Regions-of-Interest-driven Medical Image Compression

Mohsen Firoozbakht

Submitted in partial fulfilment of the requirements

Of Kingston University for the degree of

Doctor of Philosophy

2014

ABSTRACT

Advances in medical imaging technologies, particularly magnetic resonance imaging and multi-detector Computed Tomography (CT), have resulted in substantial increase in the size of datasets. In order to reduce the cost of storage and diagnostic analysis and transmission time without significant reduction in image quality, a state of the art image compression technique is required. We propose here a context based and regions of interest (ROI) based approach for the compression of 3D CT images and in particular vascular images, where a high spatial resolution and contrast sensitivity is required in specific areas. The methodology is developed based on the JPEG2000 standard to provide a variable level of compression in the (x,y) plane as well as in the z axis. The proposed lossy-to-lossless method compresses multiple ROIs depending on the degrees of clinical interest. High priority areas are assigned a higher precision (up to lossless compression) than other areas such as background. ROIs are annotated automatically. The method has been optimized and applied to the vascular images from CT angiography for peripheral arteries and compared with a standard medical image codec on 10 datasets regarding image quality and diagnostic performances. The average size of the compressed images can be reduced to 61, 60, 66, and 89 percent with respect to the lossless JP2K, Lossless JP3D, Lossless H.264, and original image respectively with no remarkable impairment for the diagnostic accuracy based on visual judgment of two radiologists.

Publication Resulting from this Work

1. M. Firoozbakht, J. Dehmeshki, M. Martini, H. Amin, M.E. Dehkordi, A. Jouannic, S.D. Qanadli, "A new lossy to lossless approach for compression of vascular medical imaging," *10th annual meeting of the union of vascular societies of Switzerland*, Lausanne, November 2009.

2. M. Firoozbakht, J. Dehmeshki, M. Martini, H. Amin, M. Dehkordi, A. Jouannic, S.D. Qanadli, "Compression of digital medical images based on multiple regions of interest," *The Fourth International Conference on Digital Society (ICDS)*, St. Maarten, Netherlands Antilles, February 2010.

3. Y. Ebrahimdoost, J. Dehmeshki, T.S. Ellis, M. Firoozbakht, A. Jouannic, S.D. Qanadli, "Medical Image Segmentation Using Active Contours and a Level Set Model: Application to Pulmonary Embolism (PE) Segmentation," *The Fourth International Conference on Digital Society (ICDS)*, St. Maarten, Netherlands Antilles, February 2010.

4. M. Firoozbakht, J. Dehmeshki, M. Martini, H. Amin, M. Dehkordi, A. Jouannic, S.D. Qanadli, "A multi scale approach for compression of vascular imaging," *International conference on Image and Video Processing and Computer vision (IPVCV'10)*, Orlando, FL, USA, July 2010.

5. Y. Ebrahimdoost, J. Dehmeshki, T.S. Ellis, M. Firoozbakht, A. Jouannic, S.D. Qanadli, "3D Pulmonary Embolism (PE) Segmentation of CTA images Using Level Sets," *International Conference on Image Processing, Computer Vision, & Pattern Recognition (IPCV'10)*, Las Vegas, USA, July 2010.

6. M. Firoozbakht, J. Dehmeshki, M. Martini, S.D. Qanadli, "A new multi level adaptive compression approach for CT angiography in teleradiology," *98th SSR-SGR Annual Meeting of the Swiss Society of Radiology*, Interlaken, Switzerland, June 2011.

7. M. Firoozbakht, M.G. Martini, S.D. Qanadli, S.Zahedi, and J. Dehmeshki, "A Multi Scale Adaptive Compression Approach with Application to Peripheral Artery CT Image," *International Conference on Image Processing, Computer Vision, & Pattern Recognition (IPCV'12)*, Las Vegas, USA, July 2012.

8. C. Nakajo, M. Firoozbakht, J. Dehmeshki, R. Meuli, S.D. Qanadli, "A 3D multi level adaptive compression approach for CT angiography," *99th SSR-SGR Annual Meeting of the Swiss Society of Radiology*, Zurich, Switzerland, May 2012.

9. M. Firoozbakht, M.G. Martini, S.D. Qanadli, S. Zahedi, and J. Dehmeshki, "A Multi Level Adaptive Compression Approach for 3D CT Images," *10th IASTED on Visualization Imaging and Image processing (VIIP'12)*, Banff, Calgary, Canada, July 2012.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBRIVATIONS

3D	Three Dimensional
1D	One Dimensional
2D	Two Dimensional
AVC	Advance Video Coding
bpp	bits per pixel
CABAC	Context-based Adaptive Binary Arithmetic Coding
CAD	Computer Aided Detection system
CAT	Computerised Axial Tomography
CAVLC	Context Adaptive Variable Length Coding
CPBAShift	Compensation-based Partial Bitplane Alternating Shift
CR	Compression Ratio
CT	Computed Tomography
CTA	Computed Tomography Angiography
DCT	Discrete Cosine Transform
DICOM	Digital Imaging and Communication in Medicine
DPCM	Differential Pulse Code Modulation
DWT	Discrete Wavelet Transform
EBCOT	Embedded Block Coding with Optimized Truncation
FDA	Food and Drug Administration
FRExt	Fidelity Range Extensions
Gbps	Gigabit per second
GUI	Graphic User Interface

HEVC	High Efficiency Video Coding
HU	Hounsfield Units
ICT	Irreversible Component Transformation
isodata	iterative self organizing data
JP2K	JPEG2000
LAN	Local Area Network
LMSE	Laplacian Mean Square Error
MAE	Mean Absolute Error
MCT	Multiple Component Transformation
MD	Maximum Difference measurement
MIP	Maximum Intensity Projection
MOS	Mean Opinion Score
MRA	Magnetic Resonance Angiogram
MRI	Magnetic Resonance Imaging
MSE	Mean Square Error
MV	Motion Vector
NAE	Normalized Absolute Error
NAL	Network Abstraction Layer
PACS	Picture Archiving and Communications Systems
PAD	Peripheral Arterial Disease
PET	Positron Emission Tomography
PROI	Primary Regions of Interest
PSBShift	Partial Significant Bitplanes Shift
PSNR	Pick Signal to Noise Ratio
QOS	Quality of Service
RCT	Reversible Component Transformation
RD	Rate Distortion
ROI	Region of Interest
SC	Structural Content
SCMShift	Selective Coefficient Mask Shift
SROI	Secondary Regions of Interest
SSIM	Structural SIMilarity
TCQ	Trellis Coded Quantization

UQI	Universal Image Quality Index
VCL	Video Coding Layer
VLC	Variable Length Coding

ACKNOWLEDGMENTS

I would like to thank my major supervisors, Professor Jamshid Dehmeshki, Dr Maria G. Martini, and Professor Saleh D. Qanadli for taking me under their academic guidance, support and professional advice during my degree.

I would like also thank to Dr. Nikajo for assessing the quality of medical compressed images.

To Dr. Siamak Azadolmolky, thank you for your support and valuable discussions.

I would like to thank my parents for their encouragement and supporting me during my studies. Finally, I would like to express my sincere appreciation to my wife, Sara, for her support and encouragement.

This work has been partially funded by the Azad University-South Tehran Branch and Azad University in Oxford.

1. INTRODUCTION

The size of datasets generated from medical imagery of various modalities such as Computed Tomography Angiography (CTA) and Magnetic Resonance Angiogram (MRA) reaches amounts, which do not allow easy archive and transmission by Picture Archiving and Communications Systems[1] (PACS) [1] [2]. A single dataset of medical imaging contains a few hundred or thousands of digital images. The on-going developments in the area of medical imaging and dynamic imaging significantly increase the size of these datasets. Research shows that the number of datasets increased dramatically between 1996 and 2010 in the USA by using diagnostic imaging[2] [3].

For instance, a 1GB dataset of one CT angiography for Peripheral Arterial Disease (PAD) contains around 2000 medical images with an average size of 512KB per image. Considering the number and increasing size of these datasets, imposes a challenge for proper data transmission. Therefore, in order to properly visualize digital images on diagnostic or review monitors in a timely fashion, the data transfer rate is becoming very crucial and important. Although 1 gigabit per second (1 Gbps) Ethernet is typically being

---

[1] PACS is a comprehensive computer system which contributes to the creation, distribution and archiving of digital medical images in the medical enterprise. The system is highly integrated with image acquisition devices (modalities), a network system, an archive device, and viewing monitors and is usually related closely to the hospital or radiology information system.

[2] Diagnostic imaging is a process in which doctors use medical images to specify the source of medical problems in patients.

installed for Local Area Networks (LANs) of new hospitals, there are many hospitals in which their existing LANs are operating based on 100 megabits per second (100 Mbps) Ethernet technology. Even when such a 100 Mbps Ethernet line is on a dedicated switch line, the typical effective data transfer rate (i.e., throughput) is not more than 40 Mbps. This means that it takes around 250 seconds to send a medical dataset of 1 GB from the PACS archive to a viewing monitor. In addition, there is a trend towards accessing medical images from remote sites such as home offices or mobile hand-held devices. Since the connection bandwidth of these devices to the hospitals' PACS servers is usually much lower than LAN connections, the required transmission time becomes even longer.

In addition to the required time for transmission of medical images across computer networks, long-term storage of huge amounts of data (i.e., medical imaging datasets) becomes more of a challenge. In order to decrease the size of medical imaging datasets, compression technologies can be applied to reduce the size of medical images for both storage and transmission across networks with different bandwidths [4]-[7].

In general, there are two categories of compression techniques: (1) Lossy and (2) Lossless. In lossy methods, some information (i.e., image details) is lost as the high compression ratio is the main goal of compression. In lossless techniques, the exact original image is reconstructed from the compressed one. Thus, the compression rate is modest in the latter. Several lossy and lossless compression techniques have been proposed in the literature [8]-[24].

Although lossy image compression techniques are potentially good candidates to decrease the size of medical imaging datasets, the loss of some clinical (and possibly critical) information prohibits the use of this technique for medical images [25] [26]. To take advantage of both, high compression rate of lossy methods and preserving clinically important information during the compression process, a region of interest (ROI) based coding method is required to avoid the loss of important information in parts of an image, which are more important than the others.

A number of ROI based coding techniques have been proposed by researchers [17]-[24]. These techniques have two disadvantages for medial image compression: (1) The segmentation of ROIs is not fully automated; therefore, it could be extremely time consuming to annotate the ROIs by a radiologist and (2) For implementing those methods, some definite modifications on the standard image compression are required. This modification necessarily changes, at least in part, the standard of compression algorithms approved in the Digital Imaging and Communication in Medicine[3] (DICOM) standard. The DICOM standard committee is highly conservative about adding new schemes due to the incremental benefit of slightly improved performance, or an extra feature, rarely justifies the risk of compromising interoperability. Medical devices are also highly regulated (e.g., by the Food and Drug Administration (FDA) in the US); therefore, they are also expensive to change given the burden of required testing and documentation **[27]** **[28]**. Conformance to the standard is mandatory for medical image compressions, which have been integrated into pre-existing PACS.

In order to increase the compression efficiency while properly addressing the above mentioned issues, a fully automated ROI based medical image compression in compliance with FDA regulation should be developed. The only compression standard, which includes ROI feature and also has been supported in the DICOM standard, is JPEG2000 **[29]** **[30]**. In this standard, ROIs and the rate of compression are typically identified manually, so a lot of interactions and manual interventions are needed by the user to view medical image datasets, which contains up to 2000 slices. Therefore, developing a fully automated ROI based image compression is highly desirable.

1.1 Aim and Objectives

The aim of this research is to design and develop a fully automated context based and ROI based approach for compression of digital vascular images in general and Peripheral Arterial Disease (PAD) in 3D CTA (Computed Tomography Angiography) in particular, with minimum interactions between user and system. In the proposed method, the Regions Of Interest (ROI) are areas, corresponding to a higher degree of clinical interest,

[3] DICOM is an International standard for storing, transmitting and printing any information related to medical imaging. It defines the digital file formats for medical images generated during many healthcare-related imaging "modalities" such as computed tomography, ultrasound, and magnetic resonance.

which will be identified based on contextual information given by a radiologist or anatomy. The main objectives of this research include the following:

- Automatic segmentation of ROIs
- A fully automated context based and ROI based compression approach, which includes in an automatic rate control mechanism
- Developing a robust performance evaluation schema (Test-bed) to compare the proposed method against other standard compression technologies on patient datasets

1.2 Summary of Contributions

The main contributions of this thesis are summarized as follows:

I have designed and developed a fully automated context based and regions of interest based system. The proposed system is utilized to compress 3D digital medical images that can keep the full skeleton of medical images while the primary regions of interest (PROI), which are critical for medical diagnosis, are maintained intact during the compression process. This means that no information is lost from PROI during the compression process. The secondary regions of interest (SROI) are defined as part of the medical images, which their loss can be tolerated during the diagnosis process. Therefore, for the secondary regions of interest, diagnostically irrelevant information that has no impact on diagnosis decision is lost.

The interactions and manual interventions with the user are minimized using the following two proposed algorithms: (1) In the automatic rate control algorithm, the bitrate for each slice is automatically calculated with the rate considered for previous slice based on the PSNR of SROI and bitrate constant step and (2) The automatic ROIs segmentation algorithm, which consists of background & body segmentation (SROI) and peripheral artery segmentation (PROI). SROI is segmented by using an adaptive thresholding method followed by morphological operations and connected component analysis. The process of PROI segmentation is an initial segmentation of arteries from other tissues followed by a 3D region growing algorithm, in combination with morphological operators to increase the precision of segmentation. The segmentation approach includes an automatic seed point selection technique.

I have also designed a robust performance evaluation schema (test-bed) to analyze the results based on the objective and subjective quality metrics. Using a GUI designed to represent the compressed images, two groups (Lossy and Lossless) including seven scenarios are evaluated over 10 datasets, concerning the image quality and the potential impact on diagnosis accuracy.

1.3 Structure of the Thesis

This research consists of four components: (1) Volumetric codecs, (2) ROIs based segmentation, (3) ROIs based compression, and (4) Evaluation system as illustrated in Figure 1.1. The organization of this thesis is laid out based on these four components followed by a literature review in Chapter 2.

(1) Volumetric codecs: The main target of this component is to apply the two standard volumetric codecs, JP3D and H.264, on the 3D medical images (patient datasets) to prepare the results of existing techniques for comparison with our proposed method. Chapter 3 presents the architecture of three image and video compression technologies used throughout this research. It starts with JPEG2000 followed by JP3D and H.264. The results of applying these compression technologies on medical datasets are reported in Chapter 6.

(2) ROI based segmentation: This component extracts different regions of interest from the patient datasets. The extracted ROIs are passed to the ROI based compression component to be compressed with different compression ratios. The segmentation approach is divided into two parts: (1) Background and SROI segmentation followed by (2) PROI segmentation. The process of segmentation includes a 3D region growing algorithm, in combination with morphological operators and starts from automated selected seed point. Chapter 4 starts with a brief overview about PAD and CT SCAN. It describes various ROI based

segmentation techniques and also introduces the proposed ROI based segmentation approach. The results of applying the proposed approach on several medical datasets are demonstrated at the end of this chapter.

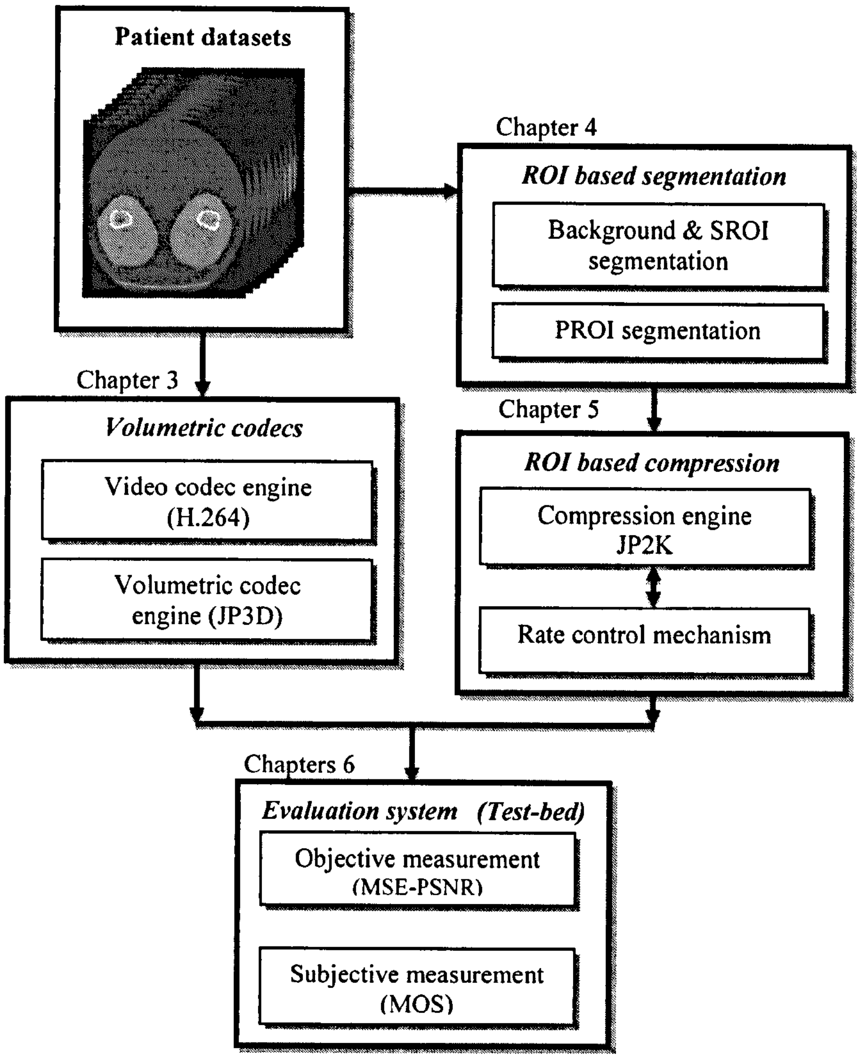


Figure 1-1 The thesis structure for the fully ROI driven medical image compression

(3) ROI based compression: This component calculates a variable level of compression in the *(x,y)* plane as well as in the *z* axis automatically for extracted regions, which are prepared from the ROIs based segmentation component, by the proposed rate control mechanism, and then images are compressed by the

Compression engine. The result passes to the Evaluation system for assessment. In Chapter 5, the proposed context and ROI based compression approach is presented in vascular medical imaging 3D CTA scan and also the proposed rate control algorithm is discussed in this chapter.

