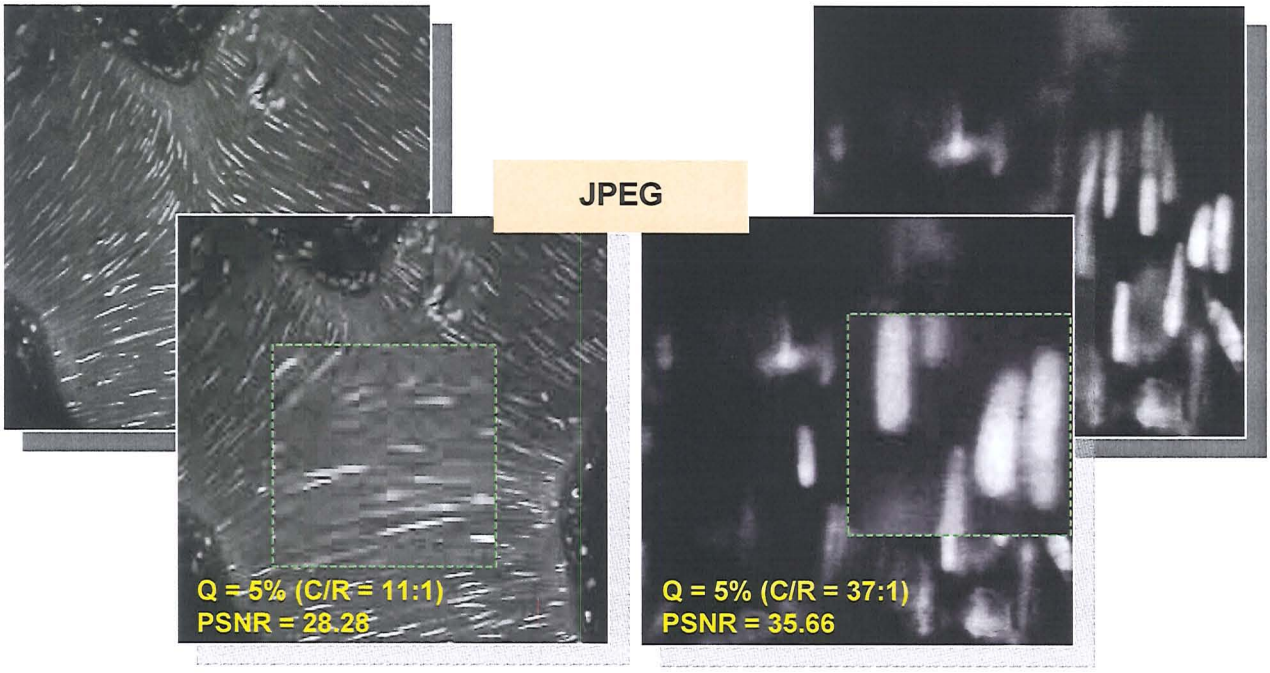


Figure 4-2: Experimental results of using JPEG on two CLSM images. The decompressed images have different visual and objective quality with the same quality setup.

4.1.2 2D-DWT

Image compression using discrete wavelet transform (DWT) is a typical application of subband coding. A subband coder performs a set of filtering operations on an image to divide it into spectral components or bands. For example, an image might be expressed as a small subimage that represents the low-frequency components which can be viewed as either a minified copy of the original and successive higher spectral bands that contain the necessary high frequency information that can be added to the low-frequency components to reproduce the sharpness of original image.

The basic idea of discrete wavelet transform is to decompose the input signal $x(n)$ into a coarse and a detail approximation through filtering with lowpass filters $h(n)$ and highpass filters $g(n)$. This detail approximation can be thought of as the additional information that must be supplied to generate a finer-scale approximation from a coarse one. According to multi-resolution analysis, a new decomposition can be applied to the coarse approximation and iterated so that the original signal is spread

into a hierarchical tree of detail functions and the coarsest approximation (see Fig. 4-3). In order to maintain the same description of the original signal, in term of sample number, at each level of decomposition the detail and the coarse functions are subsampled by a factor of two, i.e. the scale of the signal at low resolution and the detail is halved.

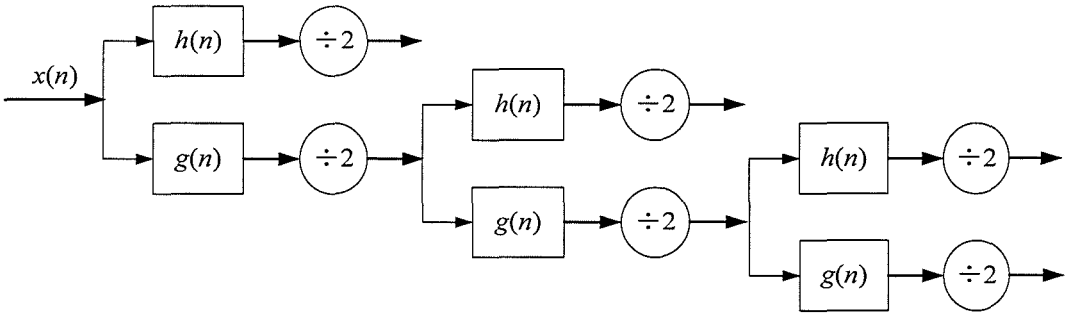


Figure 4-3: Filter bank tree with one band corresponding to coarsest approximation and several bands representing details.

In image applications, separable wavelet transforms are typically used; separability enables the 2-D transform equivalent to two separate 1-D transforms. We apply 1-D wavelet transform to rows and columns of image alternatively, and the result is a first level four-subband partition of the image into horizontal low-pass/vertical low-pass (LL1), horizontal high-pass/vertical low-pass (HL1), horizontal low-pass/vertical high-pass (LH1), and horizontal high-pass/vertical high-pass (HH1) sub-images. The LL1 image represents the smooth version whereas the other three images are the details. The previous procedure was iterated on LL1 till the desired level of decomposition as illustrated in Fig. 4-4(a). The analysis, also called discrete wavelet transform DWT and the synthesis, also called inverse discrete wavelet transform (IDWT) of a two-dimensional image signal $f(x, y)$ can be formulated by equation (4.3) and (4.4) respectively.

DWT :

$$
\begin{aligned}
F^{(0)}(n,u,v) &= \iint f(x,y)\cdot 2^n \cdot \phi(2^n x-u)\cdot \phi(2^n y-v) \Rightarrow LL(n),\\
F^{(1)}(n,u,v) &= \iint f(x,y)\cdot 2^n \cdot \phi(2^n x-u)\cdot \psi(2^n y-v) \Rightarrow LH(n),\\
F^{(2)}(n,u,v) &= \iint f(x,y)\cdot 2^n \cdot \psi(2^n x-u)\cdot \phi(2^n y-v) \Rightarrow HL(n),\\
F^{(3)}(n,u,v) &= \iint f(x,y)\cdot 2^n \cdot \psi(2^n x-u)\cdot \psi(2^n y-v) \Rightarrow HH(n).
\end{aligned}
\qquad (4.3)
$$

IDWT :

$$f(x,y) = \int_v \int_u F^{(0)}(N,u,v)\cdot 2^N \cdot \phi(2^N x-u)\cdot \phi(2^N y-v) \qquad (4.4).$$
$$+ \sum_{n=1}^{N} \int_v \int_u \begin{bmatrix} F^{(1)}(n,u,v)\cdot 2^n \cdot \phi(2^n x-u)\cdot \psi(2^n y-v) \\ + F^{(2)}(n,u,v)\cdot 2^n \cdot \psi(2^n x-u)\cdot \phi(2^n y-v) \\ + F^{(3)}(n,u,v)\cdot 2^n \cdot \psi(2^n x-u)\cdot \psi(2^n y-v) \end{bmatrix}$$

Wavelet coefficients $F^{(k)}(n,u,v)$ correspond to the projection of original signal $f(x, y)$ onto scaling function φ and wavelet function ψ, which represent the coarse information and details at scale n, respectively.

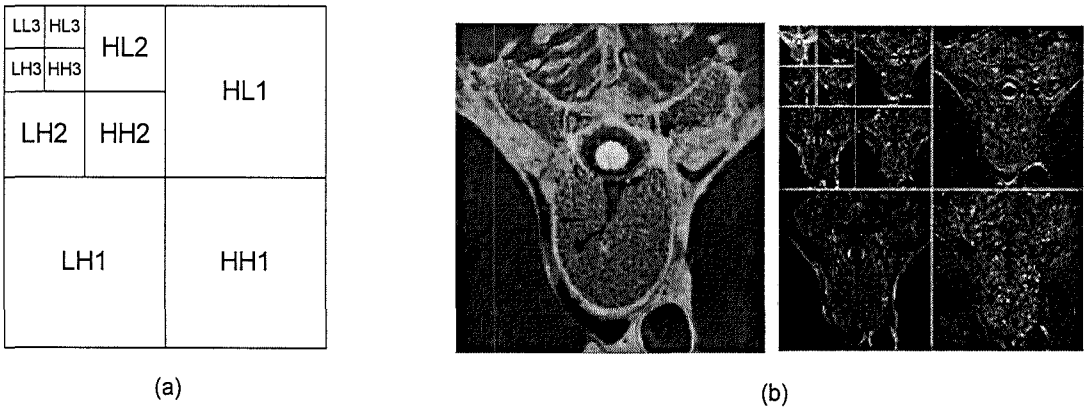


Figure 4-4: Three-level 2-D wavelet decomposition. (a) A block diagram and (b) an example on CT image (256 × 256, 8 bpp).

When the LL band is iteratively decomposed, a pyramid of subbands is formed from a coarsest scale at the top to a finest scale at the bottom. In this pyramid decomposition, each wavelet coefficient in a coarser subband is the parent of four children arranged in the form of a 2 × 2 block in the finer subband immediately below it in the pyramid at the same location, as shown in Fig. 4-4 (b). Images usually contain more energy in the lower frequencies. As a result, coarse coefficients usually have large magnitudes and the energy diminishes as we move from the top of the pyramid to the bottom. Thus the energy of a parent is indicative of the energy of its descendents. Such same-orientation ordering is one of the most desirable features in subband coding. Progressive transmission functionality is embedded by such ordering of bit stream. It has been argued that coefficient with the largest magnitude need to be transmitted first because it would provide the largest reduction in the MSE [DeVore, 1992; Said, 1996]. Furthermore, if the coefficients are represented in binary notation, the bits that

are 1's at higher bit planes provide a greater reduction in the MSE than do the '1' bits at lower bit planes when transmitted. This observation suggests that we should transmit the '1' bits at the highest bit plane first, rather than transmitting all the bits of the coefficient with the largest magnitude.

One category to order and encode the wavelet coefficients is inter-band methods designed to prioritise the significant coefficients using *zerotree*-based techniques. It has been initially investigated in [Shapiro, 1993] for two-dimensional cases. Such method, called embedded coding of zerotrees of wavelet coefficients (EZW), suggest an efficient way of ordering the bits of the wavelet coefficients for transmission. EZW is based on the observation that, if a coefficient is small in magnitude with respect to a threshold, all of its descendants are likely to be small as well. Moreover Said and Pearlman [Said, 1996] enhanced the EZW method by implementing a different ordering algorithm to convey the significance information. They referred to their work as SPIHT (set partition in hierarchical tree encoding). SPIHT produces excellent coding performance and its encoding and decoding procedures are very fast.

Another category is intra-band methods using cubic splitting combined with EBCOT (Embedded Block Coding by Optimised Truncation) [Taubman, 1999]. It has been argued that the intra-band models account for the fact that the significant coefficients are clustered in certain areas in the wavelet domain, corresponding to the edges and textural regions in the image, thus can capture most of the dependencies between the wavelet coefficients. It has been indicated that exploiting intra-band redundancies offers a better coding gain than exploiting inter-band redundancies, and that only minor gains can be obtained with composite models that capture both types of dependencies. EBCOT outperformed the other candidates by intensive objective and subjective testing, and finally was accepted in JPEG2000 standard as the major coding algorithm.

4.1.2.1 JPEG2000

JPEG2000, the latest generation of still image compression standard which started with original proposal in 1996, was finally released in 2001. The objective of

JPEG2000 is to address areas that the current JPEG standards failed to produce the acceptable quality or performance. The primary areas include:

- *Low bit-rate compression*: Current JPEG standard offers excellent rate-distortion performance at mid and high bit-rates. However, at low bit-rates, e.g. below 0.25 bpp for high-detailed gray-level images, the subjective distortion like blocky artifacts becomes unacceptable.
- *Lossless and lossy compression*: JPEG2000 supports both lossless and lossy compression by using revertible integer wavelet transform.
- *Progressive transmission*: By delicately prioritizing (ordering) the wavelet coefficients from both inter-band and intra-band, the encoded bit streams are organized in such a pattern that the image quality and resolution are progressively improved when the decoder receives more bits, and the higher the priorities of bits the more contribution to the quality improvement, and finally a lossless compressed version can be reconstructed at the decoder.
- *Region-based coding*: By defining a region-of-interest (ROI) region at the encoder, the compression routine can produce different performance in and out of this region. For instance, lossless compression can be achieve for pixels in ROI region while very high compression is performed on pixels out of this region.
- *Supporting a wider varieties of images*: The aim of the JPEG2000 is to develop a new still image coding standard for different types of still images (bi-level, gray-level, color, multicomponent, hypercomponent), with different characteristics (natural, scientific, medical, remote sensing, text, rendered graphics, compound, etc.), allowing different imaging models (client/server, real-time transmission, image library archival, limited buffer and bandwidth resources, etc.) preferably within a unified and integrated system.

Its advances in features such as progressive transmission, lossless compression and region-based coding make JPEG2000 a promising candidate for CLSM images. However, there exist some issues which are not very clear and well suited for CLSM image compression, and therefore prompt us to investigate these problems.

- *Three-dimensional image data compression*: As we described in Section 2.5, CLSM devices have three-dimensional imaging modality, which can produce a sequence of planar projections of a three-dimensional object. Pixels from the same location between consecutive images are basically correlated, while two-dimensional techniques like JPEG2000 cannot capture such correlation. Therefore a three-dimensional compressor is expected to achieve better rate-distortion performance.
- *Distortion-constraint compression*: Different from DCT, of which the basis functions are orthogonal, basis functions in wavelet transform used in JPEG2000 are biothogonal. They result in a little penalties in orthogonality while maintain the symmetry which has been indicated that symmetry is a stronger determinant of compressed image quality than orthogonality [Vetterli, 1992; Villasenor, 1995]. For biorthogonal transforms, the squared error in the transform domain is not the same as the squared error in the original image. As a result, the problem of minimizing image error is considerably more difficult than in the orthogonal case. In fact, JPEG2000 is particular designed as a rate-parameterized method instead of a distortion-parameterized method. Therefore similar to the problem in JPEG, it is difficult to achieve a target quality in one compression attempt.
- *Ringing artefacts at very low bit-rates*: In Section 4.2.2.1 we discussed the ringing artefacts coming from the coarse quantization of wavelet coefficients. Though they are not apparent in CLSM image compared to computer-generated images, it is worth attention if we can attenuate these artefacts.

We performed a similar test as Fig. 4-2 using JPEG2000 coder on the same images. We chose the compression ratio is 100:1. From the results (see Fig. 4-5) we can see that on the right side the reconstructed image still looks quite similar to the original, but on the left side the image were apparently blurred and corrupted. Therefore although JPEG2000 can precisely control the target bit rate, we have little information about the output image quality and the users have to manually balance the compression rate and image quality without explicit constraints from the encoder and hence the compression task is inconvenient to achieve a particular image quality

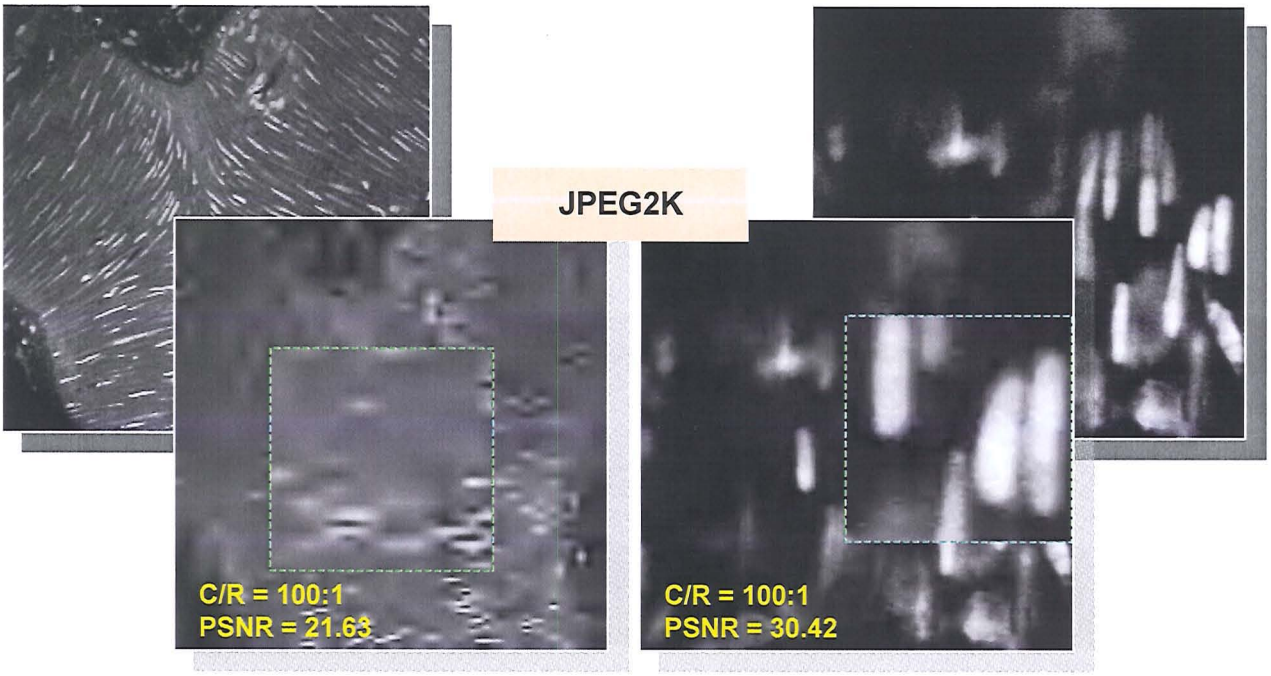


Figure 4-5: Experimental results of using JPEG2000 on two CLSM images. The decompressed images have different visual and objective quality with the same rate setup.

4.2 Three-dimensional Compression Techniques

Because three-dimensional image data can be represented as multiple two-dimensional slices, it is possible to code these images independently on a slice-by-slice basis. However, such two-dimensional methods such as JPEG and JPEG2000 do not exploit the dependencies that exist among pixel values in all three dimensions. Because pixels are correlated in all three dimensions, a better approach is to consider the whole set of slices as a single three-dimensional data set.

4.2.1 Video Compression

The development of digital video technology in the 1980s has made it possible to use digital video compression for telecommunication applications. The standard of video compression algorithms was issued by *International Committee on Telegraph and Telephones* (CCITT) in 1988 [LeGall, 1991]. The first standard was MPEG-1 and was ratified in 1992 by Moving Picture Experts Group (MPEG) committee. MPEG-1 is a standard for storage and retrieval of moving pictures and audio. The goal of MPEG-1

was to reach a medium quality video with a constant total bit-rate of approximately 1.5 Mb/sec CD-ROM playback. The picture quality of MPEG-1, however, is not suitable for broadcast applications. MPEG-2 was developed as a flexible standard providing a comprehensive tool-kit of options for broadcasting and video-on-demand. It is a standard for digital television and accommodates full television picture resolutions (including interlace) of bit-rates between 4 Mb/sec and 9 Mb/sec (for studio quality). The work on latest video standard MPEG-4 and a potential next generation standard MPEG-7 is targeting for a broader range of applications, such as mobile communications, high definition TV (HDTV) and multimedia information retrieval. Readers can find a complete reference of MPEG standards in [Watkinson, 2001]. Though there are plenty of video standards and their variations the basic idea behind a video compressor is the technique called *motion compensation.*

MPEG-1 (and -2) defines 3 types of video frames, *I*, *P* and *B* frames, which are shown in Fig 4-6. *I* (Intra) frames are coded without reference to other pictures, and they contain all the information necessary for their reconstruction by the decoder. *P* (Predicted) frames are coded from the proceeding *I* or *P* frames, using motion compensated prediction. The compression rate of *P* frame is usually higher than for *I* frames. *B* (Bi-directional or Bi-directionally predicted) frames are coded by bi-directional interpolation between surrounding *I* or *P* frames. *B* frames offer the highest compression rates. Two parameters, *M* and *N*, describe the succession of *I*, *P* and *B* frames. *M* is the distance, in number of frames, between two successive *P* frames, and *N* is the distance between two successive *I* frames, defining a *Group Of Pictures* (GOPs). An individual coding process is performed on each GOP, of which *M*, *N* can determine the compression performance.

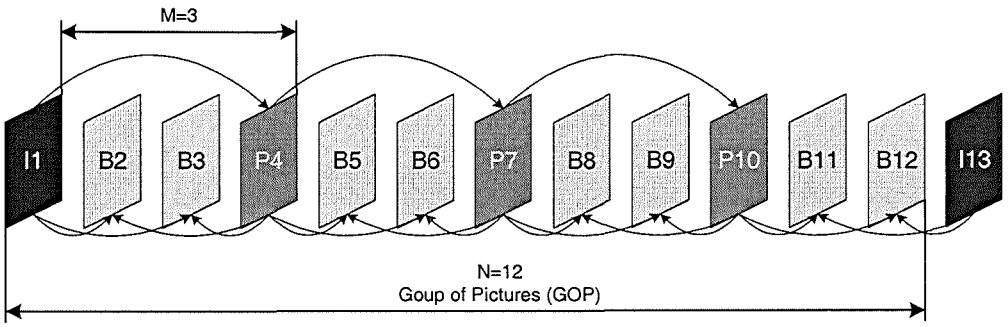


Figure 4-6: MPEG Group of Pictures (GOPs) for *M*=3 and *N*=12 comprising one *I* frame, three *P* frames and eight *B* frames.

Block-based motion estimation involves defining a motion vector, which ensures the correlation between an arrival zone on the second picture and a departure zone on the first picture, using block matching. In a given picture, a block of pixels is selected and stored as a reference. If the selected block is part of a moving object, a similar block of pixels will exist in the next picture, but not in the same place. As shown in Fig. 4-7, block matching simply moves the reference block around over the second picture looking for matching pixel values. When a match is found, the displacement needed to obtain it is used as a basis for a motion vector. The computation would become enormous when we increase the search area for fast-moving objects. It can be reduced by performing the matching in stages, where the first stage is inaccurate but covers a large motion range but the last stage is accurate but covers a small range.

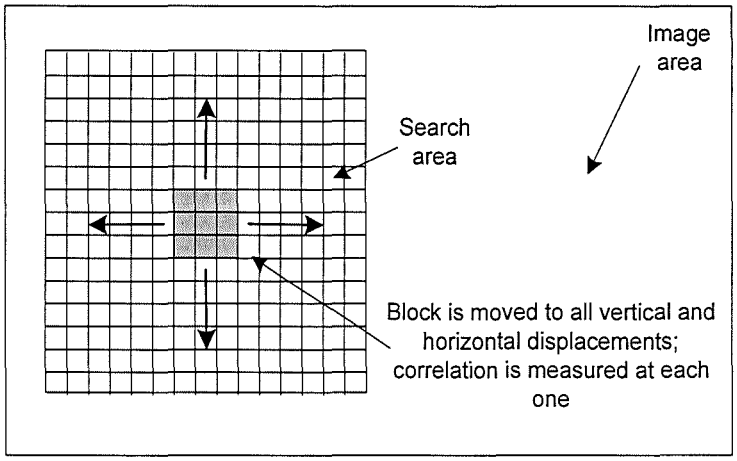


Figure 4-7: In block matching the search block has to be positioned at all possible relative motions within the search area and a correlation measured at each one.

I frames will be encoded as intra-frame coding; *P* and *B* frames will be encoded as inter-frame coding by taking the difference between the predicted (motion-estimated prediction for *P* frames and interpolation prediction for *B* frames) slices and the original ones. Both involve block-based DCT and entropy coding, like *run-length coding* (RLC) and *variable-length coding* (VLC).

MPEG coders have been investigated to code sequence of two-dimensional microscopic images [Cockshott, 2003] and medical images [Ho, 1994; Lee, 1993]. Their works demonstrated that MPEG did not make significant improvement over frame-by-frame scheme using basically two-dimensional compressor. This is because,

as we described in Section 2.4.2, CLSM devices use different imaging modality to generate an image sequence when compared to video imaging devices like video camera, and therefore both pixel intensities distributions and correlations across two consecutive frames are different between two modalities. In particular motion compensation method, which is well suited for video sequence to remove the motion redundancy, can be confused by objects moving into shade or fades [Watkinson, 2001], which is the characteristic of three-dimensional CLSM imaging when a static object is digitised as a sequence of 2D images with specific interval along the direction orthogonal to the optical plane.

Additionally in MPEG block-based DCT method like JPEG is used to encode the residual images after motion compensation. This will fundamentally introduce blocking artefacts when coarsely quantizing DCT coefficients, which are irritating to the human visual system. Moreover video coders can be used as lossy methods only, and cannot support progressive transmission. Intuitively, video coders such as MPEG family, though designed for capture inter-frame dependencies, are not good candidates for CLSM image applications.

4.2.2 3D-DCT

Volumetric coding using DCT-based techniques has been proposed and investigated in [Abousleman, 1995; Vlaicu, 1995; Lee, 1997; Schelkens, 2001]. Such JPEG-alike method was designed in order to have a good reference for DCT-based systems. The 3D JPEG-based coder is composed of a discrete cosine transform, followed by a scalar quantizer and finally a combination of run-length coding and adaptive arithmetic encoding. The basic principle is simple: the volume is divided in cubes of 8×8×8 pixels (N=8) and each cube is separately transformed using 3D-DCT, similar to a classical JPEG-coder (see Fig 4-8(a)-(b)). A DCT-transformed coefficient located at $u=(u_1, u_2, u_3)$ in the cube is computed from all the source pixels value $f(x)$ located at $x=(x_1, x_2, x_3)$ in this cube:

$$F(u) = DCT(x) = \frac{1}{\sqrt{N^3}} \sum_{x_1=0}^{N-1} \sum_{x_2=0}^{N-1} \sum_{x_3=0}^{N-1} f(x) \prod_{i=1}^{3} C(u_i) \cos\left[\frac{(2x_i+1)\pi}{2N} u_i\right] \qquad (4.5).$$

The reconstructed version of $f(x)$ can be obtained using inverse transform 3D-IDCT:

$$f(x) = IDCT(u) = \frac{1}{\sqrt{N^3}} \sum_{u_1=0}^{N-1} \sum_{u_2=0}^{N-1} \sum_{u_3=0}^{N-1} F(u) \prod_{i=1}^{3} C(u_i) \cos\left[\frac{(2x_i + 1)\pi}{2N} u_i \right] \quad (4.6),$$

$$\text{where } C(w) = \begin{cases} \sqrt{1/N} & w = 0 \\ 1 & \text{otherwise} \end{cases}.$$

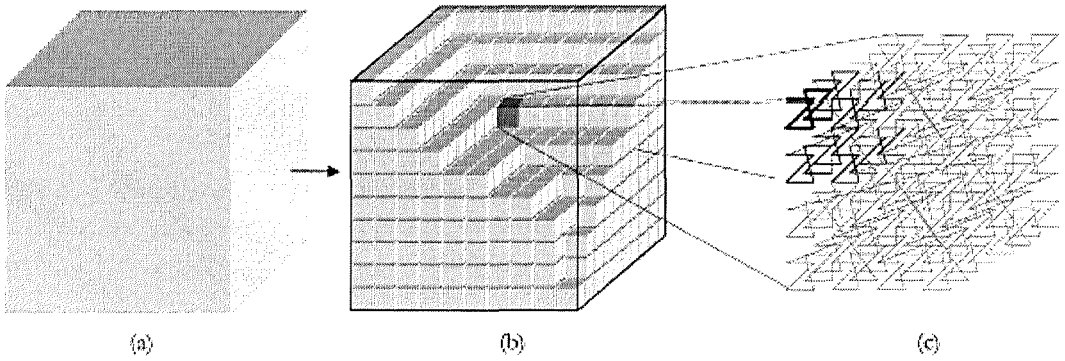


Figure 4-8: In the 3D DCT coder (a) the volume is decomposed in blocks of 8×8×8 pixels; and (b) each block is independently transformed using 3D-DCT then (c) in each DCT block the transformed coefficients are scanned according to the Morton curve, followed by quantization and entropy coding.

The quantized DCT-coefficients are scanned using a 3D space-filling curve, i.e. a 3D instantiation of the Morton-curve [Morton, 1966], to allow for the grouping of zero-valued coefficients and hence to improve the performance of the run-length coding (Fig. 4-8(c)). This curve was selected, due to its simplicity in implementation compared to that of 3D zigzag curves [Lee, 1997].

Thereafter, the DCT coefficients are quantized using a quantization matrix. In order to derive this matrix, one has to consider two options. One option is to construct quantization tables that produce an optimized visual quality based on psycho-visual experiments as they have been used in JPEG, but this approach would require elaborate experiments to come-up with reasonable quantization tables for volumetric data. The simplest solution, which is used in our implementation of 3D-DCT, is to create a uniform quantization matrix - as reported in [Schelkens, 2001]. This option is motivated by the fact that uniform quantization is optimum or quasi-optimum for most of the distributions. Actually, the uniform quantizer is optimum for Laplacian and exponential input distributions; otherwise the differences with respect to an optimal quantizer are marginal [Breger, 1972].

The 3D JPEG-like coder using block-based DCT is the first coder where real 3D volume is introduced, which treats the whole sequence as a 3D volume data and exploits the multi-dimensional redundancy within one procedure. Its effects for coding volumetric medical datasets have been investigated in [Schelkens, 2001]. They have reported that only at high bit-rates the 3D-DCT can produce good results and compete with methods using wavelet coding, and at low bit-rates, however, the performance of 3D-DCT tends to decrease fast, as it is also observed for its 2D counterparts. Moreover, the blocking artefacts observed at low bit-rates distort the visual quality. The 3D-DCT coders, like video coders, can be considered as lossy methods only, and cannot support progressive transmission.

4.2.3 3D-DWT

The wavelet transform can be extended to multiple dimensions by use of separable filters. By viewing a sequence of images as a three-dimensional image data, data along each dimension is separately filtered and downsampled with the same process. Fig. 4-9 illustrates the implementation of two levels of a three-dimensional hierarchical decomposition with separable filters. In Fig. 4-9, each subband is labelled, and the corresponding locations for storing these subbands in the output data cube are identified.

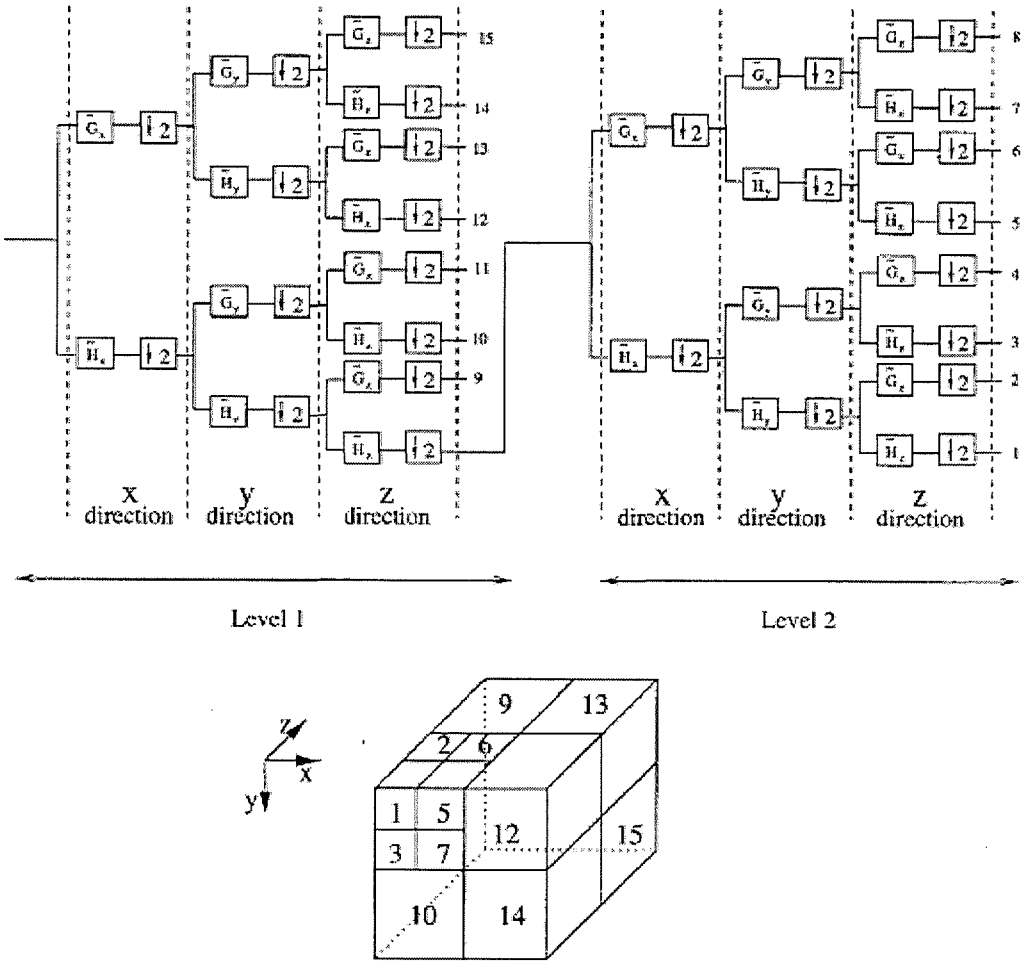


Figure 4-9: Three-dimensional wavelet analysis using separable filters.

Similar to its two-dimensional counterpart, same-orientation ordering of wavelet coefficients exists in 3D-DWT, but it is extended such that coefficients are organized in three-dimensions. Inter-band methods using *zerotree*-based techniques have been investigated, such as 3D-SPIHT [Bilgin, 2000; Kim, 1999; Kim, 2000] for three-dimensional cases. Three-dimensional intra-band methods, trying to exploit the dependencies among neighbouring wavelet coefficients from intra-band, has been extended from their two-dimensional versions and investigated in three-dimensional cases, such as 3D-EBCOT [Schelkens, 2001]. We should note that though designed for three-dimensional image data, the basic ideas of these methods are same to their two-dimensional versions. That means coefficients are processed using same routines without taking into account the potential differences of statistical characteristics from inter-frame coefficients to that from intra-frame coefficients.

Methods based on 3D-DWT have been increasingly developed and primarily investigated in coding three-dimensional medical images obtained from several imaging modalities like CT, MRI and US. It has been shown that they inherit the same features as those in 2D-DWT, such as good performance at very low bit rates, lossy-to-lossless coding functionality, quality and resolution progressiveness and precise control of bit rates. While drawbacks such as ringing artefacts at very low bit rates also exist in three-dimensional cases. These features have been considered by JPEG committee as they are working on JP3D of JPEG2000, which aims to deal with the compression of three-dimensional image data and is still under development.

4.3 Conclusions

Standardized image compression techniques, such as JPEG family and MPEG family, offers a convenient and universal environment for CLSM images. People can easily implement the algorithm or obtain the coding programs to compress and decompress these images on multiple platforms with multiple operating systems, like personal computer (PC) and handheld devices. But these techniques cannot offer excellent rate-distortion performance when working on three-dimensional CSLM images. This is because either they are single-frame-based techniques, which did not consider the potential correlations across multiple frames, like JPEG and even the latest JPEG2000; or they are using motion-compensation method to capture the inter-frame correlations, which are different from that in three-dimensional CLSM images, and therefore the efficiency is poor compared to video sequences, like MPEG family.

An increasing number of studies have been working on pure three-dimensional compressors. These methods, such as 3D-DCT and 3D-DWT, are characterized for their specific advantages and deficiencies, and motivate us to investigate the following features when coding three-dimensional CLSM images.

- *Three-dimensional architecture*: On the one hand, the compression should be a three-dimensional method, which can offer excellent coding efficiency by well capturing the inter-pixel correlations, no matter whether they are arranged in the same frame or across frames. On the other hand, since the different imaging characteristics in both the optical system and the scanning system the

intra-frame correlations could be different from that of inter-frame (see Section 2.5.1), and therefore it is worthy noting if the method can be adapted to such differences.

- *Distortion-constraint compression*: The quality is a more important factor than compression ratio in medical environments, because it is more related to the diagnostic accuracy. From the above discussion, we noticed that neither JPEG nor JPEG2000 can support distortion-constraint compression. For instance, in JPEG2000, which is a rate-parameterized compression system, there is no indication of the output quality from the control parameters in a compression task, and therefore the control of target quality would be inconvenient and not straightforward. Our method will be designed as a distortion-parameterized method, which is superior to a rate-parameterized method in controlling the compression quality. It should be emphasized that to design a quality controllable coder based on subjective image quality rating is difficult and less of generosity since the rating is primarily dependent on the knowledge and requirements from users. Therefore in our case the quality is specified by an objective distortion measure between the original and the reconstructed images, i.e. the mean-squared distortion measure discussed in Section 3.3.1.1, which has many desirable features in implementations and mathematical analyses.
- *Attenuation of artefacts at low bit-rates*: As we described in Section 3.1.2 three major types of image artefacts, blurring, blocking and ringing artefacts, exist in the reconstructed images and all of them are noticeable to human eye at low bit-rates. Blocking artefacts, primarily exist in methods based on block-based DCT, are most severe in distorting subjective image quality. Ringing artefacts in DWT-based methods, though not as severe as DCT methods in CLSM images since the presence of noise, have potential risks when post enhancement operations are performed on decompressed images. Blurring artefact caused by discarding the detailed information in small-variation regions, are of least severity in subjective quality measures. Although our method may suffer from blurring artefacts, it is expected to attenuate blocking and ringing artefacts and offer better objective and subjective rate-distortion

performance compared to methods suffering from these two artefacts, at low bit-rates.

- *Progressive transmission*: The progressiveness in quality and resolution of bit streams like JPEG2000 has more advantages in tele-communication applications than raster-ordered bit streams like JPEG. We will consider and include this feature in our method.
- *Region-based coding*: The development of computation capability makes it possible to produce and process larger volume of CLSM image data. It is worthy considering a region-based coding in CLSM applications, since it can improve the transmission efficiency of large volume image datasets by specifying the ROI regions and discarding the image quality out of these region.

Chapter 5 Compression System Design: Overview

By reviewing several standardized and state-of-art image compression techniques, we have noticed their advantages and deficiencies in three-dimensional medical image compression. To meet the goals we mentioned in Section 4.3, including better rate-distortion performance at low bit-rates and convenient control of image quality, we were encouraged to present a novel three-dimensional *Co*nfoal laser scanning *M*icroscopic *I*mage data *C*ompression system (*3D-CoMIC*), which is based on three-dimensional image pyramid transform (*3D-IPT*) and vector quantization (*VQ*).

In this chapter and the following three chapters, we describe 3D-CoMIC from an overview to specific areas where we aim to make improvements over other techniques when dealing with three-dimensional confocal microscopic images. Chapter 6 describes the issues in designing low-pass and interpolation filters, which determine the band decomposition efficiency and thus the coding performance, and how these filters are applied to images in CLSM applications. Chapter 7 and 8 are about the encoding of 3D-IPT-transformed coefficients using VQ-based techniques. Chapter 7 investigates VQ-based inter-band and intra-band encoding schemes; Chapter 8 focuses on the system optimization problems, including the bit-allocation algorithms to optimize the overall coding performance and also the improvements in computational complexities in VQ techniques.

5.1 A General View of 3D-CoMIC

Like other three-dimensional compressors such as 3D-DCT and 3D-DWT, the encoder of 3D-CoMIC accepts a sequence of same-sized two-dimensional images $\{I_i\}$ as input. A function mapping each pixel on a two-dimensional image to a voxel in a three-dimensional grid,

$$M_{\{I\}\to V} : \{V(x,y,z) = I_z(x,y), \\ x \in [0,\ldots,X-1], y \in [0,\ldots,Y-1], z \in [0,\ldots,Z-1], \\ X = \text{image.width}, Y = \text{image.height}, Z = \#\text{ of images}\} \quad (5.1),$$

is applied on a sequence of two-dimensional images $\{I_i\}$ and then obtain a three-dimensional image volume V. This process precedes our three-dimensional compression approach and then after decoding the inverse of this process is applied on the reconstructed version of V, say $\hat{V}$, to get the reconstructed version of each individual original image I_i, say $\hat{I}_i$. Since the mapping $M_{\{I\}\to V}$ and its inverse give an alternative representation between $\{I_i\}$ and V and do not involve any transforming processes or loss of information, it would be convenient to omit these two processes when describing the compression system. In such case, the input of the encoder is V and the output of decoder is $\hat{V}$. Additionally the pixel density function, which originally a two-dimensional function $I_{(x,y)}$ over (x, y) plane, now is a three-dimensional function $V_{(x,y,z)}$.

The basic ideas behind the 3D-CoMIC are subband decomposition using three-dimensional image pyramid transform and vector quantization techniques to encoding the transformed coefficients. The purpose of the former is to transform the original image signal from spatial domain to the spatial-frequency domain. It organizes the information in the image data in a more efficient way, for instance a large part of information is compacted in a small group of transformed coefficients. The latter is designed to exploit the distribution characteristics of the transformed coefficients to achieve a high coding performance. Moreover we also need to consider the distortion-control functionality when we are developing this compression system.

5.1.1 Signal Decomposition

The compression system starts from a three-dimensional image signal V as its input. V is then decomposed into several sub bands. The decomposition will generate N bands of low-pass bands $\{L_i\}$ by recursively low-pass filtering using the same low-pass filter f_{lp}. Therefore each band in $\{L_i\}$ contains a particular range of spectrum of the original signal. This process is called *subband decomposition*. The result of $\{L_i\}$ is a series of approximations of V on different scales from finest to coarsest such that L_0 is the finest representation of V which is V itself while L_{N-1} contains the lowest-frequency components of V on the coarsest scale:

$$L_i = \underset{\leftarrow\ i\ \rightarrow}{f_{lp}(\cdots f_{lp}(V)\cdots)}, \quad i = 0,\ldots N-1 \tag{5.2}$$

The purpose of the low-pass filtering is to cut off as much as possible the signals whose frequency responses belongs to (ω/2, ω] and maintain the signals whose frequency belongs to [0, ω/2]. Therefore suppose using an ideal half-band low-pass filter, the relation of frequency responses between two consecutive bands is:

$$L_{i+1}(\omega) = L_i(\omega) \cdot f_{lp}(\omega) = L_i(\omega/2) \tag{5.3}$$

To exploit the correlations of coefficients across different bands the differential signal or the high-pass bands $\{H_i\}$ are encoded instead of the low-pass bands $\{L_i\}$. This is because the frequency components in $\{L_i\}$ are overlapped such that L_i contains all the frequency components in its coarser representations L_j, $j>i$. Therefore the information between any two bands is redundant. The high-pass bands $\{H_i\}$ are computed by taking the difference between two consecutive levels of $\{L_i\}$ such that:

$$H_i = \begin{cases} L_i - L_{i+1}, & i = 0,\ldots,N-2 \\ L_i, & i = N-1 \end{cases} \tag{5.4}$$

As we will discuss in the following chapter, from the frequency analysis the high-pass bands contains the upper half-band frequencies for each low-pass band. Therefore in the ideal case there are no overlapping frequencies between any two high-pass bands and the redundancies in such representation are removed.

To reconstruct the low-pass bands $\{L_i\}$ and finally the original signal V a composition process is required for the high-pass bands $\{H_i\}$, this is referred as *subband*

composition. It starts from the coarsest level of high-pass band, say H_{N-1}; and then it recursively adds to two consecutive high-pass bands, say H_{i+1} and H_i, to obtain the low-pass band L_i. Finally the original signal V is obtained by adding L_1, which has been computed in previous recursions, with the high-pass band containing the finest high-frequency components, H_0.

Image pyramid representation is one of the good candidates for subband decomposition. It has several advantages such that it is simple in implementation, efficient in deposition performance and it is easy to be incorporated with other techniques to achieve subband-related signal processing for many applications. In the 3D-CoMIC system, we use image pyramid combined with vector quantization techniques to achieve a high-performance distortion-constraint compression of the volumetric image data.

5.1.2 Structures of VQ-based Subband Encoding

There are two forms to combine VQ-based encoding techniques with the subband structures: the opened-looped form and the closed-looped form, according to whether it employs an error-feedback scheme. In the open-looped structure the low-pass bands and high-pass bands are firstly computed, and then VQ encoding is individually performed on each level of high-pass band. At the decoder side, each high-pass band is individually reconstructed from the codes, which are generated from the encoding process on the original band at the same level, and then the low-pass bands and the final decompressed image signal are recursively reconstructed according to the previous discussion. A block diagram of the open-looped scheme is illustrated in Fig. 5-1.

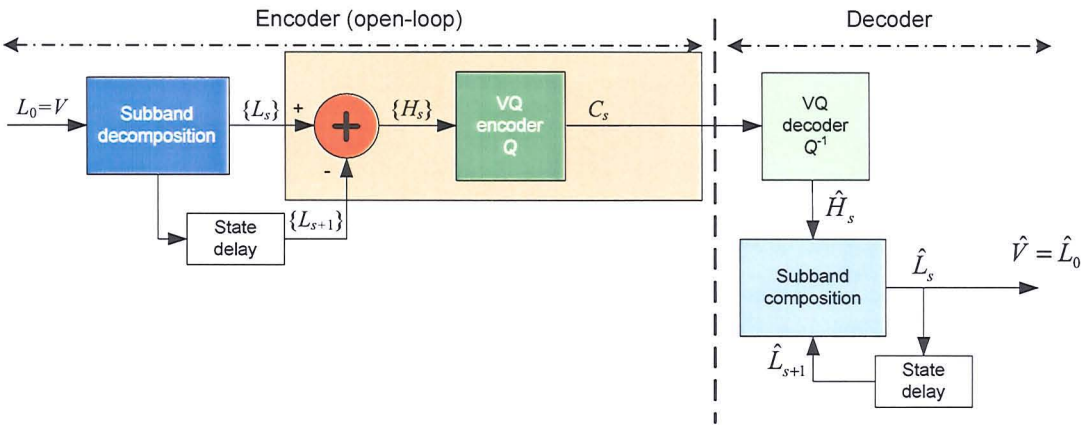


Figure 5-1: A block diagram of VQ-based subband encoding in an open-looped structure.

The control of the image quality is difficult using the open-looped structures. This is because the distortion caused by VQ-based encoding on each band will finally affect the distortion of the reconstructed finest low-pass band, the reconstructed image signal V. Moreover due to the impossibility to design an ideal low-pass filter in practices, the filters in use are all not 'perfect' in subband decomposition performance. As we will see such imperfection causes the loss of orthogonality of the decomposition, and consequently it is difficult to design a quality control scheme which can control the quality of the final reconstructed image from the encoding of these individual bands. In contrast using a closed-looped structure, the control of the final image quality can be straightforward, which has been considered in the 3D-CoMIC.

In the closed-looped form the encoding process can be described as a recursively repeated procedure consisting of three sub-processes. They are high-pass band generation, VQ-based encoding and its reverse, and reconstruction of low-pass band for compression noise recovery, as illustrated in Fig. 5-2.

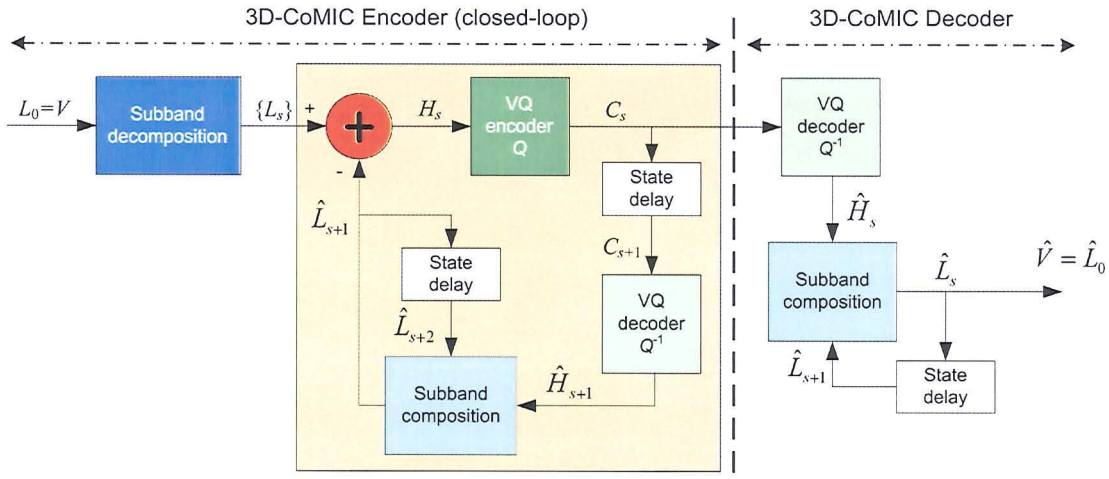


Figure 5-2: Block diagram of 3D-CoMIC encoder (left) and decoder (right).