(4) Evaluation system (Test-bed): This component analyzes the obtained results from the ROI based compression and Volumetric codec based on several objective and subjective quality measurements and furthermore, a Graphic User Interface (GUI) is designed to represent the results. Chapter 6 is assigned to this component. In this chapter, the experimental results of applying the proposed method and volumetric codecs on patient datasets are reported. A robust performance evaluation schema (Test-bed) is designed to analyze the results based on the objective and subjective quality metrics. In the evaluation system, two groups and seven scenarios are considered and also a comparison between scenarios in each group is demonstrated in this chapter.

Finally, the summary of research achievements is compiled in Chapter 7.

2. LITERATURE REVIEW

This chapter presents a literature review on the relevant themes of this thesis, which are compression technologies, segmentation techniques in vascular systems, ROI based compression technologies and evaluation of compression techniques.

2.1. Compression Technologies

The goal of image compression is to reduce the number of bits as much as possible, while keeping the resolution and the visual quality of the reconstructed image as close to the original image as possible. The general framework for image compression is represented in Figure 2.1.

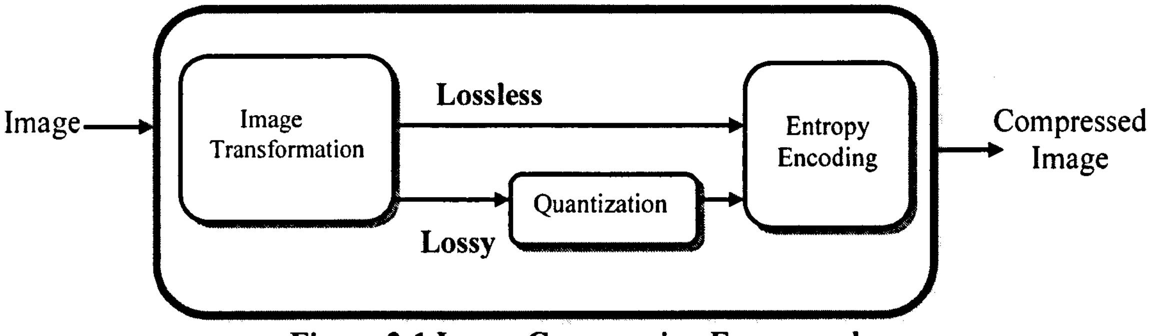


Figure 2-1 Image Compression Framework

Fundamentally, there are two types of image compression: lossy and lossless compression. In lossy compression, the image quality may be degraded in order to meet a given target data rate for storage and transmission. The applications for lossy coding are transmission of the image and video through a band limited network and efficient storage. The issue in lossy compression is how much we can reduce the degradation of the image quality given the data rate. In lossy compression, most algorithms transform pixels into a transform domain using the DCT (Discrete Cosine Transform) or the DWT (Discrete Wavelet Transform) [**31**]. The source of loss is either quantization of the transform coefficients or termination of the encoding at a given data rate. In order to meet a data rate budget, the transform coefficients are explicitly quantized with a given step size as in JPEG [**32**], JPEG2000 [**33**], MPEG-2 [**34**] and H.264 [**35**]. Implicit quantization is used in algorithms such as EZW [**36**] and SPIHT [**37**], which can be truncated at any point in the bitstream during encoding.

In lossless compression, the image after decompression is identical to that of the original. The issue in lossless coding is how much we can reduce the data rate. The main approach to lossless image compression is predictive coding or entropy encoding. For predictive coding, DPCM (Differential Pulse Code Modulation) [**38**] is often used. Linear predictors are used where the prediction is obtained by the linear combination of previously decoded neighbours. For entropy coding, Run-length coding [**39**], Huffman coding [**39**], or Arithmetic coding [**39**] is used. Context modelling can be included in the entropy encoding, which is to estimate the probability distribution function of the symbols conditioned on the context, so as to increase the compression performance. The context consists of a combination of neighbouring pixels already encountered [**40**]. The structure of predictive coding is determined by the number of neighbouring pixels used for the prediction, the weight of linear combination of neighbouring pixels, and the method of context modeling. The JPEG lossless uses DPCM and Huffman coding, or Arithmetic coding. Another method for lossless encoding is to use the reversible integer-to-integer wavelet transform [**41**]. The results of the integer wavelet transform are integers so as to recover the originals completely with the inverse integer wavelet transform.

Several standard compression technologies have been briefly reviewed in the following section.

JPEG Standard

The JPEG (Joint Photographic Experts Group) compression standard is a widely used compression method that includes both lossless and lossy techniques. It begins by breaking the image into 8×8 pixel blocks. Each image block has its DCT computed. The result is an 8×8 block of spectral coefficients with most of the information concentrated in relatively few coefficients in the upper left corner. Quantization is performed and then approximates the coefficients. An 8×8 table, called the quantization table, gives the value by which the corresponding coefficient is to be divided. The values are then converted into the nearest integer. These quantized coefficients are then reordered in a zigzag manner to group the largest values first, with long strings of zeros at the end that can be represented efficiently.

JPEG gives good compression results for lossy compression with the least complexity. However, its main disadvantage is the appearance of the blocking artifacts, especially at high compression ratios [**42**].

JPEG-LS

JPEG-LS compression algorithm is one of the standards for lossless compression of colour and grayscale images. LOCO-I (LOw COmplexity LOssless COmpression for Images) [**43**] is the algorithm at the core of JPEG-LS for lossless compression of images. It combines the simplicity of Huffman coding with the compression potential of context models.

Lossless image compression schemes often consist of two independent components: modelling and coding. The modelling components can be formulated as an inductive inference problem in which an image is observed pixel by pixel in some pre-defined order (e.g., raster-scan). The modelling component is tuned for efficient performance in conjunction with an extended family of Golomb-type codes [**44**] adaptively chosen, and an embedded alphabet extension for coding of low entropy image regions. LOCO-I algorithm attains compression ratios similar to those obtained with the state of the art schemes according to arithmetic coding [**43**].

JPEG-LS stands out as the best option when only lossless compression is of interest, providing the best compression efficiency at a low complexity but it is very susceptible to channel errors because it does not provide any error resiliency capabilities [**45**].

Basic Wavelet Compression

Although the JPEG lossy algorithm is good for general purposes, it has some drawbacks as it is applied to radiographic images. At high compression ratios, it produces prominent artifacts at block boundaries [46], and it cannot take advantage of patterns larger than the 8×8 pixel blocks. Wavelet based compression schemes usually outperform JPEG in terms of image quality at a given compression ratio, and the improvement can be dramatic at high compression ratios.

The DWT of an image is computed using a pair of high and low-pass filters with special mathematical properties [47]. Many such wavelet filters exist, but a number of groups have adopted the 9/7-tap bi-orthogonal filters, because these seem to work well in real world applications [47]. The two filters split the image into two components or sub-bands in each direction (each is half the original size). This produces four sub-band images as follows: one containing the low-frequency information, one each for the high-frequency information in the X or Y direction, and one for high-frequency information in both X and Y. The process is repeated on the low- frequency component as breaking it up into high-low and low-low components. If this process is performed n times, an n-level DWT is created. The DWT is effective for compression because it concentrates the information on a few coefficients, with most other coefficients being zero or close enough to zero that they can be considered zero without degrading the image.

Most wavelet compression algorithms compute a four or five level DWT, quantize the resulting coefficients, and efficiently encode the quantized coefficients. The quantization is performed by dividing each coefficient by a quantization parameter and rounding off to the nearest integer. Having a larger quantization parameter will result in more coefficients that are zero, and hence, will increases the compression ratio. Finally, encoding converts the coefficients into values that can be stored or transmitted efficiently.

Advanced Wavelet Techniques

The way that nonzero coefficients are encoded differentiates the advanced wavelet compression techniques. Advanced techniques capitalize on this tree based organization of the coefficients. The most well known of these techniques is Embedded Zerotree Wavelet coding (EZW) [36]. The second approach, called Set Partitioning In Hierarchical Trees (SPIHT) [37], was one of the early successful advanced wavelet techniques. It yielded significantly better results than conventional wavelet compression, with similar computational complexity. In addition to providing efficient compression, it also transmitted the compressed bitstream in which approximations of the most important coefficients (regardless of location) are transmitted first. The values of these coefficients are progressively refined, and the most important remaining information which yields the largest distortion reductions is transmitted next [37].

JPEG2000

Because JPEG was specified for computers that existed over a decade ago and also new technologies like wavelet had surpassed JPEG for many types of images, the JPEG group set out to update the standard known as JPEG2000 [33].

The JPEG2000 effort has been substantial. This group identified a number of shortcomings of the JPEG standard that JPEG2000 would address. Among these were:

1. Better performance at high compression ratios
2. A single code stream that would support irreversible and lossless compression
3. Support for many types of images (specifically including 16 bit medical images)
4. Support for many different environments (e.g., high performance local area network or low-speed wide area network)
5. Support for ROI coding

The algorithms in JPEG2000 include the best wavelet transforms and provide flexibility in the filters used and wavelet methods. It is radically different in the way it encodes information to allow seamless transition from irreversible to lossless image transmission. It allows applications to apply different compression ratios to different portions of an image. Finally, it provides mechanisms for user-specifiable pixel accuracy. Not all of these features exist in the first rollout of JPEG2000. The first step only supports wavelet image encoding. The most advanced features will be finalized as later steps.

Therefore, JPEG2000 not only removes the blocking artifacts [**46**] completely and improves the compression performance over the previous standards, but also includes a rich set of features [**45**].

JP3D

In order to support volumetric datasets, the JPEG committee decided to add a three dimensional extension, Part 10 [**48**] also referred to as JP3D [**49**], to the JPEG2000.

JP3D defines the volumetric extension for JPEG2000 that provides isotropic support for handling volumetric images with multiple components. It is specifically designed to be compatible with the other existing parts of the standard and as such offers exactly the same functionality as its two dimensional counterpart. But because JP3D properly extends the wavelet transformation and the entropy coding to three dimensions, it is able to deliver better compression results than what was previously possible by using only Part 1 [**33**] or Part 2 [**50**]. JP3D also solves the ambiguity between components and slices and handles volumetric datasets in an isotropic fashion, that is, regardless of the orientation of the data. Swapping any of the 3D axis makes no difference in compression efficiency or coding limitations. This is a huge advantage over other technologies, because it makes JP3D future proof as it is much better suited to handle high-resolution volumetric datasets.

H.264

The H.264 [**35**] is the latest generation standard for video encoding which has many goals. It provides better video quality at substantially lower bitrates, better error robustness, and better video quality at an unchanged bitrate in comparison with previous standards. The standard is further designed to give lower latency as well as better quality for higher latency. In addition, all these improvements compared to previous standards were to come without increasing the complexity of design so much that it would be impractical or expensive to build applications and systems. An additional goal was to provide enough flexibility to allow the standard to be applied to a wide variety of applications: for both low and high bitrates, for low and high resolution video, and with high and low demands on latency **[51]** **[52]**.

The core of H.264 is similar to that of all digital video coding standards but some new techniques have been used in this standard for instance motion compensation with variable block size, content adaptive binary arithmetic coding (CABAC) and multiple reference pictures(see section 3.4.2 for more details).

The H.264 represents a big leap in video compression technology with typically a 50% reduction of average bitrate for a given video quality compared with MPEG-2 and about a 30% reduction compared with MPEG-4 Part 2 **[53]**.

Discussion

Several compression technologies have been briefly reviewed in the above section. The choice of a compression technology for a particular application will depend on its requirements. In the cases where lossy compression is of interest and low complexity is of high priority, JPEG still provides a good solution **[45]**. However, it has some drawbacks when applied to medical images. It degrades ungracefully at high compression ratios, with prominent artifacts at block boundaries.

JPEG-LS stands out as the best option when only lossless compression is of interest, providing the best compression efficiency at a low complexity but it is very susceptible to channel errors because it does not provide any error resiliency capabilities. Advanced wavelet techniques yielded significantly better results than conventional wavelet compression with similar computational complexity. They appear superior to JPEG for medical images, but the lack of a single standard wavelet method has hampered comparison of results and cross validation **[54]**.

JPEG2000 and JP3D also provide the most flexible solution, combining good compression performance with a rich set of features including lossy to lossless compression, region of interest coding, error resilience, random code-stream access, and resolution scalability. Therefore, they can be good candidates for compressing medical images.

As for H.264, it represents a big leap in video compression technology with typically a 50% reduction of average bitrate for a given video quality compared with MPEG-2 and about a 30% reduction compared with MPEG-4 Part 2 and also it appears to be of interest in volumetric data encoding such as medical datasets.

Therefore, the requirements for selecting compression technologies in this research regarding their usage in medical imaging includes good compression performance with ROI features and also be supported in the DICOM standard[1]. JPEG2000, JP3D, and H.264 fulfil these requirements and will be discussed in detail in Chapter 3.

2.2. Segmentation Techniques in Vascular System

Segmentation of the peripheral arteries using CTA datasets aims to produce a more meaningful, in the sense of the clinical information provided, view of Peripheral Arterial Disease (PAD). This goal is achieved by vascular tree extraction and vessel quantification analysis. The vessel quantification analysis can be important to detect automatically the disease (occlusion in the peripheral arteries). However, it is not included in the aims of this research. The main target is to find peripheral arteries and the surrounding area as the regions of interest.

In the past two decades, there has been extensive research in the area of segmentation algorithms for vessels, whereas the topic of peripheral arteries was explored in only a limited amount of studies. Possible reasons for this discrepancy are the large size of the acquired data, the low significance of peripheral arterial disease in the medical community when compared with other ailments, the inherent computational burden, and

[1] DICOM currently supports the use of conventional JPEG standard for the lossless and lossy compression of image data, as well as JPEG-LS and JPEG2000. For video images, additional schemes supported include various levels and profiles of MPEG-2 and H.264 **[121]**.

the required high efficiency in the algorithmic development. The literature is therefore quite sparse and the existing studies mostly target specialized issues.

As such the existing vessel segmentation methods are studied and reviewed, the segmentation methods investigated are chosen based on the imaging modality. Within each imaging technique, various factors influence development of the segmentation methods. In addition, the application domain of the segmentation method is influencing parameters due to the location and structure variability of the vessels in the human body. For this reason, the segmentation techniques reviewed and presented in this section are related to the vessels in CTA datasets. Also the review is focused in particular on 3D segmentation methods.

There is no segmentation method, which can extract the entire proposed vascular structure. The reason for this is the large amount of parameters encountered in both the anatomical information (shape and abnormalities) and the imaging modality (image quality and noise). Therefore, several steps in the segmentation process are reported using various methods including, sometimes, hybrid methods **[55]**.

In the segmentation process, the initial care is given to the image examination in terms of quality, possibility of analysis and, more importantly, the possibility of improving the image quality without altering the information contained therein. This step is called pre-processing and it is used globally on the entire data. Some pre-processing algorithms are image enhancement filters, which aim to improve the signal to noise ratio **[56]**, that seek the enhancement of specific regions in the image (vessels) **[57]**, and interpolation methods **[58]** in order to create homogeneous data to create isotropic voxels. Some of these filters can be also used in the segmentation process in order to extract significant features of the vessels **[57]**.

In the segmentation process, the main step is the blood vessel delineation task. In some applications, the use of a threshold technique is able to achieve a good delineation **[59]**. In other cases, combining threshold methods with other image processing techniques can serve as a solution as in **[60]** where a threshold method and a connected component analysis are used to extract the coronary arterial tree from a 3D angiogram.

Another intensity based segmentation method is the region growing technique, which is widely used for the extraction of vessel tree. It is comparable with the threshold method being usually an intensity based technique and it also uses a connectivity criterion. The two techniques, threshold and region growing, can be used together for better results as described in **[61]**: a threshold value is manually determined and along a region growing technique is applied to segment the vessel tree in a 3D CTA. Another hybrid method is presented in **[62]** where a threshold method is used following a region growing technique, which in addition considers vessel structural information (growth orientation and size) and its smoothness acquired through morphology operations. The hybrid method was applied on cerebral vessel data from CTA.

Because the intensity and connectivity information were not enough for a successful delineation of the vessels in some applications, model based approaches were introduced and used widely in vessels segmentation. This category includes the following approaches: deformable models (active contours: snakes, geometric deformable models and deformable template matching approach), parametric models and generalized cylinders approaches. Deformable models have the advantage of adjustment to topology and shape characteristics of vessel through parametric curve fitting concept. They use an energy function defined by internal and external forces and is aimed to be minimized. The application of an active contour technique for the extraction of abdominal aorta in CTA data was successfully implemented in **[63]**. The parametric models use various parameters according to the shape of the object. Vessels are defined by circular or elliptic group of objects. In **[64]** a method for vessels reconstruction in X ray angiograms is presented. The method uses an elliptic parametric model to represent the vessels and behaves well in detecting bifurcations. For the arterial tree, successful extraction results were obtained by using template matching. This technique is then deformed by using various methods in order to match the investigated image. In **[65]** such a method is described with its application in recognition of vessels in angiograms. The model is according to the Gauss-Markov model and does not use pre-processing steps.

A plenary review of vessel segmentation techniques is presented in **[66]**. The authors classified the existing segmentation methods in six categories: pattern recognition techniques, model based approaches, tracking based approaches, artificial intelligence based approaches, neural network based approaches, and tube like object detection approaches. Each of these categories is examined in detail and the methods are described along the range of applicability.

The work of Kanistar [67] is also according to the above principle, and it investigates a variety of segmentation methods in order to segment peripheral arteries in CTA. Felkel [55] investigated the segmentation of peripheral arteries through an overview of segmentation methods of vessels in data obtained from DSA, MRI and CTA. Out of the seven methods investigated, just one was found to have applicability and successful results for PAD data. Felkel concluded that even with some adaptation of the techniques to the peripheral arteries, some of the algorithms do not offer satisfactory results. This is a valid concern considering the variability of the parameters regarding vessel anatomy and imaging modality. For example, when testing Frangi's algorithm [57], the results confirmed the aforementioned issues; initially this method was applied for MRA images, where high intensities from bones are not present, contrary to the CTA datasets. In addition, the investigated vessels had large diameter while the peripheral arteries can reach 1-2 mm in size. That being so, the combination with other methods has the potential to achieve much better results.

Discussion

The majority of the described methods have been used for vessel segmentation in CTA or MRI images. Because techniques designed for the segmentation of peripheral arteries are limited in the literature or researched for partial segments (segmentation of a specific part from the peripheral arteries), a selection of the above mentioned methods were investigated and their suitability in the case of PAD in CTA datasets.

Using intensity based approaches only, such as threshold, for segmenting peripheral arteries could result in overlap between vessels and bones due to the intensity similarity. The same problem would be encountered in using region growing methods. Morphology operations could overcome these issues due to their potential of dealing with shape and form of vessels and to the fact that they can alter the site of connectivity of different objects. Even though efficient, these operations are a computational burden considering the fact that they use a structural element, a mask, usually of small dimension, which needs to be convolved with the entire dataset. This is very inefficient and time consuming for PAD considering the large amount of images.

Image enhancement approaches represented good results in vessel segmentation, but their application was performed on MRA data. The difference in CTA images is that those blood vessels have similar intensities as the bone; therefore, the image enhancement might result in enhancing bone structures as well. The small diameter of peripheral arteries is also a drawback of this method.

Model based approaches have been widely used in the segmentation of blood vessels. It also contains a broad class of techniques, which take into consideration useful topological and structural information of the vessel. They are though computationally expensive due to the convergence criterion and in the case of PAD, which contains up to 2000 slices, can show low efficiency.

Therefore, a hybrid method will better fit the segmentation of peripheral arteries in PAD. The implemented techniques for segmenting of peripheral arteries will be presented in Chapter 5.

2.3. ROI Based Compression Technologies

ROI coding systems enable the important part of an image to encode at higher quality than the rest of an image. A number of ROI based coding approaches have been proposed in the literature [16]-[24].

The implicit ROI encoding method modifies the cost function used to allocate code block contributions to quality layers according to the ROI part rather than the whole image for distortion minimisation. Using this coding scheme, some background information and ROI appear in the early stage of decoding. However, the implicit ROI encoding method exhibits slow ROI quality improvement in progressive transmission because code blocks also contain certain amounts of background information. The main advantage of the implicit ROI coding is low complexity because no bit plane shifts occur at the encoder or decoder [16].

The Region of interest coding based on EBCOT is achieved by modifying the implicit ROI coding method. This new method reduces the priority of background coefficients in the ROI code-block without compromising algorithm complexity. It is suitable for applications in which it may be desirable to encode the ROI to a higher quality level than the background. The method improves the compression efficiency and is not compatible with ROI coding in the JPG2000 **[17]**.

The SCMShift (Selective Coefficient Mask Shift) is based on shifting the wavelet coefficients associated with the ROI to different sub-bands. The characteristics of this method are: (1) codification of multiple ROIs at various degrees of interest, (2) arbitrary shaped ROI coding, and (3) flexible adjustment of the compression quality of the ROI and the background **[23]**.

Semi-automatic region of interest identification algorithm using wavelets method is a new technique, where the texture and the edge information provided by the first level of the wavelet decomposition is used to segment the wavelet coefficients. This first level decomposition provides enough edge and texture information for image segmentation, allowing computational savings. A mask that outlines the ROI is determined based on the entropy calculation of the segmented regions. The advantage of this method is that the semi-segmentation process is entirely performed in the wavelet domain, not in the pixel domain and therefore, offering additional computational efficiency **[18]**.

The CPBAShift (Compensation-based Partial Bitplane Alternating Shift) divides all bit planes of ROI and background (BG) coefficients into two portions-Alternating Shift Portion (ASP) and Compensation Shift Portion (CSP). In ASP, the most significant bit planes of ROI and BG coefficients are shifted partially bit plane by bit plane. In CSP, the least significant bit planes of ROI and BG coefficients are scaled using compensation scheme according to the compression quality in ROI and BG **[19]**.

The Significance of Region of Interest Applied to MRI and CT Images in Teleradiology-Telemedicine is model based on Maxshift. ROI is compressed by lossless or near lossless methods while the background region is compressed with some loss of information. All regions are segmented manually **[20] [21]**.

The Partial Significant Bitplanes Shift (PSBShift) is one of ROI based coding method that combines the advantages of the two standard ROI coding methods defined in JPEG2000. The method not only supports arbitrarily shaped ROI coding without coding the shape, but also enables the flexible adjustment of compression quality in ROI and background **[22]**.

Lossy-to-lossless compression of volumetric medical images using 3D integer wavelet transforms is a method, which introduces a 3D integer wavelet packet transform structure and focuses on context modeling for efficient arithmetic coding of the transform coefficients **[24]**.

Discussion

Several ROI based coding techniques have been discussed in the above section; they have two disadvantages regarding their usage in medical image compression. Firstly, the segmentation of ROIs is not fully automated; therefore, it could be extremely time consuming to annotate the ROIs by a radiologist and secondly, for implementing those methods, some definite modifications on the standard image compression are required. This modification necessarily changes, at least in part, the standard of compression algorithms supported in the DICOM standard. The DICOM standard committee is highly conservative about adding new schemes, since the incremental benefit of slightly improved performance, or an extra feature, rarely justifies the risk of compromising interoperability. Medical devices are also highly regulated (e.g., by the FDA in the US), so they are also expensive to change given the burden of required testing and documentation **[27] [28]**. Conformance to the standard is mandatory for medical image compressions, which have been integrated into pre-existing PACS.

In order to increase the compression efficiency while properly addressing the above mentioned issues, a fully automated ROI based medical image compression in compliance with FDA regulation will be presented in Chapter 6.

2.4. Evaluation of Image Quality in ROI Based Compression Technologies

In lossy compression the reconstructed image is not the same as the original image, so individual pixel values may be different from the values in the original image. In order to evaluate the quality of the reconstructed image we need to introduce image quality metrics and also perceptual metrics, which are correlated to visual appearance. Obviously, if the human eye cannot distinguish original and reconstructed images then our compression algorithm performs very well.

Definition of perceptual metrics is a very difficult problem and it is not easy to find the point in the rate distortion[2] curve for an image where we transit from visually lossless to visually lossy compression **[68]** **[69]**. This transition point differs, depending upon the viewing angle, the viewing distance, the display medium and the viewer himself. Figure 2.1 shows the rate distortion curves for Mean Square Error (MSE) and visual appearance metrics. For high enough bitrates, there is no visual distortion. When MSE is sufficiently small, the original and decoded images are not perceptually distinguishable **[73]**.

In general, measurement of image quality can usually be classified into two categories: subjective and objective quality measurements. A subjective quality measurement, Mean Opinion Score (MOS), is truly definitive but too inconvenient, time consuming, and expensive **[70]**. Therefore, objective measurements are developed such as MSE (Mean Square Error), Mean Absolute Error (MAE), Peak Signal to Noise Ratio (PSNR), Structural SIMilarity (SSIM), Structural Content (SC), Maximum Difference measurement (MD), Laplacian Mean Square Error (LMSE), and Normalized Absolute Error (NAE) that are less time consuming than MOS but they do not correlate well with MOS **[71]** **[72]**.

[2] Rate distortion gives theoretical bounds for how much compression can be achieved using lossy data compression methods and the Rate distortion curve is a function, typically expressed as a graph, showing the accuracy of a Compressor and Decompresses (CODEC) system at various bitrates.

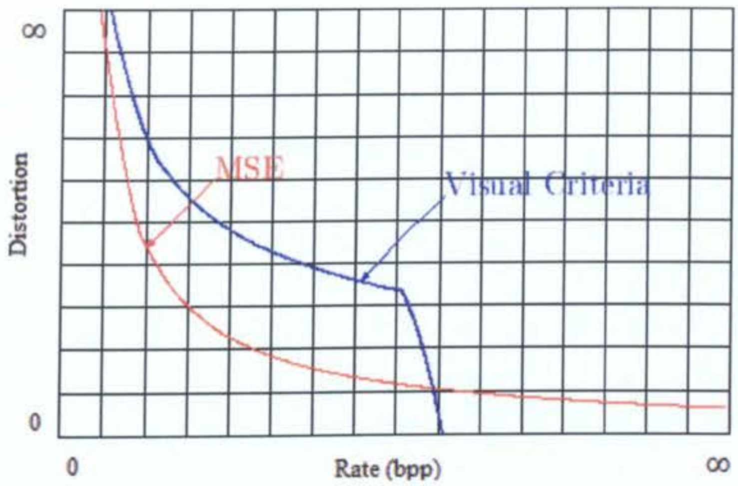


Figure 2-2 Rate distortion curves for MSE and visual appearance metrics [73]

A number of the most widely used image quality metrics are described in the following sections.

MSE and PSNR

The quality of the reconstructed images is evaluated in terms of Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR) **[68] [69]**. These objective quality metrics are the most commonly used for measuring the amount of distortion in reconstructed images.

MSE is the mean square error between two images, and is given by:

$$MSE = \frac{\sum(I_{ij} - C_{ij})^2}{N} \tag{2.1}$$

Where I_{ij} and C_{ij} represent the luminance of pixel *(i,j)* in the original and the compressed images, respectively. *N* is the number of pixels in the image.

PSNR is evaluated in decibels and is inversely proportional to MSE and defined as:

$$PSNR = 10 * log_{10}\left(\frac{(2^B - 1)^2}{MSE}\right) \tag{2.2}$$

Where B is the number of bits per pixel. Clearly, large values of MSE indicate a poor quality of the compressed image, while higher PSNR values correspond to a better image quality. MSE and PSNR have simple formulation with less computing time, have clear physical meanings, and are also mathematically convenient in the context of optimization [74] [75].

SSIM

The Structural SIMilarity (SSIM) is one of the standard criteria for the assessment of image quality and fidelity. The basic form of SSIM is very easy to understand. Suppose that x and y are local image patches taken from the same location of two images that are being compared. The local SSIM index measures the similarities of three elements of the image patches: the similarity $l(x,y)$ of the local patch luminances (brightness values), the similarity $c(x,y)$ of the local patch contrasts, and the similarity $s(x,y)$ of the local patch structures. These local similarities are expressed using simple, easily computed statistics and combined to form local SSIM **[68]**.

$$S(x,y) = l(x,y) \cdot c(x,y) \cdot s(x,y) = \left(\frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}\right) \cdot \left(\frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}\right) \cdot \left(\frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}\right) \tag{2.3}$$

Where μ_x and μ_y are (respectively) the local sample means of x and y, σ_x and σ_y are (respectively) the local sample standard deviations of x and y, and σ_{xy} is the sample cross correlation of x and y after removing their means. The items C_1, C_2, and C_3 are small positive constants that stabilize each term, so that near zero sample means, variances, or correlations do not lead to numerical instability. Actually, even if $C_1 = C_2 = C_3 = 0$, SSIM usually works quite well. This choice of constants defined the first and simplest version of SSIM with the universal image quality index (UQI) **[68]**.

The SSIM index is symmetric: $s(x,y) = s(y,x)$, so that two images being compared give the same index value regardless of their ordering. It is also bounded: $-1 < s(x,y) \leq 1$, achieving maximum value $s(x,y)$= 1 if and only if $x = y$. The SSIM index is computed locally within a sliding window that moves pixel-by-pixel across the image, resulting in an SSIM map. The SSIM score of the entire image is then computed by pooling the SSIM map, e.g., by simply averaging the SSIM values across the image.

SSIM measures similarity with greater accuracy and consistency than MSE and PSNR, but incurs greater computational cost **[74] [76]**.

MOS

As human observers make the final assessment of image quality, the most reliable judgment is based on human perception. Mean Opinion Score (MOS) is a numerical method of expressing image quality according to visual inspection and has been widely used in the literature for many years [70]-[72], [77]-[81]. In MOS rating, a group of non-experts and experts may be used. Non-experts, representing average viewers, have little or no technical background in image compression, while experts are trained individuals having much familiarity with the relevant compression technologies. The Table 2.1 shows the rating scales, which researchers have used.

Evaluation performed by the observers take two forms: Absolute and comparative. Absolute evaluation is a process whereby the observer assigns to an image a category in a given rating scale, whereas comparative evaluation is the ranking of a set of images from best to worst **[82]**. Subsequent statistical analysis can then highlight averages, variability, and other trends in the data.

$$MOS(k) = \frac{1}{M}\sum_{i=1}^{M} S(i,k) \tag{2.4}$$

Where $S(i,k)$, is rating scale of image k by viewer i, M is the number of observers.

MOS is the most reliable judgment and commonly used quality metric but too inconvenient and quite expensive in terms of time and human resources **[70]**.

Discussion

The classification of image quality metrics, objective and subjective are reviewed; moreover, several image quality measurements in each category are explained such as

Table 2-1 The rating scales, which have been used by researchers

5	Excellent	10,9	Very good
4	Good	8.7	Good
3	Fair	6,5,4	Fair
2	Poor	3,2	Bad
1	Unsatisfactory	1,0	Very bad
7	Best	3	Much better
6	Well above average	2	Better
5	Slightly above average	1	Slightly better
4	Average	0	Same
3	Slightly below average	-1	Slightly worse
2	Well below average	-2	Worse
1	Worst	-3	Much worse
1	Not noticeable but only slight improvement		
2	Just noticeable		
3	Definitely noticeable		
4	Impairment not objectionable		
5	Somewhat objectionable		
6	Definitely objectionable		
7	Extremely objectionable		

MOS, MSE, PSNR, and SSIM. It has been discussed that the subjective quality measurements are truly definitive but quite expensive in terms of time and human resources; on the other hand, the objective ones are less time consuming; however, they do not correlate well with the subjective ones.

In order to have an accurate assessment of the proposed compression method, we have decided to use both objective and subjective quality metric techniques for evaluating the reconstructed images. In the objective category method, the MSE and PSNR will be considered because they have simple formulation with less computing time, have clear physical meanings, and are also mathematically convenient in the context of optimization. MOS will be selected in the subjective classification because it is representative of the perceptual quality of the visual stimulus.

2.5. Summary

The literature review has been presented in four categories: (1) Compression technologies, (2) Segmentation techniques in vascular systems, (3) ROI based compression technologies, and (4) Evaluation of image quality in ROI based compression technologies. A summary of each category is laid out as follows:

(1) Compression technologies: Several compression technologies are briefly reviewed. Advantages and disadvantages of each one are discussed as well. The choice of a compression technology for a particular application depends on its requirements. The requirements for selecting compression technologies in this research regarding their usage in medical imaging includes good compression performance with ROI features and also should be supported in the DICOM standard. JPEG2000, JP3D, and H.264 fulfil these requirements.

(2) Segmentation techniques in vascular system: Segmentation of the peripheral arteries using CTA datasets aims to produce a more meaningful, in the sense of the clinical information provided, view of Peripheral Arterial Disease (PAD). This goal is achieved by vascular tree extraction and vessel quantification analysis. The vessel quantification analysis can be important to detect automatically the disease

(occlusion in the peripheral arteries). However, it is not included in the aims of this research. The main target is to find peripheral arteries as the region of interest and the surrounding area will be added to it for increasing the accuracy of segmentation.

The majority of the described methods have been used for vessel segmentation in CTA or MRI images based on three approaches: intensity based, image enhancement, and model based. Because techniques designed for the segmentation of peripheral arteries are limited in the literature or researched for partial segments (segmentation of a specific part from the peripheral arteries), a selection of the existing methods are investigated and their suitability in the case of PAD in CTA datasets.

Using one approach only for segmenting peripheral arteries could result in computational cost, time consumption, and overlap between vessels and bones due to the intensity similarity. Therefore, a hybrid method will better fit the segmentation of peripheral arteries in PAD.

(3) ROI based compression technologies: Several ROI based coding techniques have been discussed in this chapter; they have two disadvantages regarding their usage in medical image compression. Firstly, the segmentation of ROIs is not fully automated; therefore, it could be extremely time consuming to annotate the ROIs by a radiologist and secondly, for implementing those methods, some definite modifications on the standard image compression are required. This modification necessarily changes, at least in part, the standard of compression algorithms approved in the DICOM standard. The DICOM standard committee is highly conservative about adding new schemes, since the incremental benefit of slightly improved performance, or an extra feature, rarely justifies the risk of compromising interoperability. Medical devices are also highly regulated, so they are also expensive to change given the burden of required testing and documentation. Conformance to the standard is mandatory for medical image compressions that have been integrated into pre-existing PACS.

In order to increase the compression efficiency while properly addressing the above mentioned issues, a fully automated ROI based medical image compression in compliance with FDA regulation will be presented.

(4) Evaluation of image quality in ROI based compression technologies: The classification of image quality metrics, objective and subjective are reviewed. It has discussed that the subjective quality measurements are truly definitive but quite expensive in terms of time and human resources; on the other hand, the objective ones are less time consuming; however, they do not correlate well with the subjective ones. In order to have an accurate assessment of the proposed compression method, I have decided to use both objective and subjective quality metric techniques for evaluating the reconstructed images.