Within the light orange block in Fig. 5-2 the encoding process starts from the coarsest high-pass band, which is the same as the coarsest low-pass band after decomposition. VQ encoding is then applied on this high-pass band H_i and produces a code stream C_i for this band. The reconstructed signal $\hat{H}_i$ of H_i can be reproduced from VQ decoder, which involves the reversed operation of VQ encoding. A reconstructed version of L_i, say $\hat{L}_i$, can be computed by the subband composition process, where the required signal, its previous coarser low-pass band $\hat{L}_{i+1}$, can be obtained from the reconstruction on the previous level. Finally the reconstructed low-pass band at current recursion $\hat{L}_i$ would be used to compute the high-pass band for the next recursion H_{i-1}. This process repeated from the coarsest level to the finest level until the original image is reconstructed in the last recursion. Therefore it is clear that at each recursion the quantization noise caused by vector quantization, which is specified by the difference between the high-pass band H_i and its reconstructed version $\hat{H}_i$, is fed back to the encoding process in the next recursion. Now in the closed-looped structures the computation of the high pass band can be expressed as,

$$H_i = \begin{cases} L_i, & i = N-1 \\ L_i - \hat{L}_{i+1}, & N-1 > i \geq 0 \end{cases} \quad (5.5),$$

where the reconstructed low-pass band is computed by the subband composition of the reconstructed high-pass band and the reconstructed low-pass band from previous recursion.

$$\hat{L}_i = \begin{cases} L_i, & i = N-1 \\ \hat{L}_{i+1} + \hat{H}_i, & N-1 > i \geq 0 \end{cases} \qquad (5.6).$$

The most significant advantage of using error-feedback scheme is its possibility to control the final image quality in a more straightforward form. This is because using such scheme for a particular level the distortion of the reconstructed low-pass band is the distortion of the reconstructed high-pass band, and does not depend on the distortion from previous levels. Therefore the distortion control problem is simplified from the encoding of the whole pyramid in an open-looped structure to the encoding of only one level of the pyramid in a closed-looped structure. This will be further explained in the image pyramid form in the following discussion.

5.2 3D Image Pyramid Structures with Compression Noise Feedback

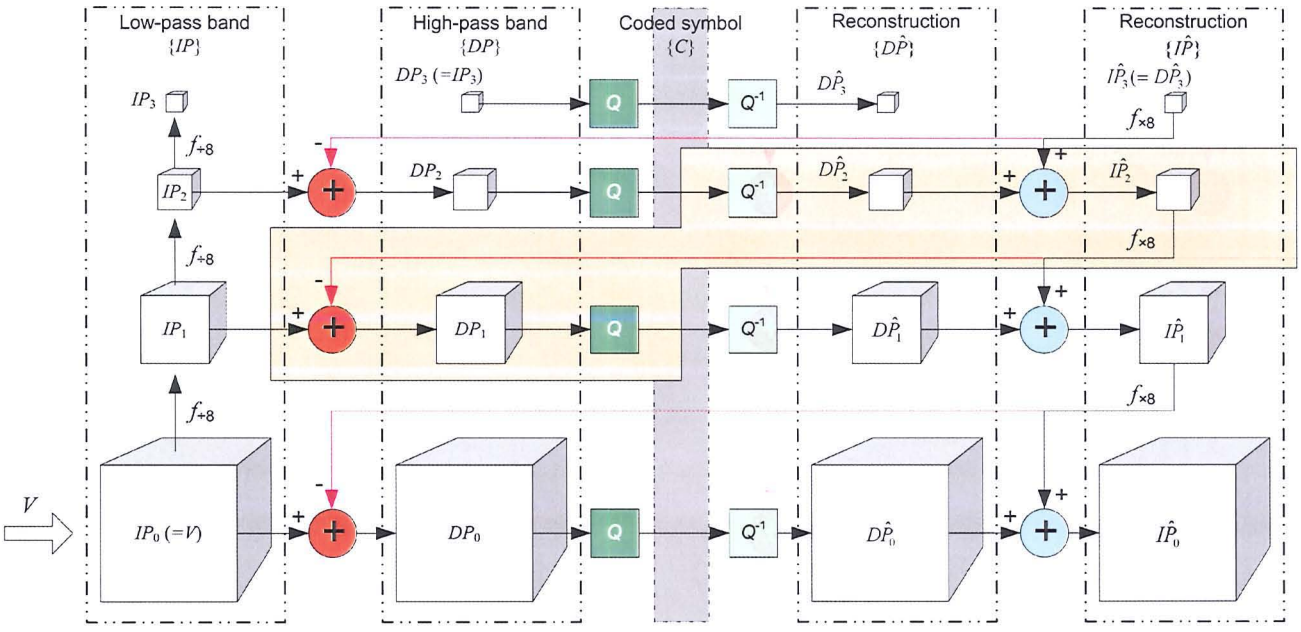


Figure 5-3: An example of encoding using four-level image pyramids with compression noise feedback.

We use image pyramid $\{IP_i\}$ structures to represent the original image signal V at different frequency scale. Each image pyramid level IP_i corresponds to a low-pass band L_i of V. Since V is three-dimensional, the low-pass filter we used is designed in

three dimensions. It is the cross-product of three identical one-dimensional half-band filters. The frequency response of IP_i is:

$$\begin{aligned} IP_i(\omega_x,\omega_y,\omega_z) &= IP_{i-1}(\omega_x,\omega_y,\omega_z)\cdot f_{lp}(\omega_x,\omega_y,\omega_z) \\ &= IP_{i-1}(\omega_x/2,\omega_y/2,\omega_z/2) \end{aligned} \quad (5.7).$$

According to *Nyquist* sampling theory [Castleman, 1995], IP_i can be sub-sampled on each dimension without introducing unwanted alias belonging to (ω/2, ω) and losing any information belonging to [0, ω/2]. Therefore the size of IP_i is reduced by a factor of two in each dimension and finally 1/8 as that of IP_{i+1}. We use a reduction function $f_{\div 8}$, which consists of a halfband low-pass filter followed by a sub-sampling filter, to compute $\{IP_i\}$. According to Eq. (5.2), an N-level image pyramid $\{IP_i\}$ is a series of low-pass filtered approximation of the original image computed by recursively low-pass filtering the previous approximation followed by sub-sampling. This can be expressed as:

$$IP_i = \begin{cases} V, & i = 0 \\ f_{\div 8}(IP_{i-1}), & 0 < i \le N-1 \end{cases} \quad (5.8).$$

IP_{i+1} is a size-reduced version of IP_i. Before we can compute $\{H_i\}$ by taking the difference of two consecutive image pyramid levels, IP_{i+1} needs to be interpolated to recover to the same size as IP_i by an expansion function $f_{\times 8}$, which consists of a up-sampling filter followed by an interpolation filter.

$$IP_i^* = f_{\times 8}(IP_i) \quad (5.9).$$

Now we can build a difference pyramid $\{DP_i\}$ corresponding to the high-pass band $\{H_i\}$. From previous discussion, according to (5.5) and (5.6), in the encoding-error-recovery scheme $\{DP_i\}$ is computed by taking the difference of IP_i and $\hat{IP}_{i+1}^*$, which is the interpolated version of the reconstruction of IP_{i+1}:

$$\hat{IP}_i = \begin{cases} Q^{-1}(Q(IP_i)), & i = N-1 \\ f_{\times 8}(\hat{IP}_{i+1}) + \hat{DP}_i, & N-1 > i \ge 0 \end{cases} \quad (5.10),$$

where $\hat{DP}_i$ is the reconstructed version of DP_i generated from the VQ decoder:

$$\hat{DP}_i = Q^{-1}(Q(DP_i)) \quad (5.11).$$

Moreover according to Eq. (5.5) the original difference pyramid $\{DP_i\}$ can then be expressed as:

$$DP_i = \begin{cases} IP_i, & i = N-1 \\ IP_i - f_{\times 8}(I\hat{P}_{i+1}), & N-1 > i \geq 0 \end{cases} \quad (5.12).$$

An example of encoding procedure using four-level image pyramid transform and vector quantization with compression error feedback is shown in Fig. 5-3. Operations are illustrated as the same colour as that in Fig. 5-2. A four-level image pyramid corresponding to the low-pass bands is initially constructed using (5.8). The encoding is recursively repeated from level 3 to level 0. The light orange block includes an individual encoding sub procedure as described in Fig. 5-2. Firstly the difference pyramid of current level i is computed by using (5.10) - (5.12), which, except for DP_3, consists of the difference signal between IP_i and $I\hat{P}_{i+1}^{*}$. Then the VQ encoding is performed on the current level of difference pyramid DP_i and produces a sub set C_i of the entire coded symbols $\{C\}$ for an individual encoding process on each level. Finally the coded symbol is both transmitted to the decoder for decompression and stored for the reconstruction of $I\hat{P}_i$, which would be used to compute next-level difference pyramid DP_{i+1}.

From Fig. 5-2 and Fig. 5-3 we noticed that the decoding process is embedded in the encoding process. This is because according to (5.10) – (5.12) before we can compute the difference pyramid for the next level DP_{i-1}, we have to decode from the coded symbols C_i to reconstruct the decoded version of the current-level difference pyramid $D\hat{P}_i$, and then reconstruct the decoded version of the current-level image pyramid $I\hat{P}_i$ by adding $D\hat{P}_i$ with the interpolated version of the reconstructed image pyramid of previous level $I\hat{P}_{i+1}$.

Such closed-looped image pyramid encoding structure inherently supports convenience control of the reconstructed image quality. Consider $I\hat{P}_i$ is the reconstruction version of IP_i at level i; and we define some unbiased bivariate function $D(\cdot)$, such as MSE, as the distortion measure between the original signal and its reconstruction. By rewriting (5.12), we will have:

$$
\begin{aligned}
& D(IP_i, I\hat{P}_i) \\
&= D([DP_i + f_{\times 8}(I\hat{P}_{i+1})], I\hat{P}_i) \\
&= D([DP_i + f_{\times 8}(I\hat{P}_{i+1})], [D\hat{P}_i + f_{\times 8}(I\hat{P}_{i+1})]) \\
&= D(DP_i, D\hat{P}_i) \\
&= D(DP_i, Q^{-1}(Q(DP_i)))
\end{aligned}
\qquad (5.13).
$$

Therefore at each level the reconstructed quality of the image pyramid is only determined by how well we encode the difference pyramid on the same level DP_i and does not depend on the encoding on previous levels. This is one of the most desirable properties of a closed-looped encoding structure compared to an open-looped encoding structure, where the reconstruction quality depends on the encoding of all the previous levels. Furthermore when we applied this rule on the bottom level we can see the distortion between the original image and its reconstruction is only determined by the encoding of the bottom-level difference pyramid, say DP_0. Therefore the quality control problem is now simplified as a distortion-constraint encoding on the bottom-level difference pyramid. The lower the distortion of the vector quantization on this level, the lower the distortion between the original image and its reconstruction, and then the higher the quality of the decompressed image.

As we know, the purpose of subband decomposition or image pyramid transform in image compression applications is to reduce the redundancy between neighbouring pixels, which often exhibit strong correlations; therefore to compact and organize the information in a more efficient way. For example, in 3D-IPT the encoding begins with the top-level difference pyramid DP_{N-1} and continues from top level down to the bottom level. For image signals, especially CLSM images, they are characterised by large low variation regions, which are often regarded as more obvious features to human eyes than the high frequency signals such as noise. Such signal, after the Fourier transform, can be well approximated using a small number of coefficients compared to the original signal. Therefore the 3D-IPT decomposes the image into a way such that the higher the pyramid levels the more likely will they represent the major features in the image. Thus coefficients from higher pyramid levels are more important in contributing to decreases in distortion. The bottom levels for example DP_0, although it has the same number of coefficient as the original, can be highly compressed since most of the coefficients produce trivial improvements in

decompressed image quality and thus can be coarsely quantized and even discarded. Moreover since the number of coefficients in each level is recursively reduced as a factor of 1/8 from bottom to top, the coding complexity in both computation and storage is greatly reduced. So in 3D-CoMIC we prioritize the pyramid-transformed coefficients from top to bottom. In the sense of rate-distortion performance the image-pyramid-transformed coefficients are organized such that encoding the higher pyramid levels will produce larger amount of decrement of distortion in the reconstructed image, than do that in lower pyramid levels, when the quantizers for these levels are working at the same rates. Therefore the closed-looped pyramid encoding scheme, which is the core of the 3D-CoMIC approach, can not only support convenient control of the reconstructed image quality by using the error-feedback scheme, but also achieve high rate-distortion performance by using the image-pyramid-based subband decomposition to compact the redundant information for efficient encoding.

We noticed from (5.8) to (5.12) that there are two important parameters controlling the subband decomposition performance in the image pyramid structure, one is the reduction function $f_{\div 8}$ and the other is the expansion function $f_{\times 8}$. More precisely they are the half-band low-pass filter in reduction function and the interpolation filter in expansion function. We will give more details in Chapter 6 regarding the design of these two filters, their band-filtering performance and how they will be implemented in 3D-CoMIC.

5.3 Encoding in Image Pyramid Structures

The encoding of the image-pyramid-transformed coefficients can either be lossy or lossless. In lossless compression, previous works [Wang, 1991; Holding, 1995] were focused on image pyramid structures without error feedback and entropy coding the coefficients for each level of difference pyramid $\{DP_l\}$ based on their probabilistic distributions. Different from other orthogonal transform like DCT, image pyramid transform based on band filtering produces more coefficients than that of original signals. Suffering from such redundancy it cannot achieve satisfactory results and compete with non-redundant transform coding, like reduced difference pyramid

coding [Holding, 1995] and differential pulse code modulation (DPCM). Recently, Aiazzi *et al.* has shown that image pyramid coding can obtain higher compression ratio with the error feedback scheme [Aiazzi, 1997]. In their work, lossy compresses were performed for all the levels of difference pyramid using scalar quantization with quantization step greater than one, except for the bottom level, where a unit step quantizer is selected to ensure the perfect reconstruction of bottom level of difference pyramid and therefore a lossless compression of the original image.

As we discussed in Section 3.1 lossless compression although ensures perfect reconstruction of the original images, will suffer from the small compression factor. Such contradiction is more severe in medical imaging applications. People prefer no quality degradation, whereas higher compression ratios are unachievable because of the existence of noise which is the determinant of compression efficiency [Roger, 1994]. Therefore we are encouraged to work on lossy compression. It offers more flexible control of a compression task, for instance it can meet various requirements on compression ratio and image quality. Moreover results from many subjective evaluations [Perlmutter, 1998; Benoit-Cattin, 1997; Cosman, 1993] have shown that when compressed at very high qualities the degradation is trivial and the diagnostic accuracies will not be affected.

Like many other transform coding techniques, in our method the loss of information occurs in one step, called *quantization*. Its performance directly affects the compression ratio and quality and therefore is the major part of a lossy compression system. It can be briefly described as a function that maps high rate digital coefficient magnitudes into a relative small number of symbols [Gray, 1998]. This operation is nonlinear and noninvertible; it is "lossy". The conversion can operate on individual sample (*scalar quantization* (SQ)) or groups of samples (*vector quantization* (VQ)). As it has been indicated in information theory [Shannon, 1959; Gray, 1998], VQ captures higher order distribution characteristics of samples than SQ does, and therefore outperforms SQ in the sense of rate-distortion performance, especially at low rates. We therefore mainly consider VQ as the quantizer.

The early works of vector quantizations in subband coding can be traced back till 1980's, when VQ was firstly introduced in image pyramid structures [Wang, 1989]. They simply used a fixed-rate VQ to encode each pyramid level separately, without employing any inter-band information or considering the non-stationary distribution of coefficients in each pyramid level. With more intensive researches on VQ approaches and the booming investigation on subband analysis especially wavelets transform in image applications, a variety of VQ-based subband image compression systems have been proposed. A brief survey can be found in [Cosman, 1996]. It has been shown that better rate-distortion performance can be achieved by using intra-band VQ structures with inter-band information and variable-rate vector quantizers than using only intra- or inter-band VQ structures and fixed-rate vector quantizers.

In our method vector quantizers are designed with the constraint of distortion to meet the requirement of convenient quality control. Its corresponding distortion-to-rate performance is measured by at least how many bits are needed to represent the original image while the distortion of the decoded image is not greater than a predefine distortion target. Moreover to exploit the non-stationary distributions of intra-band coefficients in three dimensions, VQ is preceded by classifications. It will improve distortion-rate performance and also reduce the computational complexities. Chapter 7 will cover these issues, including intra-band classified vector quantization and inter-band encoding of classification map. We will develop a novel vector quantizer – the cross-band classified vector quantizer to capture the inter-pixel dependencies from both intra-band and inter-band according to the distributional characteristics of the pyramid coefficients.

The image pyramid transform hierarchically decomposes signal into several bands or layers. The encoding of each band may affect the rate-distortion performance of the whole system. In our case such influence is in an indirect form. For instance, with error feedback scheme distortions from one level will bring its coding error to its next level, although could be recovered but increase the difficulties when encoding its next level and finally may decrease the performance as well. The encoding processes across these bands are dependent. Issues regarding bit allocation problems should be investigated. In our image pyramid coder, it includes two aspects. One is the bit-

allocation across pyramid bands or layers. In the sense of rate-distortion performance this can also be expressed as how we control the number of bits allocated for the encoding of each pyramid level to minimize the total number of bits required, while under the constraint that the distortion of the reconstructed image is no more than the target distortion level. The other aspect is how we allocate bits among vector quantizers to maximize the rate-distortion performance when encoding intra-band coefficients using classified VQ. These two bit allocation problems relate to the performance optimization of our compression system and they be discussed in Chapter 8.

Chapter 6 3D Image Pyramid Decomposition

Considerable interest has arisen in recent years regarding new subband transform techniques that specifically address the problems in image compression and many other image processing applications like feature detection and pattern analysis. These techniques come under the headings of multiresolution analysis, time-frequency analysis, image pyramid transform and wavelet transform [Pittner, 1993].

In the beginning of this chapter we review how subband transforms can be applied in image compression here we mainly focus on image pyramid transform. For simplicity we restrict most of our attention to the one dimensional case. It should be noted that they can be generalized to higher dimensions for image applications based on separable properties.

In what follows we discuss the implementation of three-dimensional image pyramids in our 3D-CoMIC coder. We begin with the design of two band filters: one for decomposition and the other for composition. Then we illustrate the construction of image pyramids and difference pyramids using these two filters. Issues regarding performance and complexity will be discussed.

6.1 Subband Coding and Image Pyramid Transform

6.1.1 Subband Analysis

Subband analysis has been long used in the area of speech analysis and acoustic signal processing. The motivation of subband analysis is to interpret the signal in a more straightforward way not only in frequency domain but also in time domain. For an acoustic signal a simple application, for example, could be the analysis of a song to determine what pitches it contains (frequency information) and when these pitches occur (time information). It is different from Fourier analysis which of course reflects the entire content of the signal, but often not in a way that is easily interpreted [Castleman, 1995]. In a Fourier transform position information, for example, is encoded in the phase spectrum in a complicated way. While the amplitude spectrum may show distinct peaks due to each of the transient signal components, this is reliable only for transient detection when those components are large enough in amplitude and duration to dominate the spectrum.

In subband analysis the original signal is, in parallel, convolved with a bank of bandpass filters for decomposing into components at different frequencies. The output for each bandpass filter identifies the corresponding frequency component of the original signal and also the location of these frequency responses. Thus we have a means of analysing the transient components of the signal in both time (location) and in the frequency domain.

In image applications many important features, such as edges, are highly localized in spatial position. They can be viewed as transient components since nonzero values occur only during a short interval. Such components do not resemble any of the Fourier basis functions, and they are not represented compactly in the transform coefficients (i.e., the frequency spectrum), as discussed above. Subband transform captures time-frequency representation. This makes it superior to Fourier coding for compressing images containing transient or localized components.

6.1.2 Subband Coding

Subband coding seeks to decompose a signal (or an image) into narrow-band components and represent these in such a way that it is possible to reconstruct the original signal without error [Woods, 1986]. Recall from Fig. 4-3 the basic structure is recursively decomposing the original signal using a halfband lowpass filter and a halfband highpass filter. The decomposition is iteratively applied on the coarser approximation of the previous result and finally generates a series of hierarchically high-passed components and the coarsest approximation of the original.

Woods described the implementations of subband coding for images. Here we simplify his ideas by describing in one dimensional case and focus the decomposition for one step, since the complete decomposition can be achieved by repeating the same process hierarchically. Consider a bandlimited signal $f(t)$, that is,

$$Fourier\{f(t)\} = F(s) = 0, \quad \text{for} \mid s \mid \geq s_{\max} \tag{6.1}$$

We can sample the signal to generate its discrete version $f(i)$ (i=0,1,...,N-1) with sample frequency $s > s_N$ ($s_N = 2s_{\max}$) according to Nyquist sampling theory to guarantee the perfect of reconstruction of $f(t)$, where s_N is the Nyquist frequency.

To compute the halfband low-passed component of $f(i)$, it is convolved with an ideal lowpass band filter $h_0(i)$ with frequency stops at $s_N/2$,

$$h_0(t) = \text{sinc}(\pi \cdot s_N \cdot t) \text{ and } H_0(s) = \Pi(\frac{s}{s_N}),$$

$$\text{where } \text{sinc}(x) = \frac{\sin(x)}{x} \text{ and } \Pi(x) = \begin{cases} 1 & |x| < \frac{1}{2} \\ \frac{1}{2} & |x| = \frac{1}{2} \\ 0 & |x| > \frac{1}{2} \end{cases} \tag{6.2}$$

The halfband lowpass filtered signal $g_0(i)$ conserves all the information of $f(i)$ except for the high-frequency component whose frequency is above $s_N/2$. This is a low-resolution (blurred) version of $f(i)$. It retains the basic shape of $f(i)$, but has lost the details. Since $g_0(i)$ has no energy above $s_N/2$, it could be subsampled by a factor of two without introducing aliasing. This process is called *subsampling* or *decimation*.

Now we can use only $N/2$ samples to perfectly represent the halfband low-passed component of $f(i)$. To see this, notice that we could recover $g_0(i)$ from its subsampled signal $g_0(2 \cdot i)$. A simple way could be (1) upsampling $g_0(2 \cdot i)$ by inserting the zero-valued odd-numbered samples, (2) filtering that signal with $2h_0(i)$. This will reconstruct the spectrum, and hence recovering $g_0(i)$. This process is called *interpolation.*

Computation of upper halfband of $f(i)$ is similar with that of lower halfband. We can isolate the high-frequency energy with an ideal halfband highpass band filter $h_1(i)$,

$$h_1(t) = \delta(t) - \text{sinc}(\pi \cdot s_N \cdot t) \text{ and } H_1(s) = 1 - \Pi(\frac{s}{s_N}) \qquad (6.3).$$

The filter produces a signal, $g_1(i)$, whose spectrum is the same as $f(i)$ when $s > s_N/2$ and zero otherwise. This signal contains exactly the high-frequency information that was eliminated from $f(i)$ by the lowpass filter (6.2), or the difference between $f(i)$ and $g_0(i)$. Again $g_1(i)$ can be subsampled by two without introducing aliasing. Since

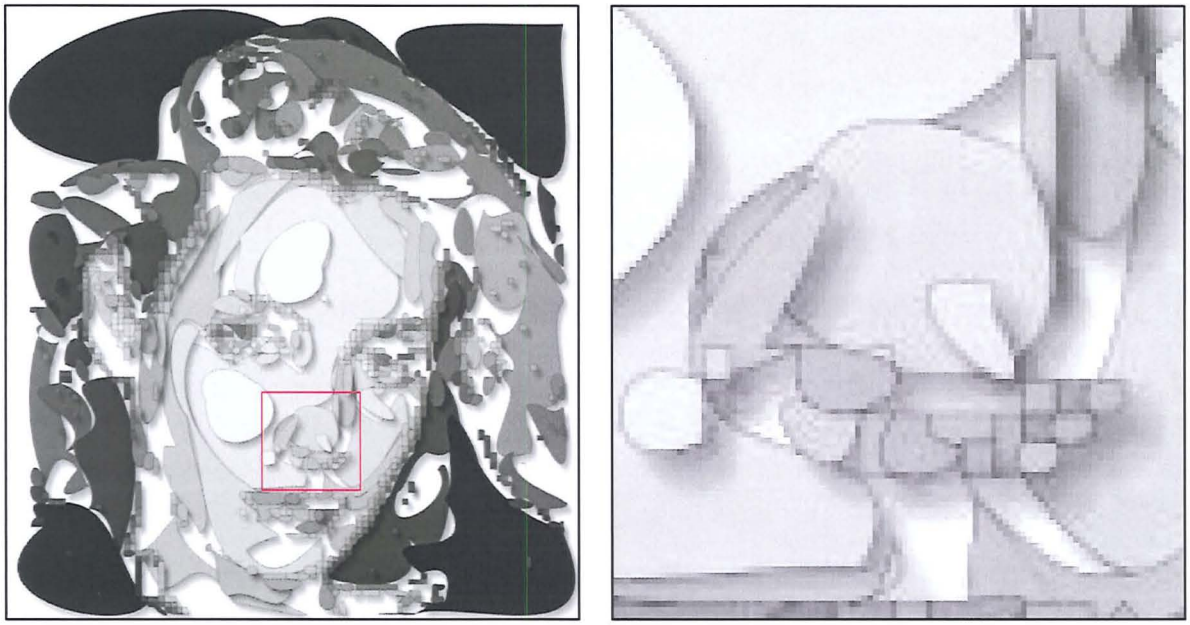
$$H_0(s) + H_1(s) = 1 \qquad (6.4),$$

the original signal $f(i)$ can be perfectly reconstructed from $g_0(i)$ and $g_1(i)$. Thus we have an invertible representation of the signal in terms of two subsampled discrete filter outputs, and it is without redundancy (i.e., exactly N coefficients).

6.1.3 Laplacian Pyramid Coding

In image analysis the *multiresolution* approach in general gives a hierarchical representation of an image on different size scales, and aims to analysis features embedded in each scale. Fig. 6-1 gives an example illustrating the approach. The left image has printing size about 8×8 cm^2. The right image is the region, specified by the red rectangle, of the left image scaled to the same size. When viewed from a normal distance of approximately eight times the image width (≈ 64 centimetres), the left image gives an impression of the face of a beauty. The right image however, contains some irregular geometry objects like circle and block. This can be interpreted that at large scales (left image), major features such as shadings and strong edges make more of an impression on the human eye while details such as individual geometry objects,

although they are elements of the image, fall below the resolving power of the eye. At smaller scales (right image), details become visible and large features are lost.



(a) (b)

Figure 6-1: Illustration of multiresolution approach. (a) The original image constructed by geometry elements like blocks and circles. (b) The red region from the left image scaled to the same size.

Image pyramid structures have a variety of applications in multiresolution analysis, depending on their purpose with different specifications in filter types and scaling factors. For example, 1/2-scaled Laplacian image pyramids have been used for lossy image compression [Burt, 1983; Aiazzi, 1997] and image restoration [Ranganath, 1991]; while 1/1.5-scaled ones for stereo image registration [Jin, 1988]. Other filters, for example 1/3-scaled median-filtered image pyramid for noise reduction [Melnik, 2001] and 1/2-scaled mean-filtered image pyramid for lossless image compression [Aiazzi, 1996], also received plenty of attention.

In image compression the motivation of the image pyramid transform is the decomposition of signals into several band components with minimal frequency overlap, while on the other hand the original signal can be reconstructed from these components. The ideal bandpass filter with rectangular shape can achieve this without any overlapping and therefore the decomposed components are orthogonal with each other. For image signal characterised with discrete and finite samples, however, it is

impossible to implement such ideal filter since it requires convolving with an infinite function – sinc(x). Therefore people began to investigate its alternatives.

Firstly introduced by Burt and Adelson [Burt, 1983] and then widely used and enhanced by [Holding, 1995] and [Aiazzi, 1997] was a pyramid coding scheme based on the Gaussian function. The image is lowpass filtered with a Gaussian-like kernel, and the result is subtracted from the original image. The high-frequency detail in the image is retained in this difference image. The lowpass filtered image can then be subsampled without loss of detail. This process iteratively involves at each step the construction of a subsampled Gaussian-filtered image and its post-interpolation difference image, as is shown in Fig. 6-2.

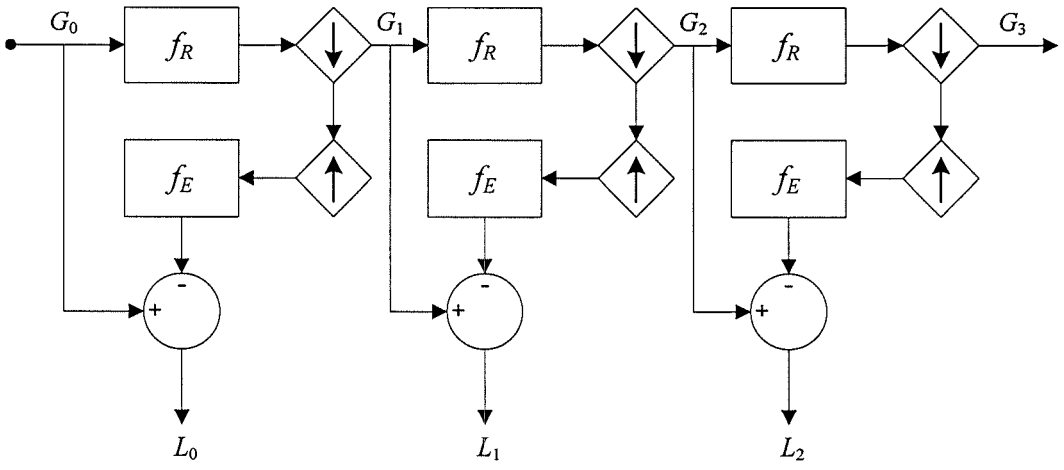


Figure 6-2: Construction of Gaussian pyramid $\{G_k\}$ and Laplacian pyramid $\{L_k\}$.

Consider N-level decomposition and G_0(i, j) be the original image, f_R(i, j) be a Gaussian-shaped lowpass filter for removing aliasing before subsampling (the *reduction* filter) and f_E(i, j) be Gaussian-shaped lowpass filter for removing aliasing after upsampling (the *expansion* filter), then at each step the image is decomposed into half-resolution low-frequency $\{G_k\}$

$$G_k(i,j) = \text{Reduction}(G_{k-1}) = [G_{k-1} * f_R](2i,2j), \quad k = N-1,\ldots 1 \tag{6.5}$$

and full-resolution high-frequency components $\{L_k\}$

$$L_k(i,j) = G_k - \text{Expansion}(G_{k+1}) = G_k(i,j) - [G_{k+1} * f_E](i,j) \tag{6.6}.$$

After N iterations the encoded image pyramid consists of the high-frequency components $\{L_k, k=1,\ldots,N\text{-}1\}$ and the lowest-frequency component G_{N-1}. Each $\{G_k\}$ including the original image can be perfectly reconstructed from L_k and G_{k+1}.

From [Burt, 1983] and [Aiazzi, 1997], we noticed that each $\{L_k\}$ is the difference of two images obtained by convolving a single image with Gaussian of single and double width. This is equivalent to convolving the image with the difference of two Gaussians, which, in turn, approximates the "Laplacian of a Gaussian" highpass filter; hence the name chosen for such pyramid structures (see Fig. 6-3).

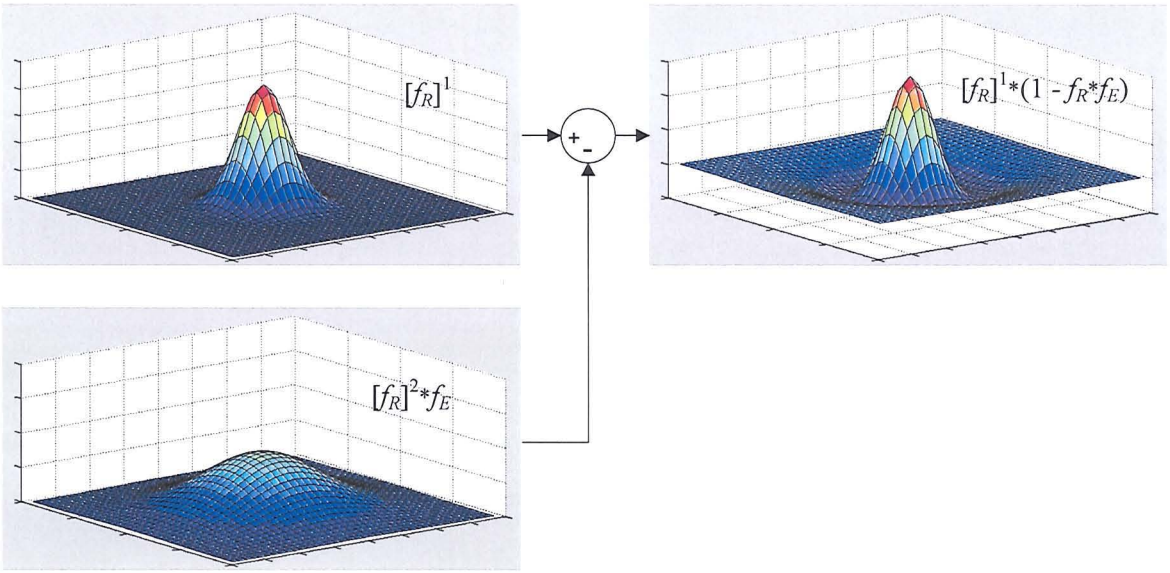


Figure 6-3: An example of two-dimensional Laplacian of two Gaussian filters. One specified by f_R; and the other is specified by f_R and f_E, with approximately double width as the former.

6.2 Reduction and Expansion Filter Design

The design of the reduction filter f_R and expansion filter f_E should consider the following aspects.

1. *Good band-pass performance*: Effectively remove the upper halfband information while retain the lower halfband signal.
2. *Separability*: Simplify the implementation of higher-dimensional filters, like three-dimensional in our case. Filters can be designed and evaluated on one dimension first then extended to higher dimensions.
3. *Symmetry*: Guarantee the performance on marginal samples where only partial information is available.
4. *Low computational complexity*: We restrict the size of filters to reduce the computational complexity in image pyramid transform, especially significant for large data set (volumetric image data).

The Gaussian-like band-pass filters mentioned above, although are easy to implement and have very low computation complexity, are not well behaved in the sense of bandpass. This is because the spectrum of a Gaussian function is also a Gaussian, which is not "steep" to give good stopband and passband performances at $s_N/2$. By noticing from Eq. (6.2) the frequency response of a rectangular function is a *sinc* function. We therefore turned to a *sinc*-like filter.

6.2.1 Reduction Filter

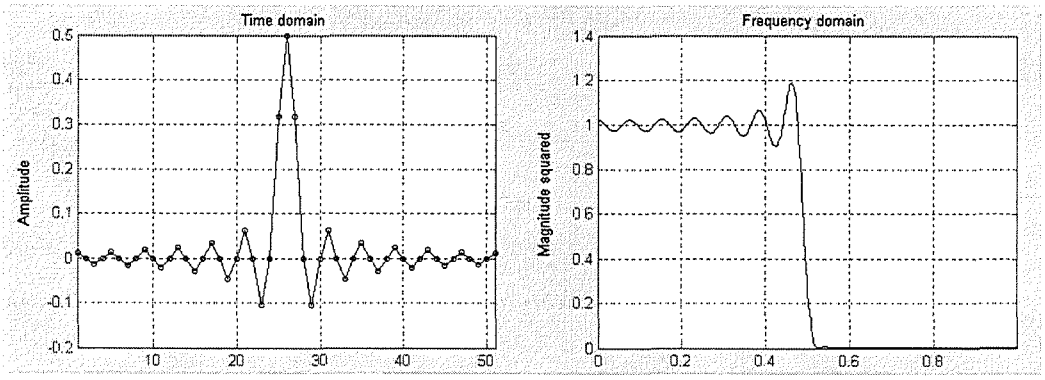
Without loss of generalization we consider the spectrum within $[-\pi, \pi]$. Consider the ideal, or brick-wall-like, digital lowpass filter with a cutoff frequency at s_0. This filter has magnitude 1 at all frequencies with magnitude less than s_0, and magnitude 0 at frequencies with magnitude between s_0 and π. From [Oppenheim, 1989], we found its impulse frequency $h_0(n)$ is

$$\begin{aligned} h_0(n) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} H_0(s)e^{jsn}ds \\ &= \frac{1}{2\pi}\int_{-s_0}^{s_0} e^{jsn}ds \\ &= \frac{s_0}{\pi}\,\mathrm{sinc}(\pi\frac{s_0}{\pi}n) \\ &= \frac{1}{2}\mathrm{sinc}(\frac{\pi}{2}n), \quad s_0 = \frac{\pi}{2}, \quad n \in Z_{(-\infty,+\infty)} \end{aligned} \tag{6.7}.$$

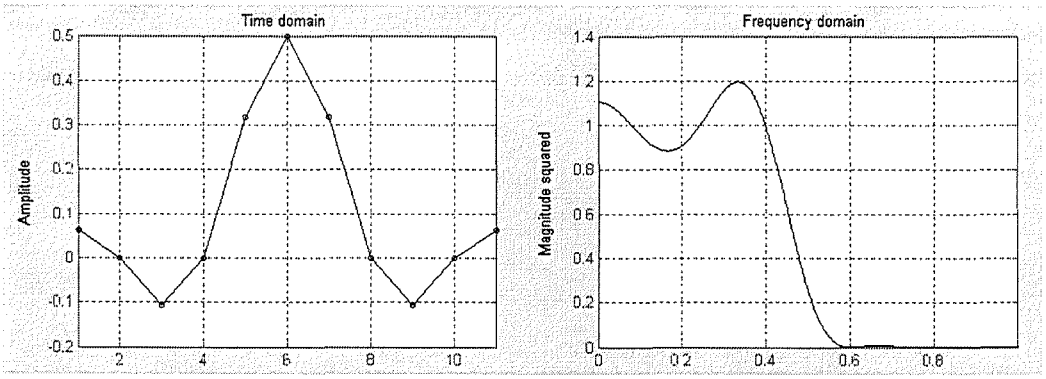
This filter is not implementable since its impulse response is infinite and noncausal. To create a finite-duration impulse response, we truncate it by applying an N-sized window such that we retain N samples (odd) of the central section of h_0. By substituting $\mathrm{sinc}(x)$ with (6.2) and truncate the central section, an N-tap lowpass filter $h_1(N)$ can be computed as

$$\begin{aligned} h_1(N) &= \left\{ \frac{1}{2}\mathrm{sinc}\,(\frac{\pi}{2}n), \quad n \in Z_{\left[-\frac{N-1}{2},\frac{N-1}{2}\right]} \right\} \\ &= \left\{ \frac{\sin(\frac{\pi}{2}n)}{\pi n}, \quad n \in Z_{\left[-\frac{N-1}{2},\frac{N-1}{2}\right]} \right\} \end{aligned} \tag{6.8}.$$

By Parseval's theorem [Kaplan, 1992], h_1 is the length N filter that best approximates the ideal lowpass filter, in the integrated least squares sense. Fig. 6-4 illustrates the impulse response and the spectrum of two $h_1(N)$ with N=51 and 11, respectively.



(a)



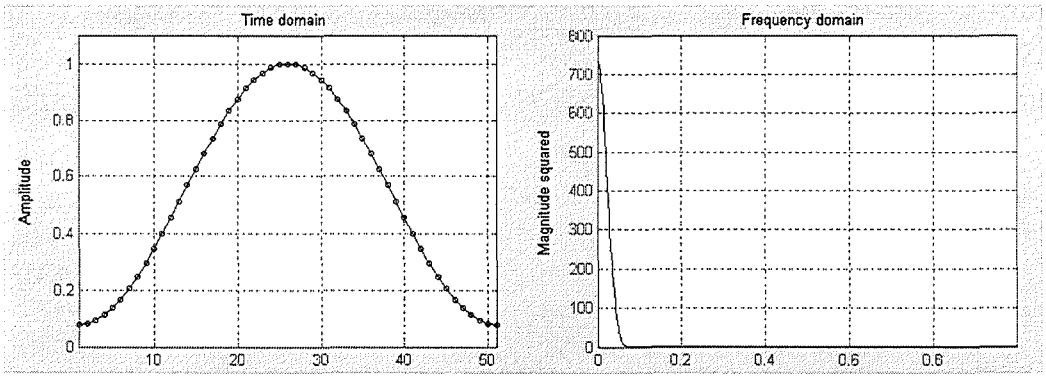
(b)

Figure 6-4: The impulse responses of $h_1(N)$ in time domain (left side) and their spectrums (right side), with (a) N=51; and (b) N=11.

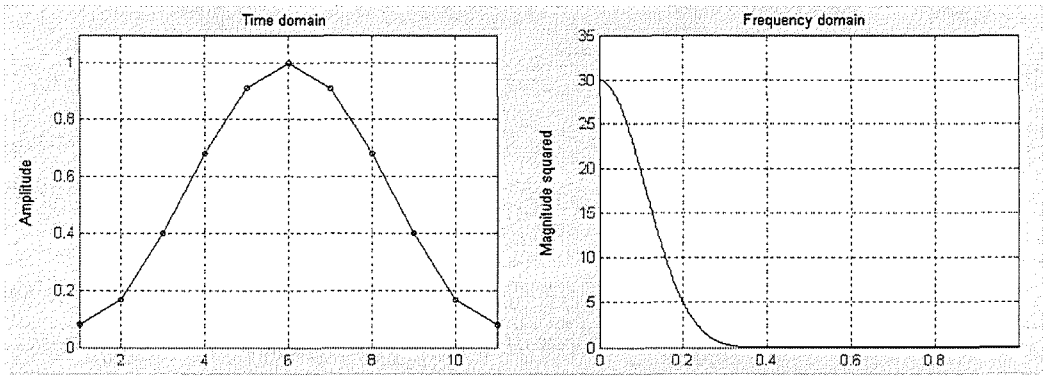
In Fig. 6-4 we noticed the ringing and ripple in the spectrum for both Ns, especially near the band edge on the bandpass side. This is referred as "Gibbs effect" and it does not vanish as the filter length increases. Since the multiplication operation in the time domain causes a convolution in the frequency domain, investigations have been working on a filter φ, whose frequency response has smoothing property. Hamming filter [Oppenheim, 1989], or Hamming window in discrete form, can achieve this. The hamming window with size N in time domain can be computed as,

$$\varphi(N) = \left\{0.54 - 0.46\cos(2\pi\frac{n}{N-1}), \quad n = 0,\ldots,N-1\right\} \qquad (6.9).$$

Figure 6-5 gives two examples of hamming window $\varphi(51)$ and $\varphi(11)$. Both when observed in time domain have approximately Gaussian shapes. The width of, for example $Fourier[\varphi(51)]$ is smaller than that of $Fourier[\varphi(11)]$. This corresponds with the narrower ripple width in $Fourier[h_1(51)]$.



(a)



(b)

Figure 6-5: Two hamming window $\varphi(N)$ with window size (a) N=51; and (b) N=11.

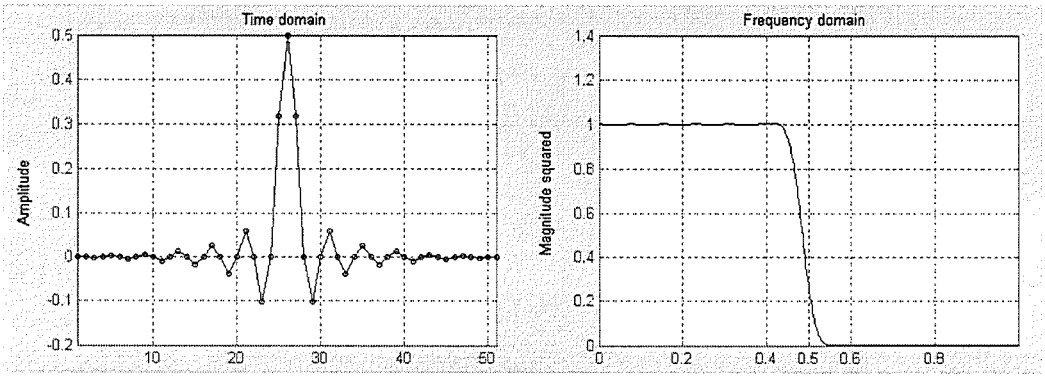
By multiplying h_1 with φ, both are specified by the same filter size N, we have a way of computing N-tap halfband lowpass filter h_2.

$$h_2(N) = h_1(N) \cdot \varphi(N)$$

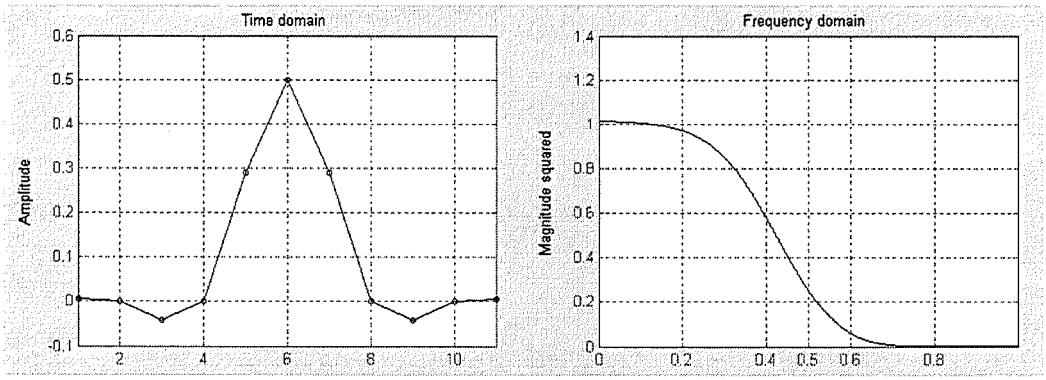
$$= \left\{ \frac{\sin(\frac{\pi}{2} i)}{\pi i}, i \in Z_{\left[-\frac{N-1}{2}, \frac{N-1}{2}\right]} \right\} \cdot \left\{ 0.54 - 0.46\cos(2\pi \frac{j}{N-1}), j \in Z_{[0,N-1]} \right\} \quad (6.10)$$

$$= \left\{ \frac{\sin(\frac{\pi}{2} i)}{\pi i} [0.54 - 0.46\cos(2\pi \frac{2i+N-1}{2N-2})], \quad i \in Z_{\left[-\frac{N-1}{2}, \frac{N-1}{2}\right]} \right\}$$

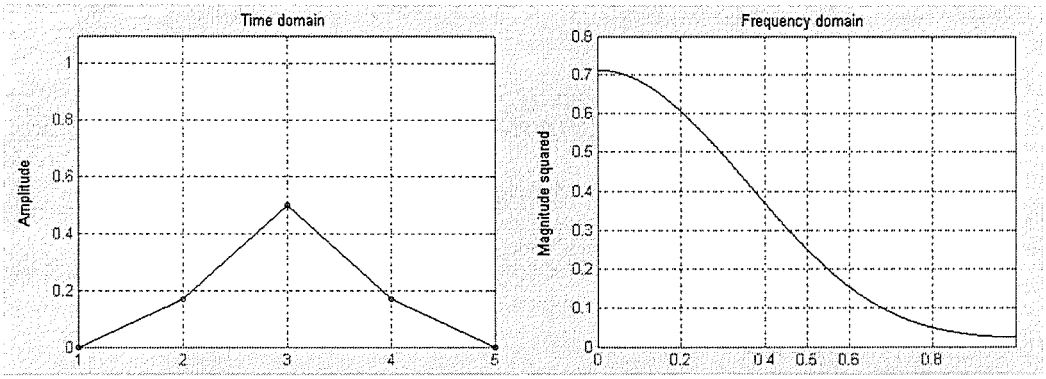
We illustrates three examples of $h_2(N)$ with N=51, 11, and 3, respectively in Fig. 6-6. Compared with Fig. 6-4, we can see that huamming filter greatly reduces the ringing effect. This improvement is at the expense of transition width such that the convolution decreases the ramp from passband to stopband.