In the objective category method, MSE and PSNR are considered because they have simple formulation with less computing time, have clear physical meaning, and are also mathematically convenient in the context of optimization. MOS is selected in subjective classification because it is representative of the perceptual quality of the visual stimulus.

3. COMPRESSION TECHNOLOGIES

This chapter presents the architecture of three image and video compression technologies used throughout this research. It starts with JPEG2000 follow by JP3D and H.264.

3.1. JPEG2000 Standard

One of the most popular compression standards in the medical community is JPEG2000 standard. There are some real incentives behind the development of the JPEG2000 standard. It provides higher compression efficiency than the other standards, (i.e., JPEG standard) and supports the following rich set of features **[10]**, [29], [30], [83].

- Improved compression efficiency
- Lossy to lossless compression
- Multiple resolution representation
- Embedded bitstream (progressive decoding and SNR scalability)
- Tiling
- Region of interest (ROI) coding
- Error resilience
- Random code-stream access and processing
- A more flexible file format

The terminology used in this chapter is as follows:

- Tile: consists of a whole image or a rectangular (non overlapping) sub-image

- Component: (normally) a single colour plane of an image
- Resolution level: a collection of wavelet sub-bands that have the same span with respect to the original samples
- Code-block: a rectangular grouping of wavelet coefficients from the same sub-band and tile component
- Precinct: a sub-division of a tile-component within a resolution
- Layer: a collection of encoded bit planes or sub bit planes, from one or more code-blocks of a tile component (layers have an order for encoding and decoding that must be preserved.)

The encoding process consists of the following stages: First, for each colour component of the input image, the pixel data is transformed using reversible or irreversible wavelet transformation and an orientation tree frequency sub-band structure is generated. The wavelet transform coefficients are then quantized into integer indices. Afterwards, the indices of each frequency sub-band are divided into small code blocks (e.g., 32x32 pixels) and bit plane coding is performed in each code block independently. The coded data constructs several quality layers. Finally, the code blocks are also grouped into precincts with a nominal size for each frequency sub-band. The code coming from each precinct, layer, resolution level, and component will be wrapped into a packet and all the packets are organized to form the final bitstream in a certain progressive order. Block diagram of this standard is shown in Figure 3.1.

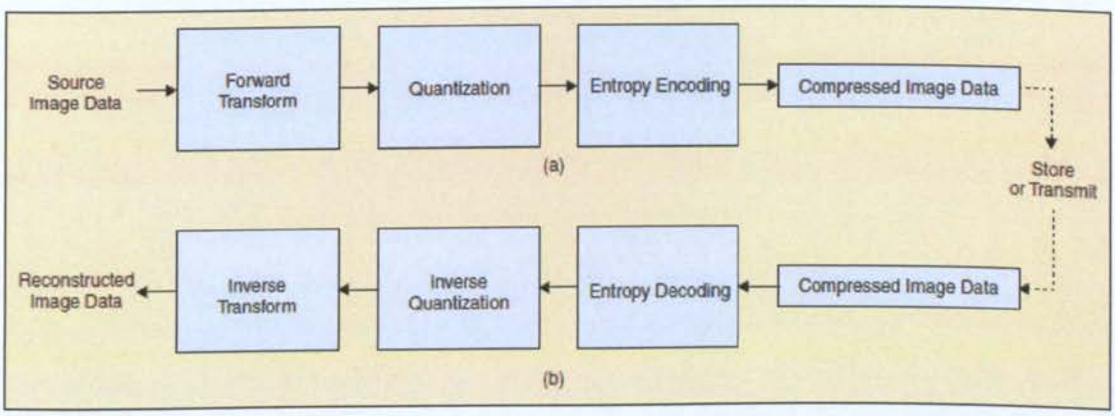


Figure 3-1 Encoding process [29]

Some features of this standard have been described in details in the following sections.

3.1.1. Tiling

The term "tiling" refers to the partition of the original (source) image into rectangular non overlapping blocks (tiles), which are compressed independently, as though they were entirely distinct images. All operations, including component mixing, wavelet transform, quantization, and entropy coding are performed independently on the image tiles (Figure 3.2). The tile component is the basic unit of the original or reconstructed image. Tiling reduces memory requirements, and since they are also reconstructed independently, they can be used for decoding specific parts of the image instead of the whole image. All tiles have exactly the same dimensions, except probably be those at the boundary of the image. Arbitrary tile sizes are allowed up to the entire image (i.e., the whole image is regarded as one tile). Components with different subsampling factors[1] are tiled with respect to a high-resolution grid, which ensures spatial consistency on the resulting tile components. As expected, tiling affects the image quality both subjectively and objectively. Smaller tiles create more tiling artifacts compared to larger tiles. In other words, larger tiles perform visually better than smaller tiles.

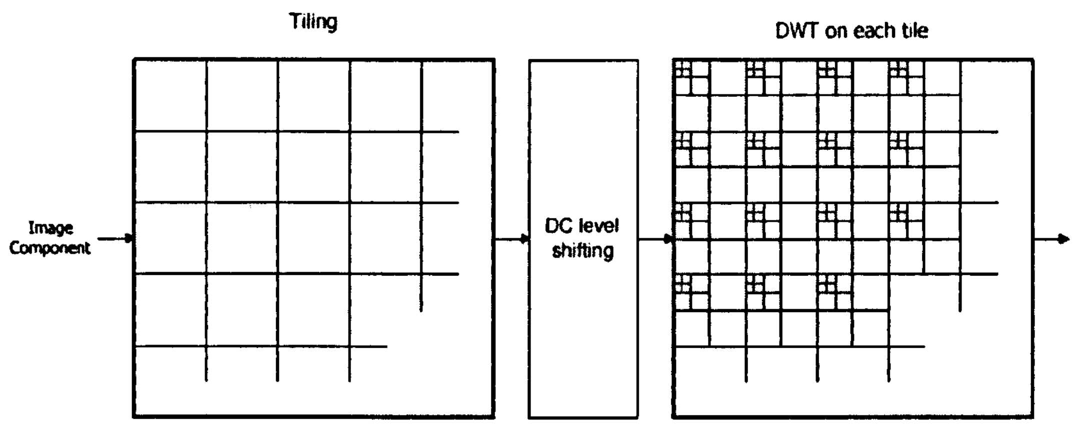


Figure 3-2 Image tiles [84]

[1] The subsampling factors for a component indicate the scaling factor between the component dimensions and the image dimensions. If an image has dimensions X_{Image} by Y_{Image}, and a component has subsampling factors X_{Subs} by Y_{Subs}, the dimensions of the component will be the ceiling of X_{Image}/X_{Subs} by the ceiling of Y_{Image}/Y_{Subs}. For example, suppose an image has an overall size of 512 by 512 samples. An image component that has subsampling factors of 4 by 4 will have dimensions 128 by 128 samples.

3.1.2. Component Transformations

JPEG2000 supports multiple-component[2] images. Different components neither need not have the same bit depths (number of bits per pixel) nor need to be all signed or unsigned. For reversible (i.e., lossless) systems, the only requirement is that the bit depth of each output image component must be identical to the bit depth of the corresponding input image component. Component transformations improve compression and allow for visually relevant quantization (This feature is optional). The standard supports two different component transformations, one irreversible component transformation (ICT) that can be used for lossy coding and one reversible component transformation (RCT) that may be used for lossless or lossy coding (in addition to encoding without colour transformation).

The block diagram of JPEG2000 multiple-component encoder is depicted in Figure 3.3. Without restricting the generality, only three components are shown in the figure. These components could correspond to the RGB[3] of a colour image. Since the ICT may only be used for lossy coding, it may only be used with the 9/7 irreversible wavelet transform (see also section 3.1.3).

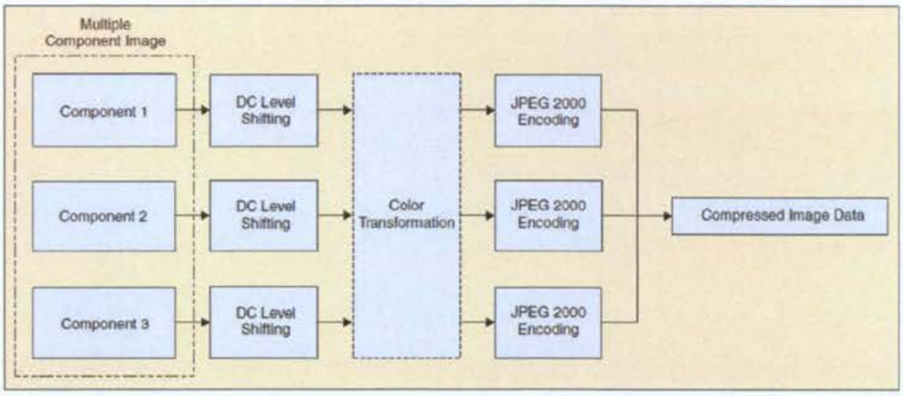


Figure 3-3 The JPEG2000 multiple-component encoder [29]

[2] The input images to JPEG2000 may contain one or more components. Although a typical colour image would have three components (for example, red-green-blue (RGB) [**122**] or luminance, chroma blue, chroma red (YC_bC_r)), up to 16384 components can be specified for an input image to accommodate multi spectral or other types of imagery.

[3] Colour images are displayed in different styles. The most classical style is the RGB representation where each colour channel (red, green, blue) is encoded in a separate array containing the respective colour intensity values.

The forward and the inverse ICT transformations are achieved by means of (3.1) and (3.2) respectively [**85**].

$$\begin{pmatrix} Y \\ C_b \\ C_r \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.16875 & -0.33126 & 0.5 \\ 0.5 & -0.41869 & -0.08131 \end{pmatrix} \cdot \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{3.1}$$

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 1.0 & 0 & 1.402 \\ 1.0 & -0.34413 & -0.71414 \\ 1.0 & 1.772 & 0 \end{pmatrix} \cdot \begin{pmatrix} Y \\ C_b \\ C_r \end{pmatrix} \tag{3.2}$$

Since the RCT may be used for lossless or lossy coding, it may only be used with the 5/3 reversible wavelet transform (see also section 3.1.3). The RCT is a decorrelating transformation, which is applied to the three first components of an image. Three goals are achieved by this transformation, namely, colour decorrelation for efficient compression, reasonable colour space with respect to the human visual system for quantization, and ability of having lossless compression, i.e., an exact reconstruction with finite integer precision. For the RGB components, the RCT can be seen as an approximation of a YUV transformation. All of the three components shall have the same sampling parameters and the same bit depth. There shall be at least three components if this transformation is used. The forward and inverse RCT is performed by means of (3.3) and (3.4), respectively, where the subscript r stands for reversible and $\lfloor\ \rfloor$ is for floor function [**85**].

$$\begin{pmatrix} Y_r \\ V_r \\ U_r \end{pmatrix} = \begin{pmatrix} \left\lfloor \frac{R+2G+B}{4} \right\rfloor \\ R - G \\ B - G \end{pmatrix} \tag{3.3}$$

$$\begin{pmatrix} G \\ R \\ B \end{pmatrix} = \begin{pmatrix} Y_r - \left\lfloor \frac{U_r+V_r}{4} \right\rfloor \\ V_r + G \\ U_r + G \end{pmatrix} \tag{3.4}$$

An effective way to reduce the amount of data in JPEG is to use an RGB to YC_rC_b decorrelation transform followed by subsampling of the chrominance (C_r,C_b) components. This is not recommended for use in JPEG2000, since the multi resolution nature of the wavelet transform may be used to achieve the same effect. For example, if HL, LH, and HH frequency sub-bands of a component wavelet decomposition (see section 3.1.3 for details) are discarded and all the other frequency sub-bands are retained, a 2:1 subsampling is achieved in the horizontal and vertical dimensions of the component.

3.1.3. Discrete Wavelet Transforms (DWT)

Wavelet transform is used for the analysis of tile components into a series of decomposition levels. These decomposition levels contain a number of frequency sub-bands, which consist of coefficients that describe the horizontal and vertical spatial frequency characteristics of the original tile component.

To perform the forward DWT, the standard uses a one dimensional (1D) sub-band decomposition of a 1D set of samples into low-pass and high-pass samples. Low-pass samples represent a down-sampled low-resolution version of the original set. High-pass samples represent a down-sampled residual version of the original set needed for the perfect reconstruction of the original set from the low-pass set. The 1D DWT can be easily extended to two dimensions (2D) by applying 1D transforms in the horizontal and vertical directions as shown in Figure 3.4. This results in four smaller image blocks; one with low resolution (LL), one with high vertical resolution and low horizontal resolution (HL), one with low vertical resolution and high horizontal resolution (LH), and one with all high resolution (HH). This process may be repeated on the low-resolution image block. This procedure is called dyadic decomposition. In Part I of JPEG2000 standard only power of 2 decompositions are allowed in the form of dyadic decomposition. Figure 3.5 shows a 3-level, dyadic decomposition and the corresponding labelling for each sub-band.

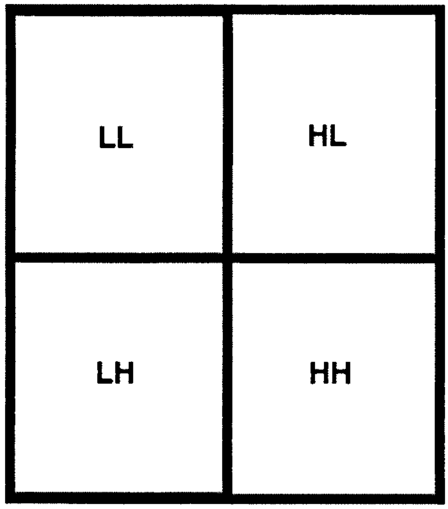


Figure 3-4 2D Wavelet decomposition level

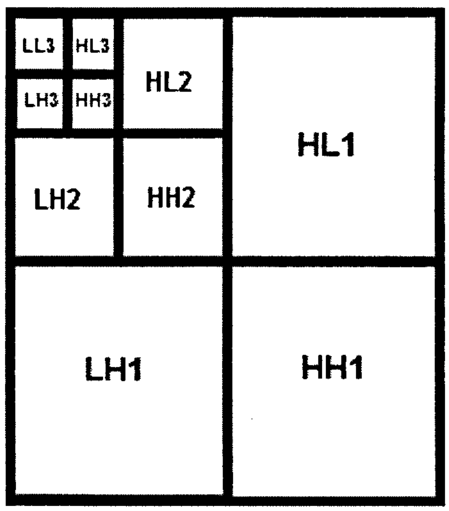


Figure 3-5 The sub-band labelling scheme for a 3-level dyadic decomposition

The DWT can be irreversible or reversible. The default irreversible transform is implemented by means of the Daubechies 9-tap/7-tap filter. The default reversible transform is implemented by means of the Le Gall 5-tap/3-tap filter.

The standard can support two filtering modes: convolution based and lifting based. Convolution based filtering consists in performing a series of dot products between the two filter (9/7 filter, 5/3 filter) masks and the extended 1D signal[4]. Lifting based filtering consists of a sequence of very simple filtering operations for which alternately odd sample values of the signal are updated with a weighted sum of even sample values, and even sample values are updated with a weighted sum of odd sample values. For the reversible (lossless) case, the results are rounded to integer values. The lifting based filtering for the 5/3 analysis filter is achieved by using equation (3.5) and (3.6) **[86]**.

$$y(2n+1) = x_{ext}(2n+1) - \left\lfloor \frac{x_{ext}(2n)+x_{ext}(2n+2)}{2} \right\rfloor \tag{3.5}$$

$$y(2n) = x_{ext}(2n) - \left\lfloor \frac{y(2n-1)+y(2n+1)+2}{4} \right\rfloor \tag{3.6}$$

Where x_{ext} is the extended input signal, y is the output signal, and n is sample value of signal. The 5/3 filter allows repetitive encoding and decoding of an image without any loss. Of course, this is true when the decompressed image values are not clipped when they fall outside the full dynamic range (i.e., 0-255 for an 8 b/p image). Traditional wavelet transform implementations require the whole image to be buffered and the filtering operation to be performed in vertical and horizontal directions. While filtering in the horizontal direction is very simple, filtering in the vertical direction is more cumbersome. Filtering along a row requires one row to be read; filtering along a column requires the whole image to be read. The line based wavelet transform overcomes this difficulty, providing exactly the same transform coefficients as the traditional wavelet transform implementation. However, the line based wavelet transform alone does not provide a complete line based encoding paradigm for JPEG2000. A complete row based coder has to take also into account all the subsequent coding stages up to the entropy coding.

3.1.4. Quantization

After transformation, all coefficients are quantized. Uniform scalar quantization with dead-zone about the origin is used in Part 1 and trellis coded quantization (TCQ) **[86]** in

[4] Before any wavelet transform is computed for a tile, the pixels of the tile may have to be extended by a simple extension method termed periodic symmetric extension **[87]**. This method is used to extend the signal at both boundaries by inculding the mirror images of the samples around the boundaries .

Part 2 of the standard. Quantization is a process by which the coefficients are reduced in precision. This operation is lossy, unless the quantization step is 1 and the coefficients are integers, as produced by the reversible integer 5/3 wavelet. According to the equation (3.7) [**88**], each of the transform coefficients $a_b(u,v)$ of the sub-band b is quantized to the value $q_b(u,v)$ in which (u,v) is the pixel position in an image.

$$q_b(u,v) = sign(a_b(u,v)) \left\lfloor \frac{|a_b(u,v)|}{\Delta_b} \right\rfloor \tag{3.7}$$

The quantization step-size Δ_b is represented relative to the dynamic range of sub-band b. In other words, the JPEG2000 standard supports separate quantization step-sizes for each sub-band. However, one quantization step-size is allowed per sub-band. The dynamic range depends on the number of bits used to represent the original image tile component and on the choice of the wavelet transform. All quantized transform coefficients are signed values even when the original components are unsigned. These coefficients are expressed in a sign-magnitude representation prior to coding. For reversible compression, the quantization step-size is required to be one.

3.1.5. Framing

After quantization, each sub-band is divided into rectangular blocks, i.e., non overlapping rectangles. Three spatially consistent rectangles (one from each sub-band at each resolution level) comprise a packet partition location or precinct. Each precinct is further divided into non overlapping rectangles, called code blocks, which form the input into the entropy coder (Figure 3.6).

The size of the code block is typically 64 × 64 and no less than 32 × 32. Within each sub-band, the code blocks are visited in raster order. These code blocks are then coded a bit plane at a time starting with the most significant bit plane with a nonzero element to the least significant bit plane. Each code block is coded entirely independently, without reference to other blocks in the same or other sub-bands, something that is in contrary to the approach adopted by the zero-tree coder described by Usevitch [**87**]. This independent embedded block coding offers significant benefits, such as spatial random access to the image content, efficient geometric manipulations, error resilience, parallel computations during coding or decoding, etc. The individual bit planes of the coefficients in a code block are coded within three coding passes. Each of these coding passes collects contextual information about the bit plane data. The arithmetic coder uses this contextual information and its internal state to generate a compressed bitstream. Different termination mechanisms allow different levels of independent extraction of this coding pass data.

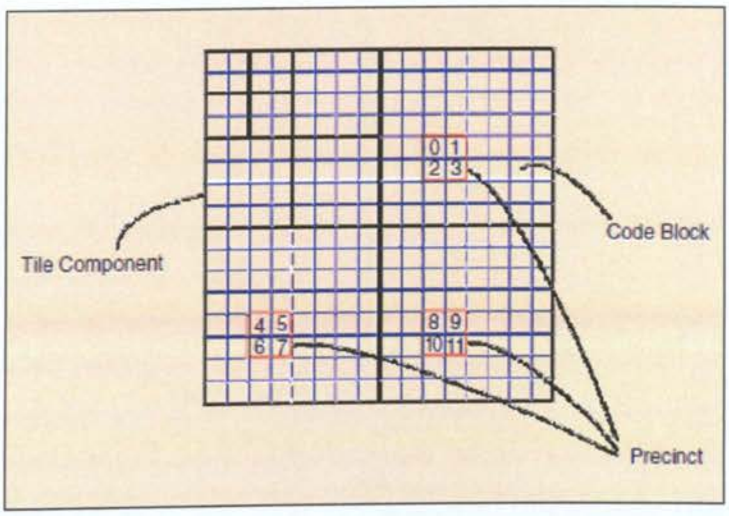


Figure 3-6 Partition of a tile component into code blocks and precincts [88]

Each bit plane of a code block is scanned in a particular order (Figure 3.7). Starting from the top left, the first four bits of the first column are scanned and then the first four bits of the second column, until the width of the code block is covered. Then the second four bits of the first column are scanned and so on. A similar vertical scan is continued for any leftover rows on the lowest code blocks in the sub-band **[29]**. This stripe height of four has been carefully selected to facilitate efficient hardware and software implementations. Each coefficient bit in the bit plane is coded in only one of the three coding passes, namely the significance propagation, the magnitude refinement, and the cleanup pass. For each pass, contexts provided to the arithmetic coder are created.

During the significance propagation pass, a bit is coded if its location is not significant, but at least one of its eight-connected neighbours is significant. Nine context bins are created based on how many and which ones are significant. If a coefficient is significant then it is given a value of 1 for the creation of the context, otherwise it is given a value of 0. The mapping of the contexts also depends on which sub-band (at a given decomposition level) the code block is in. The significance propagation pass includes only bits of coefficients that were insignificant (the significance bit has yet to be encountered) and have a nonzero context.

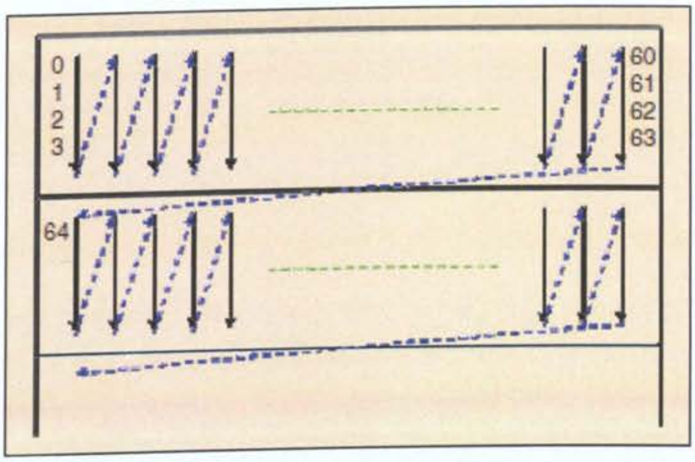


Figure 3-7 Scan pattern of each bit plane of each code block [88]

All other coefficients are skipped. The context is delivered to the arithmetic decoder (along with the bitstream) and the decoded coefficient bit is returned. The second pass is the magnitude refinement pass. During this pass, all bits that became significant in the previous bit plane are coded. The magnitude refinement pass includes the bits from coefficients that are already significant (except those that have just become significant in the immediately preceding significance propagation pass). The context used is determined by the summation of the significance state of the horizontal, vertical, and diagonal neighbours. These are the states as currently known to the decoder, not the states used before the significance decoding pass. Further, it is dependent on whether this is the first refinement bit (the bit immediately after the significance and sign bits) or not.

The final pass is the clean-up pass in which all bits not encoded during the previous passes are encoded (i.e., coefficients that are insignificant and had the context value of zero during the significance propagation pass). The cleanup pass not only uses the neighbour context, like that of the significance propagation pass, but also a run-length context. Run length coding occurs when all four locations in the column of the scan are insignificant and each has only insignificant neighbours.

3.1.6. Rate Control

Entropy coding is achieved by means of an arithmetic coding system that compresses binary symbols relative to an adaptive probability model associated with each of 18 different coding contexts. The MQ [40] coding algorithm is used to perform this task and to manage the adaptation of the conditional probability models. This algorithm has been selected in part for compatibility reasons with the arithmetic coding engine used by the JBIG2 [40] compression standard and every effort has been made to ensure commonality between implementations and surrounding intellectual property issues for JBIG2 and JPEG2000. The recursive probability interval subdivision of Elias coding [40] is the basis for the binary arithmetic coding process. With each binary decision, the current probability interval is subdivided into two subintervals, and then the code-stream is modified (if necessary). Therefore, it points to the base (the lower bound) of the probability subinterval assigned to the symbol, which occurred. Since the coding process involves addition of binary fractions rather than concatenation of integer code words, the most probable binary decisions can often be coded at a cost of much less than one bit per decision. As mentioned above, JPEG2000 uses a very restricted number of contexts for any given type of bit.

This allows for rapid probability adaptation and a decrease in the cost of independently coded segments. The context models are always reinitialised at the beginning of each code block and the arithmetic coder is always terminated at the end of each block (i.e., once, at the end of the last sub bit plane). This is also useful for error resilience.

In addition to the above, a lazy coding mode is used to reduce the number of symbols that are arithmetically coded. According to this mode, after the fourth bit plane is coded, the first and second pass are included as raw (uncompressed data), i.e., the MQ coder is bypassed, while only the third coding pass of each bit plane employs arithmetic coding. Lazy coding has a negligible effect on compression efficiency for most natural images, but not for compound imagery.

3.1.7. ROI Coding

The functionality of ROI is important in applications where certain parts of the image are of higher importance than others. In such a case, these regions need to be encoded at higher quality than the background. During the image transmission, these regions need to be transmitted first or at a higher priority, (for example in the case of progressive transmission). The ROI coding scheme in Part 1 of the standard is based on MAXSHIFT method [**85**]. The ROI coding scheme in Part 2 of the standard is based on the General Scaling method [**85**]. The principle of General Scaling and MAXSHIFT methods is to scale (shift) coefficients so that the bits associated with the ROI are placed in bit planes higher than those associated with the background as shown in Figure 3.8.

General Scaling method can support only regular shapes such as ellipses and rectangles for ROI, but MAXSHIFT method does not support different ROIs with different compression rates [**29**], [**85**], [**89**].

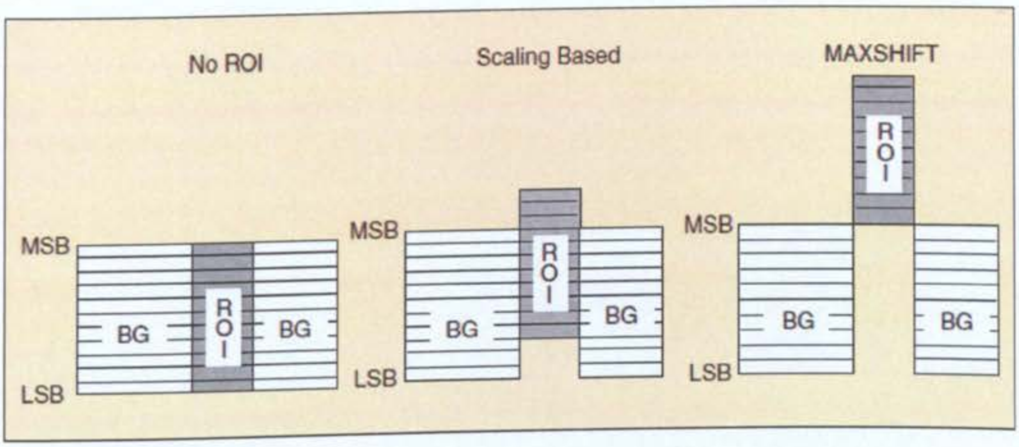


Figure 3-8 Scaling of ROI coefficients [85]

The General Scaling method is implemented as follows:

1. The wavelet transform is calculated.

2. If an ROI has been chosen, a mask (ROI mask) is derived, indicating the set of coefficients required for up to lossless ROI.

3. The wavelet coefficients are quantized. Quantized coefficients are stored in a sign magnitude representation. Magnitude bits comprise the most significant part of the implementation precision used (one of the reasons for this is to allow for downscaling of the background coefficients).

4. Coefficients outside the ROI are downscaled by a specified scaling value.

5. The resulting coefficients are progressively entropy encoded (with the most significant bit planes first).

The decoder reverses these steps to reconstruct the image (step 2 is still performed before step 3). The general scaling method requires the scaling value assigned to the ROI and the coordinates of the ROI to be added to the bitstream. The decoder performs also the ROI mask generation but scales up the background coefficients in order to recreate the original coefficients.

In the MAXSHIFT method, the encoder scans the quantized coefficients and chooses a scaling value s such that the minimum coefficient belonging to the ROI is larger than the maximum coefficient of the background (non ROI area, background). The decoder gets the bitstream and starts decoding. Every coefficient that is smaller than s must belong to the background and is therefore scaled up. The only thing that the decoder needs to do is upscaling of the received background coefficients.

3.1.8. Error Resilience

Error resilience is one of the most desirable properties in mobile and Internet applications. JPEG2000 uses a variable-length coder (arithmetic coder) to compress the quantized wavelet coefficients. Variable-length coding is known to be prone to channel or transmission errors. A bit error results in loss of synchronization at the entropy decoder and the reconstructed image can be severely damaged. To improve the performance of transmitting compressed images over error prone channels, error resilient bit stream syntax and tools are included in the standard. The error resilience tools deal with channel errors using the following approaches: data partitioning and resynchronization, error detection and concealment, and Quality of Service (QoS) transmission based on priority.

Error resilience is achieved at the entropy coding level and at the packet level. Entropy coding of the quantized coefficients is performed within code blocks. Since encoding and decoding of the code blocks are independent processes, bit errors in the bitstream of a code block will be restricted within that code block. To increase error resilience, termination of the arithmetic coder is allowed after every coding pass and the contexts may be reset after each coding pass. This allows the arithmetic decoder to continue the decoding process even if an error has occurred. The lazy coding mode is also useful for error resilience. This relates to the optional arithmetic coding bypass in which bits are fed as raw bits into the bitstream without arithmetic coding. This prevents the error propagation types which variable length coding is susceptible to it **[30]**.

At the packet level, a packet with a resynchronization marker allows spatial partitioning and resynchronization. This is placed in front of every packet in a tile with a sequence number starting at zero and incremented with each packet.

3.1.9. Scalability

Scalable coding of still images means the ability to achieve coding of more than one level of quality and/or resolutions simultaneously. Scalable image coding involves generating a coded representation (bitstream) in a manner that facilitates the derivation of images of more than one level of quality and/or resolution by scalable decoding. Bitstream scalability is the property of a bitstream that allows decoding of appropriate subsets of the bitstream to generate complete pictures of quality and/or resolution commensurate with the proportion of the bitstream decoded. Decoders of different complexities (from low performance to high performance) can coexist for a scalable bitstream. While low performance decoders may decode only small portions of the bitstream producing basic quality, high performance decoders may decode much more and produce significantly higher quality. The most important types of scalability are SNR scalability, spatial, and resolution scalability. The JPEG2000 compression system supports scalability. A key advantage of scalable compression is that the target bitrate or reconstruction resolution need not be known at the time of compression. A related advantage of practical significance is that the image need not be compressed multiple times to achieve a target bitrate, as is common with the JPEG compression standard. An additional advantage of scalability is its ability to provide resilience to transmission errors, as the most important data of the lower layer can be sent over the channel with better error performance, while the less critical enhancement layer data can be sent over a channel with poor error performance. Both types of scalability are very important for Internet and database access applications and bandwidth scaling for robust delivery. The SNR and spatial scalability types include the progressive and hierarchical coding modes defined in the JPEG, but they are more general.

3.2. Volumetric Data Encoding

The 2D compression technologies compress an image by exploiting the data redundancy within the image (i.e., intra-slice correlation) in horizontal and vertical directions only, whereas the 3D ones or the volumetric data encoders compress the datasets additionally in the third dimension through exploitation of the data redundancy between adjacent images (i.e., inter-slice correlation) **[13]**. In this section, two main technologies, JP3D and H.264 that have been used throughout this research to compress 3D medical images (volumetric datasets) are described.

3.2.1. JP3D

In order to support volumetric datasets, the JPEG committee decided to add a three dimensional extension, Part 10 **[48]**, to the JPEG2000, also referred to as JP3D **[49]** .

JP3D is envisioned as a pure extension of Part 1 **[33]** and Part 2 **[50]** of JPEG2000, which means any compliant Part 1 or Part 2 code-stream will be considered as a compliant JP3D code-stream. Especially, it will accept all the existing capabilities and syntax in Part 2 for multi-component images, even though it will provide alternatives or extensions to some of those capabilities. Within these constraints, the goal is to provide a specification for 3D datasets as isotropic as possible. For example, the project will try to present identical processing capability in all three dimensions. However, Part 1 and Part 2 code-stream syntaxes make a distinction between the two spatial axis and the cross-component axis.

Figure 3.9 illustrates the essential building blocks of a JP3D encoder, depicting the same architecture as a Part 1 encoder **[33]**. Clearly, the blocks vary internally from their two-dimensional (2D) counterparts.

The building blocks, as given in Figure 3.9, consist of pre-processing, discrete wavelet transform (DWT), quantization, tier-1 coding, and tier-2 coding. The 3D input image can be observed as a set of 2D image slices. Furthermore, each slice is simply a 2D image, having one or more components.

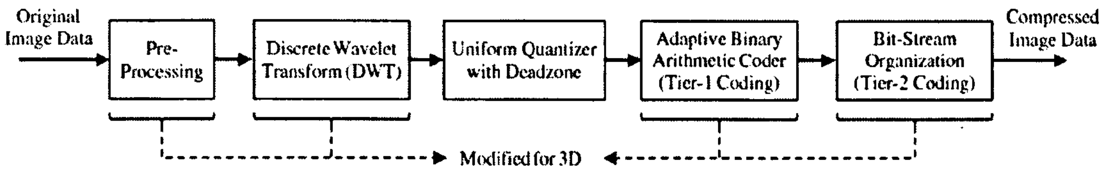


Figure 3-9 Encoder block schema [49]

The horizontally aligned X axis in each slice defines the column index, while the vertically aligned Y axis defines the row index in a slice. Besides the slices, the Z axis, orthogonal to the X and Y axes, is acceptable as a slice index (see Figure 3.10). Whereas the X axis is connected to the horizontal direction and the Y axis with the vertical direction, the Z axis is linked with what is called the axial direction. The 3D input image data for JP3D, as with 2D images, may contain one or more components, with a maximum of 16 384 (2^{14}) components.

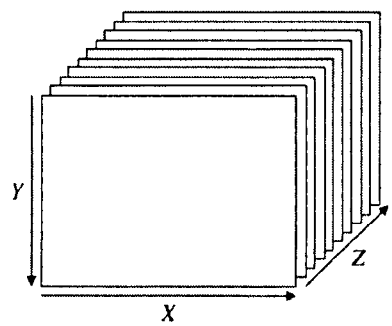


Figure 3-10 3D input image data [49]

Each component has sample values within integer typed with a bit depth in the range of 1 to 38 bits. Signed samples with bit depth of *B* would drop in the discrete interval *[−2B−1, 2B−1 − 1]* while unspecified samples would be in this range *[0, 2B − 1]*. The bit depth and sign specifications of the samples can be different for each component. Therefore, to sum up, the sample data typing for 3D input image data is precisely the same as 2D input image data.

3.2.1.1. Pre-processing

The first pre-processing step, in JPEG2000 Part 1, partitions the input image data into the same sized rectangular and no overlapping tiles. Only tiles on image borders can be varied in size. Tile sizes can be varied from one pixel to a full-image size. Tiling in JP3D is simply extended to three dimensions. Thus, tiles are not rectangular anymore, but as a substitute, become cubical subvolumes of the original input image. From Part 1 **[33]** point of view, tiles are handled independently both by the encoder and decoder.