(a)



(b)



(c)

Figure 6-6: Three examples of ringing reduced halfband lowpass filters h_2 obtained by convolving h_1 with φ in the frequency domain. The filter length is (a) 51, (b) 11 and (c) 5, respectively.

From Fig. 6-6 we noticed that the frequency response of $h_2(N)$ produces a more accurate approximation of the ideal halfband lowpass filter with the increase of filter

length N. Considering the bandpass performance and computational complexity mentioned above, we finally choose $h_2(11)$ as the kernel of reduction filter f_R in our three-dimensional image pyramid transform. Therefore f_R in one dimension is:

$$f_R^{(1)} = h_2(11) = \begin{bmatrix} 0.0051 \\ 0 \\ -0.0422 \\ 0 \\ 0.2903 \\ 0.5 \\ 0.2903 \\ 0 \\ -0.0422 \\ 0 \\ 0.0051 \end{bmatrix}^T \tag{6.11}$$

Figure 6-7 illustrates the impulse response and the spectrum of the 5-tap Burt's kernel with generating parameter a=0.6, which has been evaluated in [Burt, 1983] and [Aiazzi, 1997] that a=0.6 produce the best performance in the sense of minimizing the zero-order entropy. Comparing the Burt's kernel with $h_2(5)$ (Fig. 6-6(c)) of the same length we noticed that Burt's kernel produces higher lowpass bandpass performance. This is because $h_2(5)$ transfers only approximately 70% of low-frequency energy. That means pixel intensity of the lowpassed image will be reduced, even in the stationary region, and therefore it produces a lowpassed image with more blur and less contrast than that using Burt's kernel. However compared with $h_2(11)$ (Fig. 6-6(b)), the latter outperforms Burt's kernel since more energy will be stopped at cut-off frequency (0.5). People [Wolberg, 1990; Castleman, 1995] have shown that the failure to fully suppress the stopband, usually known as *frequency leakage*, allows the offending frequencies to fold over into the passband range and tends to be more serious since they actually corrupt the components in the original signal and are visually perceived more readily. Therefore the folded high-frequency energy will bring more severe artefacts to the next-level image pyramid due to subsampling, and eventually affect the difference signal and decrease the coding efficiency.

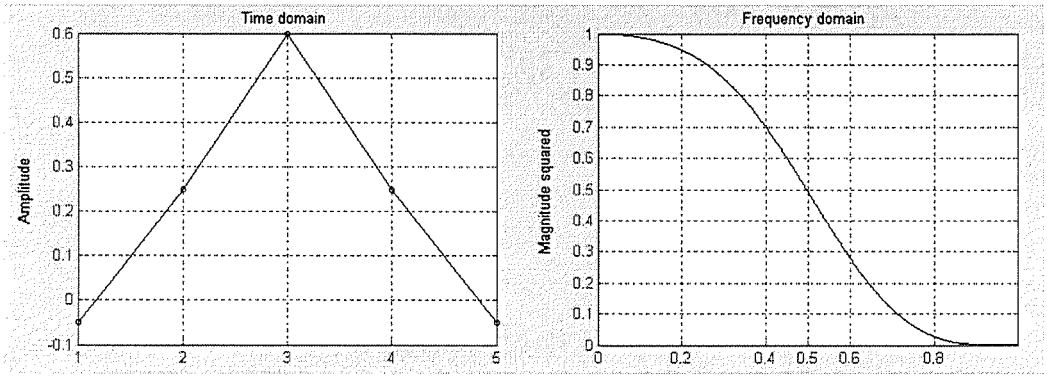


Figure 6-7: The impulse response and the spectrum of the 5-tap Burt's kernel with a=0.6 [Burt, 1983].

6.2.2 Expansion Filter

It follows from the above that in the frequency domain the goal of an expansion filter (or interpolation operation) is to recover the halfband lowpass filtered signal from its subsampled version. Since upsampling folds at frequency s_0 all the frequencies whose magnitudes are less than s_0 to the upper halfband, the expansion filter needs to lowpass the folded spectrum and cutoff high frequency component at s_0. While in the time domain it is to interpolate or estimate the values of missing samples from known ones given the spatial relationship between these samples.

A major part of image interpolation techniques were investigated in the time domain. This is because it is more straightforward and convenient to model the image in the time domain and as Muresan [Muresan, 2001] stated that any successful image interpolation method requires a good image model. Among these techniques are standard linear and cubic interpolations, where they assume that image is locally cubic, edge-directed interpolation, where in [Li, 2001] the assumption is that locally each pixel can be represented as a linear combination of the known neighbouring pixels, and interpolation using optimal recovery [Muresan, 2001; Shenoy, 1992; Michelli, 1976], where they assumed that image had been passed through a known low-pass filter, before decimation. The latter two methods are based on learning. Consider an image with size $w \times h$ is about to be interpolated to size $\alpha w \times \alpha h$, $\alpha > 1$. In Li's method the correlation coefficients between interpolated pixel and its causal neighbouring pixels are computed by learning from the training pixels in a training window. These pixels were from the original image and were assumed to exhibit

similar correlations to their neighbouring pixels on original scale as those of the interpolated pixel on scale $1/a$. While Muresan's idea is to use the original image, or a collection of other similar images, to learn the form and the coefficients of a polynomial interpolation function.

People also investigate the interpolation methods in frequency domain. In [Burt, 1983], they utilize the same kernel (Fig. 6-7) for both reduction and expansion. Aiazzi [Aiazzi, 1997] designed separate kernel for expansion. It is a 7-taps parametric kernel, which defined as:

$$f_{E,Aiazzi}^{(1)} = \begin{bmatrix} 0.5-b \\ 0 \\ b \\ 1 \\ b \\ 0 \\ 0.5-b \end{bmatrix} \qquad (6.12),$$

where b is an adjustable parameter that determines the shape of the frequency response. Moreover they have pointed out that the value b=0.5 and b=0.5625 yield linear and cubic interpolation, respectively.

Although the adaptive training-based interpolation methods like [Li, 2001] and [Muresan, 2000] outperforms the non-adaptive methods like cubic, we finally choose (6.12) with b=0.5625 as the kernel of expansion filter f_E in our three-dimensional image pyramid transform. That is,

$$\begin{aligned} f_E^{(1)} &= f_{E,Aiazzi}^{(1)}, b = 0.5625 \\ &= \begin{bmatrix} -0.0625 \\ 0 \\ 0.5625 \\ 1 \\ 0.5625 \\ 0 \\ -0.0625 \end{bmatrix}^T \end{aligned} \qquad (6.13).$$

We made this judgement based on three aspects. The first one is the high computation costs for training-based methods. As we discussed above adaptability is achieved by

acknowledging the non-stationary distribution of pixel intensities, such as edges, and learning the local features to predict the missing data. It is at the expense of introducing much more computational complexities. For instance in our implementation of cubic and complexity-reduced edge-directed interpolation [Li, 2001], the latter was about twenty times slower than the former, given a typical edge-rich microscopic image. Further such problem will be more severe when working with three-dimensional image data, and the prediction model of the causal pixels will be more sophisticated compared to their model in two-dimensional case. Since the interpolation process is at the decoder side, where we would prefer a high-speed decoding to suit the applications like browsing, this problem keeps us from choosing these high complexity methods.

The second reason is that although the fact that apparent improvement of adaptive methods over non-adaptive methods has been reported, such evaluations are mainly subjective. This means the former can recover the 'zigzag' patterns appeared on strong edges using cubic interpolations and therefore please the human eyes with smooth contours, however when interpolating from a lowpassed subsampled image they are not optimum in the sense of minimizing the differences between the original and the interpolated image. Actually in some cases the former will produce an 'edge-sharpened' image compared to the original. This will introduce unexpected high-frequency components when computing difference pyramid and decrease the coding efficiency.

Additionally as reported by Aiazzi [Aiazzi, 1997] the primary transform gain[1] of image pyramid transform is achieved by the reduction filter rather than the expansion filter. In their experiments they use Fig. 6-7 as the reduction kernel and (6.12) the expansion kernel. The curvature of the function of entropy versus a in the reduction kernel with optimum fixed b value is larger than that of entropy versus b in the expansion kernel with optimum fixed a value, which implies that after we design a good reduction filter the impact of interpolation is not significant.

[1] A measurement of a transform specifying the performance in compacting information into a small number of coefficients.

6.2.3 Implementations in Three Dimensions

In thee-dimensional image pyramid transform the reduction and expansion filter are implemented in three dimensions. The filter is specified by a three-dimensional matrix,

$$f^{(3)} = f(x,y,z), \quad x \in Z_{[0,L_x-1]},\ y \in Z_{[0,L_y-1]},\ z \in Z_{[0,L_z-1]} \tag{6.14},$$

where L_x, L_y and L_z are sizes of filter in x, y and z dimensions, respectively. Since the filter is symmetric, L_x, L_y and L_z are odd and the filter is centred at the position $(\frac{L_x-1}{2}, \frac{L_y-1}{2}, \frac{L_z-1}{2})$. The filtering of a three dimensional image signal is the convolution of two three-dimensional matrix, the image data V and the filter $f^{(3)}$, where V is defined by Eq. (5.1).

$$\begin{aligned} V'(x,y,z) &= V * f^{(3)} \\ &= \sum_{z=0}^{L_z-1}\sum_{y=0}^{L_y-1}\sum_{x=0}^{L_x-1} V(x+i-\frac{L_x-1}{2}, y+j-\frac{L_y-1}{2}, z+k-\frac{L_z-1}{2}) \times f(i,j,k) \end{aligned} \tag{6.15}.$$

According to the aforementioned desired features of the filters we consider $f^{(3)}$ is separable; that is $f^{(3)}$ can be computed by the cross product of three one-dimensional filter f_x, f_y and f_z. They are separately designed for one-dimensional signal V_x, V_y, V_z, respectively, which are obtained by raster-scanning V with different ordering, for instance in V_x the scan order is $x \rightarrow y \rightarrow z$. Since in a three-dimensional image pyramid structure, subsampling and interpolation are performed simultaneously in all directions with same reduction or expansion factor, the three one-dimensional filters are basically the same. So the three-dimensional reduction filter f_R and expansion filter f_E will be,

$$f_R = f_R^{(1)} \otimes f_R^{(1)} \otimes f_R^{(1)} \tag{6.16}$$

and

$$f_E = f_E^{(1)} \otimes f_E^{(1)} \otimes f_E^{(1)} \tag{6.17}$$

respectively, where $f_R^{(1)}$ and $f_E^{(1)}$ are computed from (6.11) and (6.13).

One of its advantages using separable filter is the reduction of computational complexity of filtering compared to that in non-separable cases. According to (6.15),

for non-separable filtering the convolution of one sample involves $L_x \times L_y \times L_z$ times[2] multiplication and addition operations, and therefore the total number of operations will be $X \times Y \times Z \times L_x \times L_y \times L_z$, where X, Y, Z are sizes of V in x, y and z dimensions. However using separable filter, (6.15) can be decomposed as three separate consecutive one-dimensional filtering operations. That is

$$\begin{aligned} V'(x,y,z) &= V * f^{(3)} \\ &= (((V * f_x) * f_y) * f_z) \end{aligned} \qquad (6.18).$$

Each convolution needs $X \times Y \times Z \times L_\alpha$ operations, where L_α could be L_x, L_y or L_z. So the total number of operations for separable cases is $X \times Y \times Z \times (L_x + L_y + L_z)$. Consider the reduction filter (6.11), for example, according to (6.15) and (6.18) the computation complexity using separable filter would be only $3 \times 11/11^3 \approx 1/40$ as that using non-separable filter.

6.3 Computing Image and Difference Pyramid

In Chapter 5 we have clearly described how to compute image pyramid and difference pyramid and formulate the procedures in (5.9) ~ (5.13). We noticed there are four primary operations involved in the computations. They are reduction function $f_{\div 8}$, the expansion function $f_{\times 8}$, the addition operation '+' and the subtraction operation '-'. Although they are well described in theory, some issues regarding implementing them into executable program need to be considered.

6.3.1 Integer Valued Pyramid

The reduction filter and expansion filter in (6.11) and (6.13) will produce floating point values. If we represent the pyramid coefficients with floating point data type, which of course increase the precision of computation, it will however considerably increase computational complexity and storage requirement. Since the original CLSM image data are in most cases integer valued and we noticed from (6.11) and (6.13) that the filters are magnitude conserved, that is $\sum f^{(1)}(i) = 1^3$, therefore the filtered

[2] Here we ignore the marginal samples where only a partial of samples bounded by the signal dimensions are involved.

[3] When using (6.13) to perform interpolation of an upsampled image, the centre coefficient will always multiply with a zero-valued sample; therefore contribute nothing to the convolution result.

coefficient will not exceed the range of the original signal. Consider using IEEE-format 32-bit floating-point data type, which is very typical in many operating systems, the data range is from $1.175494351\times10^{-38}$ to $3.402823466\times10^{+38}$. This will greatly reduce the efficiency of computation and storage since the coefficients magnitudes will usually range from 0 to 255, or 0 to 65535 at most. Moreover in the view of information theory, such representation is redundant, and thus decreasing the coding efficiency and introducing more difficulties in analysing the signal such as entropy. We therefore turned to study the integer valued pyramid.

In an integer valued pyramid, *roundoff* operation is performed on each filtered coefficients, where the roundoff operator accepts any types of number (integer, float etc.) and outputs and integer valued number which is closest to the input. As it has been investigated in [Aiazzi, 1997], although small roundoff errors are introduced in integer valued pyramids, this error can be recovered at the next lower pyramid level with error-feedback scheme. Since the original signal or the bottom level coefficients are integer valued, the perfect reconstruction is still guaranteed. Therefore in our three-dimensional image pyramid structures all coefficients are integer valued. To achieve this, apart from filtering and resampling operations in the reduction function $f_{\div 8}$ and the expansion function $f_{\times 8}$, we also include roundoff operations.

$$A_R = f_{\div 8}(A) = \text{subsampling}(\text{roundoff}(f_R(A))) \quad (6.19),$$

$$A_E = f_{\times 8}(A) = \text{roundoff}(f_E(\text{upsampling}(A))) \quad (6.20).$$

6.3.2 Data Types in Pyramids

Consider the analogue-to-digital component of a CLSM device quantize the signal into I_L intensity levels, which is ranged from 0 to I_L-1. For raw data each colour component of each pixel will take up at least

$$b = \lceil \log_2(I_L) \rceil \quad (6.21)$$

bits in computer memory. We learned from Chapter 3 that in our experiment CLSM images are all consisting one colour component with I_L=256 intensity levels. Each pixel can be represented by an eight-bit integer value. Therefore we use a three-

dimensional BYTE array for the raw image data, where in C language BYTE data type is defined as:

$$\text{BTYE} \equiv \text{unsigned int\_8} = Z_{[0,255]} \tag{6.22}$$

As we discussed above, the coefficients after reduction function will not exceed the range as that of the original signal, so the coefficients transformed from the raw image data can be represented with BYTE data type as well.

It needs more considerations when we compute the reconstructed image pyramid $\{ \hat{IP}_i \}$. Noticed from (5.11) $\hat{IP}_i$ might be different from IP_i because of the quantization errors from VQ. More precisely the data range might exceed [0, 255], say overflow (>255) or underflow (<0), especially when the coefficients in IP_i is close to the lower bound or upper bound and the quantization quality is poor. To consist with the same data range as that of the original signal, we have to restrict the range of the reconstructed signal and therefore *truncation* operations are needed:

$$\hat{IP}_i(x) = \begin{cases} 255, & \hat{IP}_i(x) > 255 \\ 0, & \hat{IP}_i(x) < 0 \\ \hat{IP}_i(x), & \text{otherwise} \end{cases} \tag{6.23}$$

Such truncation is more reasonable in reconstructing the original signal than forcing the value to BYTE data type using data type conversion operator in C. In the latter the conversion is performed by discarding the higher bits of a number when it exceeds the specified data range. Suppose the value of an original coefficient is 240 and the quantization error is +35, then we have a reconstructed coefficient whose value is 240+35=275. In the former case, the final value will be 255 according to (6.23), which the difference to the original is +15; while in the latter case, we rewrite 275 in binary mode: 275 = 0x100010011, then after discarding the most significant bit using data type conversion routine in C, the final value is 0x00010011=19, which the difference to the original is -221. Moreover truncation has a promising property, which as illustrated in the former case it reduces the effect quantization error from +35 to +15. This quantization errors recovery feature makes the distortion between the original image pyramid and its reconstruction consistently smaller than that from the quantizer. Therefore the final distortion in the pyramid encoder is strictly bounded by the quantization error.

From (5.13) we learned that the difference pyramid $\{DP_i\}$ is computed by subtracting an expanded reconstructed image pyramid from its next lower original image pyramid. Since the data types of both original image pyramid and its reconstructed version are BYTE, the data range of the coefficients in difference pyramid can be from -255 to 255. According to (6.21) we need at least 9 bits for each coefficient, therefore in C language the data type with smallest data range which cover from -255 to 255 is SHORT:

$$\text{SHORT} \equiv \text{signed int\_16} = Z_{[-32768,32767]} \tag{6.24}.$$

As in reconstructed image pyramid, the reconstructed difference pyramid also suffer from 'overflow' and 'underflow' (here the upper and the lower bounds are 255 and -255, respectively) because of the quantization errors. Similar truncation operation is performed on $D\hat{P}_i$:

$$D\hat{P}_i(x) = \begin{cases} 255, & D\hat{P}_i(x) > 255 \\ -255, & D\hat{P}_i(x) < -255 \\ D\hat{P}_i(x), & \text{otherwise} \end{cases} \tag{6.25}.$$

Finally we illustrate the data types and their valid data ranges of each type of pyramid in Fig. 6-8.

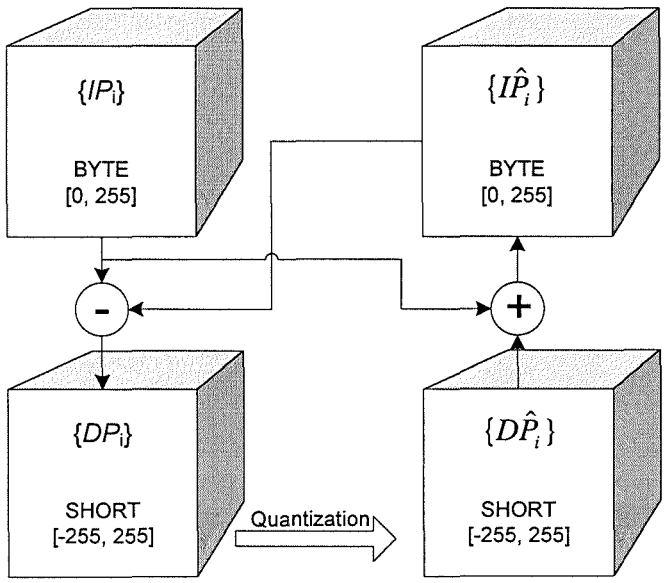
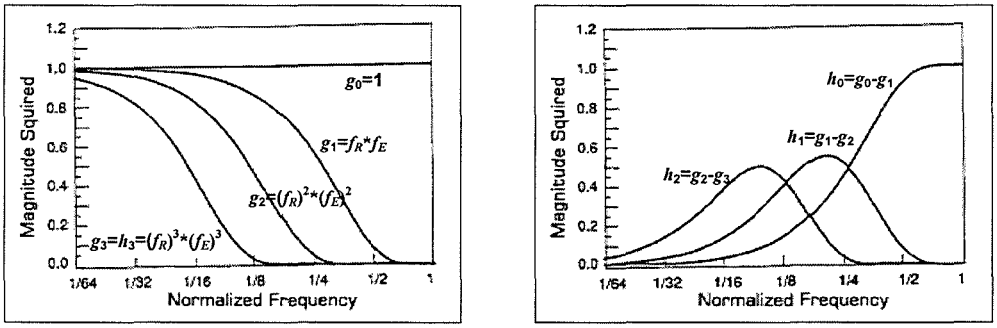


Figure 6-8: The data type and valid data range of each type of pyramid.

6.4 Desired Properties of 3D-IPT for Image Coding

After the pyramid transform using the aforementioned reduction filter and expansion filter, we can obtain an image pyramid and its corresponding difference pyramid. The encoding process is performed on the difference pyramid, because the difference pyramid has much less overlapped frequency range than the image pyramid, and therefore greatly removes the redundancy between two consecutive lowpassed bands. Fig. 6-9 gives an example of filters' frequency responses $\{g_i\}$ and $\{h_i\}$ applied in a four-level image pyramid and its difference pyramid respectively, formulated according to (6.5) and (6.6) using the proposed reduction and expansion filters.



(a) (b)

Figure 6-9: (a) Four equivalent filters $\{g_i\}$ forming a four-level image pyramid. (b) Three equivalent filters $\{h_i\}$ forming three levels of difference pyrmid.

We notice that each image pyramid records the frequency components below the cutoff frequency, which is recursively reduced from the bottom to top, for instance, in Fig. 6-9(a) it is about 1, 1/2, 1/4, and 1/8 from G_0 to G_3. Such representation is very redundant since at each level the lowpassed band contains all the information of bands after it. However the situation in difference pyramid is different. In Fig. 6-9(b) we have seen that the coefficients in difference pyramid contain the frequency components which are ranged only between the cufoff frequencies of current level and the next upper level, say [1/2, 1] as h_0, so resemble features using highpass bandpass filters.

6.4.1 An Example of Decomposition using 3D-IPT

Before we start to describe the desired features of three-dimensional image pyramid transform we give an example of decomposing a volumetric image data using this

transform. The original volume data comprise of 16 consecutive frames, frame 11^{th} to 26^{th}, from B3CGP data set (Section 2.3.1.1). The volumetric data are decomposed into four bands using a four-level image pyramid transform according to (6.5) and (6.6) using the proposed reduction and expansion functions (6.19) and (6.20). The subsampling is performed on each direction therefore the dimensions of the pyramids are: 16×256×256 for level 0 (bottom level), 8×128×128 for level 1, 4×64×64 for level 2 and 2×32×32 for level 3 (top level). The image pyramid and difference pyramid are illustrated in Fig. 6-10 and Fig. 6-11 respectively.

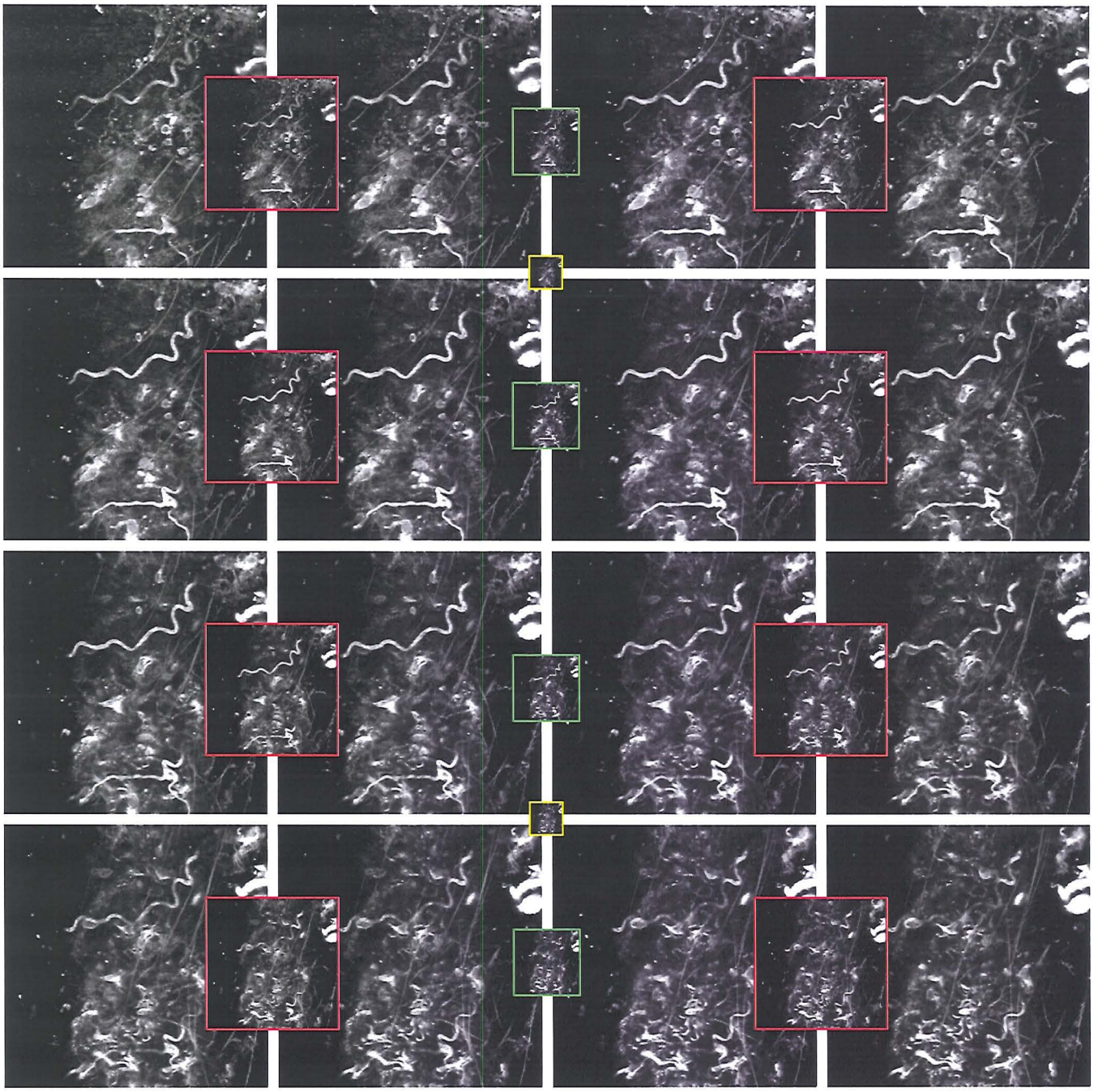


Figure 6-10: An example of four-level 3D-IPT on a 16-frame sequence. The pyramid levels from 0 to 3 are illustrated as images with no border, red border, green border and yellow border, respectively.

The z index, or the frame index, is arranged in ascending order from top to bottom and from left to right. The level 0 pyramids are 16 images with full resolution. The level 1 pyramids are 8 images with half resolution and red border. Recursively the level 2 pyramids are 4 images with quarter resolution and green border. Finally the level 3 pyramids are 2 images with one-eighth resolution and yellow border. On each

pyramid level every image is placed in the middle of two finer-resolution images from the next lower level, for instance, one green image is placed between two red images. Such pattern indicates that in image pyramid each image is the subsampled version of two consecutive images from its next lower level. One should note that although we placed the image as the same pattern as that used illustrating the difference pyramid (Fig. 6-11), it is not sensible to interpret the relations of images between two consecutive levels. Actually as it will be discuss later the correlations between them are not apparent; more specifically they are independent but not orthogonal [Ranganath, 1991].

From Section 6.3.2 we have learned that data ranges of coefficients in difference pyramid, except the top level, are [-255, 255]. To illustrate these values using 8-bits gray scale image they are preceded with a translation and truncation process: the coefficients are added by an offset 128 and then truncated to closest valid values if they are out of range. That is

$$v' = \begin{cases} 255, & v+128 > 255 \\ 0, & v+128 < 0 \\ v+128, & \text{otherwise} \end{cases} \qquad (6.26),$$

where v is the original value of a coefficient and v' is the processed value of v for display purpose. Now the around zero-valued coefficients will be displayed as intensities around mid-gray level and for those coefficients whose values are greater than 127 or lower than -128 will be reassigned as 255 and 0 respectively.

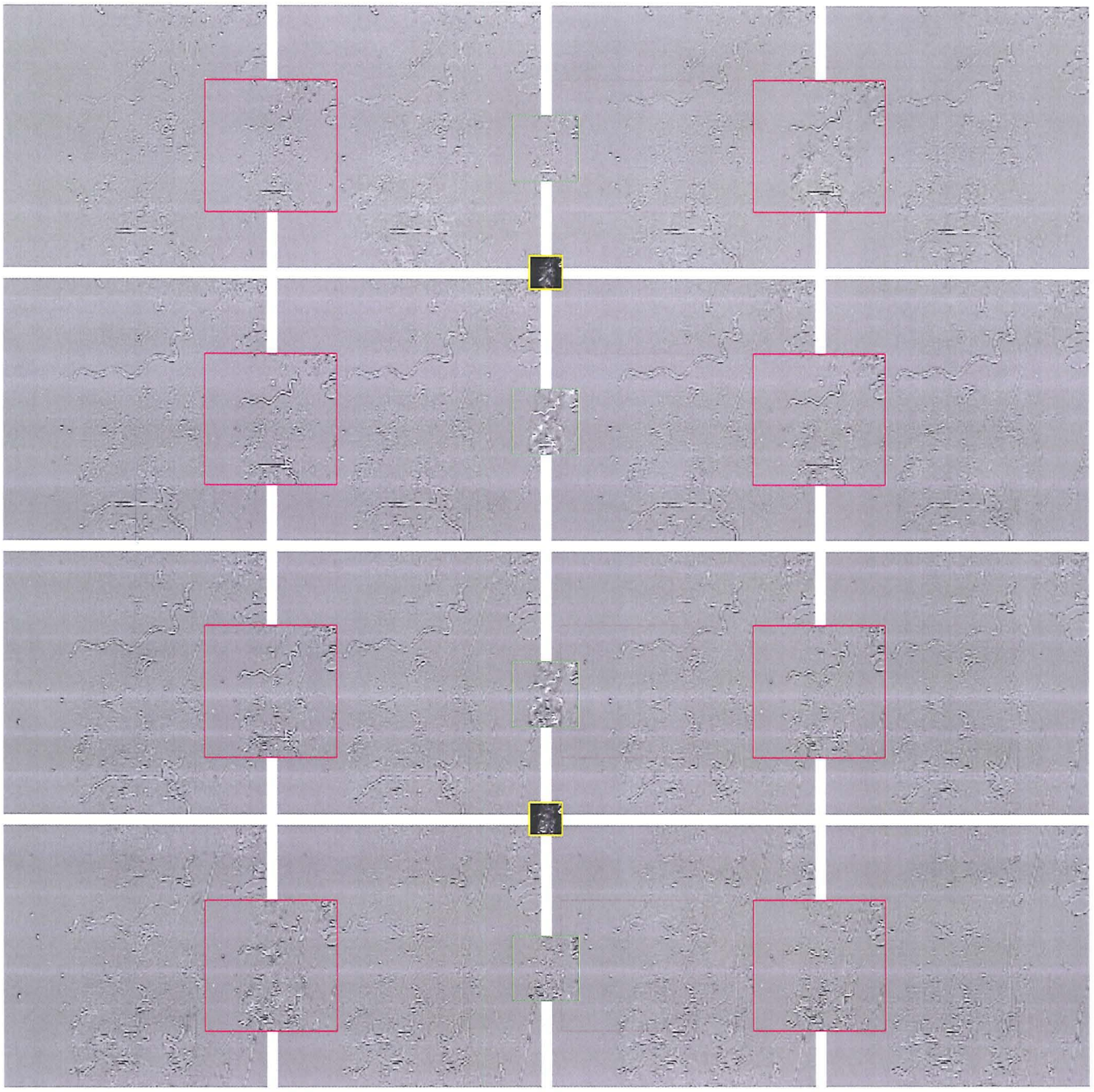


Figure 6-11: The four-level difference pyramid of Fig. 6-10. Same as previous, the pyramid levels from 0 to 3 are illustrated as images with no border, red border, green border and yellow border, respectively.

6.4.2 Characteristics of Difference Pyramid

Because of the subband decomposition properties of image pyramid transform and the non-stationary distribution of image pixels, coefficients in the difference pyramid are correlated. An efficient compression method can exploit these properties by

considering the correlations between coefficients both within each pyramid level, like an intra-band scheme, and across pyramid levels, like an inter-band scheme.

1. Prioritized hierarchical structures

From Fig. 6-9(b) we noticed that coefficients in each difference pyramid level contain the frequency components which are ranged within a specific narrow bandwidth. We have learned that CLSM images are usually characterised by large areas of low-variation regions and structural information, such as edges, usually take a small part of the whole image, therefore a large part of coefficients in difference image pyramid are around zero (mid-gray intensity in Fig. 6-11) and only those areas whose frequency components fall into the range are characterised as large magnitudes. Moreover from Fig. 6-11 we can see that the level of details is increased as we go through the pyramid from top to bottom but the variance is decreased. This is the effect of band filtering and subsampling. According to subband decomposition theory strong (coarser) features exist only in higher pyramid levels, which occupy a much larger areas in the original image than the detail features, therefore generating more large-magnitude coefficients in higher levels than those in lower levels. The subsampling process folds the 'leaked' high-frequency components in stopband to passband and thus repeatedly introducing aliasing into higher pyramid levels. Therefore as to the coefficients from each pyramid level, their variance is increased and the inter-pixel correlations are decreased from the bottom level to top. These facts are illustrated in Fig. 6-12 by studying the histogram of coefficients in each pyramid level.

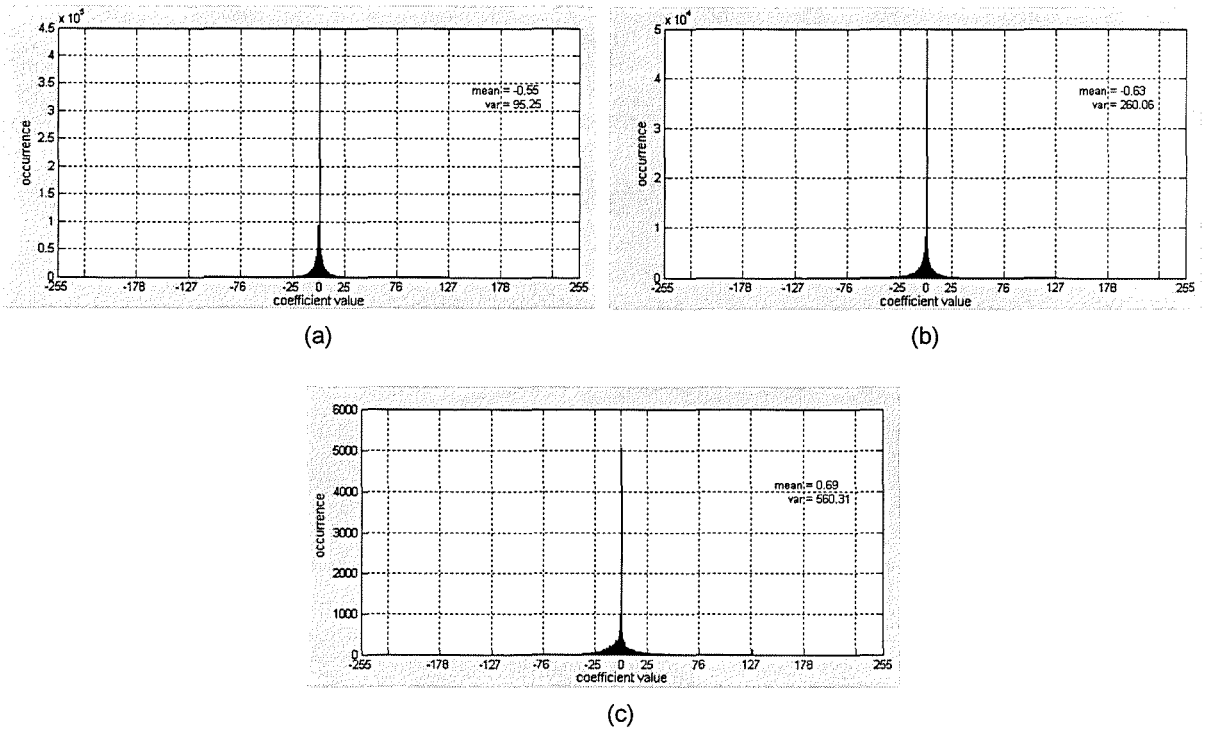


Figure 6-12: The histograms of coefficients in (a) level 0, (b) level 1 and (c) level 2.

We can see that the histogram is a symmetry peak with its means around zero. The variance is depicted as the width of the peak. The coefficients in each pyramid level have different statistical characteristics and they are prioritized in the sense of rate-distortion performance. Therefore a coding approach should consider such difference and for instance, different quantizers should be designed and used for each pyramid level separately.

2. *Non-stationary distribution of intra-band coefficients*

As we have learned in the first section of this chapter, one of the advantages of subband coding is that the local information, such as magnitude and position of edges, can be clearly interpreted by the distribution of its coefficients. Therefore the distribution of coefficients in the difference pyramid is related to that of coefficients in image pyramid. This can be illustrated, for example on the bottom level of pyramids, in Fig. 6-10 and 6-11. In the fifth image (the leftmost in the second row) in image pyramid, we observed a strong curve in the upper-mid part depicting the distribution of a long artery. While in the corresponding image in the difference pyramid we can observe that non-zero-valued coefficients in this region distribute as

the same pattern. So different from sinusoid function based transforms, such as Fourier transform and DCT, where the position information is not obvious, the distribution of coefficients in 3D-IPT is foreseeable. As we will discuss in the following chapter, a classification approach can capture such non-stationary distribution of intra-band coefficients and then improve the coding efficiency.

3. Similar distribution of inter-band coefficients along same orientation

The symmetry property of the reduction and expansion filter guarantee the spatial invariance of the filtered signals. Although the pixels after filtering have changed their values, their positions are unchanged. This makes the filtered images self-aligned with each other. Features existed in one image pyramid level will, if observable, present in its next levels at the same position without introducing any displacement and shape distortion. So in Fig. 6-11, we can observe the similar long-curve pattern in both the 5^{th} and 6^{th} images in the bottom level and the 3^{rd} image in the next upper level as well and they are located at the same position, or orientation, relative to the upper-left corner of the image. Briefly speaking, we can describe this property as if we observed a pattern of distribution of coefficients in pyramid level i, it is very likely we can expect to observe, at the same orientation, the similar pattern in its next lower levels ($<i$). This property is very important in subband coding in that by this means the coding efficiency of the bottom layer(s) can be greatly improved. Actually most of the wavelet-based approaches exploit such inter-band dependencies directly or indirectly. Therefore we are going to consider this property in our method and investigate in the context of three-dimensional image pyramid transform and vector quantization.

Chapter 7 Vector Quantization in 3D Image Pyramid Structures

In previous chapters we described the implementation and the properties of three-dimensional image pyramid transform. We learned that significant coefficients are primarily concentrated in coarser bands while the contributions of coefficients in finer bands are trivial. We also learned that the coefficients in each band is not stationary, but usually distributed in such a way that smooth regions in the original image are transformed into small magnitude coefficients and large variation regions are transformed into large magnitude coefficients; that is coefficients in each band are still correlated, more or less, with each other. Moreover we noticed that structural information, e.g. edges, exist in consecutive bands and are represented as similar patterns across these bands in the same orientation. We therefore attempt to capture these correlations among coefficients within the band (intra-band) and across the band (inter-band) by investigating a hybrid encoding method, cross-band classified vector quantization encoding (CBCVQ), which is based on vector quantization (VQ) techniques.

Before we start describing our hybrid encoding method, we begin with a preliminary section regarding the quantization theories. Then we review several vector quantization techniques, where the discussions are mainly focused on the performance in subband coding of each quantizer. To give a clear impression of the CBCVQ it is

organized in three subsections according to the encoding order. They are low-energy coefficients quantization, cross-band classification and intra-band classified vector quantization.

7.1 Preliminary Knowledge in Quantization Theory

7.1.1 Definitions

Generally, a quantizer can be defined as consisting of a set having finite number, say N, of *partitions* or *cells* $S = \{S_i; i \in Z_{[0,N-1]}\}$ together with a *codebook* C, a set having the same number of *reproduction values* or *levels* or *codewords* $C = \{y_i; i \in Z_{[0,N-1]}\}$, so that the overall quantizer Q is defined by $Q(x) = y_i$ for $x \in S_i$, which can be expressed concisely as

$$Q(x) = \sum_i y_i 1_{s_i(x)} \tag{7.1}$$

where the indicator function $1_{s_i(x)}$ is 1 if $x \in S_i$ and 0 otherwise [Gray, 1998]. Fig. 7-1 gives an example of quantization of one-dimensional and two-dimensional data. For one-dimensional data, the quantizer is usually called *scalar quantizer*, the cell S_i might take the form $S_i = (a_i, a_{i+1}]$, where the a_i is called threshold or boundary of the cell, and the y_i in S_i is the reproduction value of the cell. For k-dimensional data (k>1), like two-dimensional, the quantizer is called *vector quantizer*, and the cell S_i is a region encompassed by lines (in the case of multi-dimensional, it is a hyper-volume encompassed by hyper-planes).

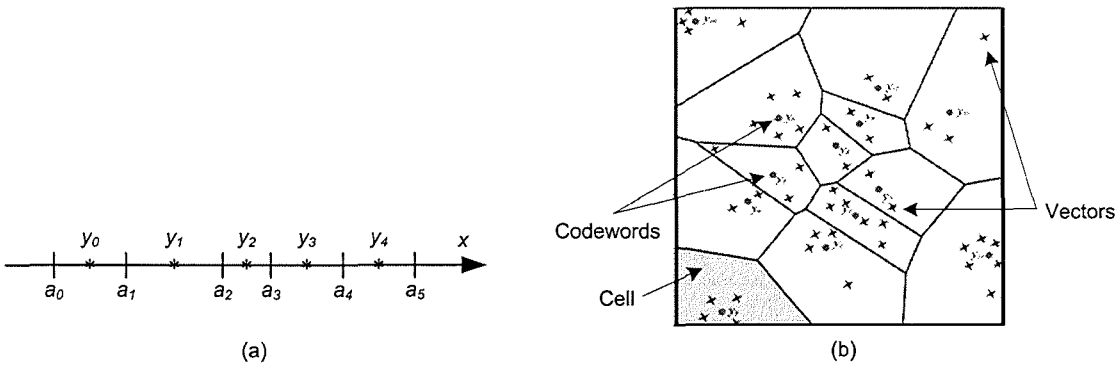


Figure 7-1: Examples of (a) 1-D and (b) 2-D quantization process.

In image compression a typical VQ is depicted in Fig. 7-2, where it firstly involves a *vector forming* process - an input vector x_i is constructed from a group of pixels, usually neighbouring pixels defined by a regular-sized blocks; and then the input vector is quantized by being encoded into an integer-valued binary index i_i, which is also referred as *quantization code* and serves as an index for the y_i in codebook C. The compressed signal is then represented by these indices $\{i\}$ and the compressed representation requires fewer bits than the original. The decoder usually only involves 'look-up' operation once we have specified the codebook.

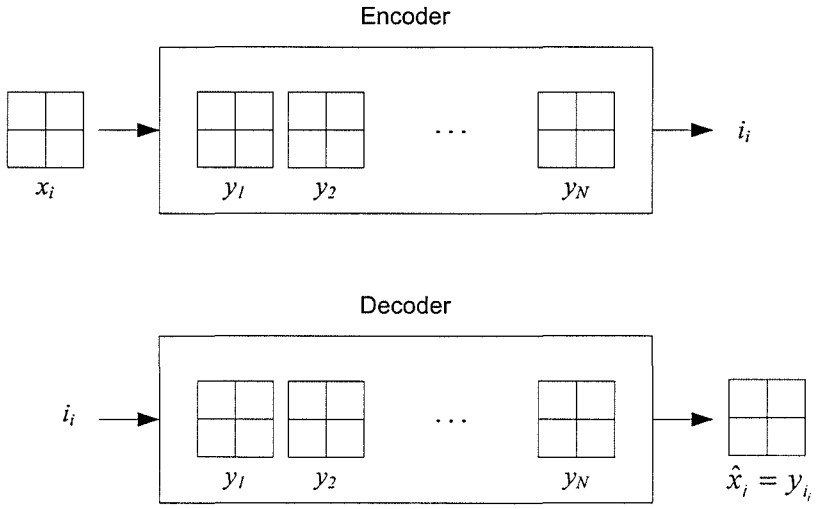


Figure 7-2: A typical vector quantizer.

There are two identifiers for each quantizer: *rate* and *distortion*. Rate, referred as R_Q or simply R, can be defined as in average the number of bits needed to represent each input vector being quantized. If the code has a *fixed rate* of b bits per input vector, then each index i has same length b. With a *variable rate* code, the indices $\{i\}$ have variable length, and b is their average length. The distortion, referred as D_Q or simply D, describes the quality of a quantizer as the goodness of the resulting reproduction in comparison to the original. It can be defined as the average distortion (or weighted averaged distortion) between the input vectors $\{x_j\}$ and their reproductions $\{\hat{x}_j\}$ specified by both the encoded index i and the codebook C.

$$\begin{aligned} D_Q &= D(\{x_j\},\{\hat{x}_j\}) = \frac{1}{n}\sum_{j=0}^{n-1} d(x_j,\hat{x}_j) \\ &= \frac{1}{n}\sum_{j=0}^{n-1} d(x_j,Q(x_j)) = \frac{1}{n}\sum_{j=0}^{n-1} d(x_j, y_i\big|_{Q(x_j)=y_i}) \end{aligned} \tag{7.2}$$

where $d(\cdot)$ can be any discrete and bivariate distortion measures, for instance MSE distortion measure. We, now, can describe the goal of a quantizer as to encode the data from a source, characterized by its distribution characteristics, into as few bits as possible (i.e. with small rate) in such a way that a reproduction may be recovered from the bits with as high quality as possible (i.e. with small distortion) [Gray, 1998].

7.1.2 Vector Quantizers Design

The design of a vector quantizer involves two tasks. One is the design of codebook, which is more frequently referred as *codebook training*. It begins with specifying either a probabilistic model, i.e. a Gaussian source with independent and identically distributed (i.i.d.) samples, or a training set, i.e. a group of typical images. The output codebook is a set of codewords, which is defined as the term $\{y_i\}$ in Eq. (7.1). If the input vector set $\{x\}$ can be well described (modelled) by a probabilistic model we can derive a codebook from the model directly, which can be expected as a good representation for $\{x\}$ and in this case the codebook is usually referred as a *general codebook* or *global codebook*; while otherwise in case that it is inaccurate or difficult to model $\{x\}$ we have to take $\{x\}$ as the training set and design a codebook specifically for $\{x\}$, which is a *specific codebook*. According to the discussion in Section 3.1, the optimum rate-distortion performance can be described by formula 3.1 and 3.2. Since it is impracticable to give the distribution model for a natural image, the optimum rate-distortion performance is never achievable in practice. However the more accurate to model the distribution of the source the closer to the optimum the actual rate-distortion performance is. The specific codebook can adapt to $\{x\}$ with different distributions and therefore outperform general codebook in the rate-distortion sense. However it is with the expense of high complexity in both computation and storage since each codebook has to be separately designed and stored for different $\{x\}$.

The second task is *codeword searching*, which involves the choice of a mapping rule to map an input vector x to a unique entry in the codebook. It is defined as the indicator function $1_{s_i(x)}$ in Eq. (7.1) such that each input vector will be uniquely associated with one partition S_i and use the codeword corresponding to this partition as the representative to the input vector. The output can be the selected codeword or a

binary index i indicating the index of this codeword in the codebook. The latter is preferable to the former in compression applications since such representation requires fewer bits. The searching rule is determined by the structure of the codebook. For instance an exhaustive search is usually applied on codebooks without a particular structure and a binary search can be used on tree-structured codebooks. According to different codebook structures and the corresponding searching methods, a vector quantizer can be classified as *unstructured* quantizer or *structured* quantizer.

7.1.2.1 Unstructured VQ

'Unstructured' means no specific structure of the codewords in the codebook. When encoding a source vector, an exhaustive search of the codebook finds which codewords having the smallest distortion measure to the source vector. It can be expressed as:

$$\text{Exhaustive Search}: \text{index } i \,\Big|\, d(x, y_i) \le d(x, y_j), \forall j \qquad (7.3).$$

Unstructured VQ has high computational complexity when encoding, but it can guarantee the optimum solution, at least locally, in the sense of minimizing the average distortion.