Following the first step, the pre-processing steps are the same to those steps performed for 2D images when using JP3D. The DC level shift **[49]** step for unsigned samples does not change, when having more than two dimensions and the point-wise inter component transformation step. The Multiple Component Transformation (MCT), also known as the last step, allows a conversion between RGB and YUV colour spaces. The components must have identical bit depths not only for 2D images but also for 3D ones.

Each tile component (i.e., each component of each tile) after the pre-processing is handled separately. Thus, the following sections will only be in reference to a single tile component.

3.2.1.2. The 3D Discrete Wavelet Transform (3D DWT)

The same two wavelet kernels, by default, are provided in JP3D as in JPEG2000 Part 1, i.e. the reversible (5, 3) wavelet transform and the irreversible (9, 7) wavelet transform. The transform kernel extension of JPEG2000 Part 2 **[50]** is used to identify custom transformation kernels in JP3D. A 1D decomposition divides signal into two frequency bands, i.e., one low-pass band and one high-pass band. Regarding separability property of the wavelet transform kernels, a 1D DWT can be easily expanded to two or more dimensions. The 3D DWT, for JP3D, is constructed from independent 1D DWT steps along the three spatial directions, X, Y, and Z. Therefore, the 3D DWT is easily a 2D DWT with an additional 1D DWT beside the Z axis.

In two dimensions, the word 'column' describes a collection of all points that have a common X coordinate, while 'row' specifies all points that have a common Y coordinate.

However, in three dimensions, the meaning of the words 'row' and 'column' is less clear. In the following text, a 'row' defines a set of all points with a common Y and Z coordinate and a 'column' defines a set of all points with a common X and Z coordinate. In this thesis, let 'z-row' to define a set of all points that have a common X and Y coordinate. Therefore, in 3D rows, columns, and z-rows each classifies lines in the horizontal, vertical, and axial directions respectively.

In the first step of the 3D wavelet decomposition, 1D forward DWT is applied to each row of z dimension of the original volume (Figure 3.11a), effectively decomposing the input along the Z axis. The volume is now decomposed into two sub-bands as illustrated in Figure 3.11b. In the second step, 1D forward DWT is applied to each column of the filtered and subsampled data to decompose along the Y axis. Therefore, the original volume is filtered into four sub-bands, as shown in Figure 3.11c. In the final step, 1D forward DWT is applied for the third time to each row of the filtered and subsampled data to decompose along the X axis effectively into eight sub-bands as shown in Figure 3.11d. The order in which 1D wavelet transforms are applied does not matter in theory. But regarding possible rounding errors it is recommended that the specified order of transforms is respected, particularly if lossless decoding is desired.

Eight sub-bands with eight corresponding sets of wavelet coefficients are created by the one-level decomposition. Each sub-band illustrates filtered and subsampled version of the original volume. JP3D has a dyadic decomposition pattern similar to JPEG2000 Part 1. In other words, by applying the same wavelet filtering steps, further decomposition of the image volume is achieved over and over, as described above, to the lowest frequency sub-band until this sub-band becomes too small. As input, each decomposition level obtains the lowest frequency sub-band and decomposes it into eight smaller sub-bands.

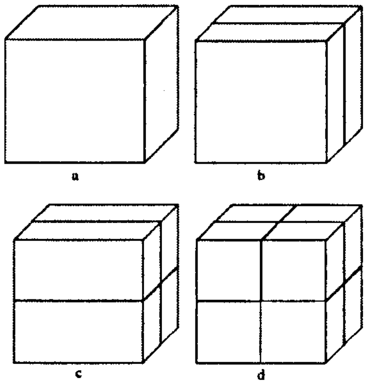


Figure 3-11 3D DWT decomposition [49]

Figure 3.12 shows a 2-level decomposition of a volumetric image and the corresponding labelling for each sub-band. The lowest frequency sub-band named LLL band signifies that low-pass filtering was applied to each spatial direction (X, Y, Z respectively). The character H indicates that high-pass filtering was used in a particular direction. The number received by each sub-band, denotes the decomposition level by which it was created. The number of decompositions in JP3D for the X and Y directions do not necessarily have to be equivalent, unlike JPEG2000 Part 1. The number of decompositions in the X or Y directions can be differed from those in the Z direction with 3D. As a result, it is possible that a specific decomposition level less than eight sub-bands is created.

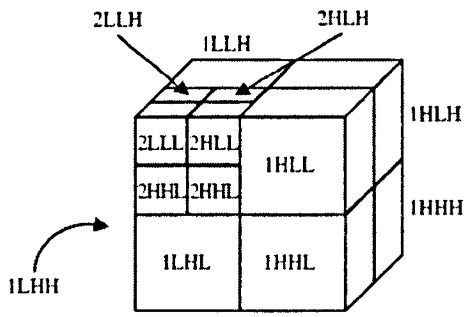


Figure 3-12 2-level decomposition [49]

The sub-band label, which indicates that no filtering was applied in a particular direction, is the character X. During the design of JP3D, a variable number of decompositions in each of the three spatial directions were added, because, in spite of the fact that JP3D targets isotropic data behaviour, most volumetric datasets have different imaging properties along the slice axis compared to those in the slice plane. Different imaging properties are usually caused by the technical and physical limitations of image scanners; the image resolution across slices is much lower than the image resolution within the slices, for instance. The user is able to compensate these kinds of discrepancies by the flexibility provided in the decomposition pattern of JP3D. Furthermore, to protect the isotropic behaviour of the JP3D codec, the option of variable decomposition levels was made available for all spatial directions.

The irregular (1, 2, 2) decomposition pattern has been presented in Figure 3.13, where a single decomposition is done in the X direction, and two decompositions are performed in Y and Z directions. The decomposition process is effectively created by performing wavelet transforms in all three spatial dimensions in the first eight sub-bands at the decomposition level 1. After that, for further decomposition the 1LLL sub-band is used; however, the wavelet transform is only performed in Y and Z direction. Therefore, four new sub-bands are created at decomposition level 2, i.e., 2XLL, 2XHL, 2XLH, and 2XHH. After the requested number of decomposition levels is reached in a specific direction, it will no longer be considered for wavelet transform. The wavelet decomposition process automatically introduces a multi resolution representation of the original image volume similar to JPEG2000 Part 1. Therefore, a volume at a specific resolution is needed to be reconstructed by resolution levels which contain those sub-bands. To reconstruct the lowest resolution version of the original volume, sub-bands from resolution level 0 are necessary. Resolution level 0 is the only resolution level which in practice contains exactly one sub-band. Adding sub-bands from the next resolution level allows a higher resolution version of the original image to be reconstructed. A full-image resolution of the original volume can be reconstructed with all sub-bands from all resolution levels. Resolution levels might become somewhat confusing to the reader, when a volume is decomposed using an irregular pattern (i.e., the number of decomposition levels is not equal in all spatial directions). Conversely, the spatial direction with the highest amount of decomposition levels determines the number of resolution levels in a decomposed volume.

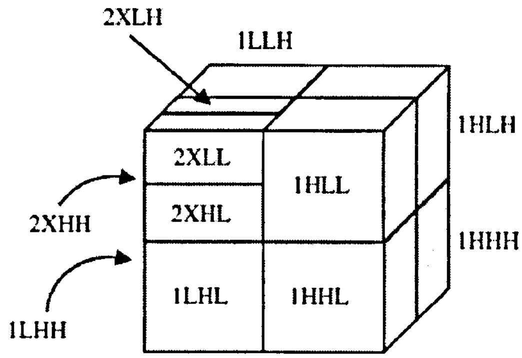


Figure 3-13 A (1, 2, 2) irregular decomposition pattern [49]

At last, the ability of allowing various wavelet kernels in each spatial direction was also added to JP3D. Nevertheless, the definition of resolution levels and sub-bands or the wavelet decomposition process is not changed by this feature.

3.2.1.3. Quantization

The dimensionality of the data is not dependent on wavelet coefficient quantization, thus the quantization strategy in JPEG2000 Part 1 and JP3D is the same (see section 2.3.4 for more details). For this reason, the wavelet coefficients per sub-band are quantized by the encoder after the 3D DWT step just like in JPEG2000 Part 1.The encoder emphasises the importance of specific sub-bands, or adds an extra but easy-to-implement mechanism for bitrate control has been allowed by quantization.

3.2.1.4. Bit Modelling and Entropy Coding

Afterwards, to create the compressed bitstream, the quantized wavelet coefficients are entropy encoded. Each sub-band is encoded separately from the other sub-bands as in JPEG2000 Part 1. Thus, the encoder and decoder can deliver better error resilience and higher flexibility in arranging the progression order of the bitstream. In addition, the process of entropy coding is referred to as tier-1 coding. For the entropy coding step, JP3D uses a 3D version of the EBCOT (Embedded Block Coding with Optimized Truncation) algorithm. Each sub-band is partitioned into smaller units, named code-block and encodes each of these code-blocks separately by EBCOT algorithm. JP3D code-blocks are 3D cuboidal volumes as shown in Figure 3.14; however, the code-blocks in JPEG2000 part 1 are 2D rectangular entities. On the other hand, a 3D code-block can be seen as a series of 2D code-block slices, with XY plane oriented slices and the slice index in the Z direction. The dimensions of a 3D code-block can be freely selected with some limitations at the encoder.

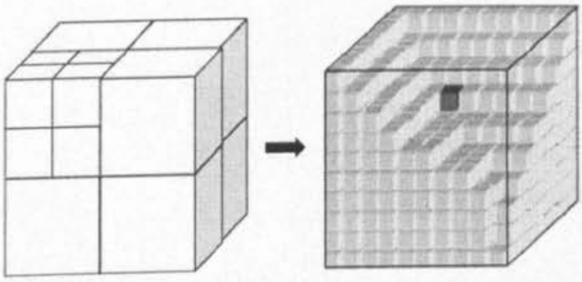


Figure 3-14 Code-block partitioning in 3D [90]

3.2.1.5. 3D EBCOT

As explained before, one of the goals of JP3D is to use as many methods as possible from JPEG2000 Parts 1 and 2 without any changes. Hence, 3D EBCOT differs slightly from its 2D counterpart. Wavelet coefficients in each 3D code-block have been entropy encoded into an embedded code-stream. The coefficients are scanned in a bit plane by bit plane order in the 3D EBCOT algorithm in which starting with the most significant bit plane and proceeding until the least significant bit plane. When, the coefficients have a bit depth of N bits, the bit depth level N − 1 is equal to the most significant bit plane. However, the least significant bit plane (zero) is situated at the bit depth level 0. Therefore, basically, bit plane P shows the bits at bit position P of all the wavelet coefficients of the code-block. The coordinates of bits in a bit plane and their respective coefficients are spatially the same. Significantly, in JP3D the bit planes are 3D bit volumes which can be described as a series of 2D bit plane slices. Figure 3.15 shows the scanning order inside each bit plane follows a bit plane slice by bit plane slice order, where the bit plane slices are situated in the XY plane.

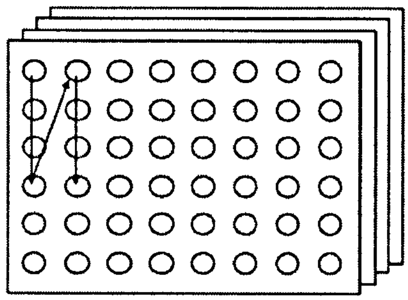


Figure 3-15 3D stripe based scanning pattern [49]

The bit plane slice with the lowest Z coordinate is scanned and then the stripe based scanning pattern is used inside each bit plane slice, which is similar to JPEG2000 Part 1.

As mentioned above, 3D EBCOT is the same as the 2D EBCOT described in the JPEG2000 Part 1 standard. In fact, the 3D extended scanning pattern in bit planes is the only significant difference between the two EBCOT.

A fractional bit plane coding process is applied by EBCOT to use the same three coding passes as in JPEG2000 Part 1 for encoding each bit plane. The bits in the bit plane are scanned one by one during each pass in the stripe based scanning order, where each bit is encoded by only one of the three coding passes. These coding passes work in the same way as in 2D EBCOT internally. The bits of coefficients, which are not yet significant but have the same time at least one significant immediate neighbour are encoded by the significant propagation pass. On the other hand, the bits of coefficients, already significant, are encoded by the magnitude refinement pass. In the end, all the bits that were not yet coded by the other two passes are encoded by cleanup pass. The three coding pass, propagation pass, refinement pass, and cleanup pass are called coding primitives, which are smaller building blocks that send out bit values, each associated with a specific context label, to the context based adaptive binary arithmetic entropy coder or simply MQ-coder. The bit values encoded by the MQ-coder contain the actual content information needed to reconstruct the code-block wavelet coefficients at the decoder side.

A bitstream, which has been delivered by the fractional bit plane coding process, can be pruned at multiple positions for each code-block. A possible truncation point is introduced by each coding pass. This provides flexibility for encoder in terms of rate/distortion (R-D) allocation of the compressed code-stream, facilitating the best possible R-D control strategy.

3.2.1.6. Bitstream Organization

Bitstream organization starts after completing entropy coding in JP3D. Various independent code-block bitstreams are reorganised into logical units called packets, during this step. Each packet usually encloses small pieces of multiple code-block bitstreams. The packets are consequently, regarding the type of application, joined in a particular order to a final JP3D code-stream. The organization of the bitstream in JP3D and JPEG2000 Part 1 [33] are similar.

3.2.2. H.264

The newest entry in the series of international video coding standards is H.264 / MPEG-4 (Part 10) Advanced Video Coding (usually referred as H.264/AVC) [48], which is the most powerful and state-of-the-art standard. In this standard the coding efficiency has been improved by an average factor of at least two over MPEG-2 standard, which is the most widely used video coding standard today. A new amendment was added to this standard, named the Fidelity Range Extensions (FRExt, Amendment 1) that reveals even further coding efficiency against MPEG-2, potentially by as much as 3:1 for some key applications [35] [91].

3.2.2.1. H.264/AVC History

A wide range of applications has been considered for H.264, from video conferencing to entertainment, streaming video, surveillance, military applications, and digital cinema. Three basic feature sets were established to address these application domains called profiles: the Baseline, Main, and Extended profiles [92]. The Baseline profile was designed for reducing complexity and providing high robustness and flexibility for use over a broad range of network environments. The Main profile was designed to emphasise compression coding efficiency capability and the Extended profile was designed to combine the robustness of the Baseline profile with a higher degree of coding efficiency and greater network robustness. Moreover, a new extension, FRExt, was added to the three profiles described in the following section (this extension is used in this research because it supports more than 8 bits accuracy in input images required in medical images).

3.2.2.2. The FRExt Amendment

The initial H.264/AVC standard (as it was completed in May of 2003) principally focused on "entertainment-quality" video, while having a broad range of applications, based on 8 bits per sample and 4:2:0 chroma sampling. The most demanding professional environments have not been supported by this standard; moreover, the highest video resolution has not been considered in this standard. For some applications such as content-contribution, content-distribution, studio editing, and post-processing, it may be necessary to use:

- More than 8 bits per sample of source video accuracy (CT images employ 12 bits per pixel)
- Colour representation which higher resolution than what is typical in consumer applications (i.e., to use 4:2:2 or 4:4:4 sampling as opposed to 4:2:0 chroma sampling format)
- Very high bitrates
- Very high resolution
- Very high fidelity – even representing some parts of the video losslessly
- No colour-space transformation rounding error
- RGB colour representation

The continuation of the joint project was defined to add new extensions to capabilities of the original standard for the needs of these most demanding applications. These extensions, originally known as "professional" extensions, were eventually renamed as "fidelity range extensions" (FRExt) to better indicate the spirit of the extensions **[93]**.

The FRExt project produced a set of four new profiles collectively called the High profiles:

- The High profile (HP), supporting 8 bit videos with 4:2:0 sampling, addressing high-end consumer use and other applications using high-resolution video without a need for extended chroma formats or extended sample accuracy.
- The High 10 profile (Hi10P), supporting 4:2:0 video with up to 10 bits of representation accuracy per sample.
- The High 4:2:2 profile (H422P), supporting up to 4:2:2 chroma sampling and up to 10 bits per sample.
- The High 4:4:4 profile (H444P), supporting up to 4:4:4 chroma sampling, up to 12 bits per sample, and additionally supporting efficient lossless region coding and an integer residual colour transformation for coding RGB video while avoiding colour-space transformation error. All of these profiles support all features of the prior Main profile, and additionally support an adaptive transform block size and perceptual quantization scaling matrices.

3.2.2.3. Coding Tools

At a basic overview level, the coding structure of this standard is similar to that of all prior major digital video standards (H.261, MPEG-1, MPEG-2/H.262, H.263 ,or MPEG-4 Part 2). The architecture and the core building blocks of the encoder are shown in Figure 3.16 and Figure 3.17, indicating that it is also based on motion-compensated DCT like transform coding. Each picture is compressed by partitioning it as one or more slices; each slice consists of macroblocks, which are blocks of 16x16 luma samples with corresponding chroma samples. However, each macroblock is also divided into sub-macroblock partitions for motion-compensated prediction. The prediction partitions can have seven different sizes –16x16, 16x8, 8x16, 8x8, 8x4, 4x8, and 4x4. In the past standards, motion compensation used the entire macroblocks or, in the case of newer designs, 16x16 or 8x8 partitions so the larger variety of partition shapes provide enhanced prediction accuracy. The spatial transform for the residual data is then either 8x8 (a size supported only in FRExt) or 4x4. In past major standards, the transform block size has always been 8x8 so the 4x4 block size provides an enhanced specificity in locating residual difference signals. The block size used for the spatial transform is always either the same or smaller than the block size used for prediction. The hierarchy of a video sequence, from sequence to samples is given by:

Sequence [pictures [slices [macroblocks [macroblock partitions [sub-macroblock partitions [blocks [samples]]]]]]].

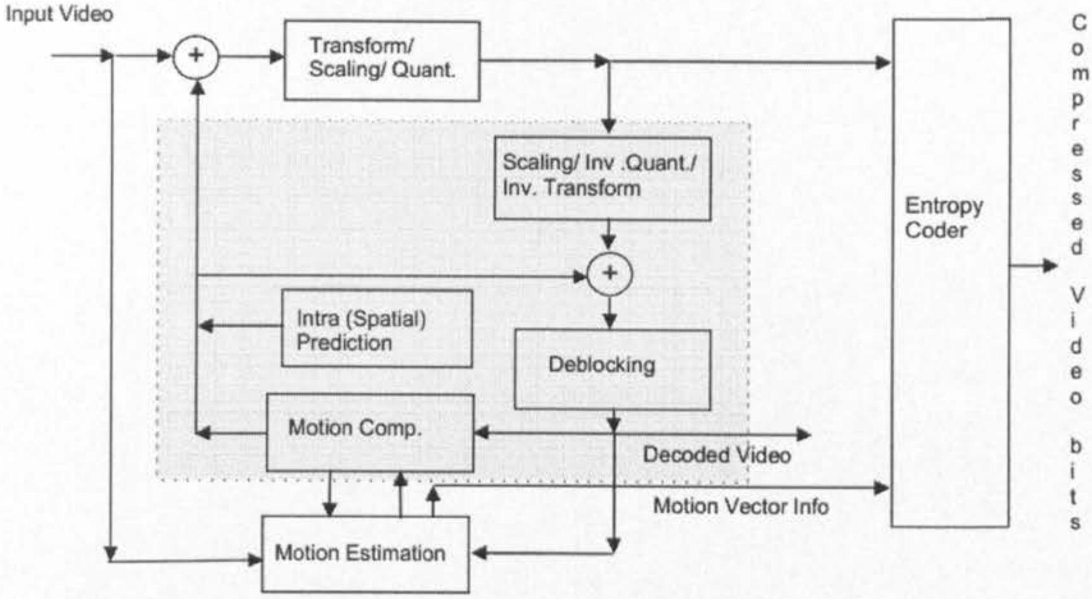


Figure 3-16 High-level encoder architecture [51]

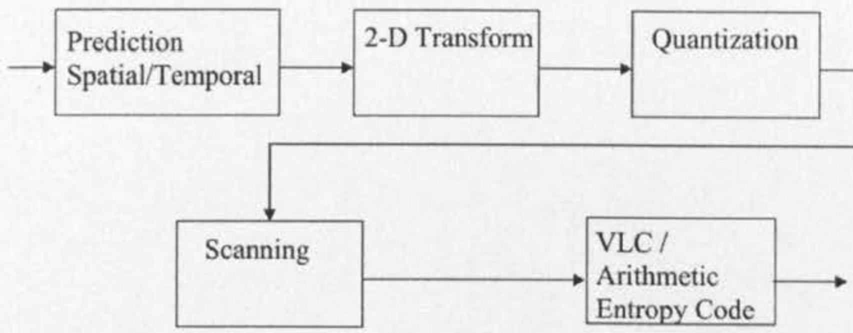


Figure 3-17 Higher-level encoder block diagram [51]

Furthermore, there may be additional structures such as packetization schemes, channel codes, etc., which relate to the delivery of the video data, not to mention other data streams such as audio. As the video compression tools primarily work at or below the slice layer, bits associated with the slice layer and below are identified as Video Coding Layer (VCL) **[91]** and bits associated with higher layers are identified as Network Abstraction Layer (NAL) **[91]** data. VCL data and the highest levels of NAL data can be sent together as part of one single bitstream or can be sent separately. The NAL is designed to fit a variety of delivery frameworks (e.g., broadcast, wireless, and storage media). Here, we only discuss the VCL which is the heart of the compression capability.

While an encoder block diagram is shown in Figure 3.16, the decoder conceptually works in reverse, comprising primarily an entropy decoder and the processing elements of the region shaded in Figure 3.16.

In the first version of the standard, only the 4:2:0 chroma format (typically derived by performing an RGB to YC_bC_r colour-space transformation and subsampling the chroma components by a factor of 2:1 both horizontally and vertically) and only an 8 bit sample precision for luma and chroma values was supported. The FRExt amendment extended the standard to 4:2:2 and 4:4:4 chroma formats and higher than 8 bit precision, with optional support of auxiliary pictures for purposes such as alpha blending composition.

The basic unit of the encoding or decoding process is the macroblock. In 4:2:0 chroma formats, each macroblock consists of a 16x16 region of luma samples and two corresponding 8x8 chroma sample arrays. In a macroblock of 4:2:2 chroma format video, the chroma sample arrays are 8x16 in size and in a macroblock of 4:4:4 chroma format video, they are 16x16 in size.

Slices in a picture are compressed by using different coding tools, for instance: Intra Spatial Prediction, Inter Temporal Prediction, Lossless Capability, Deblocking Filter, Lossless Entropy coding, Error Resilience Tools, and different colour formats.

Of course, each slice need not use all of the above coding tools. Depending on the subset of coding tools used, a slice can be of I (Intra), P (Predicted), B (Bi-predicted), SP (Switching P) or SI (Switching I) type. Pictures may contain different slice types, and come in two basic types: reference and non-reference pictures. Reference pictures can be used as references for interframe prediction during decoding of later pictures (in bitstream order), but non-reference pictures cannot. (It is noteworthy that unlike prior standards, pictures that use bi-prediction can be used as references just like pictures coded using I or P slices.) In the next section, we describe the coding tools used for these different slice types.

3.2.2.3.1. I-slice

In I-slices, pixel values are first spatially predicted from their neighbouring pixel values. After spatial prediction, the residual information is transformed using a 4x4 transform or an 8x8 transform (FRExt-only) and then quantized. In FRExt, the quantization process supports encoder-specified perceptual based quantization scaling matrices to optimize the quantization process according to the visibility of the specific frequency associated with each transform coefficient. Quantized coefficients of the transform are scanned in one of the two different ways (zig-zag or field scan) and are compressed by entropy coding using one of two methods CAVLC or CABAC.

The structure of the codec for the I-slice and the key differences between P and B-slices are discussed in following section.

Intra Spatial Prediction

Three basic types of intra spatial prediction are defined to exploit spatial correlation among pixels:

- Full-macroblock prediction for 16x16 luma or the corresponding chroma block size
- 8x8 luma prediction (FRExt-only)
- 4x4 luma prediction

In 16x16 spatial prediction mode, the luma values of an entire 16x16 macroblock are predicted from the pixels around the edges as shown in the Figure 3.18. Prediction can be made in one of the four different ways: (1) vertical, (2) horizontal, (3) DC, and (4) planar. In the vertical and horizontal predictions, the luma values of a macroblock are predicted from the pixels just above or left of the macroblock respectively. In DC prediction, the luma values of the neighbouring pixels are averaged and that average value is used as a predictor. In the planar prediction, it is assumed that the macroblock covers diagonally increasing luma values and the predictor is formed based on the planar equation.

In 4x4 spatial prediction mode, the luma values of 4x4 blocks are predicted from the neighbouring pixels above or left of the block and 9 different directional ways of prediction are allowed (see Figure 3.18). Each prediction direction specifies a particular set of spatially dependent linear combinations of previously decoded samples for use as the prediction of each input sample.

8x8 luma intra prediction (available only in FRExt profiles) uses basically the same concept as 4x4 prediction, but with a block size of 8x8 rather than 4x4 and with low-pass filtering of the predictor to improve prediction performance.

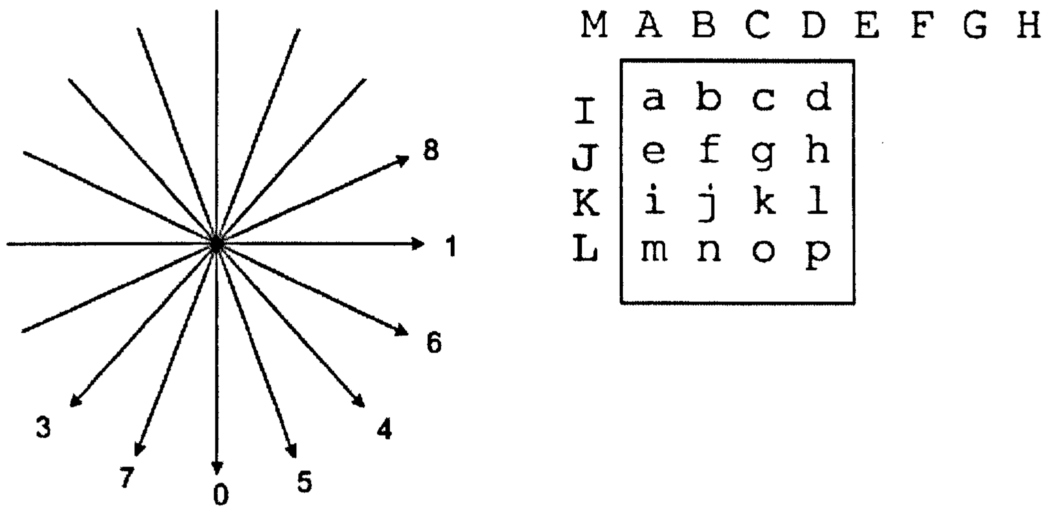


Figure 3-18 Spatial prediction of a 4x4 block [51]

Transform and Quantization

After spatial prediction, a transform is applied to decorrelate the data spatially. There are several unique features about the transform selected for this coding standard. Two features are described: (1) The transform is designed to be so simple that it can be implemented using just a few additions, subtractions, and bit shifts and (2) It is the first video standard fundamentally based on an integer inverse transforms design for its main spatial transforms, rather than using a floating point inverse transform. The forward transform that will typically be used for encoding is also an integer transform. A significant advantage of the use of an integer is that, with an exact integer inverse transform, there is no possibility of a mismatch between the encoder and decoder, unlike MPEG-2 and ordinary MPEG-4 Part 2.

While the macroblock size remains at 16x16, they are divided up into 4x4 or 8x8 blocks. Furthermore, a 4x4 or 8x8 block transformation matrix T_{4x4} or T_{8x8} is applied to every block of pixels, as given by **[94]**:

$$
T_{4\times4} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \quad T_{8\times8} = \begin{bmatrix} 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 \\ 12 & 10 & 6 & 3 & -3 & -6 & -10 & -12 \\ 8 & 4 & -4 & -8 & -8 & -4 & 4 & 8 \\ 10 & -3 & -12 & -6 & 6 & 12 & 3 & -10 \\ 8 & -8 & -8 & 8 & 8 & -8 & -8 & 8 \\ 6 & -12 & 3 & 10 & 10 & -3 & 12 & -6 \\ 4 & -8 & 8 & -4 & -4 & -81 & -8 & 4 \\ 3 & -6 & 10 & -12 & 12 & -10 & 6 & -3 \end{bmatrix} \tag{3.8}
$$

The 4x4 transform is remarkably simple, and while the 8x8 transform (used in FRExt profiles only) is somewhat more complex, it is still remarkably simple when compared to an ordinary 8x8 IDCT. The transform T is applied to each block within the luma (16x16) and chroma (8x8, or in FRExt: 8x16 or 16x16) samples for a macroblock by segmenting the full sample block size into smaller blocks for transformation as necessary.

In addition, when the 16x16 Intra prediction mode is used with the 4x4 transform, the DC coefficients of the sixteen 4x4 luma blocks in the macroblock are further selected and transformed by a secondary Hadamard transform using the H_{4x4} matrix shown below (note the basic similarity of T_{4x4} and H_{4x4}). The DC coefficients of the 4x4 blocks of chroma samples in all macroblock types are transformed using a secondary Hadamard transform as well. For 4:2:0 video, this requires a 2x2 chroma DC transformation specified by the Hadamard matrix H_{2x2} (Equation 3.9); for 4:4:4, the chroma DC uses the same 4x4 Hadamard transformation as used for luma in 16x16 intra mode; and for 4:2:2 video, the chroma DC transformation uses the matrices H_{2x2} and H_{4x4} to perform a 2x4 chroma DC secondary transformation.

$$
H_{2\times2} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad H_{4\times4} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \tag{3.9}
$$

The coefficients produced by the transformation are quantized using a quantization control parameter that can be changed for every macroblock. The parameter can take one of the 52 possible values when video format supports 8 bits per decoded sample. When supporting greater bit depth video content, FRExt expands the number of steps by 6 for each additional bit of decoded sample accuracy. Importantly, the quantization step-sizes are not linearly related to the quantization parameter (as in all prior standards), but vary in such a way that the quantization step size exactly doubles for every 6 increments of the quantization parameters.

A default relationship is specified between the quantization step sizes used for luma and chroma, and the encoder can adjust this relationship at the slice level to balance the desired fidelity of the colour components. With the FRExt amendment, the encoder can also balance the C_b and C_r fidelity separately relative to each other.

Perceptual Based Quantization Scaling Matrices

The new FRExt amendment adds support for a feature that had been a mainstay of prior use in MPEG-2 – namely, perceptual based quantization scaling matrices. The encoder can specify, for each transform block size and separately for intra and inter prediction, a customized scaling factor for use in inverse-quantization scaling by the decoder. This allows tuning of the quantization fidelity according to a model of the sensitivity of human visual system to different types of error. It typically does not improve objective fidelity as measured by MSE (or, equivalently, PSNR), but it does improve subjective fidelity, which is really the most important criterion. Default values for the quantization scaling matrices are specified in the standard, and the encoder can choose to instead use customized values by sending representation of those values at the sequence or picture level.

Scanning

If a macroblock is compressed using the 4x4 transform in frame mode, the quantized coefficients of the transform are scanned in a zig-zag fashion shown in Figure 3.19a. This scan ordering is designed to order the highest-variance coefficients first and to maximize the number of consecutive zero-valued coefficients appearing in the scan.

If a macroblock is compressed using the 4x4 transform in the field mode, the scanning order of the coefficients is modified to be more efficient for field scanning as shown in Figure 3.19b – reflecting the decreased correlation of the source data in the vertical dimension. For other block sizes such as 8x8, the same basic concepts are applied with similar scan specified for each block size.

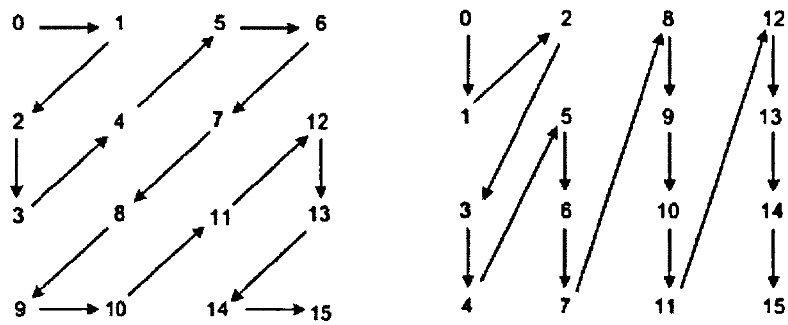


Figure 3-19 Coefficient scanning order in (a) Frame and (b) Field mode [94]

Entropy Coding

Entropy coding is a term referring to lossless coding techniques that replace data elements with coded representations, which in combination with the previously described predictions, transformations, and quantization, can result in significantly reduced data size. This standard uses two type of entropy coding: Context Adaptive Variable Length Coding (CAVLC) and Context Adaptive Binary Arithmetic Coding (CABAC).

- **CAVLC:** The principle idea of this method is that when the data elements to be coded occur with unequal frequencies; frequently occurring elements can be assigned to very short codes, while infrequent elements can be assigned longer codes and therefore this method is similar to Huffman coding. Furthermore, CAVLC uses zig-zag scanning with adaptive variable length coding. There are two motivations for CAVLC: (1) It can provide better code efficiency and (2) Lower demand of memory consumption over the universal variable length coding standards (see [35] for details).

- **CABAC:** It is the first successful arithmetic coding method deployed in the video coding standard, with considerable compression improvement compared to the previous entropy coding tools. The CABAC method has been shown to increase compression efficiency by roughly 10-15% relative to CAVLC mode, although it is significantly and computationally more complex [91]. The steps in the CABAC entropy coding method are shown in Figure 3.20. In the first step, a suitable model is chosen according to a set of past observations of relevant syntax elements; this is called context modelling. Different models are maintained for each syntax element. If a given symbol is non-binary values, it will be mapped onto a sequence of binary decisions, so-called bins, in the second step. This binarization is done according to a specified binary tree structure similar to VLC code. Then each bin is encoded with an adaptive binary arithmetic coding (BAC) engine using probability estimates that depend on the specific context (see [95] for details).

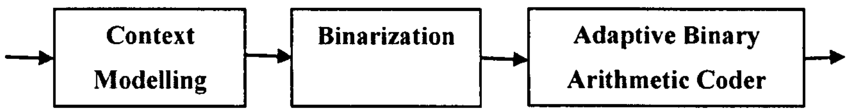


Figure 3-20 Generic block diagram of CABAC entropy coding scheme [94]

Lossless Macroblock Modes

When the fidelity of the coded video is high (i.e., when the quantization step size is very small), it is possible in certain very rare instances of input picture content for the encoding process to actually cause data expansion rather than compression. Furthermore, it is convenient for implementation reasons to have a reasonably low identifiable limit on the number of bits necessary to be processed in a decoder in order to decode a single macroblock. To address these issues, the standard includes a "PCM" macroblock mode in which the values of the samples are sent directly – without prediction, transformation, or quantization. An additional motivation for support of this macroblock mode is to allow regions of the picture to be represented without any loss of fidelity.