There are many approaches to design an unstructured codebook. The most common approaches are to apply a clustering algorithm such as the Lloyd (k-means) algorithm. The generalized Lloyd algorithm (GLA), as the first attempt on unstructured codebook design, was initially conducted by Stuart Lloyd in an unpublished technical report written in 1957 (eventually published in 1982 [Lloyd, 1982]). He found necessary and sufficient conditions for a fixed-rate quantizer to be locally optimal. The conditions can be simply stated as, the quantizer partition must be optimal for the set of reproduction levels, and the set of reproduction levels must be optimal for the partition. These conditions are the necessary conditions for any optimal quantizer. He applied these conditions in the case of MSE distortion, and found, the first condition implies a minimum distance or nearest neighbour (NN) quantization rule, which chooses the closest reproduction level to the input sample being quantized, and the second condition implies that the reproduction level corresponding to a given partition is the conditional expectation or *centroid* of the samples in their *voronoi* region. The

voronoi region of a particular codeword y_i is defined as the set of all vectors in $\Re^k$ for which y_i is the nearest vector. In the case of using squared distortion measure it can be expressed as,

$$V_i = \{x \mid \| y_i - x \|^2 < \| y_j - x \|^2, i \neq j\} \quad (7.4).$$

Accordingly he developed an iterative codebook training algorithm, usually referred as generalized Lloyd algorithm (GLA), for a given source whose distribution is defined by a training set. In this algorithm it iteratively improves a codebook by alternately optimizing the partitions or voronoi regions for each codeword and the codeword for each partition.

Algorithm 7.1: Generalized Lloyd Algorithm (GLA)

Step 1. Beginning with an initial collection of reproduction levels: let $C_0 = \{y_0^{(0)}, y_1^{(0)}, \ldots, y_{N-1}^{(0)}\}$ as the initial codebook, where N is the number of level desired. Set iteration index $m = 0$.

Step 2. Optimizing the partition for these levels by using a minimum distortion mapping: compute the voronoi region V_i of each codeword y_i according to (7.4). Compute average distortion D_m using (7.2).

Step 3. Optimizing the codeword $y_i^{(m)}$ associated with each partition V_i by replacing the old codeword with the centroids of current partition: Update C_m to C_{m+1}, as $C_{m+1} = \{y_i^{(m)} \mid y_i^{(m)} = centroid\ of\ V_i^{(m)}\}$. Compute average distortion D_{m+1} using (7.2).

Step 4. If $\frac{D_m - D_{m+1}}{D_m} < \varepsilon$, where ε is a predefined threshold, we then terminate the iteration with the final codebook C_m; otherwise, set $m=m+1$, go to Step 2.

The GLA is a decent algorithm when run on either a probabilistic model or on a training set and hence it can always be used to improve a codebook in terms of reducing distortion. However the results of GLA are sensitive to the choice of initial codebook. Starting from different initial codebooks may have the final codebook converge to different extremes and therefore the GLA is locally optimal. People have proposed many algorithms regarding the choice of the initial codebook. For example an initial codebook can be a vector set randomly selected from the training set, or an

output codebook computed from another vector quantizer. Among them LBG-VQ [Linde, 1980] is the most widely used alternative to the GLA.

In LBG a "splitting" method was developed to solve the initial codebook selection problem, wherein one first designs a codebook with two codewords that are locally optimised using GLA; then double the codebook size by adding a new codeword near each existing codeword, then runs GLA again, and so on and the process is repeated until the desired number of codewords is obtained. Referred to Fig. 7-3 the algorithm is summarized below.

Algorithm 7.2: LBG algorithm (LBG-VQ)

Step 1. Initialization: Given a set of source vectors $T = \{x_0, x_1, \ldots x_{n-1}\}$, where n is the number of source vectors. Fixed $\omega > 0$ to be a "small" number.

Step 2. Obtaining initial codebook having only one codeword: Let $N = 1$, N is the number of codeword obtained. Set iteration index $i = 0$. Set initial codebook

$$C^{(0)} = \{y_0^{(0)} \mid y_0^{(0)} = \frac{1}{n}\sum_{m=0}^{n-1} x_m\}$$ as current codebook.

Step 3. Splitting: For $j = 0, 1, \ldots, i$, updating current codebook by splitting each codeword into two new codewords.

$$C^{(i+1)} = \{y_0^{(i+1)}, y_1^{(i+1)}, \ldots, y_{2i+1}^{(i+1)} \mid \begin{matrix} y_{2j}^{(i+1)} = (1+\omega) y_j^{(i)} \\ y_{2j+1}^{(i+1)} = (1-\omega) y_j^{(i)} \end{matrix}, \quad j = 0, \ldots, i\}$$

Set $N = 2N$ and $i = i+1$.

Step 4. Optimizing current codebook $C^{(i)}$ using GLA.

Step 5. Iteration: If N equals the number of codeword desired, we terminate the iteration; otherwise go back to Step 3.

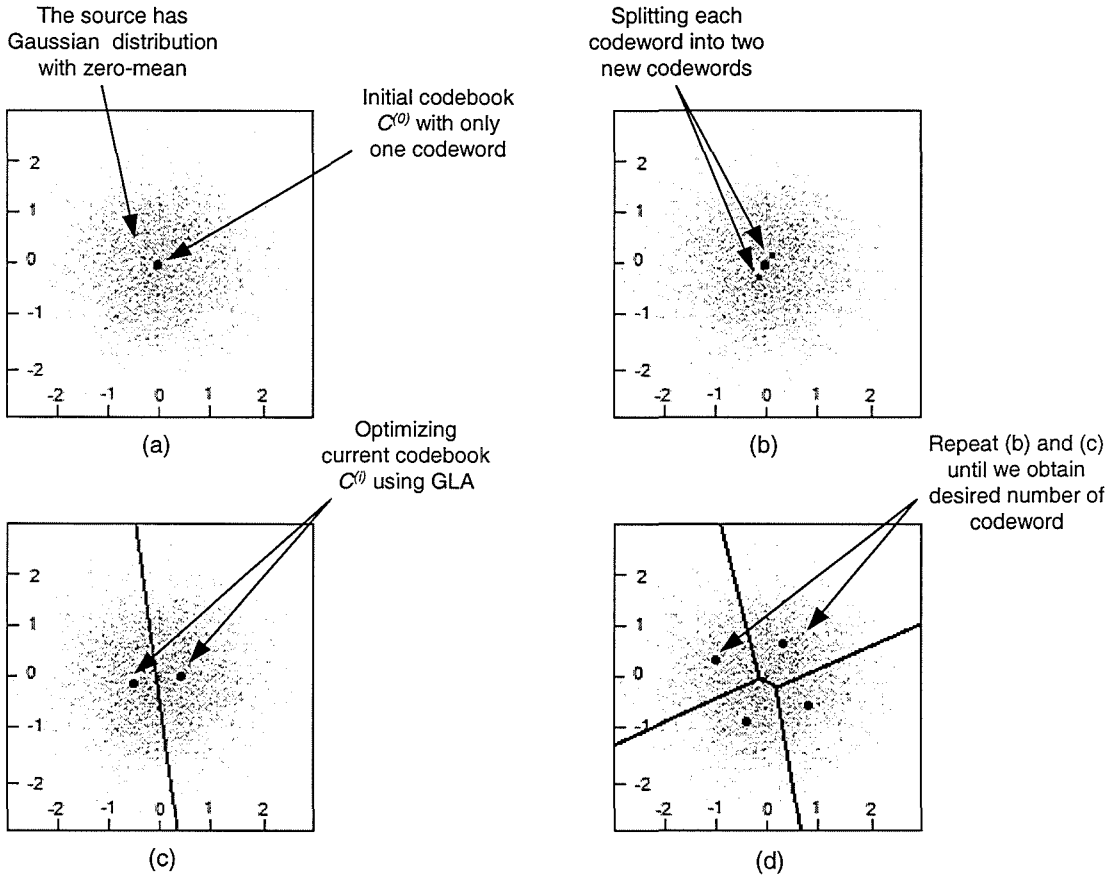


Figure 7-3: An example of training a four-entry codebook using LBG on zero-mean Gaussian-distributed samples.

Although LBG is still a locally optimum solution and depends on the way we split a codeword the sensitivity of initial choice is less than that in GLA since the bias of initial selection is restricted by the results from previous iteration and it can be rectified by performing GLA in the current iteration. In Section 8.2.1 we will investigate the initial codebook selection problem. Moreover we have proposed a novel initial codebook selection scheme, based on the distribution of both the previous codebook and the training set. It further reduces the bias in the initiative selection and therefore can produce a higher quality codebook than the conventional LBG-VQ [Linde, 1980].

Unstructured quantizers produce optimum solutions for fixed-rate quantization. In images signal samples are non-stationarily distributed and correlated, the quantization codes will resemble such distributions and correlations exist, especially at high rates. Therefore one can theoretically improve the coding efficiency and capture the

dependencies between quantization codes by using zero order entropy coding to exploit their non-stationary distributions and higher order entropy coders for inter-codes correlations. In practice, however, the inter-codes correlations are not easily modelled, especially for quantizing large-dimensional vectors at low and moderate rates. Alternatively people have proposed many VQ systems which incorporate fixed-rate quantizers with other techniques, such as transform coding and predictive coding or in constraint structures. The purpose of these methods is to exploit the dependencies between vectors and their corresponding codewords in a more straightforward form, which on one hand can improve the coding efficiencies over the ordinary VQ and on the other hand in some cases can reduce the complexity in training and encoding. This leads to structured vector quantizers.

7.1.2.2 Structured VQ

In structured quantizers either the partitions or the codebooks have been organized in a structural form, which lead themselves to fast searching techniques and, in some cases, greatly reduce the storage complexity of codebooks. It is worth noting that structured vector quantizers, in general, are not optimum. Nevertheless, in practice it is often of interest to optimise over a constrained subset of possible codewords rather than over all of them. Unstructured quantizer may prove expensive in computation, and added structure may provide gains in practical simplicity that more than compensate for loss of optimality [Gray, 1998], [Gersho, 1992]. The structured quantizers have been intensively investigated. There is no general rule to tell which technique is better than others, but they are application dependent. In general the achievements of using structured VQ can be categorized in three aspects.

Searching Complexity: For example Tree-structured VQ (TSVQ) [Riskin, 1991], Multi-stage VQ (MSVQ) [Juang, 1982; Ho, 1988] and Hierarchical VQ (HVQ) [Chang, 1985]. In these approaches the codebook is highly structured, for example it is embedded in a tree in TSVQ and lookup tables in HVQ. The searching rule is defined by the structures. Compared with full-search unstructured VQ its complexity is usually reduced, for instance the complexity is linearly increased in TSVQ and constantly in HVQ with respect to the increased bit rates, which are in contrast with exponential increment in the unstructured VQ. However since the codewords are not

optimal to the training set and the searching cannot ensure finding the nearest codeword in the codebook, they suffer the loss of quality in encoding and also the additional storage space for the codebook. They may be preferable in real-time applications, like real-time video coding [Cockshott, 1999], but in medical imaging applications most of the compression tasks are off-line, therefore encoding speed is not a critical concern. Moreover since in our method some fast search algorithms have been used in unstructured VQ and also thanks to the great improvement of processing capacity of current typical PC, the encoding speed may not be a crucial drawback of using unstructured VQ. For instance, in the following discussion we can see that encoding an image volume with size 256×256×144 takes roughly 30 seconds using unstructured VQ while roughly 5 seconds using structured VQ but with the penalty of 20% increment in average distortion.

Inter-vector Correlation: For example Finite-state VQ (FSVQ) [Aravind, 1986; Kim, 1988], Trellis Coded VQ (TCVQ) [Ungerboeck, 1987] and Predictive VQ (PVQ) [Hang, 1985]. From the Shannon coding theory we are aware that the larger the vector sizes the more the inter-symbol correlations we can capture and so the higher the performance of vector quantization. However in practice, it is difficult to quantize large-sized vectors. Therefore the purpose of these approaches is trying to capture the inter-vector correlations while avoid the increment of vector size. For instance similar to the traditional DPCM, in PVQ a residual vector is firstly computed from the difference between the actual vector and its prediction, which is usually made from the previous outputs, and then the residual vector is quantized. In FSVQ and TCVQ there are multiple codebooks and a more complicated 'prediction' is inferred using the *next-state* rule to tell the encoder and decoder the appropriate sub codebook for the next vector. The next-state rule can be any information available to both the encoder and the decoder, for instance, if the decoding process is included in the encoder, both the encoder and the decoder can have the same previously-reconstructed vectors from nearby locations.

A variety of state functions have been investigated [Chang, 1993; Chang, 1996b; Wei, 2000]. However most of these works are in original image domain; while in the transformed domain, such as subband-decomposed signals, the improvement of

FSVQ and PVQ is not as significant as that in image domain [Chang, 1996b; Tao, 2004]. This is because transform coding has already decorrelated the neighbouring vectors and the inter-vector correlations are not as strong as those in image domain. Moreover extra computations are introduced in the decoder due to the state function. This may weaken the advantages of fast decoding of VQ. Therefore these approaches, such as FSVQ, are not good candidates of VQ in our image pyramid coding.

Non-stationary Distribution: For example Classified VQ (CVQ) [Ramamurthi, 1986] and Multi-stage VQ (MSVQ) [Juang, 1982; Ho, 1988]. The improvement of using these approaches is to design a variable-rate quantizer, which can be adapted to the non-stationary distribution of a source. At same distortion level, it outperforms fixed-rate unstructured VQ, which have to be designed at higher rate and introduce more complexity.

The encoding and decoding procedure of an M-class CVQ are illustrated in Fig. 7-4. In this method the input vector x is subjected to a classifier Ω which generates an index i_c, an integer having M discrete values, say 0~M-1, which identifies which sub quantizer Q_m to encode x. The output corresponding to x thus consists of two parts: the index i_c specifying the sub quantizer and the index i_{Q_m} specifying the selected codeword from the sub quantizer Q_m. The first part or i_c is typically referred to as side information whose code length is computed as,

$$r(i_c) = \lceil \log_2 M \rceil \tag{7.5}$$

in the case of fixed length. The decoder reverses the encoding process as firstly it uses i_c to identify which quantizer will be used through Ω^{-1} then retrieves the corresponding codeword from its codebook according to index i_{Q_m}. Therefore the decoder has to maintain necessary information of each sub quantizer to guarantee the correct decoding, e.g. the information of M codebooks.

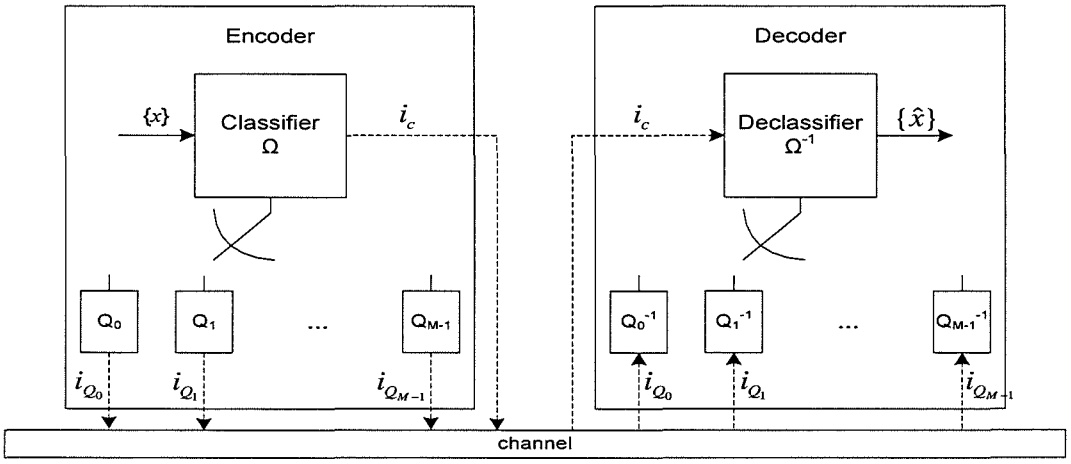


Figure 7-4: A simply block diagram of classified vector quantizer.

The benefit of CVQ can be described in the following example. Suppose we have two sources s_0 and s_1 with infinite samples. These two sources have identical distributions, for instance uniform distribution, but different variations, say δ_0^2 and δ_1^2. Now we generate a new source s having n finite samples from s_0 and s_1, where each sample is selected from s_0 or s_1 and finally we have n_0 samples from s_0 and n_1 samples from s_1. Then we use same quantizer to quantize s, s_0 and s_1 at the same distortion. From Section 3.1 we have learned that the rate-distortion function for these three sources would be:

$$\begin{cases} s_0: & r_0 = \delta_0(d) \propto \frac{1}{2}\log(\frac{M \cdot \beta \cdot \delta_0^2}{d}) \\ s_1: & r_1 = \delta_1(d) \propto \frac{1}{2}\log(\frac{M \cdot \beta \cdot \delta_1^2}{d}) \\ s: & r = \delta(d) \propto \frac{1}{2}\log(\frac{M \cdot \beta \cdot \delta^2}{d}) \end{cases} \quad (7.6),$$

respectively. Here we define the classification gain g_c as

$$g_c = \frac{n_0 \cdot r_0 + n_1 \cdot r_1}{(n_0 + n_1) \cdot r} \quad (7.7),$$

which is the ratio of the rate using classification, where each sample is encoded by $\delta_0(d)$ or $\delta_1(d)$ according to which source it is from, to the rate without using classification, where each sample is encoded by $\delta(d)$. Therefore the smaller the magnitude of g_c, the higher the compression ratio can be achieved using classified

encoding, and the more the significance of the classification gain. Substitute (7.6) into (7.7) the classification gain would be

$$g_c = \frac{(\delta_0^2)^{\frac{n_0}{n_0+n_1}} \cdot (\delta_1^2)^{\frac{n_1}{n_0+n_1}}}{\frac{n_0 \cdot \delta_0^2 + n_1 \cdot \delta_1^2}{n_0 + n_1}} \qquad (7.8).$$

Here we can see that the classification gain is actually the ratio of the geometric mean of the variance of each source to their arithmetic mean, which is determined by δ_0^2/δ_1^2 and n_0/n_1. In Fig. 7-5 we illustrate the value of g_c with respect to these two factors. From this figure we can see that two extreme points symmetrically locate at two corners, which means significant classification gain can be achieved using classified coding when the large-variation source has small number of samples and the small-variation source has large number of samples.

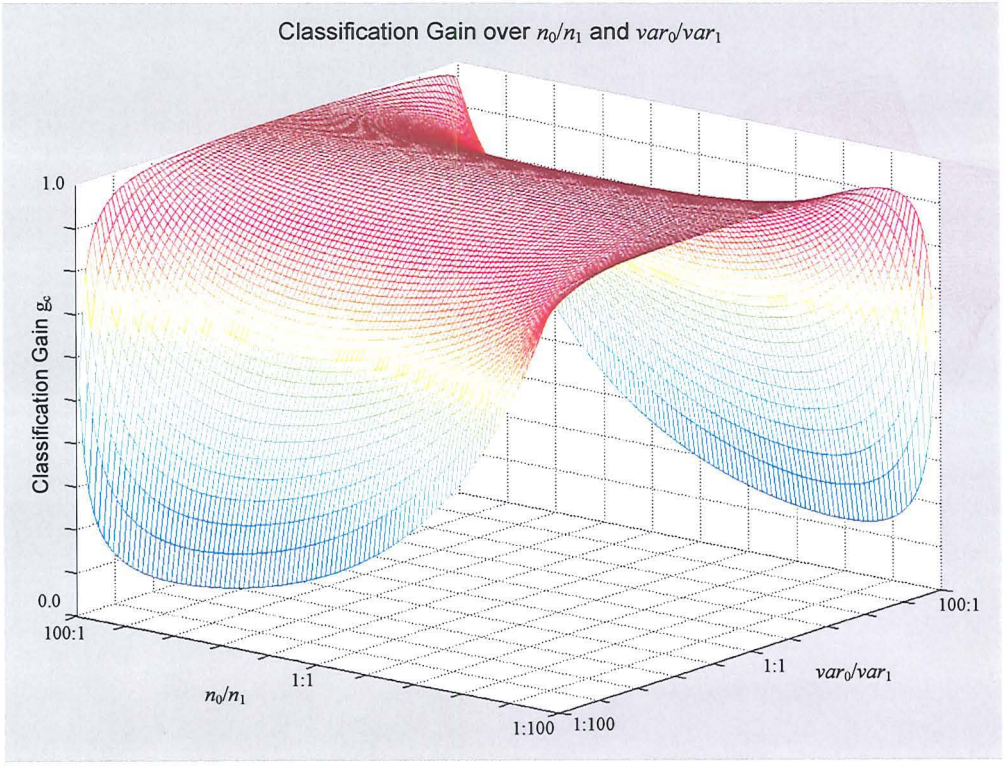


Figure 7-5: The value of classification gain varying with respect to δ_0^2/δ_1^2 and n_0/n_1.

In the image pyramid subband decomposition (see Section 6.4) the situation resembles the above example. The coefficients from a particular level can be modelled as a zero-mean Laplacian distribution and when observed locally the

coefficients also satisfy the same distribution but the variance may be different. For instance coefficients in structural regions, such as edges, have large variations while for those in the smoothed regions the variances are very small. Moreover after pyramid decomposition most of the coefficients have small variations and large-variation coefficients occupy only a small part of the image. Therefore classified VQ was considered in our method due to its advantages in coding performance in subband coding [Joshi, 1997; Yao, 1992].

7.2 Cross-band Classified VQ Encoding

7.2.1 Low-energy Coefficients Quantization

In Section 6.4.2 we noticed that smooth regions on the original image will be transformed into low-energy coefficients after pyramid decomposition, which means their magnitude is close to zero. These coefficients have much less contribution to the final image quality compared to those high-energy coefficients and have less priority. It is possible to use VQ techniques to quantize these coefficients but due to the large volume of data the expense of high computational complexity of VQ is more important than its improvement in rate-distortion performance. Therefore instead of using VQ encoding, we developed a quad-tree-like encoding approach for the low-energy coefficients which we named as *block-based splitting thresholding*.

The purpose of thresholding is to discard a large part of insignificant (low-energy) coefficients and isolate smaller entities, containing the significant (high-energy) information. The thresholding element is a block consisting of a group of neighbouring pixels rather than individual pixel. This is based on the observation that in smoothed regions neighbouring pixels tend to have similar intensities and after transformation the corresponding coefficients at the same location also tend to have similar magnitudes. Therefore a block is used to capture such similarity among these coefficients. In practice a block will be regarded as significant if it has one or more coefficients whose magnitudes exceed a threshold T. The significant block will be split into eight sub-blocks and the same significant test will be applied on each sub-block until the sub-block's size reaches the predefined smallest sub-block's size. Here we refer to these smallest sub-cubes as the leaf nodes. Such a block-based method can

capture and adapt to the irregular-shaped objects from the insignificant 'background'. In Fig. 7-6, for example, we begin with a large-size block with the initial size as 16×16×16 and it has been recognized as significant block (dark gray) since it has one or more coefficients whose magnitudes exceed the threshold. Then we split this block into eight sub-blocks and to identify in which sub-block the significant coefficients exist; the splitting and testing process is recursively performed on each sub-block until the sub-block's size reaches the size of leaf node, say 4×4×4. If we use binary '1' for significant and '0' for insignificant this 16×16×16 block will finally be encoded as an oct-tree structure by 17 bits, which would be '11000000000100000'. Therefore only the coefficients in the significant leaf nodes will be considered in the following vector quantization and for insignificant coefficients they are quantized to zero value, which is the mean of the distribution of the difference pyramid coefficients.

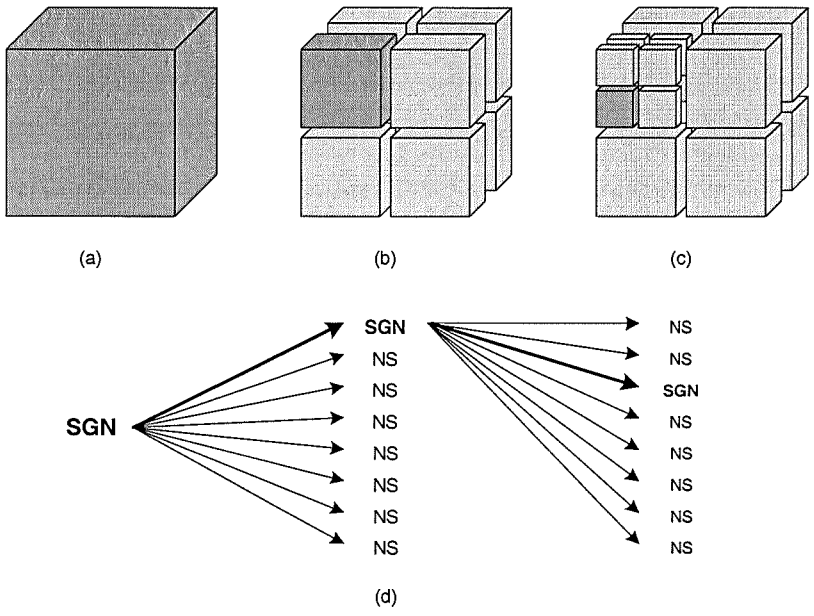


Figure 7-6: When a significant pyramid coefficient is encountered, the cube (a) is spited into eight sub-blocks (b), and further on (c) up to the leaf node level (smallest sub-cube). The result is an oct-tree structure (d) (SGN = significant node; NS = non-significant node).

Using block-based splitting thresholding to encode low-energy coefficients has three advantages. Firstly it simply discards a major part of the coefficients, which have been recognized as low priority. Although its rate-distortion performance may not be as good as other more complex approaches the difference is trivial. Therefore when we consider both the computational complexity and rate-distortion performance the

block-based splitting thresholding may be preferable to more complicated methods, such as VQ.

Secondly in the original image the noise in the smoothed regions will be transformed into low-energy coefficients after the image pyramid transform. Therefore thresholding the low-energy coefficients will inherently reduce the noise in the original image. In the following example (Fig. 7-7) a region of size 120×65 containing two nuclei (the top-left image) is used to demonstrate the effect of denoising of thresholding, which is clipped from 32^{th} frame of G27HG70 data set (see Section 2.3.1.3). The data set is firstly pyramid decomposed (the top-right image) and then on the bottom level we discard the lowest-energy coefficient using the block-based splitting threshold, where the threshold is 17; the initial block size and the leaf node size are 16×16×16 and 4×4×4, respectively (the bottom-right image). Finally we reconstruct the original image using the thresholded difference pyramid (the bottom-left image). In this example we can see that noise exist within the smoothed regions of the nuclei. Noise is then transformed into low-energy coefficients on the bottom level and the real signal is compacted in the higher pyramid levels. After thresholding low-energy coefficients are quantized to zero values and noise both from the image of nuclei and background has been removed. Therefore when doing pyramid composition the thresholded noise will not be added to the real signal, which is synthesized from the higher pyramid levels, and finally we will have a noise-reduced version of the original image.

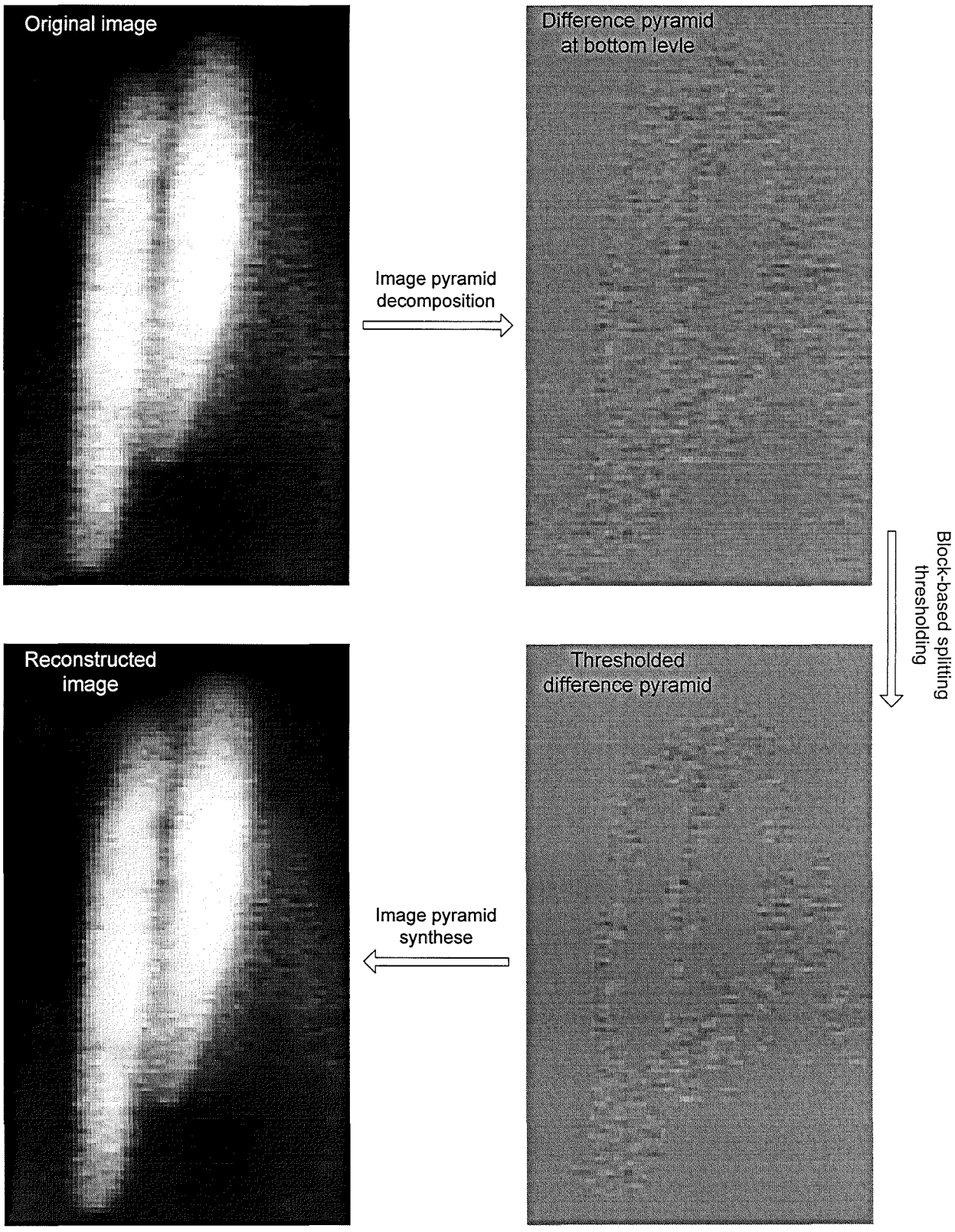


Figure 7-7: An example of denoising using block-based splitting thresholding.

Additionally after thresholding the coefficients are classified into two groups: one group consists of insignificant coefficients characterized as low-variance and another group consists of significant coefficients whose variance is much larger than the former. Separately encoding the two groups results in an effect of classified quantization as aforementioned. Therefore the coding efficiencies of both the significant group and the overall pyramid can be improved from such classification gain.

7.2.2 Cross-band Classified VQ

According to Section 6.4.2, there are two types of dependencies which could be considered and exploited by the encoder to improve the coding efficiency. One is the non-stationary distribution of intra-band coefficients; and the other is the similar distribution of inter-band coefficients crossing bands along the same directions. A cross-band classified vector quantizer (CBCVQ) has been proposed to encode the unthresholded significant blocks. The intra-band coding is based on classification according to the statistical properties of samples in each level, where an intra-band vector is firstly associated with a sub-VQ by the corresponding classification map, and then this sub-VQ is used to encode the vector based on a nearest-neighbor mapping rule. The inter-band coding is achieved by building the current-level classification map based on the encoding information from previous levels, and therefore the side information for classification map do not need to be explicitly transmitted to the decode side, but can be derived when the decoding process of the previous levels is finished.

In Fig. 7-8 we can see that the cross-band encoding can be basically divided into two parts separated by the dashed line. The first part (above the dashed line) includes the encoding of vectors in level N-2, the first high-pass band in the difference pyramid. Its intermediate outputs will be used to derive the classification maps for encoding the following lower levels. In the second part (below the dashed line) each level is encoded using a typical classified VQ, where each of them share a same classification map obtained in the first part. Therefore the cross-band gain is achieved in such a way that the overhead bits of classification information of vectors from these levels are not

explicitly transmitted to the decoder, but implicitly derived from the intermediate outputs in part one.

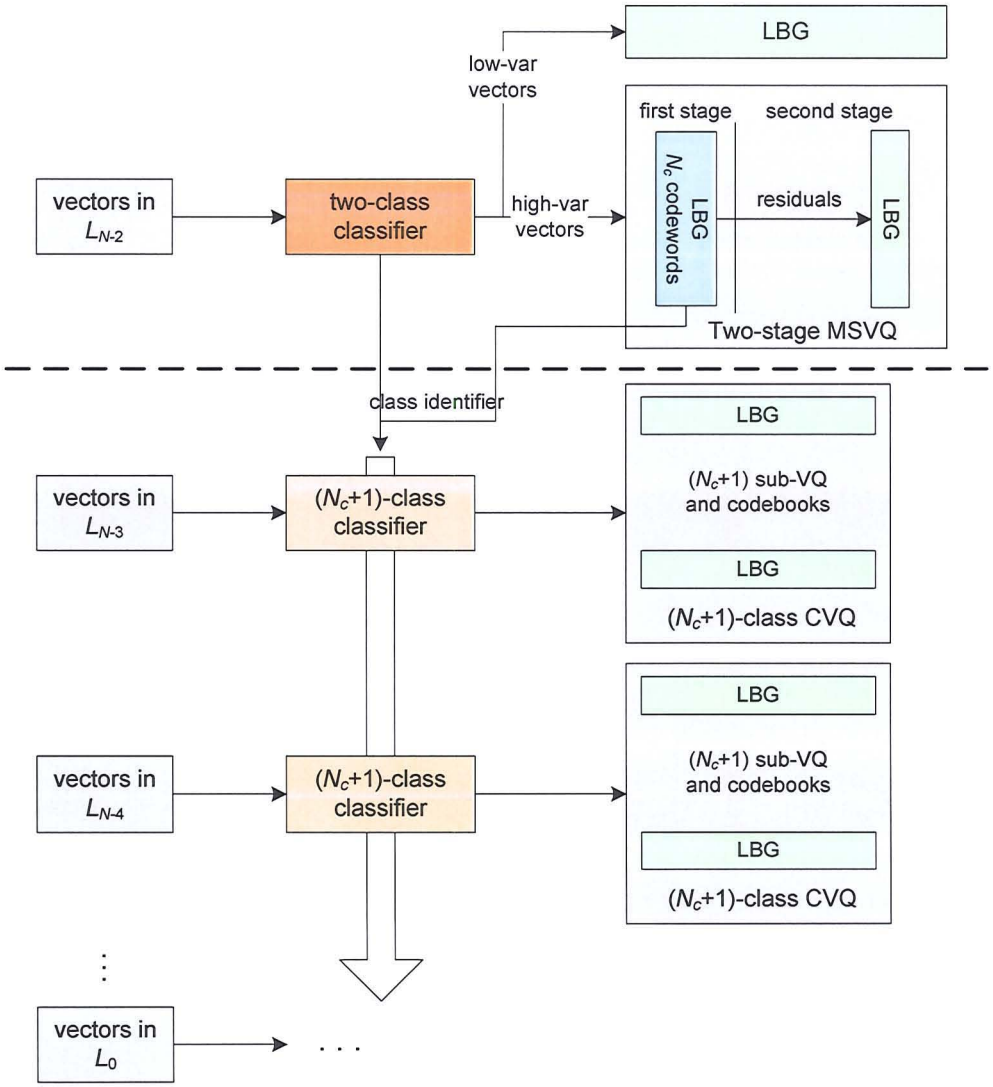


Figure 7-8: The block diagram of the cross-band classified VQ encoding.

In CBCVQ classification information are primarily generated from a VQ-based classifier (the light blue block in the first part). In this part the vector quantizer acts as, not only an encoder for the coefficients of level N-2 but also a classifier for these coefficients. After encoding process each vector in L_{N-2} will be uniquely associated to one class identifier according to the quantization rule. Actually using vector quantization as classification is natural because both techniques can be designed and implemented using methods from statistical clustering and classification rule [Cover, 1967; Cosman, 1993a]. By combining VQ with classification, the classification requires no more bits to describe than those required for compression alone.

Two issues have been considered when we use the VQ-based classification. The first one is that people [Cosman, 1993a] have found VQ has poor classification performance for low-variation vectors, which takes a larger part than the high-variation vectors in difference pyramid. Therefore a two-class classifier was preceded before the VQ-based classifier (the orange block in the first part). A vector is firstly compared with a preset threshold T_v to determine whether it is a low-variation vector or a high-variation vector. For the low-variation group it will be simply LBG coded and for high-variation group we use VQ-based classification encoding scheme.

Moreover we found that in the VQ-based classified encoding the size of the codebook (N_C), which is used to guide classification, has an impact on the coding efficiency. In previous work [Chang, 1996a] their codebook size is large, for example 256 codewords. In our experiment a large value of N_C will sometimes decrease the classification efficiency since the vectors might be 'over-classified' which lead to too few vectors in a class, for instance only one or two vectors. Therefore we introduced a two-stage multi-stage vector quantizer. The first stage uses a medium-sized codebook, for example N_C=16 codewords, to generate the classification map and pre-encoding a source vector. Then the encoding error, or the residuals, of this vector will be encoded at the second stage. With MSVQ the vector can be reasonably classified and also be high-quality encoded. So finally consider the preceded two-class classifier, for example, if we use a codebook of 16 in the first stage, we will altogether have 17 classes.

According to the third property of pyramid decomposition (refer to Section 6.4.2), the similar distribution of cross-band coefficients along same orientation, if a block of coefficients has large variation on the higher pyramid level it is very likely the variations of coefficients in the lower levels at the same location are large as well. Therefore the cross-band classification can be achieved in the following way. After the encoding on level N-2, each vector in this level will be associated with a class identifier, either from the two-class classifier or the output indices from the LBG-VQ in the first stage in MSVQ, which would be used for the following levels. If we assume that a vector block with size $v_x \times v_y \times v_z$ at the location (x, y, z) in level N-2 is associated with class c_i, then when we apply the cross-band classification rule on one

of its lower levels, say level i ($i<N$-2), the block on level i, with size $\tau v_x \times \tau v_y \times \tau v_z$ ($\tau=2^{N-2-i}$, is the scaling factor between level i and N-2) at the location (τx, τy, τz) corresponding to the same orientation as that in level N-2, will also be associated with class c_i. An example of cross-band classification rule on level N-3 is illustrated in Fig. 7-9.

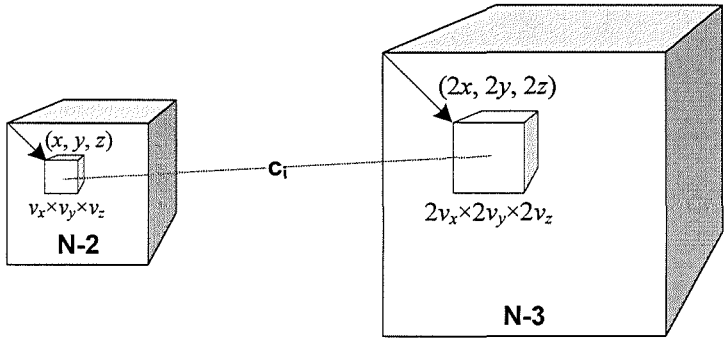


Figure 7-9: The cross-band classification rule applied on level N-3 which is the next lower level to N-2.

Fig. 7-10 – Fig. 7-11 give an example of encoding on two consecutive pyramid levels using the cross-band classified vector quantization technique. The image data are from B3CGP data set (Section 2.3.1.1) with image dimensions as 256×256×144. We build a three-level pyramid, where l_0 is the bottom level and l_2 is the top level consisting of the lowest-frequency coefficients. The parameters for thresholding and CBCVQ have been listed in Table 7-1. According to the structure of CBCVQ the encoding starts from the smallest-scaled high-frequency band, which is l_1 in this example. Although CBCVQ is applied on the entire volume, we only focus on a small part, by which we can describe how the encoding process works in three dimensions. Considering the size of each class block in l_0 is 6 in z direction, we only illustrate the results of 6 consecutive frames, frame 13 to frame 18, in l_0 and three consecutive frames, frame 7 to frame 9, in l_1. These images are also illustrated in Fig. 6-10 and 6-11 in Section 6.4.1.

Table 7-1: Specifications of the thresholding and CBCVQ in a three-level pyramid.

Level	Thresholding		Classification		Vector size
	Thr. value	Block size initial size (IS) node size (NS)	# of class	Block size	
2	T_2=0	IS: 4×4×6 NS: 2×2×3	N/A	N/A	N/A
1	T_1=5	IS: 4×4×6 NS: 2×2×3	5 (N_c=4) (T_v=60)	2×2×3	2×2×3
0	T_0=13	IS: 8×8×12 NS: 2×2×3	5	4×4×6	2×2×3

In Fig. 7-10 coefficients in l_1 are firstly thresholded using the block-based splitting thresholding method, where the thresholded coefficients are marked with blue color. Each 2×2×3 block is compared with the threshold T_1=5: if the magnitudes of all the coefficients in the block do not exceed T_1 these coefficients will be discarded and decoded as zero values; otherwise they will be recognized as significant coefficients and further encoded using CBCVQ. After thresholding vectors from l_1 will be classified into two groups, the low-variation vectors and high-variation vectors, by comparing the variance of each vector with a predetermined value T_v. High-variation vectors are those whose variance is non less than T_v (marked with green color); and low-variation vectors are those whose variance is smaller than T_v (marded with brown color). Thereafter an LBGVQ with four entries is applied on the high-variation vectors. After quantization each vector is uniquely associated with one index, say from 1 to 4 and now we have generated a classification map with five classes for l_1. This classification map will be used to guide the classification on the following level(s) according to the same-orientation classification map propagation scheme as described in Fig. 7-9.

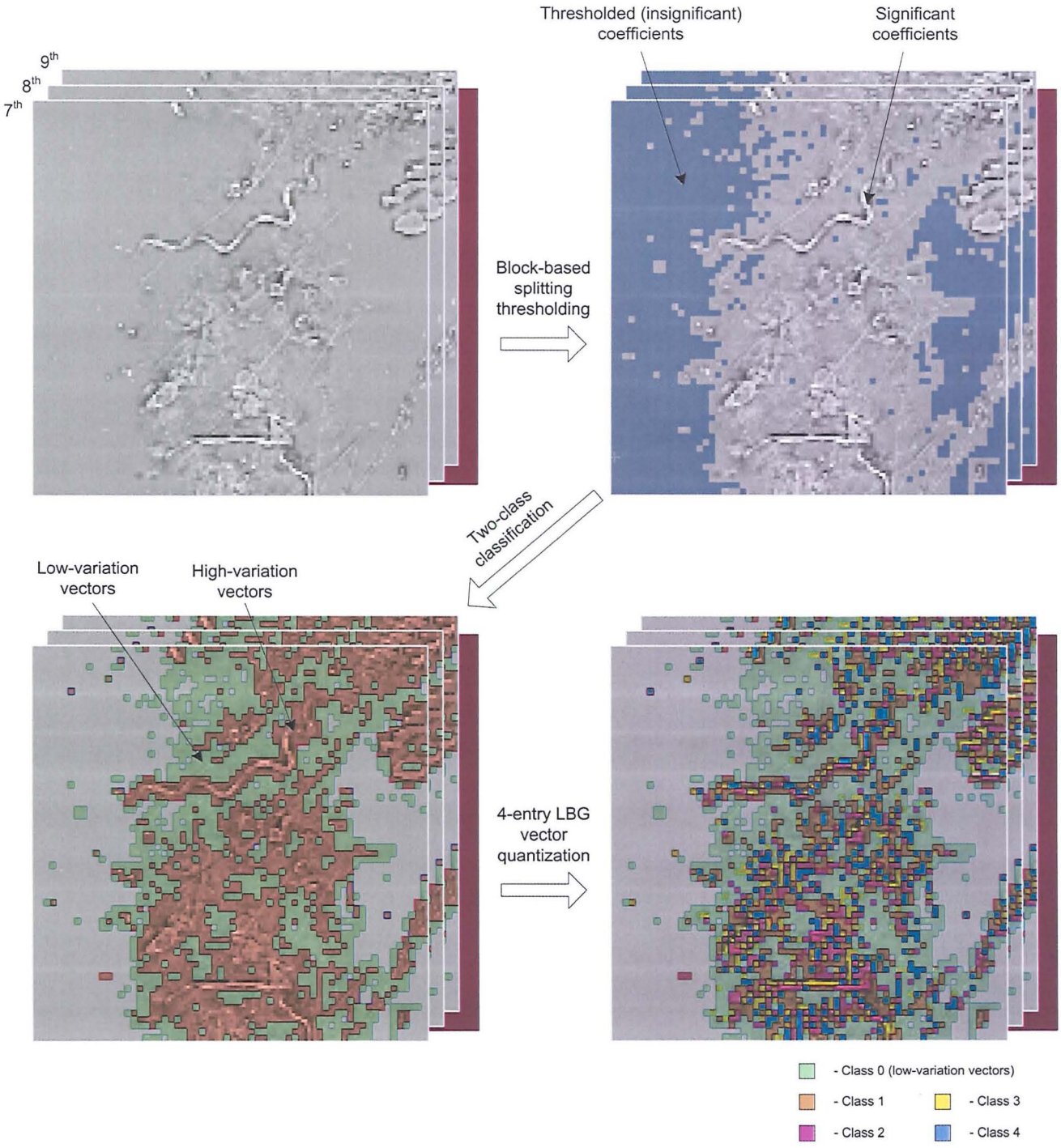


Figure 7-10: An illustration of thresholding and CBCVQ performed on level 1.

For level 0, which is below l_1, the encoding process also starts from the block-based splitting thresholding. Coefficients are block-wisely compared with the threshold T_0 and only those significant blocks will be encoded by classified VQ. The classification map is derived from that of l_1. Since the image size is doubled in each direction compared to that of l_1, the block size of classification needs to be doubled, which is

4×4×6 in l_0. One should note when using the same-orientation propagation rule we may have blocks mapped to none of the classes generated in l_1. This is because these blocks containing significant coefficients while their corresponding nodes on l_1 have been recognized as insignificant nodes and they have not been considered in classification at all. In this case we put these blocks into class 0. So the same-orientation propagation rule can be formulated as

$$class(l_k, x, y, z), k = N-3,\ldots,0,$$
$$= \begin{cases} class(l_{N-2}, \frac{x}{2^{N-2-k}}, \frac{y}{2^{N-2-k}}, \frac{z}{2^{N-2-k}}), & (\frac{x}{2^{N-2-k}}, \frac{y}{2^{N-2-k}}, \frac{z}{2^{N-2-k}}) \in \{l_{N-2,sig}\} \\ 0, & (\frac{x}{2^{N-2-k}}, \frac{y}{2^{N-2-k}}, \frac{z}{2^{N-2-k}}) \in \{l_{N-2,insig}\} \end{cases} \quad (7.9).$$

In this example the propagation of classification map from l_1 to l_0 is illustrated in Fig. 7-11.

Once we obtained the classification map for every level below N-2, the encoding on each level in the second part is a typical classified-VQ with N_c+1 classes (the light orange block in second part). One should notice that the vector size on these levels does not need to be the same as the block size for classification, but the block size has to be in multiple with vector size to make sure that each vector belongs to a unique class.

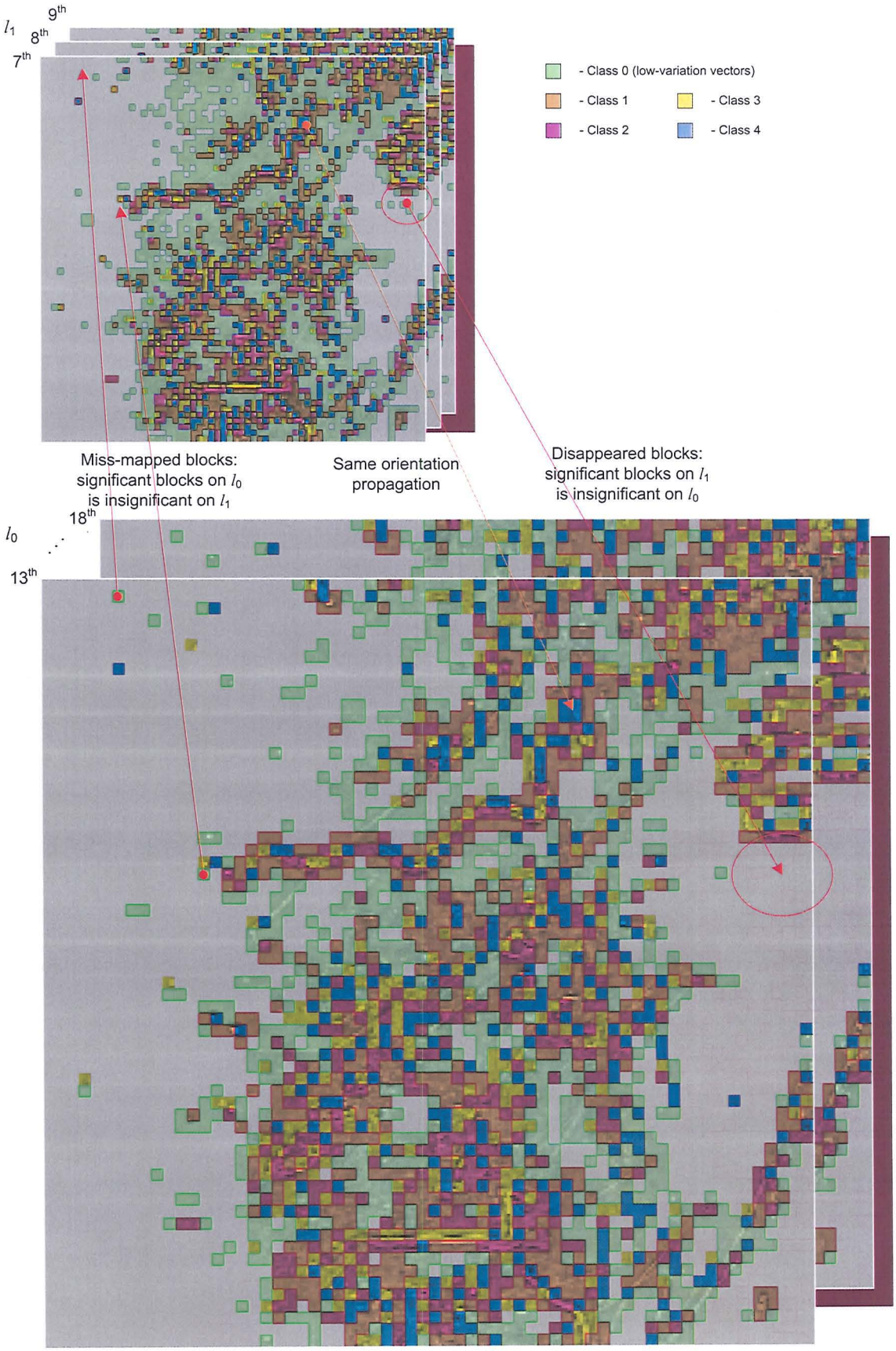


Figure 7-11: Same-orientation propagation of classification map from l_1 to l_0.

7.2.3 Vector Forming

Compared to the cross-band vector forming approaches in [Huh, 1995; Wang, 1993], each vector in our method is purely formed from intra-band coefficients, not from inter-band coefficients. Fig. 7-12 gives two examples of vector forming: Fig. 7-12(a) is cross-band vector forming method considering the inter-band coefficients, which has been used in [Huh, 1995]; and Fig. 7-12(b) is the intra-band vector forming method used in our pyramid coding. There are two reasons for choosing the intra-band vector forming. From Fig. 7-12(a), we noticed that using the cross-band vector forming method, it is difficult to control the bit allocation for each individual level. In fact all the three levels are sharing with the same bit allocation scheme, which is determined by the signal composed from three levels concurrently. As we will discussed in the following chapter, due to the different statistical characteristics between pyramid levels the bit allocation schemes for each level are different with each other. Compared to the optimal cross-level bit allocation scheme, sharing the same bit allocation may decrease the rate-distortion performance of the entire compression system. However in Fig 7-12(b) this problem can be solved since each vector quantizer quantizes intra-band coefficients from only one particular level. Therefore it is more flexible to control the bit allocation for each level.

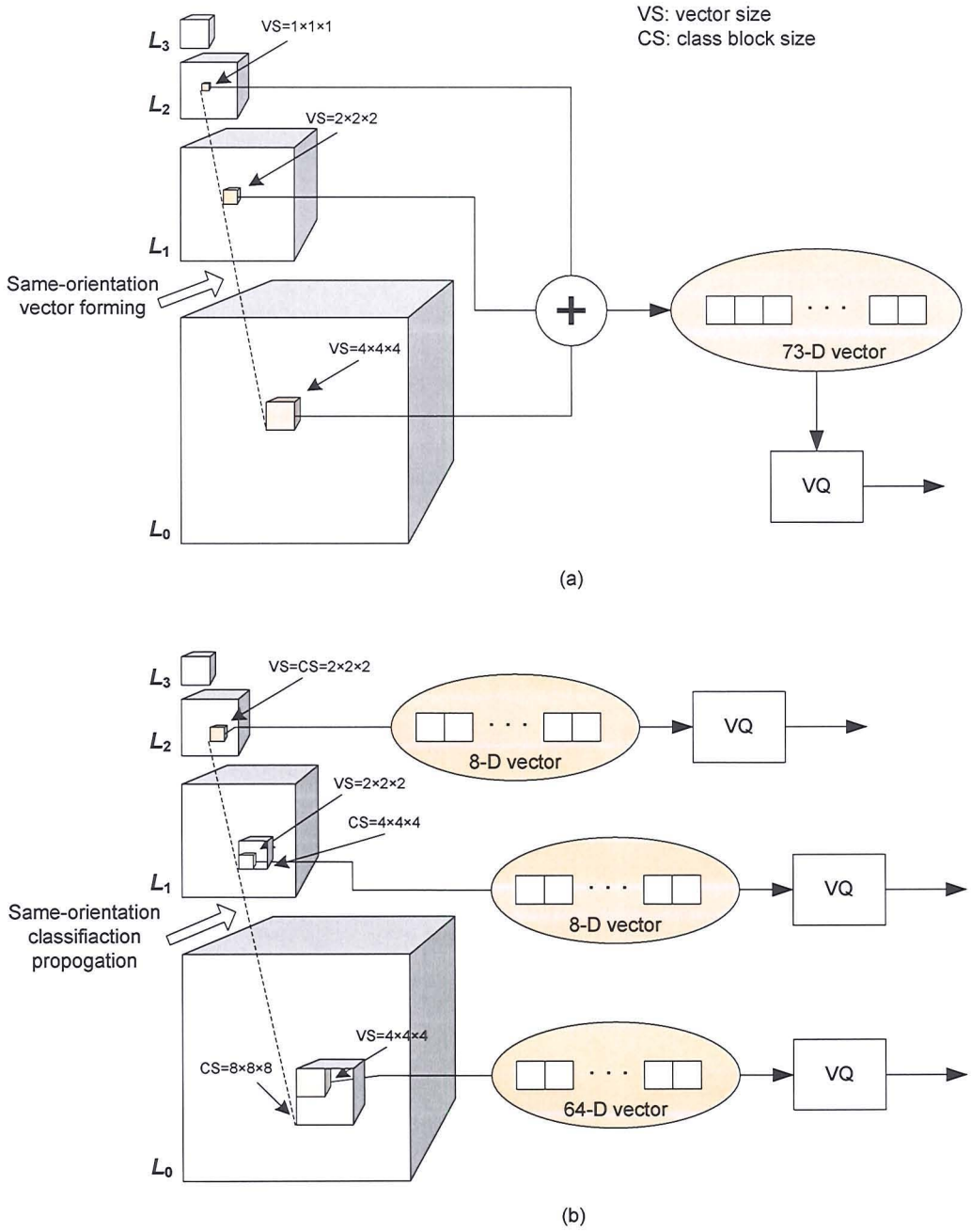


Figure 7-12: Vector forming in subband coding. (a) cross-band vector forming - a vector consists both of intra-band and inter-band coefficients; (b) intra-band vector forming - a vector consists only of intra-band coefficients.

Moreover in Huh's work they explained the purpose of using the cross-band vector forming is to capture the similarity of cross-band coefficients along the same orientation. However many people [Cosman, 1996; Schelkens, 2003] have pointed out that two issues should be considered when using such cross-band scheme. One is that it has been found both in practice [Liu, 2000] and theory [Munteanu, 1999] that intra-

band dependency is greater than the inter-band dependency in the hierarchical subband decomposition. This observation indicates that the intra-band models capture most of the dependencies between the coefficients, and that only minor gains can be obtained with composite models that capture both types of dependencies. Another issue is the cross-band similar distribution feature can be observed only when the node sizes are large [Schelkens, 2003]. In Huh's method, for instance, the node size as 1×1×1 in L_2 cannot give a good prediction of its descendent nodes in L_1 and L_0. Using large node size on L_2 can improve the cross-band prediction but the exponential increment of vector size in the follow levels will significantly increase the computational complexity and storage requirement of VQ. Therefore in recent studies the cross-band dependencies have been considered in an indirect way. For example using the cross-band dependencies to build a context model to improve entropy coding of the intra-band codes [Taubman, 2001]; or, such as Fig. 7-12(b), the cross-band correlations are captured by same-orientation classification propagation with large block size while in each class the vector quantizer can be flexibly defined without constraint of vector size.

To take into account the different distribution characteristics of coefficients along x, y and z directions (see Section 2.5.1.2), we developed an adaptive vector shape decision scheme, where we determine the shape of the vector in each vector quantizer according to the distribution characteristics of the corresponding source. For instance, suppose coefficients are grouped as 12-dimensional vectors and there are three types of vector shapes as shown in Fig. 7-13. According to Shannon coding theory, better coding efficiency can be achieved by increasing the vector size. Therefore we choose such a vector shape that along the direction of larger vector size the coefficients would have stronger correlations and smaller variations. An automatic pre-analysis of variances is used to determine which strategy will be preferable based on the above criterion. In practice we compute the variations of the probability distributions of coefficients along x, y and z directions and the shape of the vector is determined by δ_x^2, δ_y^2 and δ_z^2. For instance suppose we have $\delta_x^2 \geq \delta_z^2$, $\delta_y^2 \geq \delta_z^2$, we will use strategy 3, having largest size along z direction, to encoding this source. We should note in the cross-band classified VQ, each source consists of intra-band vectors from level i and also associated to class j.

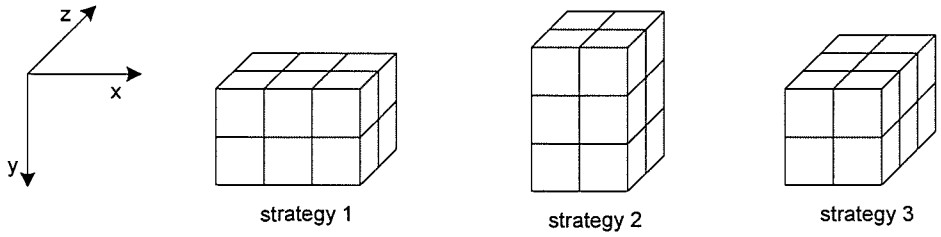


Figure 7-13: Three types of vector shape for a 12-D vector.

Chapter 8 System Optimization

So far we have described the structures of the 3D-CoMIC compression system, where two most important parts are the three-dimensional pyramid decomposition and the cross-band classified vector quantization. For each part the encoding process can be viewed as the cooperation of several sub-systems. For instance the encoding of the whole pyramid can be viewed as a sequential of operations on each level and in each level the encoding can be viewed as a parallel of operations on each sub vector quantizer. Each sub system has its own performance characteristic and all of them contribute to the overall performance of the whole system. This will raise a question: How to control the performance of each sub system such that the overall performance can be maximized? In terms of rate-distortion performance this question can be interpreted as how we control the rate (or distortion[1]) of each sub system such that the overall rate-distortion performance can be optimal. This is usually referred to as *bit allocation* problem of a compression system.

It is known that in our distortion-constraint compression system the rate-distortion performance can be measured as the final output rate of the original image while under the condition that the reconstructed image quality has to be no less than the target quality. Therefore the purpose of bit allocation in our system is adaptively and automatically controlling the rate (or distortion) of each sub system according to its rate-distortion characteristic to minimize the overall rate while under the constraint that the distortion of the reconstructed image is no more than the target distortion

[1] Without specifically mentioned, the 'distortion' is mean-square-error distortion measure (MSE).

level. Two algorithms regarding the bit allocation problems of the aforementioned two parts will be discussed in the first part of this chapter. One is for inter-band bit allocation; the other is for intra-band bit allocation.

In the second part we will address the computational complexity of the system, which is one of the issues to be considered in practice. We noticed that because it uses more computing time in training and encoding than in decoding the VQ approach is a high-asymmetric system. Therefore, focused on VQ techniques, we will discuss the efforts to improve the training and encoding processes. Compared to the conventional methods, our experimental results show that the methods we investigated have an appreciable improvement in computational cost.

8.1 Rate-Distortion Performance Optimization

In our method quantization is performed on vectors or blocks consisting of intra-level coefficients of the difference pyramid. The bit-allocation task on each level is determined by the quantizers and the characteristics of distribution of coefficients of this level only, which is independent from the other levels. Therefore the bit-allocation task of the whole pyramid can be divided into two separate aspects: the inter-level (inter-band) bit allocation and the intra-level (intra-band) bit allocation. Since the coding process starts from the pyramid decomposition and then followed by the intra-level quantization, the inter-level bit allocation is ahead of the intra-level bit allocation. Therefore in Fig. 8-1 we can see the bit-allocation process is firstly determining the number of bits allocated to each level (the blue block) and then determining how to allocate the bits assigned to a particular level to its quantizers (the light orange block). The rate-distortion performance optimization can be achieved by optimizing the bit allocation of each level and then optimizing the bit allocation of each intra-level quantizer for a particular level.

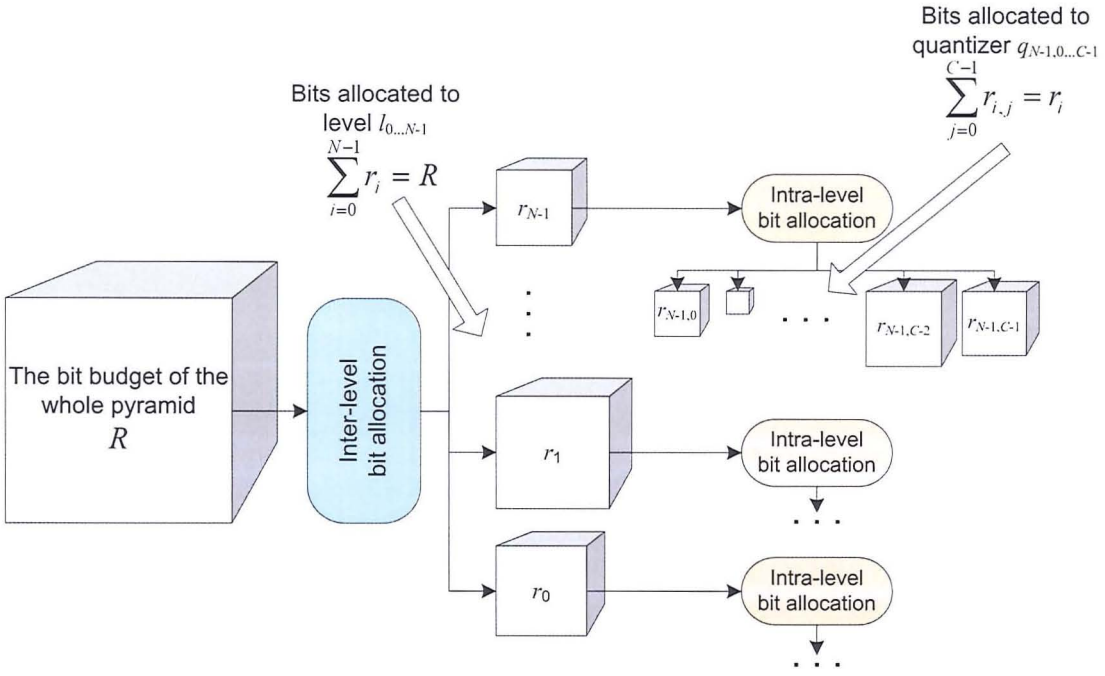


Figure 8-1: Block diagram of bit allocation scheme. It starts with inter-level bit allocation and then followed by intra-level bit allocation on each level.

8.1.1 Rate-minimizing Inter-level Distortion Control

Thanks to the error-feedback encoding scheme the final image quality is determined by the encoding of bottom level, but the coding efficiency is dependent on all the levels across the whole pyramid. The inter-level bit allocation task can be intuitively explained by the following example. Suppose we use few bits on the higher levels, the reconstructed qualities of these levels will be very poor, and according to the error feedback scheme, it will bring more errors to the following lower levels, and therefore for lower levels, they may need more bits to recover these errors to ensure the same quality level. Conversely, if we increase the rate of higher levels both the distortion of these levels and the quantization error introduced on the following lower levels can be reduced. However the expense of extra bits on the higher levels may be greater than the gain of the reduced bits on the lower levels. Therefore in both cases the bit allocation is not optimal, and some 'extra' bits are wasted. Now considering the quality-control property of our method the inter-level bit allocation problem can be described as to determine the bit allocation profile for each level of pyramid which would minimize the overall rate while under the constraint that the final reconstructed distortion is no more than the target distortion.

The problem of the bit allocation across layers in a closed-loop image pyramid has been investigated in [Horn, 1997; Aiazzi, 1997; Ramchandran, 1994]. Among these works, they describe this problem in the context of pyramid coding as determining the average number-of-bit per sample (or rate) r_i for each level which would minimize the distortion D between the reconstructed image and the original while under the constraint that the overall rate R is no more than a predefined value R_0,

$$\text{optimum rate scheme} : \arg\min\left[\sum_{i=0}^{N-1}\delta_i^{(d)}(r_i)\right]\Big|_{\sum r_i<R_0} \tag{8.1},$$

where $\delta_i^{(d)}(\cdot)$ is the rate-distortion function corresponding to the encoding of coefficients in i^{th} level. Horn found the solution and described it in the following form,

$$r_i = \begin{cases} \dfrac{1}{2}\log 2\dfrac{\eta\sigma_i^2(M_{i-1}-M_i)}{\sigma_{i-1}^2(M_i-M_{i+1})} & i>0 \\ R_0 - \dfrac{\sum_{j=1}^{N-1}M_j r_j}{\sum_{j=0}^{N-1}M_j} & i=0 \end{cases} \tag{8.2},$$

where M_i and σ_i^2 are the number of samples and their variance, in the i^{th} level of difference pyramid in the open-loop form as illustrated in Fig. 5-1, respectively; η is referred as the power transfer factor indicating how much energy from previous level has been propagated to the current level, which is only determined by the interpolation filter f_E.

In their works they found that in a closed-loop structure the optimum rate profile for each level (l_i, i=1,…,N-1), except the bottom level, does not depend on the predefined target rate R_0, but are determined by the distribution characteristics of the pyramid coefficients and the spectrum property of the interpolation filter. Once the rate profiles for these levels are determined the bits allocated on the bottom level l_0 is just the remaining bits by subtracting the total number of bits allocated for upper levels from the bit budget R_0. This is in contrast with the optimum rate scheme in an open-loop structure. In the open-loop structure the encoding of each level affects both the rate and the distortion of the final reconstructed image. Therefore the optimum rate profile in this case depends not only on the distribution characteristics of the pyramid coefficients but also the predefined target rate R_0.

However their works cannot be directly applied in our case. This is because their methods are basically rate-parameterized and in (8.2) we found the target is determined by the rate R_0. However in the case of distortion-parameterized coding, the target is determined by a particular distortion D_0; the rate, which is expected to be minimized by the optimal bit allocation scheme, is unknown. Therefore we would prefer the bit-allocation problem to be described in terms of a distortion control scheme rather than the conventional rate control schemes. This is achievable thanks to the inter-changeability between the rate-distortion function $\delta_R(D)$ and distortion-rate function $\delta_D(R)$.

The problem in our case is to determine the distortion profile d_i for encoding each level of pyramid which would minimize the overall rate R while under the constraint that the final reconstructed distortion D is no more than a predefined value D_0. Compared to the optimum rate scheme, we can describe the problem of optimum distortion scheme as,

$$\text{optimum distortion scheme} : \text{argmin}[\sum_{i=0}^{N-1} \delta_i^{(r)}(d_i)]\Big|_{d_0<D_0} \qquad (8.3),$$

where $\delta_i^{(r)}(\cdot)$ is the distortion-rate function for coefficients in level i.

To derive the solution of (8.3) from (8.2) we followed the same assumption in [Horn, 1997; Aiazzi, 1997] that the samples in difference pyramid can be modelled as a zero-mean Laplacian distribution such that,

$$d_i = \delta_i^{(d)}(r_i) = \tilde{\sigma}_i^2 2^{-2r_i} \qquad (8.4).$$

Here d_i is the mean-squared error distortion between the i^{th} level difference pyramid and its reconstruction and $\tilde{\sigma}_i^2$ denotes the variance of the difference pyramid in the closed-loop form as illustrated in Fig. 5-2. Different from σ_i^2 in (8.2), $\tilde{\sigma}_i^2$ is determined by two aspects. One is the variance of the same-level difference pyramid in the open-loop form, say σ_i^2, the other is the encoding error produced at the previous level i+1 which is going to be reported at current level i, say d_{i+1}. Moreover due to the interpolation operation, only partial of the encoding error from the previous

level will be propagated, which is determined by the power transfer factor. Therefore the variance in the closed-loop form can be estimated as:

$$\tilde{\sigma}_i^2 = \sigma_i^2 + \eta \cdot d_{i+1} \qquad (8.5).$$

By substituting (8.4) and (8.5) into (8.2) we can work out the solution to the optimum distortion profile as:

$$d_i = \begin{cases} 0 & i = N-1 \\ (\sigma_i^2 + \eta \cdot d_{i+1}) \dfrac{\sigma_{i-1}^2 (M_i - M_{i+1})}{\eta \sigma_i^2 (M_{i-1} - M_i)} & N-1 > i > 0 \\ D_0 & i = 0 \end{cases} \qquad (8.6),$$

where M_i, σ_i^2 and η are of the same meaning as those in (8.2). d_{i+1} is the optimum distortion profile of the previous level which is used to compute the optimum distortion profile d_i of current level.

The significance of Eq. (8.6) is that, considering a closed-loop encoding structure, if the encoding process on each level is precisely controlled at the distortion level specified by (8.6) then the overall rate can be minimized under the constraint that the final reconstructed distortion D is no more than a predefined value D_0. The way to achieve this objective can be explained by Eq. (8.6) itself in two steps. Firstly, the distortion of the reconstructed image only depends on the encoding quality of the bottom level, therefore specifying the bottom-level distortion profile as D_0 can guarantee that the final reconstructed distortion is strictly controlled at the target D_0. Moreover the rate minimization is achieved by considering an optimum bit allocation for other levels. From (8.6) we can see that the optimum distortion profiles between the top level and the bottom level do not depend on the target distortion D_0, which is consistent with the findings in optimum rate scheme such that the bit allocations on these levels do not depend on the user-specified target. According to the encoding-error-feedback scheme they are determined by the distribution characteristics of the pyramid coefficients, the spectrum property of the interpolation filter and the distortion profile of the previous level.

When the Eq. (8.6) is applied in our three-dimensional pyramid decomposition it can be further simplified. We have discussed the interpolation filter f_E in Section 6.2.2, which in our method is a 7-taps kernel parameterized by one parameter b and it has

the same form as in [Aiazzi, 1997]. [Crochiere, 1983; Aiazzi, 1997] investigated the power gain η in the Laplacian image pyramid form and concluded that the power gain is only related to the property of the frequency response of the interpolation filter, which can be expressed in the following form:

$$\eta \equiv \left(\frac{1}{4}\int_{-\pi}^{\pi}|\psi_E(\omega)|^2 d\omega\right)^k = \left(\frac{1}{2}\sum_{i=1}^{7}|f_E(i)|^2\right)^k = (2b^2 - b + 3/4)^k \quad (8.7).$$

Here k denotes the dimensions of the image data. For example k=2 for decomposing a single frame image and k=3 for decomposing a volumetric image data. In (6.13) we chose b=0.5625 as the interpolation filter in our 3D pyramid decomposition, therefore from (8.7) η in our method can be computed as:

$$\eta = (2b^2 - b + 3/4)^3 \cong 0.552 \quad (8.8).$$

Moreover in the 3D pyramid decomposition, since the subsampling is performed on three directions for a volumetric data, the number of coefficient between two adjacent levels has the relation that:

$$M_i = 8 \cdot M_{i+1} \quad (8.9).$$

Therefore in the 3D pyramid decomposition the factor $\frac{(M_i - M_{i+1})}{(M_{i-1} - M_i)}$ in (8.6) can be computed as:

$$\frac{(M_i - M_{i+1})}{(M_{i-1} - M_i)} = \frac{1 - \frac{1}{8}}{8 - 1} = \frac{1}{8} \quad (8.10).$$

By substituting (8.8) and (8.10) into (8.6) we can find out the solution to the optimum distortion profile in the case of the three-dimensional image pyramid decomposition as:

$$d_i = \begin{cases} 0 & i = N-1 \\ \dfrac{(\sigma_i^2 + 0.552 d_{i+1})\sigma_{i-1}^2}{4.42\sigma_i^2} & N-1 > i > 0 \\ D_0 & i = 0 \end{cases} \quad (8.11).$$

Now we describe the algorithm of the inter-level optimum distortion scheme as the following.

Algorithm 8.1: Optimum Inter-level Distortion Control

Step 1. Build an N-level open-loop difference pyramid. Compute the variance σ_i^2 of coefficients for each level in the difference pyramid.

Step 2. Set target distortion D_0 of the reconstructed image to the bottom level l_0: $d_0 = D_0$.

Step 3. Specify the distortion for the top level. Here since we used entropy coding on the coarsest pyramid level, the distortion is zero: $d_{N-1} = 0$.

Step 4. Iteratively compute the distortion profile for other levels, from l_{N-2} to l_1, using formula (8.11).

One should note that the optimum distortion profile obtained from above inter-level distortion control algorithm is based on the assumption that coefficients in difference pyramid can be modelled as a zero-mean Laplacian distribution. In practice, however, coefficients may not fit the model perfectly. This comes from many aspects, for instance, the imperfect band-stop properties of the half-band low-pass filter for downsampling and the randomly-distributed quantization noise introduced from previous level. These factors will lead to the mismatch between the actual distribution of the coefficients to the theoretic model and also affect the optimality of the distortion control algorithm. However it is very difficult to obtain the practically optimum distortion profile. Actually there is no straightforward way to pre-compute such optimum distortion profile without the practical attempts. One has to decide the optimum distortion profile from the actual encoding results. This process usually involves multiple attempts of encoding, for instance an iterative approach that repeatedly update the distortion profile until the overall rate is approaching to the minimum. Therefore its computational complexity is much higher compared to the proposed model-based distortion control algorithm. Moreover we have found that the difference between the minimized overall rate using the distortion control algorithm and that from the practical results is not significant. The degradation of optimality using model-based distortion control algorithm can be no more than 1% of the

practical result. Therefore the proposed distortion control algorithm is preferable for its effectiveness in optimization and simplicity in computation. This will be validated in Section 8.1.3.

8.1.2 Distortion-constraint Intra-level Bit Allocation

The inter-level distortion control determines a distortion profile for each level, which means each intra-band encoder is constrained at a particular quality level specified by this distortion profile. Optimum inter-band bit allocation can be achieved only when each intra-band encoder is working at its particular quality level; otherwise more bits would be necessary to ensure the same final output quality, as it is aforementioned.

In Section 7.2 we noticed that the intra-level coefficients are firstly recognized as two groups: insignificant coefficients will be thresholded and decoded as zero values while significant coefficients will be further VQ-encoded. The intra-level bit allocation method concerns only significant coefficients since they contribute more on the rate-distortion performance than those insignificant coefficients. Therefore the distortion profile of the VQ encoder is not the same as the inter-level distortion profile, which considers both the insignificant coefficients and the significant coefficients. The actual distortion profile used for constraining the VQ encoder is computed from the inter-level distortion profile by removing the factor from those insignificant coefficients. So at i^{th} level the distortion profile of the VQ encoder is,

$$d_i^{(VQ)} = \frac{d_i \cdot M_i - d_i^{(TH)} \cdot M_i^{(TH)}}{M_i - M_i^{(TH)}} \qquad (8.12),$$

where d_i and $d_i^{(TH)}$ are the pre-computed distortion profile and the distortion of the insignificant (thresholded) coefficients, respectively; and M_i and $M_i^{(TH)}$ are the number of coefficients of i^{th} level coefficients and the insignificant coefficients, respectively. More specifically, given a threshold value T_i the distortion of the thresholded coefficients can be computed by,

$$d_i^{(TH)} = \frac{1}{M_i^{(TH)}} \sum (c_{i,j})^2 \Big|_{c_{i,j} \in l_i, |c_{i,j}| \le T_i} \qquad (8.13).$$

We use classified vector quantizer for intra-band coding. Each vector is firstly classified according the classification map and associated with a unique sub vector quantizer. Different from inter-band bit allocation, where the inter-level bit allocation is dependent, the encoding procedure among these sub vector quantizer is independent with each other. So in this case the bit allocation for a particular level can be described as minimizing the average rate among these sub quantizers while under the constraint that the overall distortion from these sub quantizers is no greater than the previously-computed distortion profile $d_i^{(VQ)}$. Consider at i^{th} level, we use C_i-class classified VQ to encoding this level, so totally C_i sub-VQs, and the constraint distortion for the CVQ is $d_i^{(VQ)}$, and the intra-level bit allocation problem is to find the optimum rate profile $r_{i,j}$ for each sub quantizer $q_{i,j}$ satisfying the following condition,

$$\text{optimum rate scheme}: \text{argmin}\left[\sum_{j=0}^{C_i} r_{i,j}\right] \Bigg| \sum_{j}^{C_i} (d_{i,j} = \delta_{i,j}^{(d)}(r_{i,j})) \le d_i^{(VQ)} \qquad (8.14).$$

The solution to such multiple variable optimization problem where the variables are independent is the well-known equal-slope method. This method is commonly referred as 'Pareto Optimality' in economics [Walras, 1954] and widely used in compression applications [Cockshott, 2000]. Its intuitive explanation can be briefly described in Fig. 8-2.

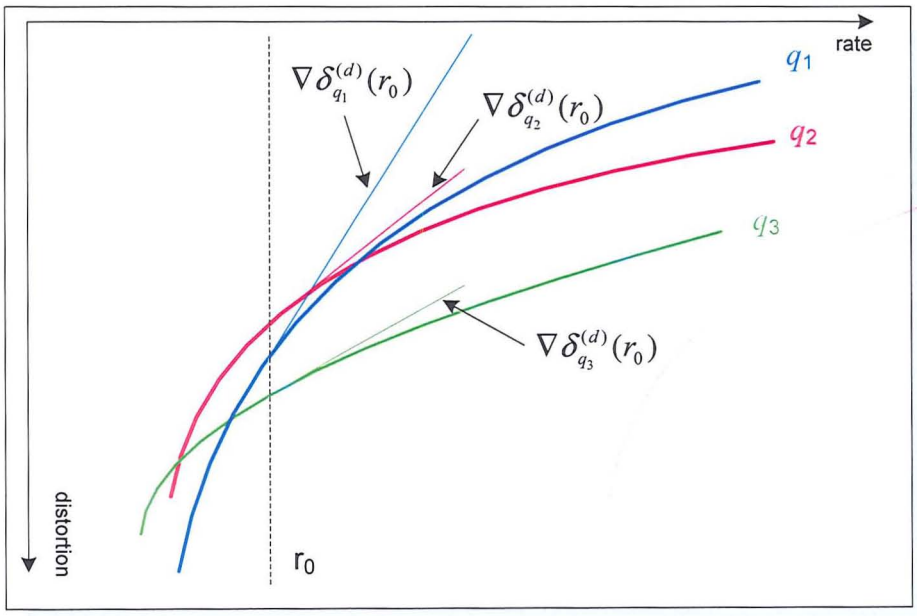


Figure 8-2: An example of the solution to the Parato Optimality.

Suppose there are three quantizers, q_1, q_2 and q_3. Each is individually encoding a part of a source without overlapping. Each quantizer is operating according to its rate-distortion function $\delta_*^{(d)}(r)$. At the moment they are all working at the same rate r_0. Now if the channel allows us to increase the overall rate slightly, which quantizer shall we choose to increase? At rate r_0, we found that q_1 (the blue curve) has the maximum magnitude of slope. So we allocate these bits to q_1 since it will produce maximum amount of distortion decrease than others.

Now we give the algorithm of using 'Pareto Optimality' to solve the optimum intra-level bit allocation problem for level i. Consider a $k_{i,j}$-dimensional sub-LBGVQ $q_{i,j}$ is designed at rate $r_{i,j}$, its normalized magnitude of slope $\varepsilon_{i,j}$ at this rate can be computed by taking the difference of distortion of $q_{i,j}$ at rate $r_{i,j}$ and $r_{i,j}$+1, corresponding to the codebook size $2^{r_{i,j}}$ and $2^{r_{i,j}+1}$, respectively; and then normalized by its percentage in the entire samples in this level:

$$\varepsilon_{i,j}(r_{i,j}) = \nabla\delta_{q_{i,j}}^{(d)}(r_{i,j}) = \frac{M_{i,j}\cdot k_{i,j}\cdot[\delta_{i,j}^{(d)}(r_{i,j}) - \delta_{i,j}^{(d)}(r_{i,j}+1)]}{\sum_{j=1}^{C_i} M_{i,j}} \qquad (8.15),$$

where $M_{i,j}$ is the number of vectors associated to the quantizer $q_{i,j}$.

Algorithm 8.2: Optimum intra-band bit allocation for sub VQs $\{q_j\}_i$ under the distortion constraint $d_i^{(VQ)}$

Step 1. (Initialization) Initialize each sub-VQ at rate $r_{i,j}$=0 with codebook size = 1. Compute $\varepsilon_{i,j}(r_{i,j})$ for each sub-VQ according to Eq. (8.15). Store $\varepsilon_{i,j}(r_{i,j})$ in an one-dimensional array E with $E[j]$=$\varepsilon_{i,j}(r_{i,j})$.

Step 2. (Stopping rule) Compute the overall average distortion d_i' as

$$d_i^{'} = \frac{\sum_{j=1}^{C_i} M_{i,j}\cdot k_{i,j}\cdot\delta_{i,j}^{(d)}(r_{i,j})}{\sum_{j=1}^{C_i} M_{i,j}} \qquad (8.16).$$

If $d_i^{'} \le d_i^{(VQ)}$ we continue to Step 3; otherwise stop, with each sub-LBGVQ $q_{i,j}$ designed at rate $r_{i,j}$.

Step 3. (Find the sub-VQ with the maximum magnitude of slope) Search E to obtain the index $j_{\max}$ corresponding to the maximum magnitude of slope produced by $q_{i,j}$ at rate $r_{i,j}$.

Step 4. (Allocate bit to the sub-VQ $q_{i,j_{\max}}$) Re-build the sub-VQ $q_{i,j_{\max}}$ by doubling its codebook size from $2^{r_{i,j_{\max}}}$ to $2^{r_{i,j_{\max}}+1}$. Update its rate as $r_{i,j_{\max}} = r_{i,j_{\max}} + 1$.

Step 5. (Update E and iteration) Replace $E[j_{\max}]$ with $\varepsilon_{i,j_{\max}}(r_{i,j_{\max}})$. Go back to Step 2.

Although each sub vector quantizer is operating with its own characteristics, such as number of associated vectors and vector dimensions, their rate-distortion performances are normalized using (8.15) and mapped to the overall rate-distortion performance of the entire sample set. Therefore at step 3 the quantizer $q_{i,j_{\max}}$ always produce maximum amount of distortion reduction in the sense of the entire sample set for each increment of rate, say Δr. As we will see that the proposed algorithm satisfies both the necessary and sufficient conditions for the optimality of the rate-distortion performance. At each iteration i, given each increment of rate, Δr, the selected quantizer $q_{j_{\max}}$ produces maximum amount of decrease in distortion, so we have,

$$\left|\Delta d_{j_{\max}}^{(i)}\right| \geq \left|\Delta d_{j}^{(i)}\right|, \forall j \qquad (8.17).$$

The final distortion after K iterations will be,

$$d^{(K)} = d^{(0)} - \sum_{i=1}^{K}\left|\Delta d_{j_{\max}}^{(i)}\right| \leq d^{(0)} - \sum_{i=1}^{K}\left|\Delta d_{j}^{(i)}\right|, \forall j \qquad (8.18).$$

If we assume the rate-distortion function of each sub quantizer is different from each other. The equivalence in (8.18) is true only when j=$j_{\max}$ at each iteration. Therefore the algorithm is sufficient for the optimality.

For the necessary condition the proposition would be if $d(K)$ is minimized at iteration K then at each iteration the selected quantizer must be the quantizer computed from the above algorithm. Now we reduce this proposition to its absurdity, of which the contradictory would be $d(K)$ can be minimized by choosing q_j other than $q_{j_{\max}}$. That is,

$$\text{contradictory}: \exists j_c, (j_c \neq j_{\max}) \wedge (d^{(0)} - \sum_{i=1}^{K}\left|\Delta d_{j_{\max}}^{(i)}\right| = d^{(0)} - \sum_{i=1}^{K}\left|\Delta d_{j_c}^{(i)}\right|) \qquad (8.19),$$

as it is illustrated in Fig. 8-3 for K=2, where the blue curve is the rate-distortion curve computed from $j_{\max}$ and the red curve is that computed from an alternative, j_c. Without loss of generosity, we suppose in the first iteration $j_c^{(1)} \neq j_{\max}^{(1)}$, and then according to the definition of $q_{j_{\max}}$, we will have $\left|\Delta d_{j_{\max}}^{(1)}\right| > \left|\Delta d_{j_c}^{(1)}\right|$. To satisfy (8.19), we will have $\sum_{i=2}^{K}\left|\Delta d_{j_{\max}}^{(i)}\right| < \sum_{i=2}^{K}\left|\Delta d_{j_c}^{(i)}\right|$. Therefore we can derive that there must exist some $j_c^{(i)}, i>1$ such that,

$$\exists i, i>2, \left|\Delta d_{j_{\max}}^{(i)}\right| < \left|\Delta d_{j_c}^{(i)}\right| \tag{8.20}$$

Now we noticed that (8.20) is in contradiction with the property we derived in (8.17), therefore the contradictory is false and the necessary condition is proved.

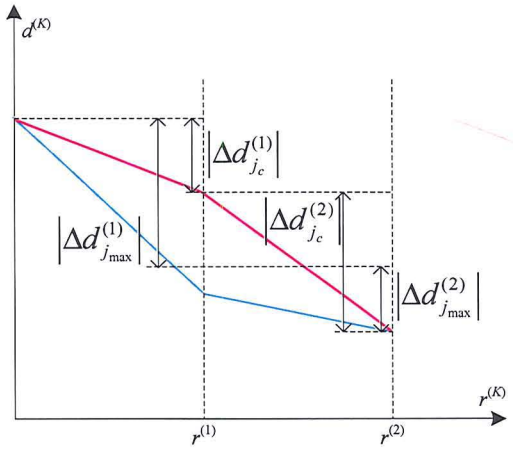


Figure 8-3: A paradox example of two optimum bit allocation scheme.

As we have learned that, in every iteration, the algorithm always allocates bits to the sub-VQ with the maximum magnitude of slope in its rate-distortion curve. The magnitude of slope can be theoretically expressed as the absolute of the first-order derivative of the rate-distortion function:

$$\begin{aligned}\text{magnitude of slope} &= \left|\frac{d}{dr}\delta^{(r)}(r)\right| = \left|\frac{d}{dr}\delta^2 2^{-2r}\right| \\ &= \left|2\ln 2 \cdot \delta^2 \cdot 2^{-2r}\right|\end{aligned} \tag{8.21}$$

From (8.21) we can see that when starting at the same rate, say r=0, the algorithm will allocate bits to the quantizer with which the associated training vectors are of large variance and then its magnitude of slope will decrease by the factor 2^{-2r}. The

allocation will continue until it finds another quantizer whose magnitude of slope is larger than that of the current one and then the bits will be allocated to the newly-selected quantizer. Therefore the allocation will end up with an 'equal-slope' pattern such that the magnitudes of slope of these sub quantizers are tending to be the same. Therefore people usually refer to this method as 'equal-slope' method.

Moreover we can also find in (8.21) that due to the equal-slope property a quantizer associated with low-variation vectors will have a smaller-sized codebook than that of a quantizer associated with high-variation vectors. It means the former will have a much smaller rate than the latter. Because low-variation vectors occupy a large part of the pyramid coefficients, the overall rate, which is averaged by the number of vectors from each class, can be smaller than that without classification. In the following example, a simple validation of rate-distortion performance of classified VQ is performed on the bottom level of the pyramid built for 'B3CGP' data set. The classified VQ has 17 classes and we use the above algorithm to control the bit allocation among these sub vector quantizers. The overall distortion is constrained at MSE=65.025 (PSNR=30.00dB). Fig. 8-4 illustrates the variance of vector set and the number of vectors of each class, where each value is normalized by the ratio to the maximum value in the group. We also listed the computed rate for each quantizer and compared the overall rate to that without using classified VQ, where in the latter an LBG-VQ was applied on the entire vector set and we iteratively increased the rate until the overall distortion is not smaller than the target one. The result indicates that quantizers associated with small-variation vectors tend to have small rates and they also occupy a large part of coefficients. Although for some quantizers the rate is high they did not significantly affect the overall rate due to the small number of vectors associated with them. Therefore in this example we have experimentally observed twice the compression ratio by using classified VQ. This is in consistent with the theoretical investigation in Fig. 7-5.

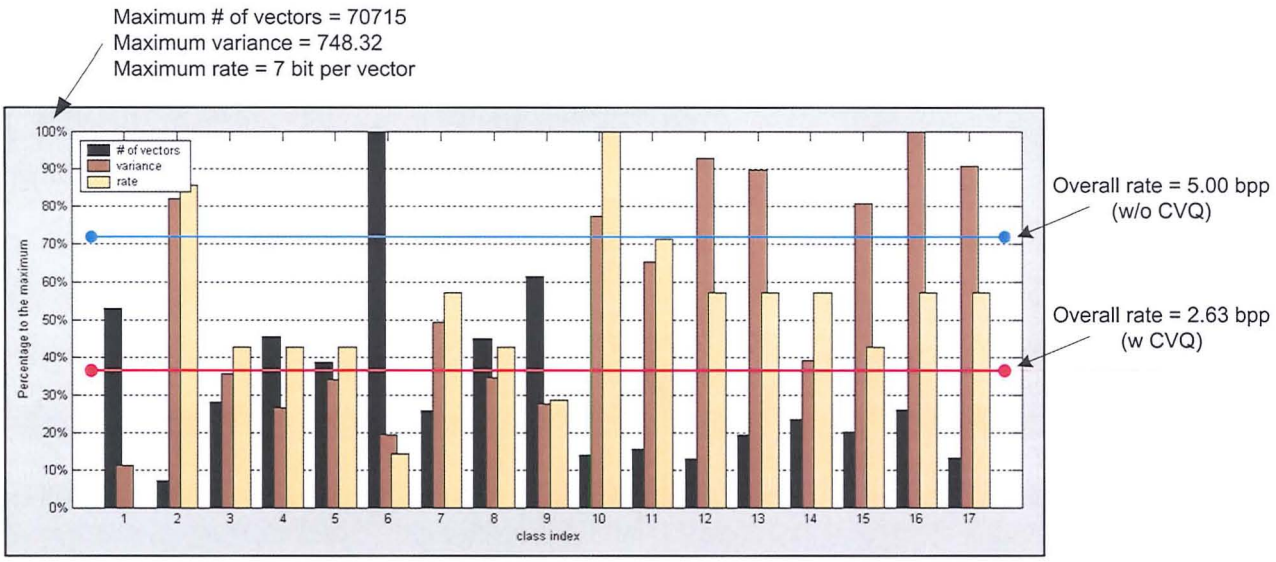


Figure 8-4: An example of bit allocation and classification gain by using a 17-class CVQ compared to the ordinary VQ. The evaluation was on the bottom level of the pyramid built for 'B3CGP' dataset.

8.1.3 Experimental Validation of Optimum Inter-level Distortion Control

It has been stated that the proposed inter-level distortion control approach may be different from the practical optimum result but the loss of optimality can be neglected with considerable gains in computational complexity. To verify this discussion an experimental test was performed on confocal microscopic image data sets. The purpose of the experiment is to find the difference between the practically optimum distortion profile, say $d_i^{(\text{prac})}$, and the theoretically optimum distortion profile, say $d_i^{(\text{theor})}$, computed from the proposed algorithm for a particular level, and also the difference of the rate-distortion performance of the coding system with respect to these two profiles. We focused on only two consecutive levels of the pyramid, the bottom level l_0 and its next above level l_1. We fix the target quality of the reconstructed image and all the variables except for the distortion profile of l_1. The quality and the bits allocated to l_1 will change as we choose different distortion profiles. Moreover according to the error-feedback scheme the variation of quality of l_1 will affect those levels below it, such as l_0, and also the bits allocation of these levels and finally the total bits of the pyramid, or the overall rate. To find the practically optimum distortion profile of l_1, we fixed the target quality of the

reconstructed image and all the variables except for the distortion profile of l_1, and then we observed the overall rate (total bits) by varying this distortion profile. The practical optimum is the profile corresponding to the minimum overall rate.

The test was performed on three CLSM data sets. They are 'B3CGP' data set, 'C5' data set and 'HUM_GLT_C' data set. For each data set we chose three different target quality levels. According to algorithm 8.1, for each data set the theoretically optimum distortion profile $d_1^{(\text{theor})}$ of l_1 can be computed by: 1) building an open-looped difference pyramid; 2) computing the variance for each level; 3) iteratively computing the distortion profile for each level, from the top level to level l_1, using formula (8.11). The theoretically optimum distortion profile of l_1 for each data set was listed in Table 8-1. From the discussion in Section 8.1.1 we learned that, in the scheme of the optimum inter-level bit allocation, these profiles of l_1 do not depend on the final target quality, but are only determined by the characteristics of the pyramid itself. Therefore for a particular data set $d_1^{(\text{theor})}$ will not change as we are working at different target quality levels.

The practically optimum distortion profile $d_1^{(\text{prac})}$ of l_1 which corresponds to the minimum overall rate is computed from the actual coding attempts. There is no straightforward indication of what profile will minimize the overall rate, nor can we find it precisely. Therefore we can only approximate it through finite samples. It is reasonable to assume that the practical profile $d_1^{(\text{prac})}$ is not far away from the theoretical profile $d_1^{(\text{theor})}$. So we computed the practical optimum distortion profile in the following way: 1) computing the theoretical profile $d_1^{(\text{theor})}$; 2) determining a testing distortion profile set of l_1, say $\{\hat{d}_1^{(\text{prac})}\}$, from $d_1^{(\text{theor})}$; 3) performing coding process for each distortion profile in $\{\hat{d}_1^{(\text{prac})}\}$; 4) observing the output overall rate with respect to these profiles and regarding the distortion profile corresponding to the minimum overall rate as the approximated practically optimum profile. The testing profile set $\{\hat{d}_1^{(\text{prac})}\}$ consists of ten consecutive integer-valued PSNR numbers with $d_1^{(\text{theor})}$ located in the middle. That is $\{\hat{d}_1^{(\text{prac})}\} = \{\lceil \text{PSNR}(d_1^{(\text{theor})}) \rceil + i, \quad i = -5 \ldots 4\}$ .In table 8-1 we listed the practically optimum distortion profiles for three date sets at

three different quality levels. In Fig. 8-5 each curve illustrates the overall rate (total bits) changing with respect to the ten distortion profiles. Therefore for one data set at a particular distortion level the practically optimum profile is the profile corresponding to the minimum on its curve.

Table 8-1: Theoretically and practically optimum distortion profile on level 1 and the corresponding overall rate (total bits) for three data sets at three quality levels.

Data sets and quality levels		**Theoretical Optimum**		**Practical Optimum**		**Bit increase (%)** $\frac{R^{(\text{theor})} - R^{(\text{prac})}}{R^{(\text{prac})}} \times 100\%$
		Dist. profile (in dB) $d_1^{(\text{theor})}$	Total bits $R(d_1^{(\text{theor})})$	Dist. profile (in dB) $d_1^{(\text{prac})}$	Total bits $R(d_1^{(\text{prac})})$	
B3CGP	Q_1=28dB	33.560	1,703,306	33.000	1,702,706	0.07%
	Q_2=30dB	33.560	2,301,008	34.000	2,304,014	0.26%
	Q_3=33dB	33.560	3,538,439	35.000	3,515,479	0.65%
C5	Q_1=30dB	35.399	1,458,247	35.000	1,454,126	0.28%
	Q_2=33dB	35.399	2,132,818	36.000	2,124,312	0.40%
	Q_3=35dB	35.399	2,794,635	36.000	2,777,778	0.61%
HUM_GLT_C	Q_1=35dB	40.180	1,686,109	38.000	1,656,324	1.80%
	Q_2=37dB	40.180	1,989,334	38.000	1,930,905	3.03%
	Q_3=38dB	40.180	2,215,832	39.000	2,198,771	0.78%

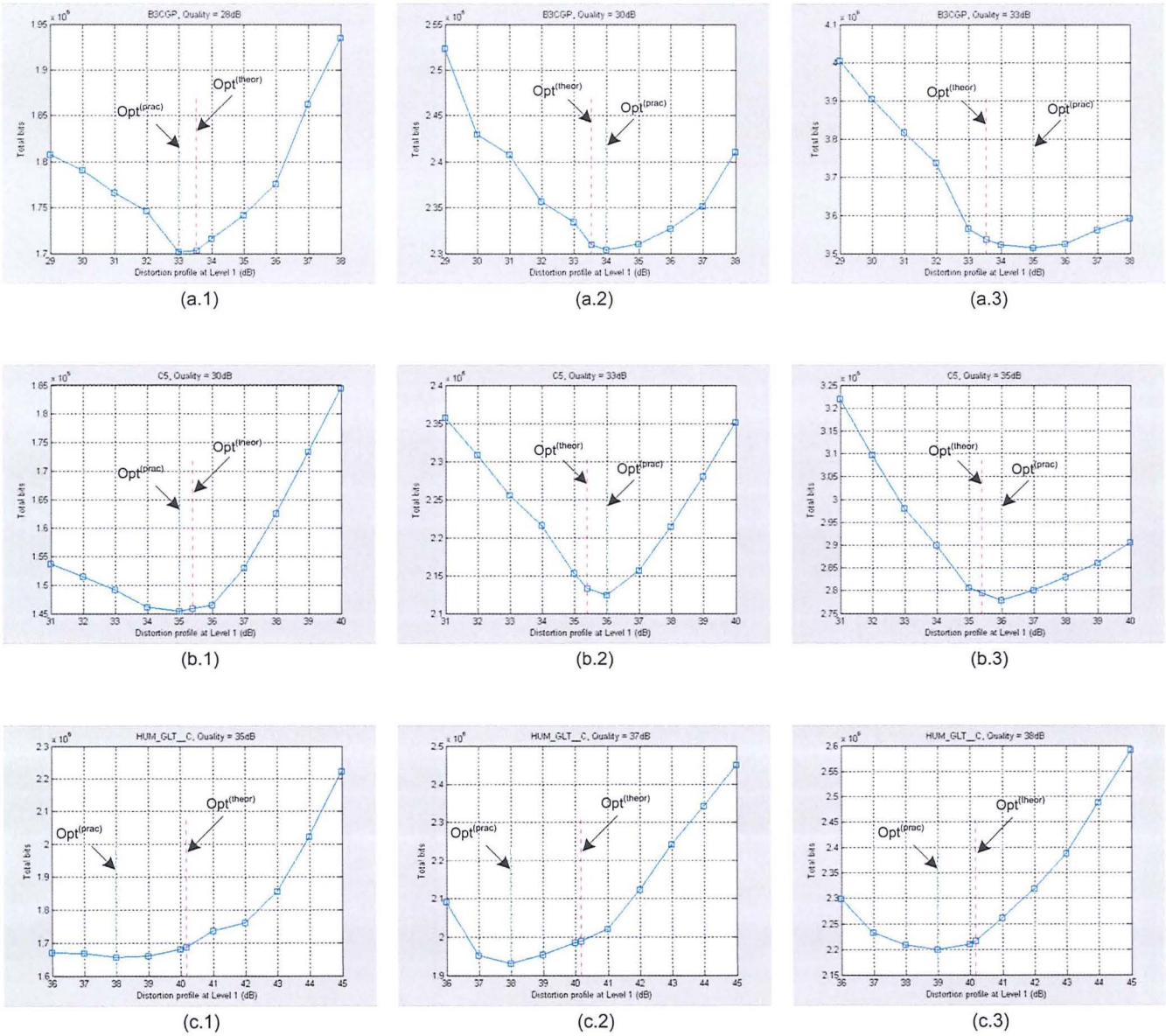


Figure 8-5: The overall rate (total bits) of the 3D-CoMIC method varying with distortion profile on level 1. The experiment was performed on three data sets, (a) B3CGP, (b) C5 and (c) HUM_GLT_C, at three quality levels.

Here we use one group of the results, the 'C5' data set at the quality level PSNR=35dB, as an example to describe how the overall rate would be affected by the distortion profile on l_1. The theoretically optimum profile for this data set is $d_1^{(\text{theor})} = 35.399\text{dB}$, so the testing profile set consists of ten distortion profiles is: $\{\hat{d}_1^{(\text{prac})}\} = \{31\text{dB},\ldots,40\text{dB}\}$. In Table 8-2 we listed the experimental results indicating how the number of bits, allocated to l_1, l_0 and the whole pyramid, vary with different distortion profile of l_1. The orange bar and the blue bar indicate the bit allocation at

the theoretically optimum distortion profile (PSNR=35.399dB) and the practically optimum distortion profile (PSNR=36.000dB), respectively. From the results we noticed that an increasing number of bits are allocated to l_1 as we decrease the distortion profile of this level since the distortion-constraint intra-level bit allocation algorithm requires quantizers to increase their rate to adapt for the decreased distortion. While on the other hand the bits allocated to l_0 are decreasing, which comes from the decreased number of vectors and the decreased rate of vector (bits per vector). This is because the smaller the distortion on l_1 the less the quantization error propagated to l_0 and the smaller the variance of the coefficients on l_0. Therefore when they are thresholded by comparing with the same threshold, there will be less number of significant coefficients and so less number of vectors. Additionally for those significant coefficients the variance will decrease as well. According to the discussion on the rate-distortion theory in Section 3.1, to achieve the same distortion level a quantizer can produce lower rates on small-variation sources. Therefore the bits per vector are reduced as the decrement of the variation of these vectors.

Table 8-2: Bits, allocated to l_1, l_0 and the whole pyramid, varying with the distortion profile of l_1, testing on C5 data set at the quality level PSNR=35dB.

Distort profile on l_1 (PSNR in dB)	**Bits allocation on l_1 (68935 vectors)**		**Bits allocation on l_0**			**Total bits of pyramid**
	Bits per vector	Total bits	Bits per vector	# of vectors	Total bits	
31.000	3.01	207,551	8.03	266,143	2,136,940	3,220,316
32.000	3.58	246,650	7.61	259,805	1,976,366	3,097,197
33.000	4.10	282,815	7.17	254,373	1,823,661	2,979,846
34.000	4.72	325,153	6.89	247,249	1,703,317	2,900,418
35.000	5.39	371,316	6.68	238,829	1,594,244	2,835,933
35.399	5.68	391,642	6.54	234,367	1,533,706	2,794,635
36.000	6.41	441,673	6.47	220,415	1,426,125	2,777,778
37.000	7.72	532,351	6.41	211,769	1,356,796	2,800,119
38.000	9.71	669,432	6.33	195,846	1,239,945	2,829,574
39.000	11.68	805,449	6.27	185,855	1,165,430	2,860,804
40.000	12.34	950,917	6.04	176,893	1,048,511	2,905,387

In Fig. 8-6 we plotted the number of allocated bits to the l_1, l_0 and the pyramid according to the results in Table 8-2. The position corresponding to the practical and the theoretical optimum distortion profiles are indicated. Although the curve is monotonically increasing for l_1 and monotonically decreasing for l_0, the results for the whole pyramid, which depend on both l_1 and l_0, present a valley-shaped curve. Minimum rate only occurs at $d_1^{(\text{prac})} = 36\text{dB}$ and for other distortion profiles the rate will increase. The theoretically minimized rate occurs at $d_1^{(\text{theor})} = 35.399\text{dB}$, where the overall rate is slightly larger than the practical optimum. The difference between the overall rates calculated at the theoretical and the practical optimum is $\frac{2794635 - 2777778}{2777778} \times 100\% \approx 0.61\%$.

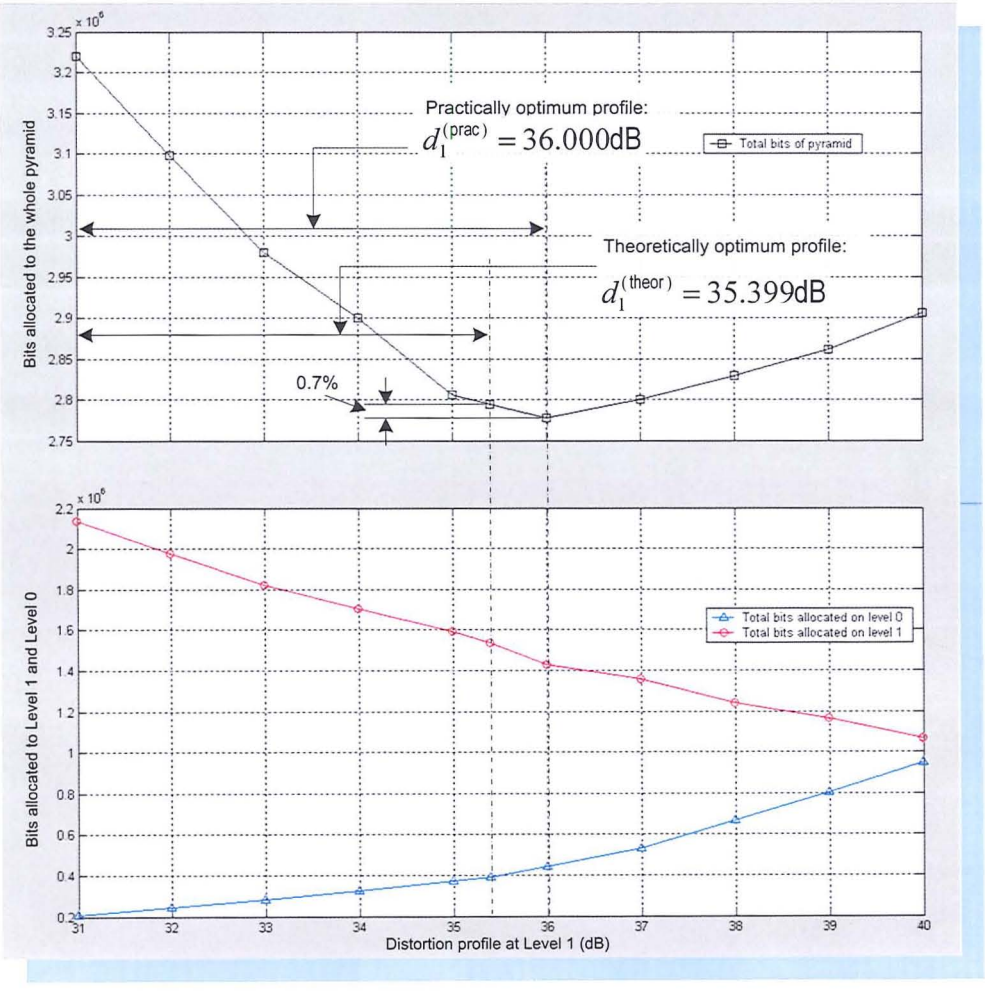


Figure 8-6: The curves of the bits allocated to the l_1, l_0 and the pyramid measured at different distortion profiles (PSNRs) on l_1. Results are from C5 data set at the quality level PSNR=35dB.

From the results in Table 8-1 we noticed that the difference between the overall rates, one is minimized by the proposed inter-level distortion control algorithm and the other is determined from practical coding attempts, is not significant, for different images at different quality levels. Most of the results demonstrate that the former has no more than 1% degradation of the optimality in rate-distortion performance. For instance in B3CGP data set at the quality level PSNR=28dB, the degradation is only 0.07%, which is very close to the practical optimum. Since the practical results are not straightforward, it is necessary to perform multiple coding attempts before one can decide the optimum performance or an automatic searching process will converge to some optimum state, and then its implementation will be very complicated and the computational complexity will be very high. Therefore considering the effectiveness in optimization and the simplicity in computation we can conclude that the proposed rate-minimizing distortion control algorithm is a good candidate for inter-level bit allocation.

The inter-level distortion control algorithm is a model-based approach. It solves the inter-level bit allocation problem by modelling the distribution of the pyramid coefficients, and therefore the solution to the optimum state can be easily and directly calculated, which is its main advantage over those iterative approaches. However when the model is not accurate, its results may be biased from the practical ones. For instance in Table 8-1, we found that the practically optimum distortion profiles on l_1 vary with the different quality levels, which, as discussed in Section 8.1.1, should not be affected by the final quality. Moreover in some cases, such as HUM_GLT_C data set at the quality levels PSNR=35dB and PSNR=37dB, the degradations to the practical optimum is about 2-3%, which are much larger than others. According to the information theory (refer to Section 3.1), the coding performance of a specific source is determined by not only the characteristics of the distribution of the source and also the performance of the compression method in encoding such source. In practice these two factors are difficult to be modelled accurately. Therefore, as it has been discussed in Section 8.1.1, when people [Horn, 1997; Aiazzi, 1997] were investigating the inter-level bit allocation problem, they always accept the following two assumptions. One is that the coefficients in the difference pyramid can be modelled as a zero-mean Laplacian distribution. This approximates the unpredictable source a mathematically

tractable and defined form. The other is that the quantizer on each level is assumed to be working at its optimum performance. This assumption removes the bias comes from the compression method.

However in 3D-CoMIC, both assumptions may be inaccurate. The imperfect band-stop properties of the half-band low-pass filter for downsampling and interpolation and the randomly-distributed quantization noise introduced according to the error feedback scheme will lead to the mismatch between the actual distribution of the coefficients to the theoretical model. On the other hand the proposed cross-band classified encoding strategy is not optimum in encoding the pyramid coefficients. For instance the VQ-based classification scheme does not optimize the classification gain [Joshi, 1997], and the efficiency in capturing the cross-band dependencies by sharing one classification map across the pyramid levels may become weak when the number of pyramid levels becomes large, such as for HUM_GLT_C data set. The inaccuracy in these factors degrades the efficiency in inter-level bit allocation. Fortunately these deficiencies can be alleviated. For instance one can improve the coding performance in classified encoding by optimizing the classification gain, and improve the efficiency in capturing the cross-band dependencies. These efforts will help to improve the accuracy of the actual process to the theoretical model and therefore reduce the bias between the practical optimum and the theoretical optimum.

8.2 Improvements in Computational Complexity of VQ

As we discussed in Section 7.1 VQ-based encoder consists of two parts – the codebook training and codebook searching. The complexities of these two parts can vary greatly according to the different VQ techniques and codebook structures used. For instance, if the codebook is designed on a probabilistic model it can be directly computed mathematically; and if the codebook is associated with a specific training set the training process can be very complicated because the entire training set would be used to obtain a codebook which is optimized for this training set and usually an iterative optimization approach is considered. In codebook searching the structure of the codebook determines the complexity. For instance a binary search is possible in

tree-structured VQ while an exhaustive search (or its variants) is required for unstructured VQ, such as LBG. In CBCVQ, a classified VQ is defined on intra-band coefficients from each level. Therefore for each encoding attempt the codebooks have to be generated from the training set, which ensures the optimality of the codebooks to the training set but with the expense of high computational complexity introduced in training. Moreover because we use an LBG-VQ in each class the complexity of codebook searching is also large.

It is worth noting that, fortunately the classified VQ can reduce the computational complexity in both codebook training and codebook searching. This is because each sub vector quantizer is separately applied on one of the groups of the entire vector set and in each group vectors are about to have similar characteristics, e.g. variations. Therefore in codebook training, like LBG algorithm, the codewords are more likely to converge to the optimum. Furthermore according to the distributional characteristics of pyramid coefficients classification tends to group large amount of small-variation coefficients into a class and large-variation coefficients occupying only a small part into another class. As mentioned before, after intra-level bit allocation the small-variation group usually associates with a small codebook, where the complexities in training and encoding are much less than the large codebook associated with the large-variation group; and the overall complexity is less than that using ordinary VQ on the entire vector set without classification. However the computational complexity can still be very high since the image dimensions dramatically increase in volumetric image data compared to two-dimensional images. For instance, for B×CGP data set, there are about $(1+\frac{1}{8})\times 256\times 256\times 180 \approx 1.33\times 10^7$ coefficients in the pyramid. Even though it is typical to discard 90% of the coefficients using the thresholding technique, there are still a large number of coefficients ($\approx 1.33\times 10^6$) to be encoded by VQ techniques. Therefore we have considered several approaches in reducing the complexity in codebook training and codebook searching.

8.2.1 Improvements in Codebook Training for LBG-VQ

From the intra-band bit allocation algorithm (Algorithm 8.2) we noticed that within each iteration we increase the rate of the selected vector quantizer by doubling its

codebook size from its previous computation. That means the new codebook do not need to be built exactly according to the LBG algorithm (Algorithm 7.2) from a null codebook, while instead it can be built based on the codebook generated in previous iterations. Suppose the quantizer q is working at rate $r^{(i)}$ in current iteration i, and now we are going to increase its rate from $r^{(i)}$ to $r^{(i)}+1$, then the corresponding codebook $C^{(i)}$ can be viewed as the current codebook in Step 3 in LBG algorithm and the new codebook $C^{(i+1)}$ can be computed just by 'splitting' $C^{(i)}$ to get doubled initial codewords and followed by GLA optimization. The new codebook would be exactly the same as we follow the LBG algorithm by starting from a null codebook because in Algorithm 8.2 the codebooks generated prior to the quantizer reaching the current rate are the same as the codebooks temporarily generated at each splitting-and-updating process in LBG algorithm. Therefore each codebook training task in Algorithm 8.2 is actually a GLA algorithm. Furthermore different from using randomly-selected codewords as the initial codebook in conventional GLA [Lloyd, 1982], we can have a good initial codebook from its previously-generated half-sized codebook.

The selection of the initial codebook is the only factor that affects the optimality of the final codebook to the training set, and therefore affects the rate-distortion performance somehow. Apart from using a 'splitting' approach similar to that in LBG [Linde, 1980] we also investigated a hyper-space splitting method to generate a good initial codebook. The former is featured for its simplicity in computation while the latter can improve the optimality compared to the former.

A. Improved Linearly-Splitting Codeword Selection Approach

In LBG method [Linde, 1980], suppose i is the current iteration and after 'splitting' process two new initial codewords are generated from each current codeword. The new initial codewords for the next iteration are computed as:

$$\{y_0^{(i+1)}, y_1^{(i+1)}, \ldots, y_{2j+1}^{(i+1)} \mid \begin{matrix} y_{2j}^{(i+1)} = (1+\omega)y_j^{(i)} \\ y_{2j+1}^{(i+1)} = (1-\omega)y_j^{(i)} \end{matrix}, \quad j = 0,\ldots,i\} \tag{8.22}$$

As to the parameter ω, they suggested using a fixed small number. However we have observed that using fixed number, although is simple in implementation, may raise some problems. For instance using small number when the codebook is not large will make the two new initial codewords cluster together and are far away from other

codewords, and therefore it may suffer from more computations since the codebook is difficult to converge by using GLA; however if we consider a large number, we cannot produce a good initial codebook either when the codebook becomes large, because the distribution of the new initial codewords may not be restricted by the codebook generated in previous iteration. Therefore in our implementation, the value of ω is adapted to the distribution of the previous codebook. Each ω_j is associated with a codeword y_j and computed as the 1/3 of the distance from a codeword y_j to its neighbouring codeword, say $y_j^{(close)}$.

$$\omega_j = \frac{1}{3}\left| y_j - y_j^{(close)} \right| \tag{8.23}.$$

Moreover the new codewords are the linear combination of y_j and $y_j^{(close)}$, not determined only by y_j in (8.22). So in each splitting process the two new codewords are:

$$\begin{cases} y_{2j}^{(i+1)} = y_j^{(i)} + \omega_j \cdot (y_j^{(close)} - y_j^{(i)}) \\ y_{2j+1}^{(i+1)} = y_j^{(i)} - \omega_j \cdot (y_j^{(close)} - y_j^{(i)}) \end{cases} \tag{8.24}.$$

In (8.23) the neighbouring codeword $y_j^{(close)}$ can be defined as the closest codeword to y_j measured with Euclidean distance. However such definition will involve a search of the entire codebook for each computation of ω_j, and therefore the computational cost would be high when the codebook becomes large. Alternatively we define the neighbouring codeword of y_j as the codeword whose distance to the origin is most similar to that of y_j. This approximation is reasonable because as we noticed that neighbouring pixels tend to have similar value in the image and the vectors are likely to distributed along the diagonal of the vector space (refer to Fig. 2-15), the codewords are likely to be distributed along the diagonal line, especially when the codebook size is large. Therefore the distance measure originally in hyper-dimensional space now can be reduced to the measure in one dimension. This property has also been exploited in fast searches of unstructured codebook, such as the pre-quantization process in K-d tree approach [Friedman, 1975]. Now we can describe the approach to find the neighbouring codeword $y_j^{(close)}$: we sorted the previous codebook according to the distance of each codeword to the origin, from

near to far to obtain a sorted codebook $\{ y_j^{(sort)} \}$ and then $y_j^{(close)}$ is selected in the sorted codebook either as the predecessor of $y_j^{(sort)}$ or its successor, depending on whose distance is most similar to that of $y_j^{(sort)}$.

Since the new initial codewords are computed by the distribution of the previous codebook, the complexity of the linearly splitting approach is determined by the size of the codebook only. Suppose the previous codebook has N_p codewords. The complexity of computing the squared Euclidean distance of each codeword to the origin is O(1) ; and then the complexity of sorting the N_p-entry codebook is $O(N_p \log N_p)$; and finally the complexity of computing the two new codewords using (8.24) is $O(N_p)$. Therefore the overall complexity would be about

$$O(N_p \cdot (1 + \log N_p)) \qquad (8.25),$$

or it needs about $O(1 + \log N_p)$ operations for every new initial codeword.

The above linear prediction of the new initial codewords from the previous codebook produces a reasonable distribution of the initial codewords. They will more likely resemble a uniform distribution than those produced by the conventional splitting method in [Linde, 1980], for instance no two initial vectors are over close. This would help the convergence of the GLA. However such linearly-splitting approach may suffer from two problems. Firstly the model we used to compute the neighbouring codeword is not accurate. In fact the way of computing $y_j^{(close)}$ did not take the angle into account, and in some cases $y_j^{(close)}$ may not be the closest vector to y_j and even far away from it, as illustrated in Fig. 8-7(a). Moreover the selection of the initial codewords only considers the distribution of the previous codebook not the distribution of the training set and the positions of the new initial codewords are not restricted. So we may get some *dummy* initial codewords which means no training vectors will be associated with these codewords when doing GLA and these codewords cannot produce reasonable results at all (see Fig. 8-7(b)). To overcome these two drawbacks, a new initial codeword selection approach based on splitting the voronoi region has been proposed.

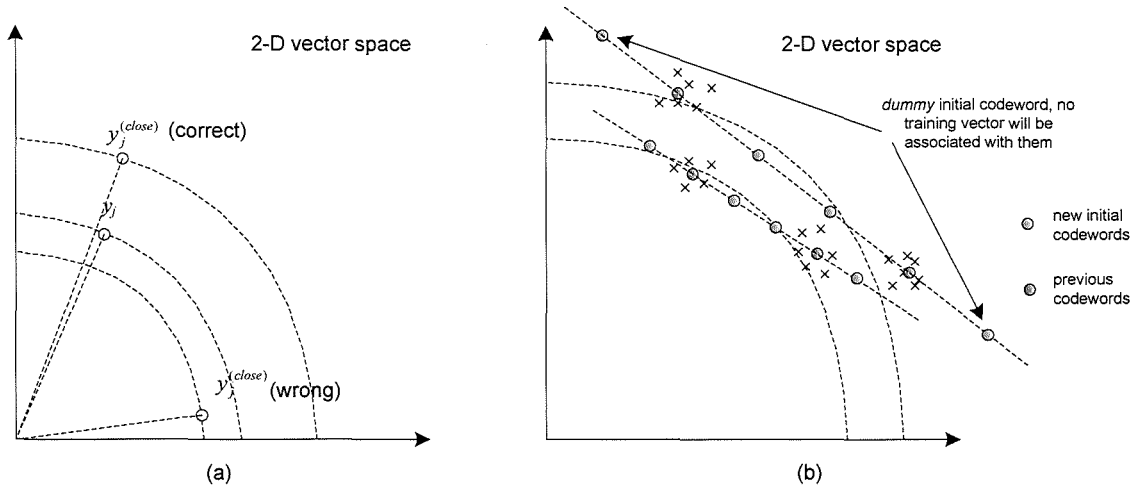


Figure 8-7: Two problems of using linearly-splitting codeword selection approach. (a) Wrong closest neighbouring codeword without considering the angle aspect; (b) Dummy initial codewords due to no restriction on their positions.

B. Initial Codeword Selection by Splitting the Voronoi Region

From Eq. (7.4) we learned that the voronoi region of a particular codeword is defined as the training vectors associated with this codeword according to the nearest neighbour rule. The previous codebook not only defines the distribution of the codewords but also the voronoi region of each codeword. Compared to the above linearly splitting approach, a preferable approach should be able to select initial codewords that are adapted to both the training set and the previous codebook. Therefore a new splitting and codeword selection approach has been investigated, which is based on the statistical characteristics of the voronoi regions.

Suppose we start from the previous codebook $\{ y_j^{(i)} \}$ and we can compute the voronoi region $V_j^{(i)}$ associated with each codeword using (7.4). Then we split each voronoi region using a hyper-plane in such a way that the hyper-plane is orthogonal to the axis of the vector space, along which the vectors belonging to this voronoi region have the maximum magnitude of variance (MMV). We call this axis as A_{mmv} and the corresponding component of each codeword is $y_{j,mmv}^{(i)}$, which is a scalar denoting the projection of $y_j^{(i)}$ onto the axis A_{mmv}. Once we have this hyper-plane orthogonal to

A_{mmv}, we can split this voronoi region into two parts, $V_{2j}^{(i)}$ and $V_{2j+1}^{(i)}$ by comparing the *mmv*-component of each vector belong to $V_j^{(i)}$ with $y_{j,mmv}^{(i)}$:

$$\begin{cases} V_{2j}^{(i)} = \{x \mid x \in V_j^{(i)} \wedge x_{mmv} \le y_{j,mmv}^{(i)}\} \\ V_{2j+1}^{(i)} = \{x \mid x \in V_j^{(i)} \wedge x_{mmv} > y_{j,mmv}^{(i)}\} \end{cases} \quad (8.26).$$

Finally each new initial codeword is the probabilistic centroid of one of the new partitions, such that:

$$\begin{cases} y_{2j}^{(i)} = \text{centroid}(V_{2j}^{(i)}) \\ y_{2j+1}^{(i)} = \text{centroid}(V_{2j+1}^{(i)}) \end{cases} \quad (8.27).$$

An example of this approach is illustrated in Fig. 8-8 on a 2-D vector space.

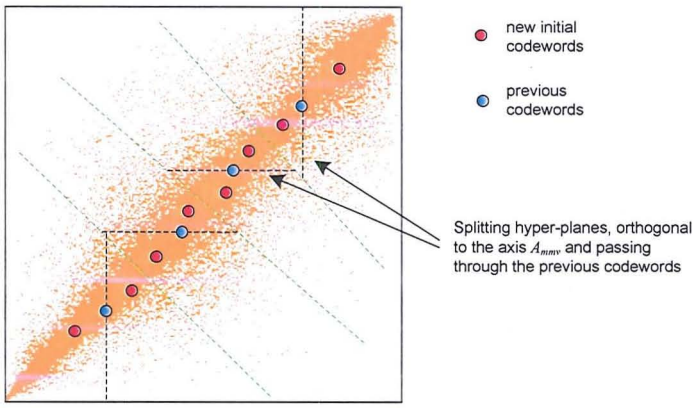


Figure 8-8: An example of hyper-plane splitting the voronoi regions for initial codewords selection.

Such voronoi-region-based initial codeword selection approach has many desired properties. Each voronoi region is computed from the previous codebook and every pair of new initial vectors are from this voronoi region, therefore this approach considers the distribution characteristics of both the previous codebook and the training set. It is preferable to the aforementioned linearly-splitting approach because of no *dummy* initial codewords. Moreover the hyper-plane orthogonal to the maximum-variance axis ensures the two new initial codewords are not too close. As we have discussed this may help the convergence of GLA. Although the two new initial codewords are not optimized for the distribution of the voronoi region (the optimization can be achieved using a two-codeword GLA approach), they are good enough as the initial codewords. Moreover the computational complexity is very low, because the splitting process is basically a floating-point data comparison operation.

As opposed to the linearly-splitting approach the complexity of the voronoi-based splitting method depends not only on the size of the previous codebook but also on the size of the training set. According to the aforementioned process, the complexity can be derived as follows. Suppose the size of the previous codebook is N_p and the training set contains N_t vectors. The computation of each voronoi region involves the searching of the closest codeword for each training vector. Suppose the number of vectors in each voronoi region is N_v ($\sum_{v=1}^{N_p} N_v = N_t$), using the exhaustive search the complexity for each voronoi region is $O(N_v \cdot N_p)$; the complexity in computing the sampled variance of vectors within each voronoi region is $O(N_v)$; and the complexity of the hyper-plane splitting process involves the comparison of a particular component of each vector to that of the centroid vector, and its complexity is $O(N_v)$; and finally the complexity of the computation of centroids of the two new partitions is $O(N_v)$. Fortunately by inspecting the GLA algorithm (Algorithm 7.1) we noticed that the voronoi regions has been computed in Step 2 in the last iteration when we get the final codebook, so the computation of the voronoi regions in this approach can be neglected. Therefore the overall complexity would be

$$\sum_{v=1}^{N_p} O(N_v) = O(N_t) \qquad (8.28),$$

or it needs about $O(\frac{N_t}{N_p})$ operations for every new initial codeword.

8.2.1.1 Complexity Analysis in Fast Codebook Training

In our method the codebook training process can be split into two parts: the initial codebook selection and the GLA applied on the initial codebook. The latter has much higher complexity than the former. Although the voronoi-region-based initial codeword selection approach may have more computation costs than the linearly-splitting approach, for example it is normal that $N_t >> N_p$ in volumetric microscopic image compression, we did not experimentally observe a significant difference of the codebook training time between two training methods using these two initial codebook selection schemes respectively. This is because the major part of the

computation is within the GLA and the difference between the complexities of two initial codeword selection approaches can be neglected. Moreover the voronoi-region-based splitting approach can produce a better distribution of the initial codewords, which will accelerate the converging speed of the GLA and thus reduce the overall training complexity.

Here we investigated the computational complexity and the rate-distortion performance with these two initial codebook selection schemes. They are: (Scheme 1) each quantizer always chooses the voronoi-region-based initial codeword selection approach when it is indicated to increase its rate; and (Scheme 2) each quantizer always chooses the linearly-splitting initial codeword selection approach when it is indicated to increase its rate. Their performance is evaluated on the bottom level of the pyramid coder performed on three different datasets. They are 'B3CGP' dataset, 'C5' dataset and 'HUM_GLT_C' dataset. For each dataset we evaluated on four different rate-distortion performance levels. A 17-class classified VQ was used to encode the bottom-level pyramid coefficients. In the first three performance levels the vector quantizers were working under the distortion constraint. The rate-distortion performance of the two schemes was measured by the rate (the number of bits) allocated on the bottom level, say l_0, at the same distortion level. The distortion-constraint intra-level bit allocation algorithm was used to control the optimum bit allocation among the sub vector quantizers in the classified VQ. In the fourth performance level the vector quantizers were working under the rate constraint, where the vector quantizers in these two schemes were working at the same rate level and the rate-distortion performance was measured by the distortion on l_0. The training speed was measured as the number of seconds performed on a 1.8 GHz Intel Pentium 4 processor executing C/C++ code, which is averaged from three repetitions. The overall results of twelve tests (three datasets × four performance levels), including the training speed and the rate-distortion performance on the bottom level, are reported in Table 8-3.

Table 8-3: The results of computational complexity in codebook training and rate-distortion performance on the bottom level, l_0, with two initial codebook selection schemes evaluated on three datasets at four rate-distortion profiles.

Datasets and R/D profiles		Scheme 1 (voronoi region scheme)			Scheme 2 (linearly splitting scheme)		
		Training time (s)	R/D performance		Training time (s)	R/D performance	
			Bits/Vec	Total bits or distortion		Bits/Vec	Total bits or distortion
B3CGP (291,054 vectors; size = 2×2×3)	PSNR=28dB	8.41	0.59	171,202 bits	9.03	0.60	174,506 bits
	PSNR=31dB	46.11	3.72	1,083,210 bits	47.98	3.78	1,100,732 bits
	PSNR=33dB	149.52	6.52	1,897,218 bits	156.17	6.63	1,929,610 bits
	rate = 10.00 (bit/vector)	397.23	10.00	MSE=23.12	399.06	10.00	MSE=25.45
C5 (235,821 vectors; size = 2×2×3)	PSNR=30dB	6.67	0.42	97,968 bits	7.20	0.43	100,332 bits
	PSNR=33dB	36.84	3.24	764,336 bits	38.17	3.28	773,494 bits
	PSNR=35dB	134.44	6.21	1,465,017 bits	148.02	6.64	1,565,957 bits
	rate = 9.00 (bit/vector)	210.47	9.00	MSE=16.75	208.91	9.00	MSE=17.71
HGC (78,697 vectors; size = 4×4×4)	PSNR=35dB	28.55	0.71	55,738 bits	29.38	0.72	56,810 bits
	PSNR=37dB	137.86	3.93	309,593 bits	151.53	4.04	317,938 bits
	PSNR=38dB	262.77	6.17	485,881 bits	280.16	6.45	507,596 bits
	rate = 10.00 (bit/vector)	473.66	10.00	MSE=7.38	470.92	10.00	MSE=8.49

By comparing the training speed between the voronoi region scheme and the linearly splitting scheme in Table 8-3 we noticed that the difference of training complexity is not significant. More specifically from the fourth performance test for each dataset, where the vector quantizers in both schemes were working at the same rate level, we found that they have very closed complexity in codebook training, where the difference is no more than 1%. According to the discussion in the beginning of this section although the complexity in initial codebook selection between these two schemes may be different, depending on the sizes of the codebook and the training set, the complexity in codebook training is primarily determined by the codebook optimization process using the GLA approach. When compared to the complexity of the GLA method, the amount of complexity gain in the initial codebook selection is too inferior to be considered. On the other hand we noticed an improvement of the

rate-distortion performance in the voronoi region scheme over the linearly splitting scheme. The former produces less distortion than does the latter, or it has higher codebook quality, where the improvement in MSE measure can be up to 15%. This comes from the fact that new codewords, generated from the voronoi region codebook selection approach which considers the distribution of both the training set and the previous codebook, have closer distribution to the finally optimized codebook than those from the linearly splitting approach. Therefore the GLA method is more likely to converge to the optimum results by using the former.

From the results of the first three performance tests we noticed that the linearly splitting scheme is always inferior to the voronoi region scheme in codebook training complexity and the rate-distortion performance for all the three datasets at different quality levels. For instance, for C5 dataset at the quality level PSNR=35dB the degradation of the former to the latter is more than 10% in complexity and 7% in rate, respectively. Such degradation actually comes from the loss of optimality in the distortion-constraint intra-level bit allocation. The linearly-splitting initial codebook selection approach decreases the optimality of the codebook to the training set in the GLA method, and therefore the 'slope' according to (8.14) will not be as accurate as that computed from those higher quality codebooks, for instance the codebooks generated using the voronoi-region initial codeword selection approach, and finally the equal-slope bit allocation algorithm would make those originally high-rate quantizers tend to work at lower rates and those originally low-rate quantizers tend to work at higher rates. This can be verified by comparing the rates allocated to each quantizer between two schemes (see Fig. 8-9). In Fig. 8-8, due to the inaccuracy in computing the maximum-slope quantizer, the linearly splitting scheme has different bit allocation to the voronoi region scheme. In this example it increased the rate of a low-rate quantizer and decreased the rates of two high-rate quantizers. However the overall rate is larger than that in the voronoi region scheme, because it failed to follow the 'optimum' path to the target distortion level. Therefore more bits and computations are 'wasted' in achieving the same distortion level by using the linearly splitting scheme. Here we can conclude that the voronoi region approach is a good candidate in selecting initial codewords. It avoids the degradation of the codebook quality in the codebook training process and consequently ensures the optimality of

the rate-distortion performance in the equal-slope bit allocation process, which is superior to the linearly splitting approach.

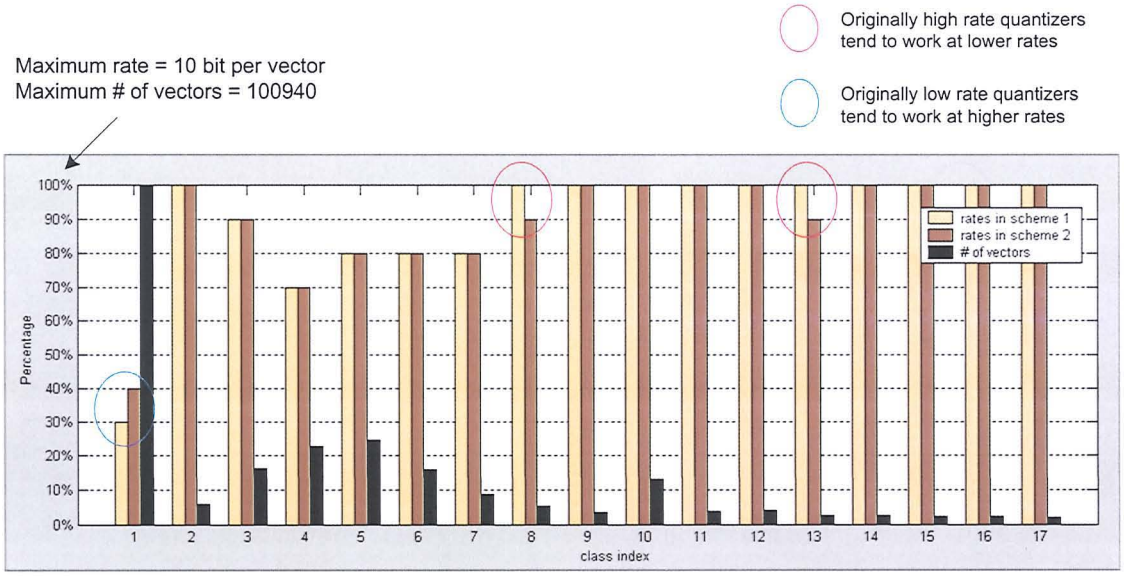


Figure 8-9: The comparison of rates of each quantizer, using scheme 1 and scheme 2 as the initial codebook selection scheme, respectively. The lose of optimality in codebook training in scheme 2 makes those originally high-rate quantizers tend to work at lower rates and those originally low-rate quantizers tend to work at higher rates.

8.2.2 Fast Searches of Unstructured Codebooks

Many techniques have been developed for speeding the exhaustive search (according to the minimum-distortion rule) of an arbitrary codebook C containing N k-dimensional codewords. A variety of these techniques are mentioned in Section 12.16 of [Gersho, 1992]. They can be coarsely categorized into two groups: one is an exhaustive search but usually introducing more constraints to the stopping rule, therefore the originally linear searching space can be narrowed and the overall complexity is reduced; in the other category the search techniques usually begin with a 'coarse' prequantization with very low complexity, and then the following exhaustive search is applied on a subset of the codebook, which is associated with one of the levels in prequantization. For the first category the search space is narrowed by introducing extra constraints but the technique does not abruptly delete any codewords from the searching space. In fact all the codewords can be considered in searching if necessary. Therefore these approaches have exactly the same results as

conventional exhaustive search but they are very likely to have reduced complexity. In contrast, for the second category, codewords are abruptly deleted from the search space by prequantization. Therefore even followed by an exhaustive search these approaches can not guarantee to find the optimum codeword because the optimum codeword may be discarded in prequantization step and would never be considered in the exhaustive search. So we refer to these approaches as 'sub-optimal' search. One of the widely used 'sub-optimal' searching methods is an approach searching in a K-d tree structure [Bentley, 1975; Friedman, 1975]. In this method before encoding each codeword is prequantized and associated with a unique cell according to the prequantization. Each cell is then called a *bucket* containing the indices of the associated codewords. Then to encode a source vector x, one applies the prequantization, finds the cell to which x should be associated, and performs an exhaustive search on the corresponding bucket for the closest codeword to x. As we can see the prequantization determines the performance in optimality of the searching: the more likely a vector is pre-quantized into the same cell to which its globally optimal codeword was associated the less degradation the encoding does.

Additionally we would like to emphasize another technique, called *partial distance search* (PDS) method [Cheng, 1984; Bei, 1985], which has gained widespread acceptance as a more efficient method than the conventional brute force searching. PDS consists of a simple modification to the way that distances are calculated. During the calculation of the distance sum, if the partial distance exceeds the distance to the nearest neighbour found so far, the calculation is aborted. This means we need an additional comparison for each element of the distance sum. In this method the gain of the complexity depends on the distribution of the source vector set and the codewords of the corresponding codebook, therefore it is not very easy to give a mathematical expression of the complexity gain. However in their works, it has been reported that this method substantially decreases the computational complexity. Moreover another contribution of this method is that it can work with other fast encoding methods whenever the minimum-distance searches are involved.

Although the sub-optimal searching is featured for its simplicity in computation, they are not considered in our vector quantization techniques. This is because they suffer

from the degradation in rate-distortion performance. Due to the incapability of accurately predicting the distribution of the training set, it is always a problem to design a good prequantizer that can predict the globally optimal codeword to an input vector x. Moreover consider the intra-level bit allocation, the inaccuracy in codebook searching may induce the inaccuracy in bit allocation and finally decrease the efficiency of the bit allocation for each quantizer and the rate-distortion performance of the overall system. In addition, we have observed that compared to the cost in training the cost in encoding is trivial, for example no more than 5 seconds to encode approximately 300,000 12-D vectors at the average rate range from 6bpp to 7bpp, which is in comparison with more than 100 seconds in training (see Table 8-3). Therefore we are focusing on the exhaustive search and its variants, which have no degradation in the optimality of the training set to the codewords.

A number of fast search techniques in the first category share the similar idea: given a source vector x, they narrow the searching space by effectively and correctly 'kick out' those codewords, which have been recognized as impossible to be the candidates of the closest codeword to x, according to the previous and current search states [Motoishi, 1984; Vidal, 1986]. The implementation of this technique can be described in the following two steps.

A. Pre-processing

In this part we need to precompute the relations of each codeword to the others, which could be further used in the next part. The Euclidean distances between all pairs of codewords are precomputed and stored in a pair-distance table (see Fig. 8-10). Consider the codebook C has N codewords, each element $T_{i,0}$ in the first column is the index of each codeword y_i. Then for the i^{th} row, starting from the second element $T_{i,1}$ the table stores the Euclidean distances of other N-1 codewords to the codeword $T_{i,0}$, ordered from closest to furthest, and also the index of these codewords. Totally there are $N \times N$ elements in the table and for each element the component $T_{i,j}$.index indicates the j^{th} closest codeword to codeword y_i and $T_{i,j}$.dis indicates the Euclidean distance from codeword $T_{i,j}$.index to y_i.

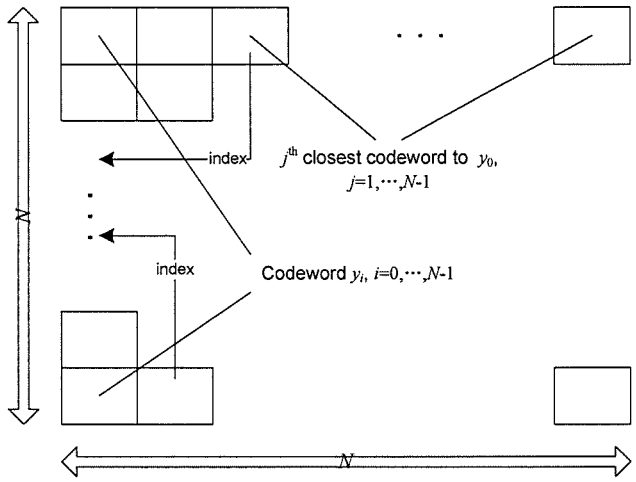


Figure 8-10: An illustration of the pair-distance table, storing the Euclidean distances between all pairs of codewords.

B. Fast Searching the Codebook

Given a source vector x to quantize, some initial codeword $\tilde{y}$ is chosen. Then all codewords y_i whose distance from $\tilde{y}$ is greater than $2\|x-\tilde{y}\|$ are eliminated from further consideration because they cannot be closer than $\tilde{y}$. Those not eliminated are successively compared to x until one that is closer than $\tilde{y}$ is found, which then replaces $\tilde{y}$, and the process continues. In this way the set of potential codewords is gradually narrowed. Moreover since we have precomputed the distance between each pair of the codewords the core computation determining which codeword should be eliminated involves only table look-up operations. Therefore the complexity is reduced compared to the conventional linear search of the codebook. More specifically, we describe the fast searching process in the following algorithm.

Algorithm 8.3: Fast search the optimal codeword in an unstructured codebook according to the nearest-neighbour rule

Step 1. (Choose an initial codeword) Given an input vector x, we choose an initial codeword from the first column in the pair-distance table $T_{i,0}$.index and we refer this codeword as $\tilde{y}$, which assumed to be the closest codeword to x so far. The choice of the initial codeword affects the efficiency of the searching. This issue will be investigated later. Set an index j=1, which indicating the position of the potential closest codeword to x in i^{th} row.

Step 2. Compute the distance between x and $\tilde{y}$: $d_{\min} = \|x - \tilde{y}\|$.

Step 3. If the following conditions are met we stop the search and output $\tilde{y}$ as the closest codeword to x; otherwise we compute the distance between x and $T_{i,j}$.index ($d_{\text{pot}} = \|x - T_{i,j}.\text{index}\|$) and continue to Step 4.

Condition 1: The distance between the current potential codeword $T_{i,j}$.index and $\tilde{y}$ is greater than the doubled distance between x and $\tilde{y}$: $T_{i,j}.\text{dis} \geq 2d_{\min}$.

Condition 2: $j>N$-1, which indicates that we have searched all the elements in i^{th} row.

Step 4. (We found a codeword which is closer than $\tilde{y}$) If d_{pot} is smaller than $d_{\min}$, we replace $\tilde{y}$ with $T_{i,j}$.index, reset j=1 and go back to Step 2; otherwise continue to Step 5.

Step 5. (Sequential search on the sorted list $T_{i,\cdot}$) Increase j by one and go back to Step 3.

In the first termination condition the idea behind is that because in each row of the pair-distance table the codewords are organized in the ascending order according to their distance to $T_{i,0}$.index, once we have found at j^{th} column that the distance between the $T_{i,0}$.index and $T_{i,j}$.index is greater than the doubled distance between x and $T_{i,0}$.index the codeword $T_{i,j}$.index is impossible to be a closer one to x than does $T_{i,0}$.index, and neither for its successions (as illustrated in Fig. 8-11). The explanation to the second termination condition is straightforward: before j becomes greater than N-1, the algorithm must have already compared the distance between $T_{i,j}$.index (j=1,…,N-1) and x to that between $T_{i,0}$.index and x and found $T_{i,0}$.index is the closest codeword to x.

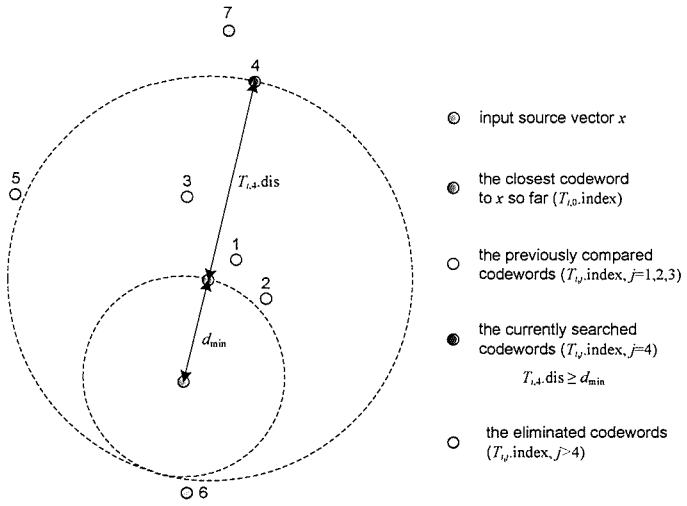


Figure 8-11: An example of the explanation to the first termination condition.

The selection of an initial codeword in Step 1 will affect the efficiency of the searching algorithm – the closer the initial codeword to the potential closest codeword to x the faster does the algorithm find the closest codeword. Conventionally a large number of initial codeword selection approaches are based on the results of the previously coded neighbouring vectors. The initial codeword is predicted from the codewords which have been associated to the neighbouring vectors of x encoded previously. This is based on the assumption that neighbouring vectors tend to have similar patterns and so do their associated codewords. However in case of subband coding, the correlations of neighbouring vectors within the band are not as strong as those in original image domain. Therefore the prediction from neighbourhoods is not assumed to be a good scheme for pyramid coefficients. Alternatively we are aware that coefficients in difference pyramid resemble a Laplacian distribution and the histogram of them has a strong peak at origin (zero value), as illustrated in Fig. 6-12. Therefore when we are considering a group of coefficients as a vector in a high-dimensional vector space, we can assume the distribution would resemble a Laplacian distribution projected onto a series of concentric hyper-spheres centered at the origin of the vector space, such that a high proportion of vectors are clustered near the center of these spheres and the further the distance to the center the more sparse the vectors are distributed. So we can follow the same technique as described in Eq. (8.24) to compute the 'closest' initial codeword such that its distance to the origin is most similar to that of the input vector x.

8.2.2.1 Complexity Analysis in Fast Codebook Searching

As we known, the conventional exhaustive search involves a linear search of the entire codebook for each source vector. Considering a codebook with size N_c is associated with a source vector set having N_s vectors, its computational complexity can be expressed as:

$$\mathrm{O}(\text{conv.exh.search}) = \mathrm{O}(N_s \cdot N_c) \qquad (8.29),$$

or $\mathrm{O}(N_c)$ per source vector. For the fast searching approach we discussed above, the encoding task includes two parts: the pre-processing (building pair-distance table) and fast encoding. In some works [Motoishi, 1984] they reported the overall computational complexity of this approach by only considering the second part. Such judgement was reasonable because they assumed a general codebook which is not designed for a specific training set. In this case the building of the pair-distance table can be viewed as an off-line task and do not need to be considered in the overall complexity. However in our situation since the codebooks are specifically designed in each coding attempt, we are using specific codebooks rather than a general codebook. Therefore we have to consider the complexity of the first part into the overall complexity.

In the first part, building the pair-distance table consists of two steps. One is the computation of Euclidean distance of each possible pair of the codewords. The complexity of this step is $\mathrm{O}(N_c^{\,2})$. The second step is for each codeword we get a sorted list of other codewords according to their distance to this codeword. The complexity of this step is $\mathrm{O}(N_c^{\,2} \log N_c)$ if we consider a 'quick sort' implementation. Therefore the complexity in the first part is:

$$\mathrm{O}(\text{fast search - building table}) = \mathrm{O}(N_c^{\,2} \cdot (1 + \log N_c)) \qquad (8.30).$$

While in the fast encoding part it is difficult to give an explicit expression of the complexity because encoding path depends on the distribution of the codewords and the source vector set. From experience it has been reported that the fast searching can achieve approximately 10 times speed up for each source vector in average [Vidal, 1986]. Therefore the overall complexity of the fast search approach is about

$$\text{O(fast search)} \cong \text{O}(N_c^{\,2} \cdot (1+\log N_c) + \frac{1}{10} N_s \cdot N_c) \qquad (8.31).$$

By comparing (8.31) with (8.29) we noticed that although the search process is much faster in the fast searching approach, if we take into account the complexity of pre-processing the overall complexity of the fast searching approach may not be persistently preferable to the conventional linear search approach, especially when the training set size is not very large relative to a large-sized codebook. This can be verified by the following experiment, where we pairwise compared the encoding time of the conventional linear search approach and the fast searching approach with different codebook sizes. The source vector sets (the training set) contains approximately 20,000 6-D vectors sampled from the top-second pyramid level applied on 'B3CGP' data set. The size of the codebook varies from 2 to 4096 by doubling the codebook size every time, and all the codebooks are specifically designed for the training set using LBG algorithm. The complexity of encoding are measured by averaging the execution time (in milliseconds) of 10 repeats on a 1.8 GHz Intel Pentium 4 processor executing C/C++ code. One should note that the partial distance search (PDS) method has been considered in the implementations of both the conventional linear search and the fast encoding. The results are reported in Table 8-4.

Table 8-4: The comparison results of codebook searching complexity using the conventional linear search approach and the fast searching approach, respectively. The source vector set contains approximately 20,000 6-D vectors. PDS method has been considered in the implementations of both approaches.

Codebook sizes	[1] Time in conv. linear search approach (ms)	Time in fast searching approach (ms)		
		[2] Building table	[3] Fast searching	[4] Overall
2	17.2	<0.1	13.3	13.3
4	20.3	<0.1	16.8	16.8
8	26.6	<0.1	17.2	17.2
16	37.5	<0.1	21.9	21.9
32	57.8	0.3	29.7	30.0
64	90.6	1.4	48.0	49.4
128	159.4	6.3	72.3	78.6
256	323.5	28.1	113.5	141.6
512	642.2	124.5	161.8	286.3
1024	1317.2	545.6	217.5.	763.1
2048	2789.0	2373.2	324.2	2697.4
4096	8937.5	10256.3	679.1	10935.4

From the column [1] we found the variation of the computation time did not follow the Eq. (8.29) exactly, which should be doubled when we doubled codebook sizes. This is because the effects of using partial distance search method. As we can see when the codebook size is small the codewords are sparsely distributed, and then when computing the distance sum the calculation is more likely to abort with fewer elements considered. Therefore the encoding time of PDS-integrated linear search approach does not increase in the way as indicated in (8.29) when the codebook sizes are small. However when the codebook becomes larger, the distribution of codewords is much denser and the density is relative to the distribution of the source vector set (the training set). It will take into account more elements before the computation of the distance sum stops since for a given source vector there are more codewords having similar distance to it than the case when the codewords are sparse. Therefore the complexity gain from the PDS method may become marginal when the codebook size is large.

Comparing the above discussion with the fourth column of Table 8-4, we can see the proposed fast searching approach decreases the complexity in nearest neighbour searching. Although the factors of the complexity gains of the fast search technique for small codebook sizes are not as much as those when the codebook becomes large, this is a bias caused by introducing the PDS method in both approaches. Therefore when the effect of PDS method becomes weak, such as the codebook size is 2048 and 4096, the fast search can achieve as high as about 10 times gain in complexity over the linear search. However as we discussed previously in our implementation the overall complexity of the fast searching approaching has to consider not only the complexity in the pure fast searching but also the complexity in computing the pair-distance table. From the third column of Table 8-4, we noticed the complexity of building table increases approximately 4 times when we doubling the codebook size. When the codebook becomes large its complexity can be so great that it becomes much larger than the complexity of the pure fast searching and finally makes the overall complexity of the fast searching approach exceed the complexity of the conventional linear search. Due to the effect of PDS, it is difficult to determine a criterion mathematically, indicating when the fast searching approach becomes inferior to the conventional linear search. Therefore the choice between these two approaches needs more consideration on not only the size of the codebook and the source vector set, but also their distribution characteristics. In general when the ratio of the source vector set size to the codebook size is large, the fast searching approach is preferable to the conventional linear search; however when this ratio becomes small, the latter maybe more efficient since the computation cost in building the pair-distance table in the former approach could be very high.

Chapter 9 System Performance Evaluation: A Comparison Study

According to the discussion in Section 4.3, we are expecting our compression system, the 3D-CoMIC approach, to have the following desired features. They are: 1) distortion-constraint compression, 2) three-dimensional encoding adapted to volumetric image data and 3) competitive rate-distortion performance and mitigating artefacts at low rates. To evaluate these aspects a comparison study has been performed by comparing the 3D-CoMIC approach with the standard and the state-of-the-art approaches reviewed in Chapter 4. The testing data are all volumetric image data sets collected from CLSM devices, which have been described in Chapter 2. The evaluations are objective. Due to the difficulties in practice subjective evaluations have not been considered. However some tentative efforts have been made in this area and they will be discussed as one of the future works in the last chapter.

This chapter starts with the discussion of comparing two compression schemes: distortion-parameterized approaches, such as the 3D-CoMIC, and the rate-parameterized approaches, such as the JPEG2000 and SPIHT. The results indicate that the distortion-parameterized approaches are more desirable in image quality control compared to the rate-parameterized approaches. In the second part three volumetric image data compression schemes, 2D-based frame-by-frame approaches, multi-frame approaches based on video coding and pure 3D volumetric compression

approaches, will be considered in the evaluation of rate-distortion performance in volumetric image data compression. Their results validate the hypothesis that in volumetric image data compression the real 3D approaches outperforms the other two schemes. In the last part of evaluation the performance of several state-of-the-art real 3D approaches, such as the methods based on 3D-DCT and 3D-DWT, are compared with our 3D-CoMIC approach. The objective measures of the reconstructed images using these methods show that the 3D-CoMIC approach has similar or even better performance compared to these approaches. Moreover at low rates, the artefacts are alleviated in our method, which may be more visually acceptable.

9.1 Distortion-Control Performance

By reviewing the discussion in Section 3.1 we have learned that an image compression approach can be designed as either distortion-parameterized or rate-parameterized. Methods from these two categories have different properties and applications. If the target file size and the bit rate are more critical, for instance in the applications like telecommunications, a rate-parameterized approach may be preferable, since the compression system can be easily adapted to the change of the target rate. However if the target quality of the reconstructed image is a more important concern, for instance in the applications in medical imaging, we can expect a distortion-parameterized approach to be more advantageous, since the target quality can be achieved with much less efforts than using the rate-parameterized approaches.

Moreover it has been recognized that the accuracy in the control of these two aspects depends on the efficiency of bit allocation algorithm of a method. The smaller the possible minimum amount of bits involved in one allocation process, the higher the accuracy of the outcome rate (or distortion) to the target rate (or distortion). For example some methods, such as JPEG2000 and SPIHT, have received considerable attention for their accurate control in bit rate. It is achieved by allowing a small amount of bits, even one bit in SPIHT method, in each allocation process. The rate is gradually increased until it meets the target rate. Therefore the control of rate in these methods can be very accurate. Similarly in our 3D-CoMIC approach, the control of the reconstructed image quality has been simplified as the control of the encoding

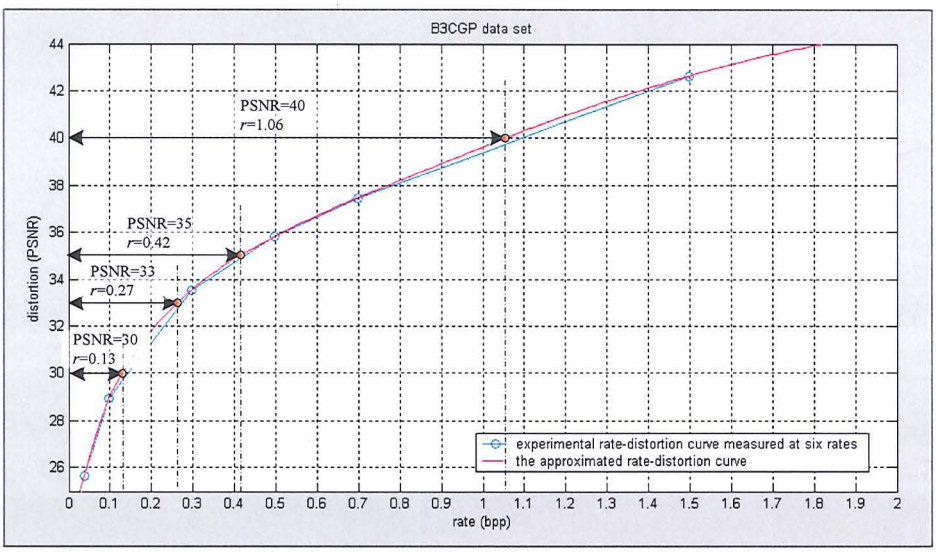
quality on the bottom level. Furthermore the intra-band bit allocation scheme allows the minimum amount of allocated bits to be associated with only one sub vector quantizer in a classified VQ. In each allocation process the magnitude of decrement of the overall distortion is determined by one sub vector quantizer when its rate is increased from r to r+1. The distortion is gradually decreased until it meets the target distortion. Therefore it can accurately control the distortion since the magnitude of decrement of the overall distortion in each bit allocation process is small.

9.1.1 Evaluation on Distortion-Control Performance

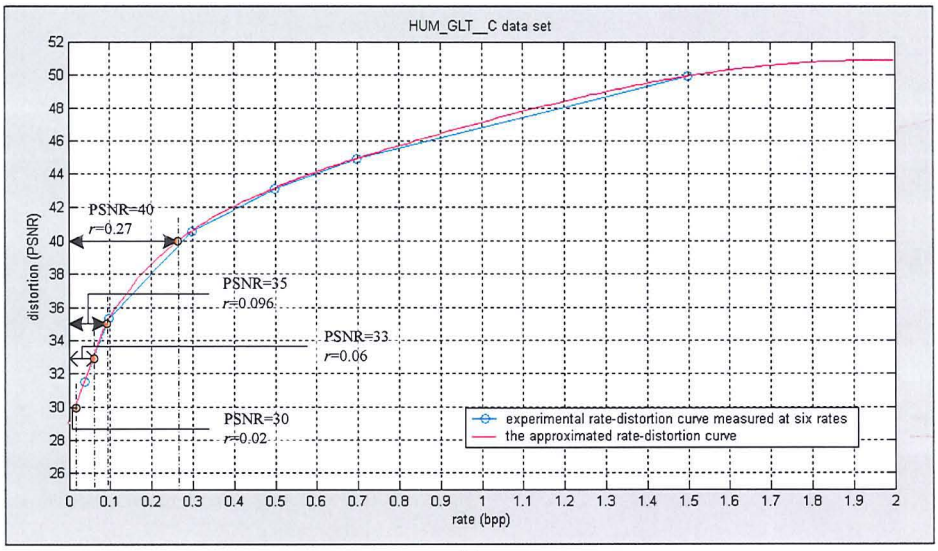
To examine the distortion-control performance of the 3D-CoMIC approach, an experiment has been made on compression of two data sets at four quality (distortion) levels. The two testing data sets are B3CGP data set from B×CGP group having many detail and structural information (see Section 2.3.1.1) and HUM_GLT_C data set from HUM_GLT_× group having large area of smoothed regions (see Section 2.3.1.4). The four quality profiles are: PSNR = 30dB, 33dB, 35dB and 40dB, which are corresponding to MSE=65.03, 32.85, 20.72 and 6.50, respectively. A comparison study has been performed by comparing our 3D-CoMIC method with another three-dimensional volumetric image data coder, the 3D-SPIHT[1]. In 3D-CoMIC we can straightforwardly specify the target distortion as one of the above four target distortion profiles. However we noticed that 3D-SPIHT is a basically rate-parameterized method and does not support distortion-control functionality. To make it work at a particular target quality we have to try multiple compression attempts and manually decide the rates corresponding to this quality. This was achieved in the following way. Thanks to its high accuracy in rate control, for each data set six rate-distortion points are firstly computed at six different rates, they are rate = 0.04bbp (bit-per-pixel), 0.10bpp, 0.30bpp, 0.50bpp, 0.70bpp and 1.50bpp. Secondly a rate-distortion curve is approximately fitted according to the above six experimental rate-distortion points, as illustrated in Fig. 9-1. In the next we manually determine the approximate rates on the curve which are corresponding to the target PSNRs and finally delicately adjust the rates to make the outcome PSNRs approach to the desired

[1] The executable program and the source code of 3D-SPIHT can be download at: http://www.cipr.rpi.edu/research/SPIHT/spiht3.html

values as close as possible, for instance falling into the range of ±2% of the target value.



(a)



(b)

Figure 9-1: The rate-distortion curves using 3D-SPIHT method on, (a) B3CGP data set and (b) HUM_GLT_C data set, approximated according to six experimental rate-distortion points. These curves will be used to guide the selection of rates corresponding to the target distortion.

Table 9-1 listed the actual PSNRs using two methods measured at four PSNR levels for each data set. We also indicated, for each PSNR level, the number of encoding attempts performed before the compressor can output the desired distortion, where for the rate-parameterized methods, such as 3D-SPIHT, it consisted of the averaged

number of attempts in computing the rate-distortion curve and the number of attempts in refining the PSNR value.

Table 9-1: The evaluation of the distortion-control performance of the 3D-CoMIC approach compared with 3D-SPIHT at four quality levels.

Data set	Target PSNR (dB)	3D-CoMIC		3D-SPIHT	
		Actual PSNR	No. of attempts	Actual PSNR	No. of attempts*
B3CGP	30	30.00	1	29.96	1.5+1
	33	33.00	1	33.09	1.5+1
	35	35.13	1	35.03	1.5+1
	40	40.10	1	40.02	1.5+3
HUM_GLT_C	30	30.30	1	30.31	1.5+4
	33	33.16	1	33.18	1.5+2
	35	35.09	1	34.88	1.5+3
	40	40.20	1	40.04	1.5+1

*: In 3D-SPIHT the number of encoding attempts consists of two part – one is the averaged number of attempts in computing the rate-distortion curve, which is 6/4=1.5 in this experiment; and the other is the number of attempts in refining the PSNR value when the approximated value is not accurate.

From Table 9-1 we noticed that in the 3D-CoMIC approach, thanks to the closed-looped pyramid coding structure and the distortion-constraint intra-level bit allocation algorithm, the outcome distortions were effectively constrained by the desired distortion levels. In fact the actual distortions are very close to the user-specified targets, where the magnitude in difference is no greater than $\frac{|30.30-30|}{30}\times 100\% = 1\%$ measured in PSNR Moreover the desired quality is achieved in just one encoding attempt and the entire encoding process is straightforward and automatic without any manual intervention. Additionally as it has been recognized that in the intra-level bit allocation scheme the allocation process will not stop until the distortion becomes no greater than the distortion profile on this level, so for the bottom level we can always expect that the actual distortion is no greater than the one specified. According to the error-feedback scheme we can derive that the overall distortion will not be greater than the target. This can be seen in this experiment that the actual PSNRs are always

higher than the targets. This property is very desirable that we can always meet the quality requirements since the outcome distortion is strictly upper-bounded by the target distortion level. Therefore we can see that the 3D-CoMIC is very efficient in distortion-control performance.

However in the rate-parameterized methods, such as 3D-SPIHT, the control of the distortion is not straightforward but redundant. Actually multiple encoding attempts and manual intervention are required. For instance, to determine the rate corresponding to the target distortion level, one needs to pre-compute the rate-distortion curve for a particular image data set, then carefully adjusts the approximated rate and tries the encoding and decoding processes multiple times until the outcome distortion becomes enough close to the target. Another problem is that if the approximated rates are in the range where the rate-distortion curve is well fitted with the experimental rate-distortion points, these rates could be accurately associated with the target distortions and one is likely to achieve the target quality in fewer attempts, for instance the results of encoding attempts using 3D-SPIHT on B3CGP data set. However if the approximated rates are out of such range, the inaccuracy of curve fitting may mislead the determination of the accurate rates, and therefore increase the complexity in searching the correct rates. For instance when we were encoding the HUM_GLT_C data set using 3D-SPIHT at PSNR=30dB quality level, the rate approximated from the curve was 0.02bpp, however after 4 attempts we found the correct rate corresponding to this target quality was 0.029bpp. Although all these tasks can be implemented as an automatic approach, the complexity in practice would be very high since there is no straightforward indication such that at which rate the system can meet a particular quality requirement. In addition when working with those rate-parameterized methods, for instance 3D-DCT, where the control of the rate is not very accurate, the situation may be more severe. This is because in these methods the actual outcome rate might be biased from the expected target rate and so they are inaccurate in rate control performance. Such inaccuracy could become 'noise' that deteriorates the efficiency of determining the 'correct' rate corresponding to the target distortion because the coder is not capable of working exactly at the expected rate, and therefore these approaches are very likely to perform more repetitive encoding and decoding attempts than those with accurate rate control.

Therefore we can draw the conclusion that the distortion-parameterized method, such as 3D-CoMIC, outperforms the rate-parameterized method, such as 3D-SPIHT, for its flexibility and simplicity in meeting a quality demand.

9.2 Performance in Volumetric Image Compression using Difference Multi-frame Coding Schemes

Confocal laser-scanning microscopy devices can produce a volumetric image of a specimen in three dimensions. In Chapter 2 we have learned that due to the characteristics of optics and the scanning techniques the imaging properties are not the same in all directions. More specifically the volumetric image can be viewed as a specimen projected onto a three-dimensional grid and the spatial characteristics of this grid is determined by the optics and the imaging techniques. In CLSM imaging, such grid is not isotropic; however the spatial characteristics within the focal plane differ from those across the focal planes. Generally the volumetric image is collected and represented in a form as a sequence of two-dimensional images such that each image is a projection of the specimen on the focal plane at a particular penetration depth. Therefore the encoding of a volumetric image is equal to the encoding of this image sequence.

Three volumetric image compression schemes have been considered in compressing a volumetric image. They are 2D-based frame-by-frame approaches, multi-frame approaches based on video coding and pure 3D volumetric compression approaches. In the first scheme each image in the sequence is individually encoded by a two-dimensional image coder without considering the inter-frame correlations between the frames. The approach considered for the evaluation of this scheme is the JPEG2000[2] image coder. As it has been reviewed in Section 4.1.2.1, JPEG2000 has so many desired properties, such as accurate control of rate and high rate-distortion performance, that it has been proposed as the next generation of the current image coding standard, the JPEG.

[2] The JPEG2000 is implemented in Anything 3D – JPEG2000 Compressor™, which is developed by *Anything 3D Corp.* (http://www.anything3D.com).

In the second scheme we considered a multi-frame approach resembling a video coder. In Section 4.2.1 we have discussed the application of video coders in the three-dimensional medical image compression, where a video coder takes the entire sequence as the input and tries to capture the inter-frame dependencies using the motion compensation technique. Although there are a large variety of video coders in use, they share the same core technique, the motion compensation, and actually each coder is just trying to adapt itself for a particular application. For instance MPEG-1 is designed for TV quality movie with image size as 352×288 and MPEG-2 is designed for studio quality movie, where the image size is 720×576. Therefore since there is no video coder specifically designed for CLSM volumetric imaging applications, we finally choose the MPEG family[3] as the video-based multi-frame approach for evaluation. This is because MPEG is a very successful example of 'motion compensation' and it has been integrated in most video editing and creating applications.

Real volumetric image data compressors have been considered in the third scheme. As opposed to the above two schemes, the input in this scheme is three-dimensional image data (a sequence of images is stacked into a volumetric image data) rather than a sequence of two-dimensional images. The processing and the encoding are in three-dimensions, which are designed to capture the inter-voxel dependencies representing the real physical system being imaged. Therefore they are assumed to be a better scheme for volumetric CLSM images than the frame-by-frame approaches and the motion-compensation-based video coders. We considered a state-of-the-art 3D-Wavelet compressor, the 3D-SPIHT, which has good rate-control performance.

9.2.1 Evaluation on Multi-frame Coding Schemes

In Section 2.5 we made a hypothesis that the gain in coding efficiency by exploiting the inter-frame dependencies depends on the distribution characteristics of volumetric image data along all three dimensions. In this experiment we considered two volumetric CLSM image data sets. One is the 'C5' data set from the 'C5' group and

[3] The MPEG-1 and MPEG-2 are implemented in VideoMach™, which is developed by *Gromada Corp.* (http://www.gromada.com/videomach.html).

the other is the 'HUM_GLT_C' data set from the 'HUM_GLT_×' group. From Table 2-1 we can see that these two data sets have different pixel dimensions, for instance each voxel in 'C5' represents a volume as 1μm×1μm×1μm of the specimen while in 'HUM_GLT_C' data set the represented volume size is 0.6μm×0.6μm×1μm. Therefore they have different distribution characteristics. Such characteristics along each direction can be statistically described by the two-dimensional histograms of the cooccurence of the neighbouring pixels introduced in Section 2.5.1.2. Following the same techniques we illustrate from Fig. 9-2(a) to Fig. 9-2(c) the histograms of the neighbouring pixels in HUM_GLT_C data set, which are sampled along x, y and z directions, respectively. We also indicate the correlation coefficients of the two neighbouring pixels in each direction. By comparing Fig. 9-2 with Fig. 2-15 we can see that the differences in pixel dimensions in two data sets result in the differences in their statistical characteristics. For instance the inter-frame correlations in HUM_GLT_C data set is weaker than that in C5 data set.

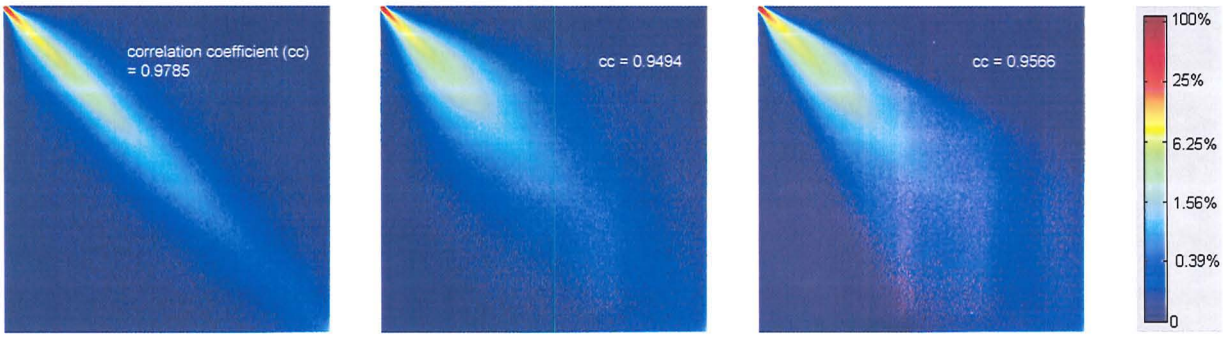
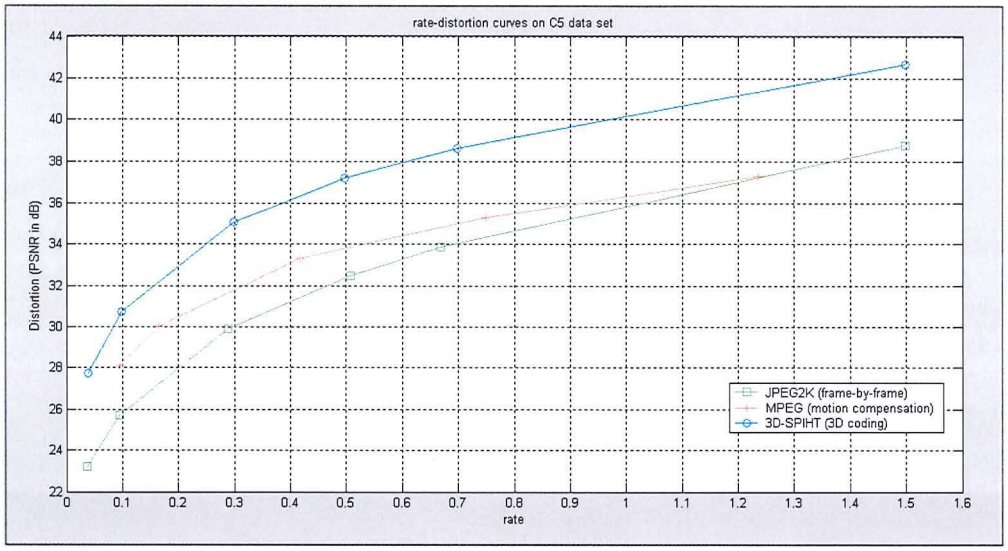


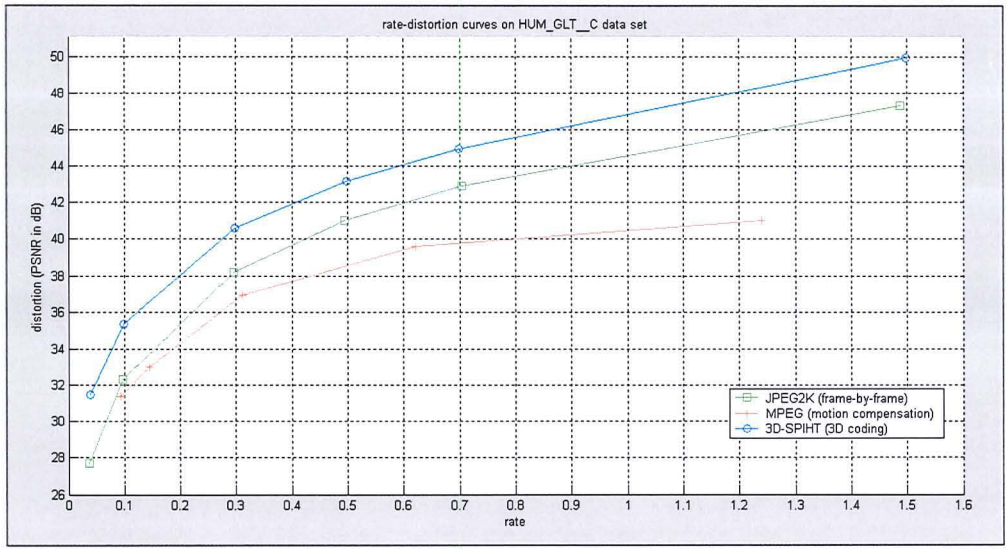
Figure 9-2: Two-dimensional histograms of the cooccurence of two neighbouring pixels from 'HUM_GLT_C' data set, which are formed by sampling along (a) x-direction, (b) y-direction and (c) z-direction. The histograms are normalized by dividing the maximum value and displayed in logarithm scale.

We noticed that the three approaches, JPEG2000, MPEG and 3D-SPIHT, are all rate-parameterized. However due to the lack of flexibility in the compression programs we cannot operate at an arbitrary rate. Therefore, in the comparison study, rather than comparing the distortions of each method at the same rate level we would like to compare their rate-distortion curve over a particular rate range. The rate range in this experiment is selected as from 0.04bpp to 1.5bpp. The rate-distortion curve is computed from the rate-distortion performances at several rate levels within the rate

range. The rate-distortion curves of three methods on C5 and HUM_GLT_C data sets are plotted in Fig. 9-3(a) and Fig. 9-3(b), respectively.



(a)



(b)

Figure 9-3: The experimental rate-distortion curves of three methods, JPEG2000, MPEG and 3D-SPIHT, which are evaluated on (a) C5 data set and (b) HUM_GLT_C data set.

In Fig. 9-3 we can see that the real three-dimensional coding method, 3D-SPIHT, consistently outperforms the 2D-based frame-by-frame method, JPEG2000 and the MPEG video coder based on motion compensation techniques, from low rates (0.04bpp) to high rate (1.5bpp). Even though the 3D-SPIHT and JPEG2000 are both based on wavelet transform coding, their performances one with and one without

exploiting the inter-frame dependencies could be very different. For instance in the C5 data set the PSNR values of 3D-SPIHT is approximately 5 dB higher than those of JPEG2000 measured at the same rates. In another word from the Fig. 9-3(a) the gain in compression ratio using the real three-dimensional coding approaches is about 2 times at high quality levels, i.e. PSNR=38dB, and as much as 5 times at low quality levels, i.e. PSNR=28dB.

Moreover by comparing the results of the MPEG approach with those of others we noticed that it is possible to improve the rate-distortion performance by considering the inter-frame correlations. For instance, in MPEG although the motion-compensated frames are encoded using DCT based techniques, where the DCT-based methods have been recognized having much poorer rate-distortion performance than the DWT-based methods, the deficiency can be alleviated due to the consideration of inter-frame correlations. Therefore in some cases, for instance the C5 data set, the coding gain from this consideration can make a DCT-based video coder outperform a high-performance DWT-based 2D image coder (see Fig. 9-3(a)). However compared with the real three-dimensional coding approaches the video coders cannot have a comparable performance to these approaches. This is because the causative process in the CLSM image and the video imaging is fundamentally different. In CLSM imaging when the optical plane moves along the optical axis orthogonal to the focal plane, the change of the light density has different characteristics to that when the object is moving parallel to the focal plane in video imaging. Therefore although the video coder considers the inter-frame dependencies its object-moving model does not fit the three-dimensional volumetric data in CLSM and the motion compensation technique cannot capture the inter-frame correlations very well.

We also found that the gains of coding efficiency by considering the inter-frame correlations depend on the distribution characteristics of a data set along its z direction. In the C5 data set voxels are more correlated along the z direction than those in the HUM_GLT_C data set. Therefore from the rate-distortion results in Fig. 9-3 we can see that in the HUM_GLT_C data set the gains from the real three-dimensional coding methods and the video coders are not as much as those for the C5 data set,

when compared to the 2D-based frame-by-frame methods. For instance, here JPEG2000 is superior to MPEG.

Another advantage of the inter-frame methods is that they tend to smooth the difference of qualities between individual frames throughout the sequence by gaining the qualities of the high-detailed images, which are sensitive to the human visual system, from the penalties of degrading some qualities of the low-detailed images, which are less-sensitive to the human visual system. As we will see this property comes from considering an efficient inter-frame bit allocation scheme, which is how a method would control the bit allocation on each frame in the situation of coding an image sequence. We noticed that in the 2D-based frame-by-frame scheme, there is no consideration of the inter-frame bit allocation, since each frame is individually encoded and no information from the encoding processes of the previous frames is used to guide the bit allocation of the encoding process of the current frame. Therefore the rate-distortion performance on each individual frame is only determined by the distribution characteristics of the pixel values of this frame, such that when working at the same rates the encoder may produce a high quality for a low-detailed image and produce a poor quality for a high-detailed image. However in the other two schemes, the cross-frame dependencies have been exploited and the bits are allocated either for a group of images, e.g. in MPEG, or for a block consisting of voxels from both intra-frame and inter-frame, e.g. in 3D-SPIHT. Therefore the distortion levels throughout the entire sequence in these schemes tend to be smoother than those in the frame-by-frame scheme. It can be observed in Fig. 9-4 illustrating the variation of distortion levels of each individual image throughout the whole sequence, where the C5 data set was compressed at the same distortion level (PSNR=30dB) using these three schemes.

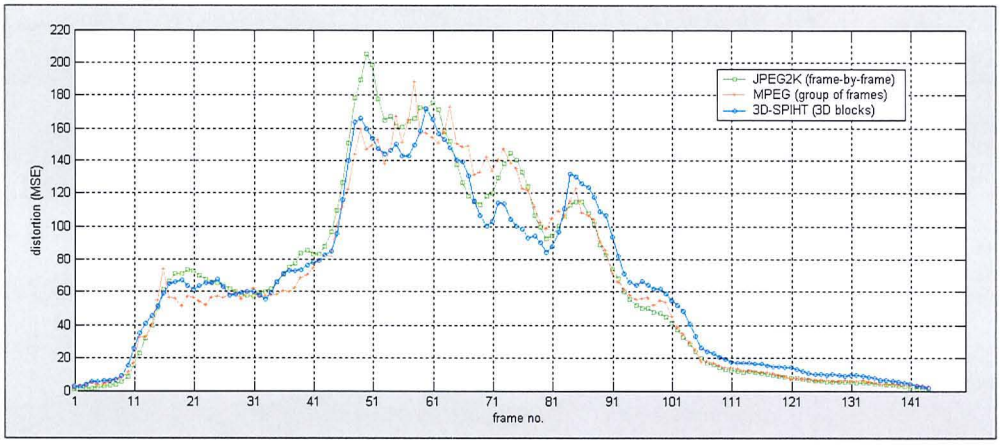


Figure 9-4: The variation of distortion levels of each frame throughout the C5 data set. The data set is compressed at the same distortion level, PSNR=30dB, using the three multi-frame coding schemes.

We can see that for images having many details, e.g. from frame 48 to frame 82, the distortion levels using 3D-SPIHT is smaller than those using JPEG2000; while for less-detailed images, e.g. from frame 83 to frame 144, the distortion levels using the 3D-SPIHT become larger than those using the JPEG2000. This indicates that in the real three-dimensional data compression methods the bits are allocated in a way that the gains in the decrement of distortions for detailed images are achieved from some penalties of distortions from those less-detailed images, and consequently they tend to smooth the distortion levels throughout the sequence. Such smoothness has an important advantage that it can improve the visual quality of the whole data set. This is because the detailed images are featured for strong structural information, such as edges, which are more sensitive to the human visual system and the decrease of distortion can be easily observed. While human visual system is less-sensitive to the smoothed regions on a less-detailed image and their penalties of distortion are not likely to be detected, and therefore the overall visual quality could be improved [Cockshott, 2000]. For instance in Fig. 9-5(a) the 3D-SPIHT produces a higher-visual-quality image for the 50^{th} frame (high-detailed image) than the others; while for the 99^{th} frame (low-detailed image) in Fig. 9-5(b) it suffers from a worse quality in objective measure, but it is not visually apparent.

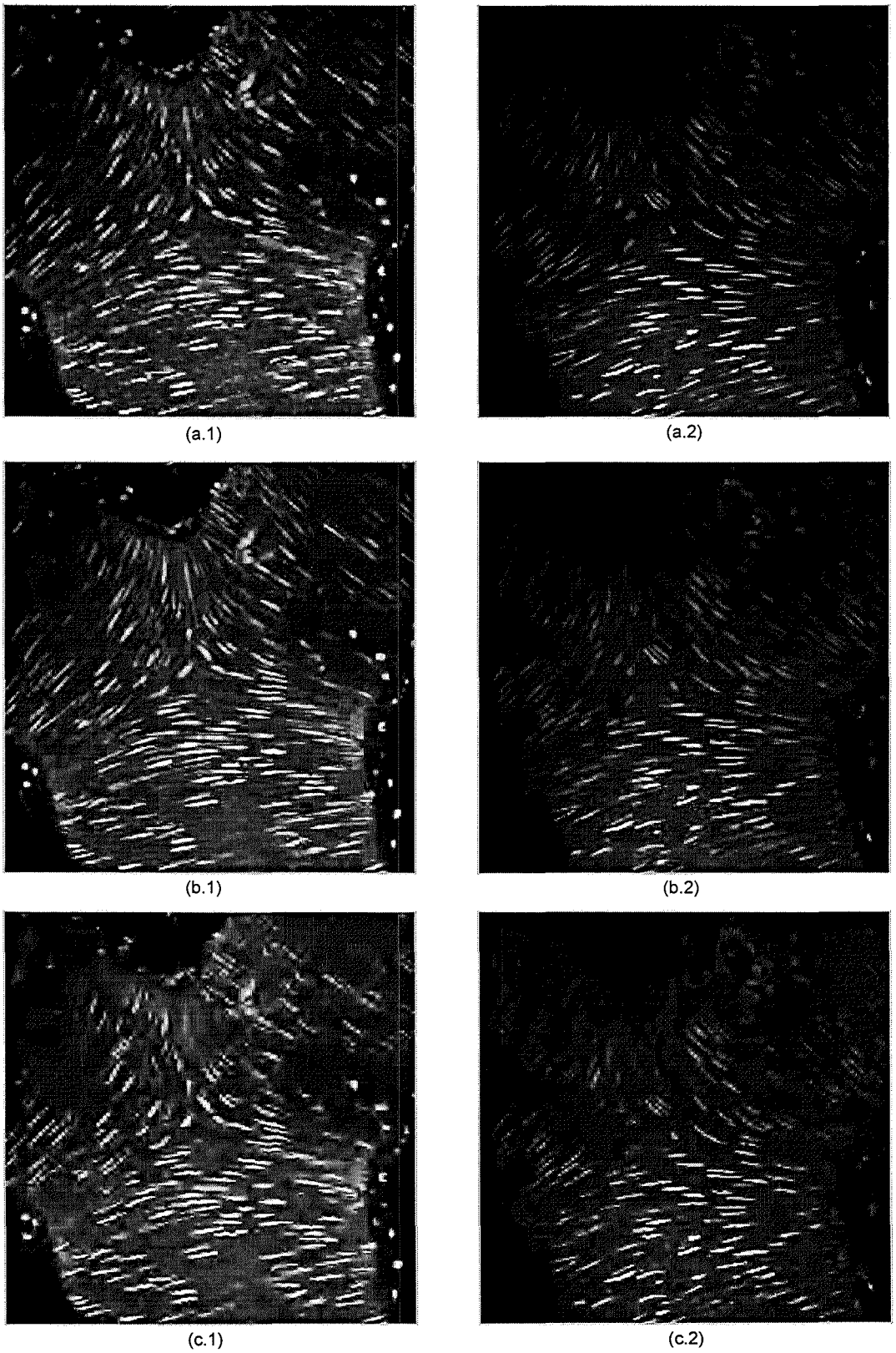


Figure 9-5: The reconstructed images of (1) the 50th frame and (2) the 99th frame using (a) 3D-SPIHT, (b) MPEG and (c) JPEG2000. Three methods have been configured to work at the same distortion level of the whole data set (PSNR=30).

9.3 Performance in Three-dimensional Encoding of Volumetric Image Data

In the last part we evaluate the coding performance of the 3D-CoMIC approach compared to the other three-dimensional approaches. We have shown that 3D approaches have advantages over the 2D image compressors and video compressors in volumetric image compression. Although there is no standard for volumetric image compression, a large amount of intensive efforts has been made in this area. As reviewed in Chapter 4 there are two state-of-the-art techniques considered in three-dimensional approaches; one is based on three-dimensional discrete cosine transform (3D-DCT) and the other is based on three-dimensional discrete wavelet transform (3D-DWT). In the following comparison study we choose a 3D-DCT[4] coder for the first category, which is implemented by Schelkens in [Schelkens, 2003]; for the second category we choose the 3D-SPIHT coder.

We evaluated the distortion-rate performance of these three approaches on three CLSM data sets. They are 'B3CGP' data set, 'C5' data set and 'HUM_GLT_C' data set. The first two datasets are characterized by high-variation regions and structural information, while the HUM_GLT_C dataset has much less details. The performances were measured at four different distortion profiles: PSNR=30dB, 33dB, 35dB, and 40dB, which are corresponding to MSE=65.03, 32.85, 20.72, and 6.50, respectively. Due to the inability to control distortion in the 3D-DCT and the 3D-SPIHT, the target distortions were achieved using the same technique as described in Section 9.1.1, where a rate-distortion curve is firstly approximated from several experimental rate-distortion point and then manually determine the rates corresponding to these target distortions and finally, if necessary, gradually refine the rate until the outcome distortion is close enough to the target.

In the 3D-CoMIC method, the number of pyramid levels has been determined such that the image size on top level is 32×32 in the x-y plane. So we built a four-level

[4] The source code and the program of 3D-DCT can be acquired by emailing the author on: Peter.Schelkens@vub.ac.be

pyramid for B3CGP and C5 data sets; while a five-level pyramid for HUM_GLT_C data set. Refer to the description in Section 7.2 we use a lossless coder, e.g. DPCM for the top level, which contains the lowest-frequency components of the original image signal. The cross-band classified vector quantizer (CBCVQ) works on all the other levels containing high-frequency components corresponding to different scales. In CBCVQ, for the smallest-scaled level, we choose codebook size N_c for the first stage in MSVQ as 16, therefore we have a 17-class classified VQ on the other levels. The high-variation/low-variation threshold T_v on this level is 60. For all the other parameters, including the magnitude of the threshold to classify the coefficients into the significant and insignificant groups, the size of the vector and the classification block on each level, we have noticed that these parameters are sensitive to the distribution characteristics of the signal. For instance according to the discussion in Section 7.2.3 it is possible to capture the higher-order inter-voxel correlations by increasing the vector size. Therefore considering the balance between the coding performance and the computation complexity we empirically choose these parameters as the following, which are listed in Table 9-2.

Table 9-2: Specifications of the 3D-CoMIC approach on B3CGP, C5 and HUM_GLT_C data sets.

Data set	Level	Thresholding		Classification		Vector size
		Thr. value	Block size initial size (IS) node size (NS)	# of class (T_v=60)	Block size	
B3CGP (4-level pyramid)	l_0	0	IS: 4×4×6 NS: 2×2×3	N/A	N/A	N/A
	l_1	1	IS: 4×4×6 NS: 2×2×3	17	2×2×3	2×2×3
	l_2	5	IS: 4×4×6 NS: 2×2×3	17	4×4×6	2×2×3
	l_3	13	IS: 8×8×12 NS: 2×2×3	17	8×8×12	2×2×3
C5 (4-level pyramid)	l_0	0	IS: 4×4×6 NS: 2×2×3	N/A	N/A	N/A
	l_1	1	IS: 4×4×6 NS: 2×2×3	17	2×2×3	2×2×3
	l_2	4	IS: 4×4×6 NS: 2×2×3	17	4×4×6	2×2×3
	l_3	11	IS: 8×8×12 NS: 2×2×3	17	8×8×12	2×2×3
HUM_GLT_C (5-level pyramid)	l_0	0	IS: 4×4×4 NS: 2×2×2	N/A	N/A	N/A
	l_1	1	IS: 4×4×4 NS: 2×2×2	17	2×2×2	2×2×2
	l_2	1	IS: 8×8×8 NS: 2×2×2	17	4×4×4	2×2×2
	l_3	4	IS: 16×16×16 NS: 4×4×4	17	8×8×8	4×4×4
	l_4	15	IS: 16×16×16 NS: 4×4×4	17	16×16×16	4×4×4

In Table 9-3 we list the actual PSNRs and corresponding rates for three methods measured at four desired PSNRs for each data set. Similar to Table 9-1 we also indicated the number of encoding attempts performed before the compressor can output the desired distortion. The distortion-rate measures of the three approaches on each data set are illustrated in Fig. 9-6, which gives a better presentation of the distortion-rate performances between them.

From the distortion-rate curves computed on each data set having different distribution characteristics, we can see that the overall rates generated from the 3D-CoMIC method are always smaller than those generated from 3D-DCT and 3D-SPIHT when we measured at the same distortion level. One should note that the parameters listed in Table 9-2 are not optimized for the distortion-rate performance, which means a higher coding performance could be achieved with considerations on these parameters. But we can conclude that the rate-distortion performance of 3D-CoMIC is comparable to the results obtained from these two state-of-the-art 3D image coders. Moreover consistent with previous discussions we observed that distortion-control performance of 3D-CoMIC is superior to those rate-parameterized methods, where the target distortion can be achieved in only one attempt using 3D-CoMIC, however multiple encoding attempts and manual intervention are required in the other two methods. This situation may become worse if the method are not capable for accurate rate control functionality, for instance we noticed that more attempts have been performed in 3D-DCT method.

Table 9-3: The distortion-rate measures of the three methods for each data set.

Data set	Target PSNR (dB)	3D-CoMIC		3D-SPIHT		3D-DCT	
		Actuall PSNR (No. of Attempts)	Rate (bpp)	Actuall PSNR	Rate	Actuall PSNR	Rate
B3CGP	30	30.00 (1)	0.12	29.96 (2.5)	0.13	30.07 (4.5)	0.21
	33	33.00 (1)	0.25	33.09 (2.5)	0.27	33.02 (4.5)	0.39
	35	35.13 (1)	0.39	35.03 (2.5)	0.42	34.97 (3.5)	0.58
	40	40.10 (1)	1.04	40.02 (4.5)	1.08	39.96 (4.5)	1.33
C5	30	30.03 (1)	0.08	29.97 (2.5)	0.09	29.99 (3.5)	0.14
	33	33.00 (1)	0.16	33.01 (2.5)	0.18	33.02 (3.5)	0.23
	35	35.02 (1)	0.28	35.08 (2.5)	0.30	34.93 (4.5)	0.34
	40	40.02 (1)	0.90	40.01 (3.5)	0.95	40.18 (4.5)	1.01
HUM_GLT_C	30	30.30 (1)	0.026	30.31 (5.5)	0.03	30.09 (6.5)	0.06
	33	33.16 (1)	0.05	33.18 (3.5)	0.06	32.95 (4.5)	0.10
	35	35.09 (1)	0.083	34.88 (4.5)	0.09	35.56 (4.5)	0.16
	40	40.20 (1)	0.263	40.04 (2.5)	0.27	40.37 (4.5)	0.38

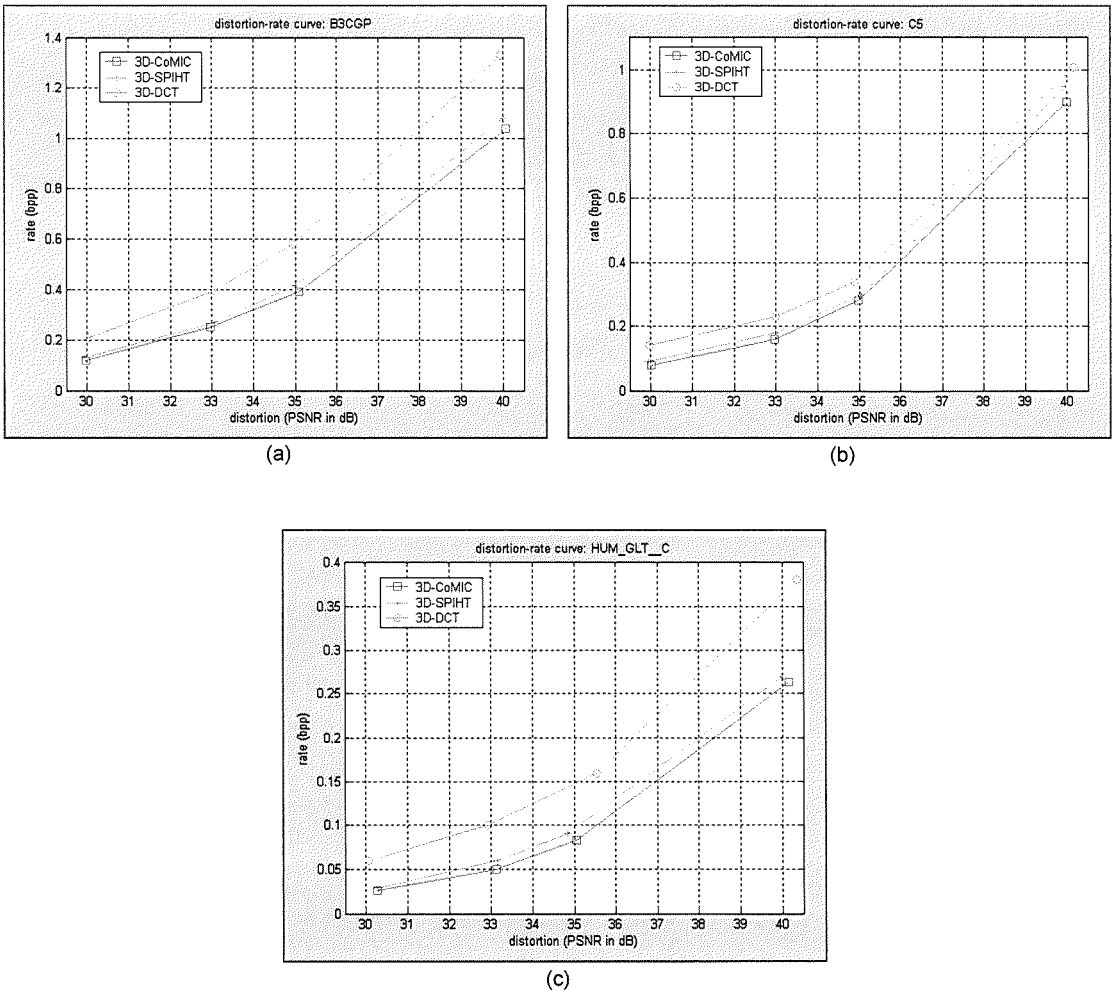


Figure 9-6: Distortion-rate curves of the three methods, 3D-CoMIC, 3D-SPIHT and 3D-DCT, computed on (a) the B3CGP, (b) the C5 and (c) the HUM_GLT_C data set, respectively.

Comparing 3D-CoMIC with 3D-SPIHT we observed that they yield close coding performance. However both methods performs better than 3D-DCT. For instance, in the HUM_GLT_C data set, the rates generated from 3D-CoMIC and 3D-SPIHT are approximately half of those generated from 3D-DCT when measured at the same distortion level. This is because, as it has been discussed in [Castleman, 1995], the multi-resolution representation and subband decomposition techniques are more suitable for image signals than the discrete-cosine transform. In fact image signals are characterized for transient signal components, of which the nonzero values occur only during a short interval and they are highly localized in spatial position. Such components do not resemble any of the DCT basis functions, and therefore they are not represented compactly in the transform coefficients. This makes the discrete-

cosine transform a less than optimal representation for compressing images containing transient or localized components.

Additionally when looking at the decompressed images we noticed that these methods suffer from different types of artefacts and presented different visual quality at same objective measurement, say PSNR, especially at low bit rates. Fig. 9-7 illustrates the 31st frame from the HUM_GLT_C data set compressed at quality PSNR=30 by three methods respectively. We also indicate the histogram of the pixel intensity for each image. We noticed that the histogram using the 3D-CoMIC method is closer to that of the original image than those using other two methods. Actually all these three methods exhibit blurring artefacts. However, additionally, the ringing artefacts, recognized as unexpected high frequency components, is apparent in 3D-SPIHT within the smooth regions around the nuclei; and in 3D-DCT the shapes of the interesting regions, such as nuclei, are distorted by blocking artefacts. These two types of artefacts are less acceptable since they are more obvious to the human visual system than the blurring artefacts.

In Fig. 9-7 we see that due to the blurring the contrast of the reconstructed image using 3D-CoMIC is not as high as that in the original image. For instance, from the histogram, we can see that the percentage of the intensity value 255 in Fig. 9-7(b) is less than that in the original image. However this problem could be alleviated by the post processing on the decompressed images. For instance in Fig. 9-8 we give an example of the post-processed results on the images in Fig. 9-7, where the image contrasts have been enhanced by the histogram equalization operation. In Fig. 9-8(b) the effects of blurring artefact on the contrast-enhanced image is not as apparent as those from ringing and blocking artefacts. However in Fig. 9-8(c) and Fig. 9-8(d) the ringing and blocking artefacts have deteriorated after contrast enhancement – the high-frequency ringing components and the discontinuities between the adjacent blocks are more apparent than they were before the contrast enhancement. Therefore we can conclude that 3D-CoMIC can effectively alleviate the artefacts on the reconstructed image when working at low rates, because its blurring artefacts have less severity than the ringing and blocking artefacts to not only the human visual system but also the image processing routines afterwards [Cockshott, 2003].

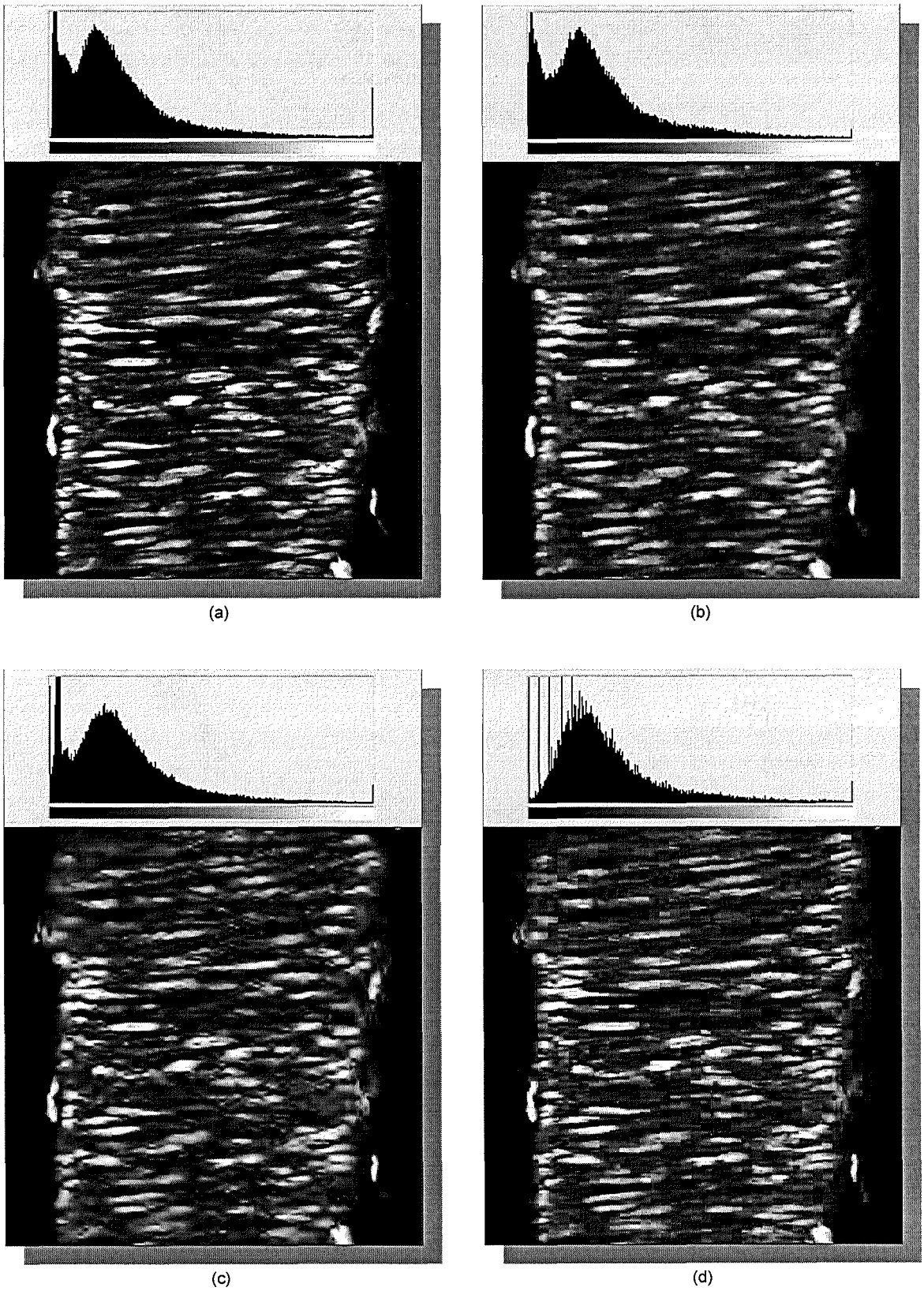


Figure 9-7: HUM_GLT_C data set compressed at PSNR=30dB, (a) The original image, (b) using 3D-CoMIC, (c) 3D-SPIHT and (d) 3D-DCT. A region of 330×330 from the 31st frame is illustrated.

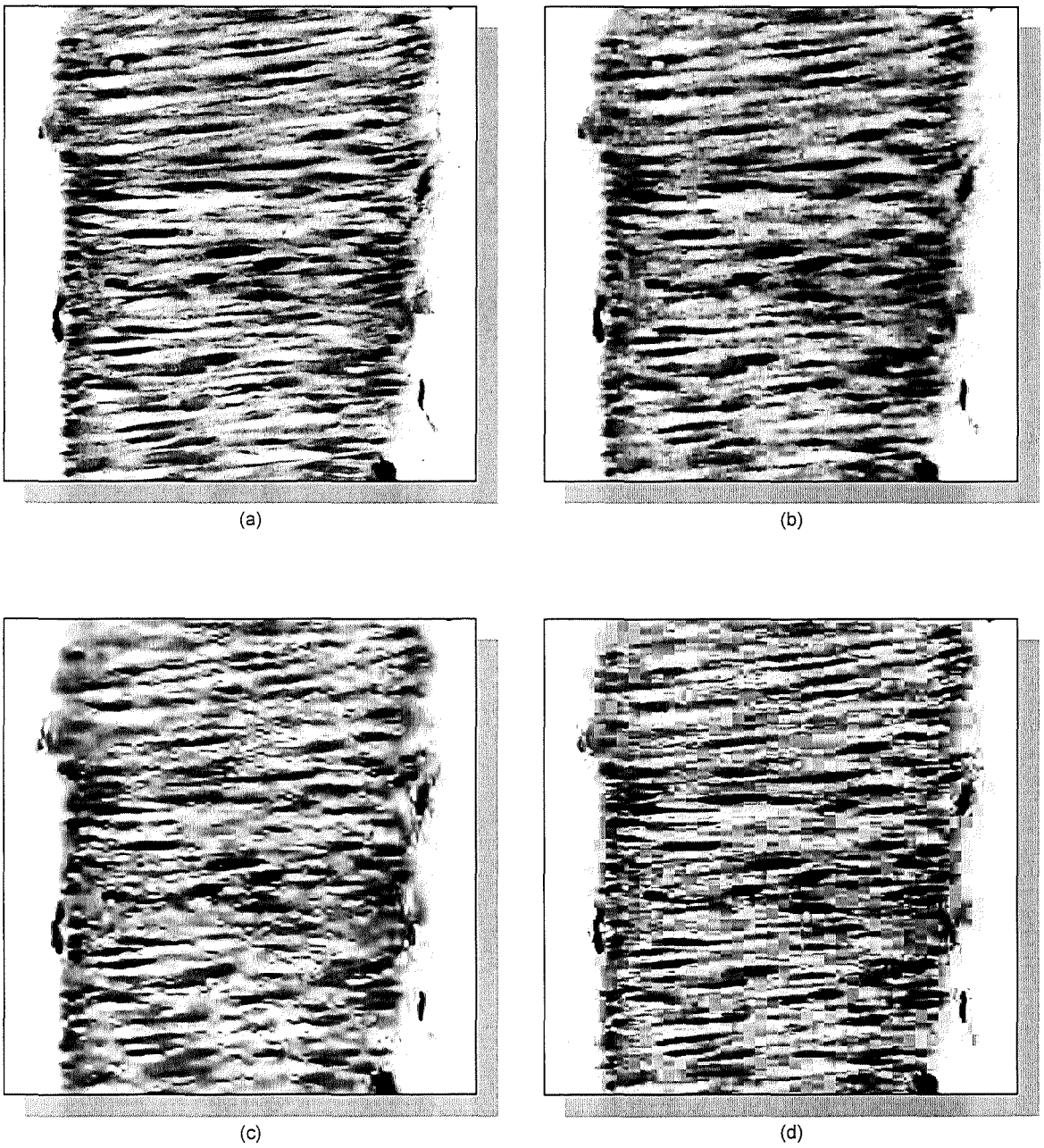


Figure 9-8: The contrast-enhanced results of the images in Fig. 9-7 after histogram equalization operation, (a) The original image, (b) using 3D-CoMIC, (c) 3D-SPIHT and (d) 3D-DCT. The image is displayed as its inverse for better visualization.

Chapter 10 Conclusions and Perspectives

In this thesis, we employ a closed-looped image pyramid structure combined with vector quantization techniques to develop a novel distortion-constraint compression system, 3D-CoMIC, which is well suited for compression applications in three-dimensional medical imaging, such as CLSM imaging. The significance of the distortion control is allowing the achievement of an arbitrary user-specified target quality. As has been indicated in the literature controlling image quality plays a more important role in medical environments than controlling compression ratios, however in the standard and even the state-of-the-art image coders the process of controlling the image quality is difficult and inconvenient. For instance neither JPEG, based on DCT technique, nor JPEG2000, using the latest wavelet coding techniques, is capable for such a compression task specified by a particular quality requirement. This is because the properties of the basis functions in these transforms obscure the relation of the distortion measures between the original image domain and the transformed domain. They are more likely to be designed as a rate-parameterized method where they can easily work at a particular rate requirement. However due to the difficulties in modelling the distribution of an image signal, it is impossible to predict the outcome distortion of a reconstructed image from its corresponding compression rate. Therefore if they are forced to work at a particular quality the compression process would be very inconvenient and repetitive, which increases the costs in computation.

We solve the distortion control problem in these rate-parameterized methods by investigating a closed-looped image pyramid structure and propose a distortion-parameterized compression system, which is able to conveniently and accurately control the outcome distortion according to the user specified quality requirement. Moreover, we propose a cross-band classified vector quantizer (CBCVQ), which is particularly adapted to capture the dependencies among the transformed coefficients in the image pyramid. We also investigate the intra-band and inter-band bit allocation problems in the closed-looped image pyramid structure and develop two bit allocation algorithms. They automatically and adaptively determine the parameters of each inter-correlated sub system to optimize the overall compression performance. These efforts to improve rate-distortion performance make our approach have excellent coding performance in three-dimensional image compressions, comparable to the state-of-the-art approaches in the literature. We conclude our contributions in this thesis as follows.

10.1 Contributions

10.1.1 Distortion-Constraint Compression

We investigate the effect of the encoding-error feedback in the closed-looped image pyramid coding from the view of distortion control performance. Such an error-feedback scheme makes it possible that the distortion measured in the image domain is directly related to that in the transformed domain. Here the expected distortion in the image domain can be explicitly derived from the quality of the reconstructed coefficients from the bottom level of the pyramid. So our method can accurately work at a particular distortion level. This solves the problem in other transform-coding approaches, such as DWT, where the distortion relation between the image domain and the transformed domain is obscure.

Coding algorithms designed in the image domain can support distortion-constraint compression as well. For example, both quad tree and vector quantization techniques can be directly applied on image domain and they can be designed as a distortion-parameterized compressor. However they cannot achieve as good rate-distortion performance as those designed in the transformed domain. This is because the

transformation can decorrelate the inter pixel dependencies at high orders, and the coefficients are organized in a more compact form to achieve higher coding efficiency. However as has been indicated in the literature capturing the high order inter-pixel dependencies in the image domain requires more costs in computation and storage, which may be impractical when the dimension order becomes very large [Shannon, 1959]. Therefore the advantages of using the closed-looped image pyramid structure are: on one hand to support distortion-constraint compression functionality; and on the other hand to achieve a high rate-distortion performance by implementing subband coding, which has been recognized for its efficiency in representing image signals.

Distortion-constraint compression opens a variety of possible research orientations in medical image compression. For instance it is possible to implement a near-lossless, or even lossless, compression, which is very desirable for diagnosis purpose in medical imaging since they can free from artefacts and will not affect the diagnostic accuracy. Moreover although currently we choose a simple objective quality measure, the MSE, to evaluate the distortion, other more sophisticated distortion measures, such as those considering the human visual model, can be directly integrated into our approach. This is due to frequency analysis in pyramid decomposition: it captures not only the frequency dependencies in the frequency domain, but also those important geometric characteristics, such as edges, in the image domain. Therefore a distortion-constraint compression in the transformed domain which has considered a human visual model could improve the visual quality of the reconstructed image.

10.1.2 High Rate-Distortion Coding Performance in Three-dimensional CLSM Image Compression

A high rate-distortion performance is achieved from the following three major considerations. A three-dimensional image pyramid transform captures the inter-pixel correlations in three dimensions and performs a multi-resolution subband decomposition to represent the image signal in a more compact form. Then a cross-band classified vector quantizer exploits the inter-band and intra-band dependencies

from the coefficients. Finally bit allocation algorithms automatically determine the parameters of each sub system to optimize the overall coding performance.

The coding efficiency of a compression algorithm generally depends on how well it can exploit the properties of the source signal. In CLSM the correlations between frames in the image sequence are neither independent nor resemble an optical flow in motion sequence, instead they describe a three-dimensional image of the real specimen and the inter-frame pixels are correlated just as those intra-frame pixels are. We therefore regard the image sequence as a three-dimensional image data and use a three-dimensional image pyramid transform to capture the inter-pixel correlations, no matter whether they are arranged as inter-plane or intra-plane. This makes our approach a pure three-dimensional image data compressor. The evaluation results in Section 9.2 verify our hypothesis that such three-dimensional compression algorithm is superior to the frame-by-frame approaches and motion-compensation-based video coders, since its model gives a more accurate representation of the real physical system being imaged.

To improve the coding efficiency of the pyramid coefficients, we also investigated the distribution characteristics of the coefficients in the image pyramid structure. Two dependencies have been recognized. They are non-stationary distribution of intra-band coefficients and the similar distribution of inter-band coefficients along the same orientation. We employ a classified vector quantizer (CVQ) to capture the former type of dependencies. It is especially suited for a non-stationary source, for instance the pyramid coefficients from intra-band, where the number of large-variance coefficients is usually much less than that of small-variance coefficients. The experimental evaluation on CLSM data sets shows that it is possible to achieve twice the compression ratio by using the classified VQ compared to the conventional VQ when they are working at the same distortion level. The idea, that is sharing the classification map across the bands to exploit the second type of dependencies, is simple, but effective. It avoids the overhead bits required to representing the classification map on each level; and on the other hand can approximate the classification information of the lower levels from that generated on the smallest-scaled high-pass band.

The investigation of bit allocation problems in our compression system is very important. From results in the experimental evaluation in Section 8.1 we noticed that the proposed bit allocation algorithms reduce the computation costs in searching the optimum state of the system. For the inter-band bit allocation problem, we extended previous work, which focused on optimum rate schemes, in the situation of the closed-looped image pyramid encoding, and developed an algorithm to determine the optimum distortion profile on each level which can minimize the overall rate. Such rate-minimizing inter-level distortion control algorithm is particularly suitable for the requirement in the distortion-parameterized compression, of which the target is to minimize the overall rate while under the constraint that the final reconstructed distortion is no more than the expected distortion. Moreover this algorithm offers a prototype, or a higher abstraction, concerning the inter-level bit allocation problem in distortion-parameterized compression using closed-looped image pyramids. Its significance is that it can still ensure rate-minimization when it is working with different intra-level compression schemes so long as they can meet the distortion requirement on each level. Therefore for further improvement one does not need to consider the effect of intra-band coding on the inter-band bit allocation, just focus on the improvement of the intra-band coding itself. The second algorithm deals with the intra-band bit allocation among individual sub vector quantizers. We successfully applied the 'Parato' optimality theory to solve the bit allocation problem among these quantizers. Moreover to meet the distortion requirement specified by the inter-level bit allocation algorithm we need to constraint the overall distortion accurately. This is achieved by allowing only a small amount of decrement of the overall distortion in each bit allocation process, which is associated with only one sub vector quantizer in a classified VQ.

10.1.3 Others

Besides the distortion-control capability and the high coding performance in three-dimensional medical image, our method has some other desired features in medical image application. Image artefacts due to the inability to reconstruct the high frequency components in a compression algorithm are the major problem that degrades the visual quality of the decompressed image. Such inability may produce

blocking artefacts in a block-DCT-based algorithm and ringing artefacts in a DWT-based algorithm. Both types of artefacts are obvious to human visual system because they may introduce high-frequency components in a smooth region. However in 3D-CoMIC these artefacts are alleviated by the closed-looped image pyramid structure. The quantization error generated at a particular level can be recovered in the coding process on its next lower level. Therefore although our image pyramid transform resembles subband decomposition, it will not suffer from the ringing artefacts apparent in conventional subband coding. In our approach the thresholding and vector quantization tend to smooth the high-frequency components and blur the contrast of the reconstructed image. However the evaluation shows that such blurring artefacts are less obvious to the human eyes than the former two types of artefacts. Moreover the blurring artefacts can be simply alleviated by the post processing on the decompressed images, for instance the contrast enhancement routine. In contrast neither blocking artefacts nor ringing artefacts can be attenuated after contrast enhancement and they may become even worse.

Additionally although they have not been investigated in the thesis, progressive coding and ROI-based coding are in principle supported in our method. By reviewing the image pyramid structure, we can see that coefficients are inherently organized in a form which is well suited for progressive transmission. Coefficients from upper levels contribute more in rate-distortion performance and take up only a small percentage of the total number of coefficients; while coefficients from lower levels contribute less but they take up a majority part of the total number of coefficients. Therefore a progress transmission can be achieve by setting upper-level coefficients higher priorities and encoding and transmitting them first, then followed by those lower-level coefficients. According to the encoding process in the closed-looped image pyramid, where the encoding process always starts from the top level and go down to the bottom level, we can see that the output bit stream is already progressively structured.

ROI-based coding could be implemented in our method as well. Different from DCT, in the image pyramid transform the distribution of coefficients is directly associated with the location of local structures in the image domain. Therefore the ROI regions defined in the image domain can delimitate in the pyramid domain to classify the

coefficients as in-region coefficients and out-of-region coefficients. We can achieve ROI coding by performing a distortion-constrain compression for the in-region coefficients and discarding the out-of-region ones. One should note that in this situation, the vector quantization and bit allocation are only considered for in-region coefficients, not the entire coefficients in previous.

10.2 Future Work

We have concluded our contributions in this thesis in the previous discussion. The three-dimensional closed-looped image pyramid coding combined with vector quantization techniques has been successfully implemented as a high coding performance distortion-constraint three-dimensional image compressor, which has shown its advantages in three-dimensional medical image compression applications. To extend our work, possible research orientations can be made in the following areas.

10.2.1 Image Quality Measure

Currently we use a numerical objective measure, MSE (or PSNR), to describe the quality of the reconstructed image. The lower the MSE value the higher the image quality. However as it has been discussed in Section 3.3.1 this objective measure only offers a fidelity measure to the original image. It does not provide trustworthy evaluation of visual quality of an image. In medical environment most of the diagnosis processes are human-guided, which involve lots of man-made decisions according to one's subjective evaluations. Subject evaluations on the decompressed images are desired. Moreover applications using various imaging modalities may have different criterions in subjective measures. Although many subjective evaluations have been reported in literature, for instance in the study of cardiac imaging using MRI, we notice few efforts have been made on the subjective evaluation of compression algorithms in CLSM imaging. Therefore it would be much appreciated for a comprehensive research study on the quality evaluation in the application of CLSM image compression. It will involve a number of quality measures, including both objective measures and subjective measures, across various compression algorithms on different types of CLSM images. According to their

different purposes the subject evaluations can be conducted in the following two categories.

A. Subjective Evaluation on 3D-CoMIC Approach

It is known that the 3D-CoMIC approach is a distortion-parameterized method that can conveniently and accurately work at a particular objective quality level. In this work the evaluation will be performed by using both an objective method and a subjective method on various types of CLSM images at different compression levels. The purpose is to determine the relations between the objective measures and the subjective measures. Eventually it is possible that the compression task can be defined by some subjective descriptions of the target quality, such as 'no visual difference', 'no obvious artefacts' and the like, which would be more attractive than a scale of objective measure in practice. By objective method some statistical indices are calculated to indicate the reconstructed image quality, such as MSE and PSNR and by subjective method viewers read images directly to determine their quality. As it has been discussed in Section 3.3.1.2 a scoring approach can be used to calculate the subjective measures. Moreover we can use the absolute scoring approach to investigate the consistency between subjective and objective measures, and use the relative scoring approach to reduce the bias of judgement from the original image itself.

B. Subjective Evaluation Across Various Approaches

The second type of examination is to investigate the visual qualities across various compression approaches on different images. It can be performed, for instance, by simultaneously presenting the reconstructed image from each compression approach at the same compression level, and comparing each reconstruction with the original image and finally ranking them from the best reconstruction to the poorest reconstruction. Furthermore the evaluation can be measured on two types of compression standards: the rate and the objective quality. In the former all compression approaches are configured to work at the same rate level. Using the rate standard we can have a subjective evaluation of the compression capability across various compression approaches for a particular image. In the latter all compression approaches are configured to work at the same objective quality level, e.g. MSE. It

removes the bias from the compression capability of an approach, and therefore gives a subjective judgement on the severity of different types of artefacts coming from various compression techniques.

10.2.2 Improvement in Coding Performance

More investigations can be made in several aspects when we attempt to improve the coding performance of our approach. One of them is an enhanced CBCVQ, which is being considered in our latest research. Currently the classification of intra-band coefficients is achieved using vector quantization, where each block is vector quantized and associated with a unique codeword index, and blocks having the same index are grouped as one class. Such classification does not optimize the classification gain. According to the discussion in Section 7.1.2.2, the classification gain only depends on the low level statistics, such as the variance and the number of samples, of each class. However in vector quantization it tries to capture the inter-block dependencies at higher levels rather than these low level features. Therefore instead of current VQ-based classification we can improve the coding performance by developing another classification technique which is designed for optimizing the classification performance.

Another potential improvement in CBCVQ is the way of capturing the cross-band dependencies. In the current CBCVQ we solve this problem by sharing a same classification map generated on the smallest-scaled high-pass band among all its following bands. However when we increase the decomposition level to deal with very large-sized images, i.e. 2K×2K, the cross-band same-orientation dependencies become weak. Sharing a same classification map may not work as well as it does when the decomposition level is not large. Therefore more investigations can be made in exploiting the cross-band classification dependencies in the image pyramid coding. For instance, some efforts have been made in literatures [Joshi, 1997], where they separately build a classification map for each subband and then using a context-based entropy coder to capture the dependencies among these maps.

Moreover parameters, including the threshold magnitudes in the block-based splitting thresholding method and the vector size and classification block size in the CBCVQ,

are currently manually determined. They may affect the rate-distortion performance and also the computation costs of our approach. For instance it is possible to improve the coding performance by increasing the vector size to capture the higher-order inter-coefficient correlations. However it will increase the computation costs in codebook training and encoding and also the storage requirement of the codebooks. It is desirable to investigate the effects of these parameters on the coding performance by considering both the rate-distortion performances and the complexities.

10.2.3 Improvement in Complexity

Due to the use of vector quantization the complexity of our method becomes an important issue in practice. As one of the contributions in this thesis, we have improved the computation costs in both codebook training and codebook searching. However the computation complexity is still a problem and the situation may become more severe when we deal with large-volume image data. On the other hand the complexity in codebook storage is also very high since in CBCVQ we separately train a codebook for each class and store each codebook individually. For instance in our experiment some quantizers associated with large-variation sources have to work at very high rates to meet the small distortion requirement, so their codebook sizes may be very large, such as 1024 entries. A possible solution to alleviate these two problems is using a model-based codebook training technique, such as Pyramid VQ (PVQ) [Fishcer, 1986]. As discussed in Section 7.1.2 in this technique the codebook is built on a particular distribution model, for instance the Laplacian distribution, rather than a specific training set. It loses the optimality of the codebook to the source compared to a specific codebook, but can decrease the complexity in both computation and storage, because it avoids the codebook training process in each coding attempt and the codebook can be designed and transmitted independent from the training set. Therefore it is worthy of consideration if one wants to balance the coding performance and the complexity.

References

Abousleman, G. P., M. W. Marcellin, et al. (1995). "Compression of Hyperspectral Imagery using the 3D DCT and Hybrid DPCM/DCT." IEEE Trans. on Geosci. Remote Sensing **33**: 26-34.

Ahmet, M. and S. Paul (1995). "Image Quality Measures and Their Performance." IEEE Trans. on Commun. **43**: 2959-2965.

Aiazzi, B., L. Alparone, et al. (1997). "Lossless Image Compression by Quantization Feedback in a Content-Driven Enhanced Laplacian Pyramid." IEEE Trans. on Image Processing **6**: 831-843.

Aravind, R. and A. Gersho (1986). Low-rate Image Coding with Finite-state Vector Quantization. Proc. IEEE Intl. Conf. on Acoust., Speech, and Signal Processing (ICASSP), Tokyo.

Arfken, G. (1985). Lagrange Multipliers. Mathematical Methods for Physicists. Orlando, FL, Academic Press: 945-950.

Arribas, S. M., C. Daly, et al. (1999). Measurements of Vascular Remodeling by Confocal Microscopy. Confocal Microscopy (Methods in Enzymology, Volume 307). P. M. Conn, Academic Press: 246-273.

Avinash, G. B. (1995). "Image Compression and Data Integrity in Confocal Microscopy." SCANNING-The Journal of Scanning Microscopies **17**: 156-160.

Bei, C. D. and R. M. Gray (1985). "An Improvement of the Minimum Distortion Encoding Algorithm for Vector Quantization." IEEE Trans. on Commun. **33**: 1132-1133.

Bentley, J. L. (1975). "Multidimensional binary search trees used for associative searching." Commun. of ACM: 209-226.

Bilgin, A., G. Zweig, et al. (1998). Efficient Lossless Coding of Medical Image Volumes using Reversible Integer Wavelet Transforms. Proc. 1998 DCC.

Blanchett, T., G. C. Kember, et al. (1998). "KLT-Based Quality Controlled Compression of Single-Lead ECG." IEEE Trans. on Biomed. Eng. **45**(7): 942-945.

Burt, P. J. and E. H. Adelson (1983). "The Laplacian Pyramid as a Compact Image Coder." IEEE Trans. on Commun. **31**: 532-540.

Castleman, K. R. (1995). Digital Image Processing, Prentice Hall.

Chang, P. C., J. May, et al. (1985). Hierarchical Vector Quantizers with Table-lookup Encoders. Proc. of IEEE Int. Conf. on Commun.

Chang, R.-F. and W.-M. Chen (1996a). "Adaptive Edge-based Side-match Finite-state Classified Vector Quantization with Quadtree Map." IEEE Trans. on Image Processing **5**(2): 378-383.

Chang, R.-F. and W.-T. Chen (1993). "Image Coding Using Variable-Rate Side-Match Finite-State Vector Quantization." IEEE Trans. on Image Processing **2**(1): 104-108.

Chang, R.-F. and Y.-L. Huang (1996b). "Finite-state Vector Quantization by Exploiting Interband and Intraband Correlations for Subband Image Coding." IEEE Trans. on

Image Processing **5**(2): 374-378.

Chen, J. and S. Itoh (1998). "A Wavelet Transform-based ECG Compression Method Guaranteeing Desired Signal Quality." IEEE Trans. on Biomed. Eng. **45**(12): 1414-1419.

Chen, W. H. and C. H. Smith (1977). "Adaptive coding of monochrome and color images." IEEE Trans. on Commun. **25**: 1285-1292.

Cheng, D. Y., A. Gersho, et al. (1994). Fast Search Algorithms for Vector Quantization and Pattern Matching. Proc. IEEE Intl. Conf. on Acoust., Speech, and Signal Processing (ICASSP), San Diego.

Christopoulos, C. (2000). JPEG2000 Verification Model 8.5. ISO/IEC JTC1/SC29/WG1, Report N1878.

Cockshott, W. P., A. Cottrell, et al. (2000). "Optimal Allocation of Bits to Frames in Compressed Video." SCI 2000: 249-255.

Cockshott, W. P. and R. Lambert (1999). "Algorithm for Hierarchical Vector Quantization of Video Data." IEE Proc. - Vision, Image and Signal Processing **146**(4): 222-228.

Cockshott, W. P., Y. Tao, et al. (2003). "Confocal Microscopic Image Sequence Compression Using Vector Quantization and 3D Pyramids." SCANNING-The Journal of Scanning Microscopies **25**(5): 247-256.

Cosman, P. C., R. M. Gray, et al. (1994). "Evaluating Quality of Compressed Medical Images: SNR, Subjective Rating, and Diagnostic Accuracy." Proc. of IEEE **82**(6): 919-932.

Cosman, P. C., R. M. Gray, et al. (1996). "Vector Quantization of Image Subbands: A Survey." IEEE Trans. on Image Processing **5**(2): 202-225.

Cosman, P. C., K. L. Oehler, et al. (1993a). "Using Vector Quantization for Image

Processing." Proc. of IEEE **81**(9): 1326-1341.

Cosman, P. C., C. Tseng, et al. (1993b). "Tree-structured Vector Quantization of CT Chest Scans: Image Quality and Diagnostic Accuracy." IEEE Trans. on Med. Imaging **12**: 727-739.

Cover, T. M. and P. E. Hart (1967). "Nearest Neighbor Pattern Classification." IEEE Trans. on Inform. Theory **13**: 21-27.

Crochiere, R. E. and L. R. Rabiner (1983). Multirate Digital Signal Processing. Englewood Cliffs, NJ, Prentice Hall.

Daly, C. (2002). Receptor Imaging. VASCAN 2000 Tech. Rep. 2.

Daly, S. (1992). "Visible Differences Predictor: An Algorithm for the Assessment of Image Fidelity." Proc. of SPIE **1666**: 2-14.

DeVore, R. A., B. Jawerth, et al. (1992). "Image Compression Through Wavelet Transform Coding." IEEE Trans. on Inform. Theory **38**: 719-746.

Elias, P. (1970). "Bounds on Performance of Optimum Quantizers." IEEE Trans. on Inform. Theory **16**: 172-184.

Eskicioglu, A. M. (1995). "Image Quality Measures and Their Performance." IEEE Trans. on Commun. **43**(12): 2959-2965.

Fischer, T. R. (1986). "A Pyramid Vector Quantizer." IEEE Trans. on Inform. Theory **32**(4): 568-583.

Friedman, J., F. Baskett, et al. (1975). "An Algorithm for Finding Nearest Neighbours." IEEE Trans. on Computers **24**: 1000-1006.

Gersho, A. (1979). "Asymptotically Optimal Block Quantization." IEEE Trans. on Inform. Theory **25**: 373-380.

Gersho, A. and R. M. Gray (1992). Vector Quantization and Signal Compression. Boston, Kluwer Academic Publishers.

Gray, R. M. and D. L. Neuhoff (1998). "Quantization." IEEE Trans. on Inform. Theory **44**: 2325-2383.

Hamid, T. (1998). DICOM Requirements for JPEG2000. ISO/IEC JTC1/SC29/WG1, Report N944.

Hang, H. M. and J. W. Woods (1985). "Predictive Vector Quantization of Images." IEEE Trans. on Commun. **33**: 1208-1219.

He, Z. and S. K. Mitra (2001). "Novel Rate-distortion Analysis Framework for Bit Rate and Picture Quality Control in DCT Visual Coding." IEE Proc. - Vision, Image and Signal Processing **148**: 398-406.

Ho, B. K. T., P. Saipetch, et al. (1994). Video Compression Algorithm for Dynamic Angiographic Images. Proc. SPIE Medical Imaging, Newport Beach, CA.

Ho, Y.-S. and A. Gersho (1988). Variable-rate Multi-stage Vector Quantization for Image Coding. Proc. IEEE Intl. Conf. on Acoust., Speech, and Signal Processing (ICASSP).

Horn, U., T. Wiegand, et al. (1997). Bit Allocation Methods for Closed-loop Coding of Oversampled Pyramid Decompositions. Proc. IEEE International Conference on Image Processing ICIP-97, Santa Barbara, CA, USA.

Hosaka, K. (1986). A New Picture Quality Evaluation Method. Proc. Int. Picture Coding Symp., Tokyo, Japan.

Houlding, D. and J. Vaisey (1995). "Low Entropy Image Pyramids for Efficient Lossless and Progressive Image Communication." IEEE Trans. on Image Processing **4**: 1150-1153.

Hu, J., N. Sinaceur, et al. (1997). Removal of Blocking and Ringing Artifacts in Transform Coded Images. Proc. IEEE Intl. Conf. on Acoust., Speech, and Signal Processing (ICASSP).

Huffman, D. A. (1952). "A Method for the Construction of Minimum Redundancy Codes." Proc. of IRE **40**: 1098-1101.

Huh, Y., J. J. Hwang, et al. (1995). "Block Wavelet Transform Coding of Images using Classified Vector Quantization." IEEE Trans. on Circuits and Systems for Video Tech. **5**(1): 63-67.

Jin, Z. (1988). On the Multi-scale Iconic Representation for Low-level Computer Vision Systems, The Turing Institute and The University of Strathclyde.

Joshi, R. L., H. Jafarkhani, et al. (1997). "Comparison of different methods of classification in subband coding of images." IEEE Trans. on Image Processing **6**(11): 1473-1486.

Juang, B.-H. and A. H. Gray (1982). Multiple Stage Vector Quantization for Speech Coding. Proc. IEEE Intl. Conf. on Acoust., Speech, and Signal Processing (ICASSP), Paris.

Kaplan, W. (1992). Advanced Calculus. Reading, MA, Addison-Wesley.

Kim, T. (1988). New Finite-state Vector Quantizers for Images. Proc. IEEE Intl. Conf. on Acoust., Speech, and Signal Processing (ICASSP), New York.

Kim, T. (1992). "Side Match and Overlap Match Vector Quantizers for Images." IEEE Trans. on Image Processing **1**(2): 170-185.

Kim, Y. and W. A. Pearlman (1999). Lossless Volumetric Medical Image Compression. Proc. SPIE on Applications of Digital Image Processing XXII.

Lee, H., D. Haynor, et al. (1992). Subjective evaluation of compressed image quality.

Proc. of SPIE - Image Capture, Formatting and Display.

Lee, H., Y. Kim, et al. (1993). "Statistical Distributions of DCT Coefficients and Their Applications to An Interframe compression Algorithm for 3D Medical Images." IEEE Trans. on Med. Imaging **12**: 478-485.

Lemke, H. U. and M. Osteaux (1993). "Special Issue: Picture Archiving and Communication Systems and Digital Radiology." Europe. J. Radiology **17**(1): 1-2.

Linde, Y., A. Buzo, et al. (1980). "An Algorithm for Vector Quantizer Design." IEEE Trans. on Commun. **28**: 84-95.

Liu, J. and P. Moulin (2000). Analysis of Interscale and Intrascale Dependencies between Image Wavelet Coefficients. Proc. of International Conference on Image Processing (ICIP).

Lloyd, S. P. (1982). "Least Squares Quantization in PCM." IEEE Trans. on Inform. Theory **28**(3): 129-137.

MacMahon, H., K. Doi, et al. (1991). "Data Compression: Effect on Diagnostic Accuracy in Digital Chest Radiographs." Radiology **178**: 175-179.

Melnik, V., I. Shmulevich, et al. (2001). "Block-Median Pyramidal Transform: Analysis and Denoising Applications." IEEE Trans. on Signal Processing **49**(2): 364-372.

Miaou, S. G. and S. T. Chen (2004). "Automatic Quality Control for Wavelet-Based Compression of Volumetric Medical Images using Distortion-Constrained Adaptive Vector Quantization." IEEE Trans. on Med. Imaging **23**(11): 1417-1429.

Minsky, M. (1957). Microscopy Apparatus. U.S.

Motoishi, K. and T. Misumi (1984). On A Fast Vector Quantization Algorithm. Proc. VIIth Symp. on Inform. Theory and Its Applications.

Munteanu, A., J. Cornelis, et al. (1999). "Wavelet-based Lossless Compression of

Coronary Angiographic Images." IEEE Trans. on Med. Imaging **18**: 272-281.

Muresan, D. D. (2001). Adaptive, Optimal-Recovery Image Interpolation. Proc. IEEE Intl. Conf. on Acoust., Speech, and Signal Processing (ICASSP), Utah.

Naveen, T. and J. W. Woods (1993). Subband finite state scalar quantization. Proc. IEEE - ICASSP.

Neuhoff, D. L. and D. H. Lee (1991). On the Performance of Tree-structured Vector Quantization. Proc. IEEE Intl. Conf. on Acoust., Speech, and Signal Processing (ICASSP), Toronto.

Nill, N. B. and B. H. Bouxas (1992). "Objective Image Quality Measure Derived from Digital Image Power Spectra." Optical Engineering **31**: 813-825.

Pawley, J., Ed. (1995). Handbook of Boilogical Confocal Microscopy (3rd ed.). New York, Plenum.

Perlmutter, S. M., P. C. Cosman, et al. (1998). "Medical Image Compression and Vector Quantization." Statist. Sci. **13**(1): 30-53.

Ramamurthi, B. and A. Gersho (1986). "Classified vector quantization of images." IEEE Trans. on Commun. **34**(11): 1105-1115.

Ramchandran, K., A. Ortega, et al. (1994). "Bit Allocation for Dependent Quantization with Applications to Multiresolution and MPEG Video Coders." IEEE Trans. on Image Processing **3**: 533-545.

Riskin, E. A. and R. M. Gray (1991). "A Greedy Tree Growing Algorithm for the Design of Variable Rate Vector Quantizers." IEEE Trans. on Signal Processing **39**: 2500-2507.

Roger, R. E. and J. F. Arnold (1994). "Reversible Image Compression Bounded by Noise." ieee tRans. on Geosci. Remote Sensing **32**(1): 19-24.

Said, A. and W. A. Pearlman (1996). "A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees." IEEE Trans. on Circuits and Systems for Video Tech. **6**: 243-250.

Sayre, J., D. R. Aberle, et al. (1992). Effect of Data Compression on Diagnostic Accuracy in Digital Hand and Chest Radiography. Proc. of SPIE on Med. Imaging VI: Image Capture, Formatting and Display, Bellingham, WA.

Schelkens, P., A. Munteanu, et al. (2003). "Wavelet Coding of Volumetric Medical Datasets." IEEE Trans. on Med. Imaging **22**(3): 441-458.

Shannon (1959). "Coding Theorems for a Discrete Source with a Fidelity Criterion." IRE Nat. Conv. Rec.: 142-163.

Shenoy, R. G. (1992). "An Optimal Recovery Approach to Interpolation." IEEE Trans. on Signal Processing **40**(8): 1987-1996.

Tao, Y. and W. P. Cockshott (2004). 3D Microscopic Image Coding by Finite-State Vector Quantization in an Enhanced Image Pyramid. SPIE on Medical Imaging 2004, San Diego, CA.

Taubman, D. and M. W. Marcellin (2001). JPEG2000 - Image Compression: Fundamentals, Standards and Practice. Hingham, MA, Kluwer Academic Publishers.

Tzou, K. H. (1987). "progressive Image Transmission: A Review and Comparison of Techniques." Optical Engineering **26**: 581-589.

Ungerboeck, G. (1987). Trellis-coded Modulation with Redundant Signal Sets, Parts I and II. IEEE Communication Magazine. **25:** 5-21.

VASCAN (2000). Three-Dimensional Analysis of Vascular Structure, Function and Receptor Distribution Using Confocal Laser Scanning Microscopy, European Commission - Shared Cost RTD project [QLG1-1999-00084].

Vetterli, M. and C. Herley (1992). "Wavelets and Filter Banks: Theory and Design." IEEE Trans. on Acoust. Speech Signal Process. **40**(9): 2207-2232.

Vidal, E. (1986). "An Algorithm for Finding Nearest Neighbors in (Approximately) Constant Average Time Complexity." Pattern Recognition Letters **4**: 145-157.

Villasenor, J., B. Belzer, et al. (1995). "Wavelet Filter Evaluation for Image Compression." IEEE Trans. on Image Processing **2**: 1053-1060.

Wallace, G. K. (1991). "The JPEG Still Picture Compression Standard." Commun. of ACM **34**(4): 30-44.

Walras, L. (1954). Elements of Pure Economics. Ill: Irwin, Homewood.

Wang, L. and M. Goldberg (1989). "Progressive Image Transmission Using Vector Quantization on Images in Pyramid Form." IEEE Trans. on Commun. **37**(12): 1339-1349.

Wang, L. and M. Goldberg (1991). "Comparative Performance of Pyramid Data Structures for Progressive Image Transmission." IEEE Trans. on Commun. **39**: 540-548.

Wang, X. and S. Panchanathan (1993). "Wavelet Transform Coding using NIVQ." Proc. of Visual Commun. and Image Processing - SPIE **2094**: 999-1009.

Watkinson, J. (2001). The MPEG Handbook: MPEG-1, MPEG-2, MPGE-4, Focal Press.

Webb, R. H. (1996). "Confocal Optical Microsocpy." Reports on Progress in Physics **59**(3): 427-471.

Wei, H.-C., P.-C. Tsai, et al. (2000). "Three-sided Side Match Finite-state Vector Quantization." IEEE Trans. on Circuits and Systems for Video Tech. **10**(1): 51-58.

Weinberger, M., G. Seroussi, et al. (1996). LOCO-I: A Low Complexity, Context-based, Lossless Image Compression Algorithm. Proc. IEEE DCC'96, Snowbird, Utah,

U.S.

Weinberger, M., G. Seroussi, et al. (2000). "The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS." IEEE Trans. on Image Processing **9**(8): 1309-1324.

Wilhelm, P., D. R. Haynor, et al. (1990). "Lossy Image Compression for Digital Medical Imaging Systems." Proc. of SPIE **1232**: 348-358.

Wittenberg, U. (1993). "Applications of the JPEG Standard in A Medical Environment." Proc. of SPIE **1977**: 121-127.

Wolberg, G. (1990). Digital Image Warping, IEEE Computer Society Press.

Wong, S., L. Zaremba, et al. (1995). "Radiologic Image Compression - A Review." Proc. of IEEE **83**(2): 194-219.

Woods, J. W. and S. D. O'Neill (1986). "Subband Coding of Images." IEEE Trans. on Acoust. Speech Signal Process. **34**: 1278-1288.

Wu, X. and N. Menon (1996). "CALIC-A Context-based Adaptive Lossless Image Codec." IEEE Trans. on Commun. **45**(4): 437-444.

Yao, S. and R. J. Clarke (1992). "Image Sequence Coding using Adaptive Vector Quantization in Wavelet Transform Domain." Electronics Letters **28**(17): 1566-1568.

Ziv, J. and A. Lempel (1978). "Compression of Individual Sequences Via Variable-Rate Coding." IEEE Trans. on Inform. Theory **24**: 530-536.