3.2.2.3.2. P-slices

In P-slices (predictively coded, or inter slices), temporal (rather than spatial) prediction is used by estimating motion between pictures. Innovatively, motion can be estimated at the

16x16 macroblock level or by partitioning the macroblock into smaller regions of luma size 16x8, 8x16, 8x8, 8x4, 4x8, or 4x4.

A distinct motion vector can be sent for each sub-macroblock partition. The motion can be estimated from multiple pictures that lie either in the past or in the future in display order. The selection of which reference picture is used is done on the macroblock partition level (so different sub-macroblock partitions within the same macroblock partition will use the same reference picture). A limit on the number of pictures used for the motion estimation is specified for each level. To estimate the motion, pixel values are first interpolated to achieve quarter-pixel accuracy for luma and up to 1/8th pixel accuracy for chroma. Interpolation of luma is performed in two steps – half-pixel and then quarter-pixel interpolation. Half-pixel values are created by filtering with the kernel [1 -5 20 20 -5 1]/32, horizontally and/or vertically. Quarter-pixel interpolation for luma is performed by averaging two nearby values (horizontally, vertically, or diagonally) of half pixel accuracy. Chroma motion compensation uses bilinear interpolation with quarter-pixel or one-eighth-pixel accuracy (depending on the chroma format). After interpolation, block based motion compensation is applied. As noted, however, a variety of block sizes can be considered, and a motion estimation scheme that optimizes the tradeoff between the number of bits necessary to represent the video and the fidelity of the result is desirable.

If a macroblock has motion characteristics that allow its motion to be effectively predicted from the motion of neighbouring macroblocks, and it contains no non-zero quantized transform coefficients, then it is flagged as skipped. This mode is identified as the Skip mode. Note that, unlike prior standards, non-zero motion vectors can be inferred when using the Skip mode in P slices.

In addition to the use of motion compensation and reference picture selection for prediction of the current picture content, weighted prediction can be used in P slices. When weighted prediction is used, customized weights can be applied as a scaling and offset to the motion-compensated prediction value prior to its use as a predictor for the current picture samples. Weighted prediction can be especially effective for such phenomena as "fade-in" and "fade-out" scenes.

After the temporal prediction, the steps of transform, quantization, scanning, and entropy coding are conceptually the same as those for I-slices for the coding of residual data. The motion vectors and reference picture indexes representing the estimated motion are also compressed. To compress the motion vectors, a median of the motion vectors from the three neighbouring macroblock partitions or sub-macroblock partitions – left, above and above right or left – is obtained and the difference from this median vector and the value of the current motion vector is retained and entropy coded. Similarly, the selected reference frame indexes are also entropy coded.

3.2.2.3.3. B-slices

In B-slices, two motion vectors, representing two estimates of the motion per macroblock partition or sub-macroblock partition are allowed for temporal prediction. They can be from any reference picture in future or past in display order. Again, a constraint on the number of reference pictures that can be used for motion estimation is specified in the Levels definition. A weighted average of the pixel values in the reference pictures is then used as the predictor for each sample.

B-slices also have a special mode (Direct mode) in which the motion vectors for a macroblock are not explicitly sent. The encoder can specify in the slice header either for the decoder to derive the motion vectors by scaling the motion vector of the co-located macroblock in another reference picture or to derive it by inferring motion from spatially neighbouring regions.

The weighted prediction concept is further extended in the case of B slices. In addition to its use for scaling and offsetting a prediction value, in B slices weighted prediction can enable encoder adjustment of the weighting used in the weighted average between the two predictions that apply to bi-prediction. This can be especially effective for such phenomena as "cross-fades" between different video scenes, as the bi-prediction allows flexible weighted blending of content from such scenes.

Unlike prior standards, pictures coded using B slices can be used as references for decoding of subsequent pictures in decoding order (with an arbitrary relationship to such pictures in display order).

3.2.2.3.4. SP and SI Slices

Switching P (SP) and Switching I (SI) slices are close cousins of the usual P and I slices, utilizing either temporal or spatial prediction as before; however, their main virtue is that they can allow reconstruction of specific exact sample values, even when using different reference pictures or a different number of reference pictures in the prediction process. The main usefulness of this property (which naturally comes at some cost in coding efficiency when compared to the usual P and I slices) is to allow bitstream switching and to provide additional functionalities such as random access, fast forward, reverse, and stream splicing. These tools are only available in the Extended Profile.

3.2.2.3.5. Deblocking Filter

As shown in Figure 3.16, H.264/AVC uses an in-loop deblocking filter to reduce the blockiness introduced in a picture. The filtered pictures are used to predict the motion for other pictures. The deblocking filter is an adaptive filter that adjusts its strength depending upon compression mode of a macroblock (Intra or Inter), the quantization parameter, motion vector, frame or field coding decision and the pixel values. When the quantization step size is decreased, the effect of the filter is reduced, and when the quantization step size is very small, the filter is shut off. The filter can also be shutoff explicitly or adjusted in overall strength by an encoder at the slice level.

3.2.2.3.6. Error Resilience Tools

A number of features of the codec design are designed to enable recovery of video fidelity in the presence of network transmission errors or losses. For example, the NAL design, with its highly robust treatment of sequence and picture header content (more properly called sequence and picture parameter sets in the standard), establishes a high degree of robustness. The basic slice structure design adds further robustness, as each slice is designed to be completely independent of all other slices of a picture in the basic decoding process. No content of any slice of a picture is used for the prediction of syntax elements or sample values used in the decoding process of other slices in the picture. Additionally, the encoder can select to specify that the prediction of intra macroblock sample values in P and B slices will not use spatial neighbours that were not also coded in intra modes – adding further robustness against temporal error propagation. The multiple-reference-picture support can also be used by an encoder to enable further resilience against data losses and errors (basically by avoiding the use of any pictures as reference pictures in the prediction process if the fidelity of those pictures may have been adversely affected by transmission errors or losses).

Going beyond these basic features that are an inherent part of the design, there are essentially four additional tools that are specified in the standard for further protection of the video bitstream from network transmission problems, which may occur for example as a result of congestion overloads on wired networks or due to channel errors in wireless networks. These tools are: (1) Flexible Macroblock Order (FMO), (2) Arbitrary Slide Order (ASO), (3) Redundant Slices (RS), and (4) Data Partitioning (DP). FMO can work to randomize the data prior to transmission, so that if a segment of data is lost (e.g. a packet or several packets), the errors are distributed more randomly over the video pictures, rather than causing corruption of complete regions, making it more likely that relevant neighbouring data is available for concealment of lost content. ASO allows slices of a picture to appear in any order for delay reduction (particularly for use on networks that can deliver data packets out of order). RS offers more protection by reducing the severity of loss using redundant representations of pictures. DP separates the coded slice data into separately decodable sections according to how important each type of data is to the resulting picture fidelity.

3.2.2.3.7. Colour Space and Residual Colour Transform Support

Like spatial transforms, colour transforms to date have generally used floating point operations and have thus been prone to rounding errors. Typically, video is captured and displayed using RGB colour space, but these components are typically highly correlated. Further, the human visual system seems better match to luma (brightness) and chroma (hue and saturation) representations, rather than RGB. The usual approach has been to perform a colour transformation for example RGB to YC_bC_r before compression, and then code the video in the YC_bC_r domain.

There are two problems with this approach. The first is that since the samples are actually represented using integers, rounding error is introduced in both the forward and inverse colour transformations. The second is that, because the above transformation was not originally designed for digital video compression, it uses a sub-optimal tradeoff between the complexity of the transformation (with difficult-to-implement coefficient values such as 0.2126 and 0.0722) and coding efficiency. Focusing on the second problem first, the FRExt amendment adds support for a new colour space called YC_gC_o (where the "C_g" stands for green chroma and the "C_o" stands for orange chroma), which is much simpler and typically has equal or better coding efficiency.

3.2.2.4. Profiles and Levels

H.264/AVC contains a rich set of video coding tools while not all the coding tools are required for all the applications. For example, sophisticated error resilience tools are not important for the networks with very little data corruption or loss. Forcing every decoder to implement all the tools would make a decoder unnecessarily complex for some applications. Therefore, subsets of coding tools called profile are defined. A decoder may choose to implement only one subset (profile) of tools or choose to implement some or all profiles. The following three profiles were defined in the original standard, and remain unchanged in the latest version:

- Baseline (BP)
- Extended (XP)
- Main (MP)

Table 3.1 gives a high-level summary of the coding tools included in these profiles. The Baseline profile includes I and P slices, some enhanced error resilience tools (FMO, ASO, and RS), and CAVLC, where as it does not contain CABAC entropy coding. The Extended profile is a super-set of Baseline, adding B, SP, SI slices and interlaces coding tools to the set of Baseline profile coding tools and adding further error resilience support in the form of data partitioning (DP) and it does not include CABAC. The Main profile includes I P and B-slices, CAVLC and CABAC. It does not include enhanced error resilience tools (FMO, ASO, RS, and DP) or SP and SI-slices. Interest to Main profile, which received a great deal of initial implementation interest for entertainment-quality consumer applications, now seems to be waning due to the new definition of the High profile in FRExt.

Table 3-1 Profiles in original H.264/AVC standard

Coding tools	Baseline	Main	Extended
I and P slices	x	x	x
CAVLC	x	x	x
CABAC		x	
B slices		x	x
Enhance error resil. (FMO, ASO, RS)	x		x
Further enhance Error resil. (DP)			x
SP and SI slices			x

The New High Profiles Defined in the FRExt Amendment

The FRExt amendment defines four new profiles:

- High (HP)
- High 10 (Hi10P)
- High 4:2:2 (Hi422P)
- High 4:4:4 (Hi444P)

All of these profiles also support monochrome coded video sequences, in addition to typical 4:2:0 video. The difference in capability among these profiles is primarily in terms of supported sample bit depths and chroma formats. However, the High 4:4:4 profile additionally supports the residual colour transform and predictive lossless coding features not found in any other profiles.

3.3. Summary

The architecture of three image and video compression technologies used throughout this research has been reviewed. The results of applying these compression technologies on medical datasets are reported in Chapter 6. A summary of each technology is laid out as follows:

(1) JPEG2000: It is one of the most popular compression standards in the medical community. There are some real incentives behind the development of the JPEG2000 standard. It provides higher compression efficiency over the previous standards and supports a rich set of features including tilling, lossy to lossless compression, region of interest coding, error resilience, random code-stream access, and resolution scalability. These features and encoding processes were reviewed in detail in the related section of this chapter.

(2) JP3D: It defines the volumetric extension for JPEG2000 that provides isotropic support for handling volumetric images with multiple components. It is specifically designed to be compatible with the other existing parts of the standard and as such offers the exact same functionality as its 2D counterpart. But because JP3D properly extends the wavelet transformation and the entropy coding to 3D, it is able to deliver better compression results than what was previously possible by using only Part 1 or Part 2 of JPEG2000. Therefore, it can be a good candidate to be used in volumetric datasets such as CT datasets.
The building blocks of JP3D were also discussed. It started with pre-processing step (tiling and component transformation) followed by 3D DWT, quantization, 3D EBCOT, and bitstream organization.

(3) H.264: It is the newest entry in the series of standard for video encoding. This initiative has many goals. It provides good video quality at substantially lower bitrates than previous standards and better video quality at an unchanged bitrate. The standard is further designed to give lower latency as well as better quality for higher latency. In addition, all these improvements compared to previous standards were to come without increasing the complexity of design. It provides enough flexibility to allow the standard to be applied to a wide variety of

applications. Furthermore, a new amendment is added to this standard, FRExt, for further coding efficiency against previous video coding.

The core of H.264 was discussed as well. It was similar to that of all digital video coding standards but some new techniques have been used in this standard, for instance motion compensation with variable block size, content adaptive binary arithmetic coding (CABAC) and multiple reference pictures. Furthermore, these techniques were reviewed and the coding structure, types of slices, and type of profiles were discussed.

4. ROI[1] BASED SEGMENTATION APPROACH

This chapter first gives a medical background, which includes Computed Tomography (CT) and Peripheral Arterial Disease (PAD). In the sequel, various regions of interest (ROI) based segmentation techniques, which can be used for detecting PAD, are described and also the proposed ROI based segmentation approach is introduced. This approach has been implemented for detecting peripheral arteries and body as ROIs over 3D CTA images. The flowchart of the proposed segmentation approach as well as the corresponding section associated with each step is depicted in Figure 4.1.

The proposed segmentation approach is divided into two parts: (1) Background & Body Segmentation (SROI segmentation) and (2) Peripheral Artery Segmentation (PROI Segmentation). The process of segmentation for the first part consists of approximate extraction of body region, removing unnecessary areas in an image and adding inner holes to the body region using a thresholding method, morphological operations, and connected component analysis. In the second part, the segmentation of Peripheral Artery includes an initial segmentation of arteries from other tissues followed by a 3D region growing algorithm, in combination with morphological operators and starts from automatic seed point selection.

The extracted regions (segmented PROI & SROI) are passed to the ROI based compression component, introduced in Chapter 6, to be compressed with different compression ratios. The segmentation of ROIs has been illustrated by several examples at the end of this chapter. The evaluation of these results in terms of Diagnostic Mean Opinion Score (MOS - see section 2.4 for detail) test has been covered in Chapter 6.

---

[1] Region of Interest

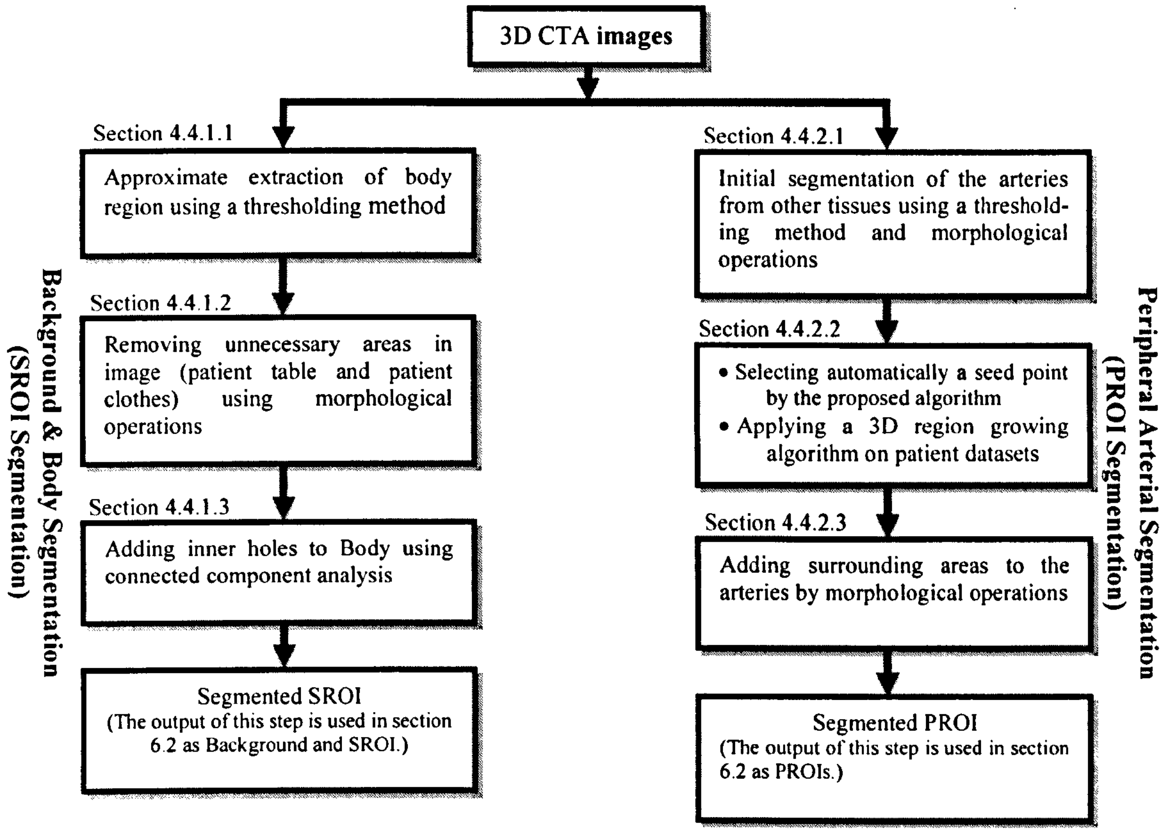


Figure 4-1 Flow chart of the proposed segmentation approach

4.1. Medical Background

This section presents a brief description of medical imaging modality, Computer Tomography which is an imaging diagnostic tool in different diseases such as Peripheral Arterial Disease and it followed by a medical background about PAD.

4.1.1. Computed Tomography

Computer Tomography (CT) scan formerly referred to as Computerised Axial Tomography (CAT) scan is an X-ray procedure that combines many X-ray images with the aid of a computer to generate cross-sectional views and, if needed, three-dimensional images of the internal organs and structures of the body. A CT scan is used to define normal and abnormal structures in the body and/or to assist in procedures by helping to guide accurately the placement of instruments or treatments [96] [97].

CT scanner measures the local X-ray attenuation coefficients of the tissue volume elements, also refer to as voxels, in an axial slice of the patient's anatomy. The attenuation coefficients are translated into grayscale values (CT value) of the corresponding picture elements (pixels) in the displayed 2D image of the slice. The numeric value I_{ij} assigned to a pixel (i,j) corresponds to the average X-ray attenuation $\mu_{(x_i,x_j)}$ within its associated pixel (x_i, x_j), after normalisation to the attenuation properties of water.

$$I_{ij} = 1000 \frac{\mu_{(x_i,x_j)} - \mu_{water}}{\mu_{water}} \quad [97] \tag{4.1}$$

Where μ_{water} is the attenuation coefficient of water and I_{ij} is measured in Hounsfield Units (HU). Pixel values are stored as integers, in the range -1024 to 3071 HU, corresponding to 4096 different values. By definition, HU for water (I_{ij} = 0) and HU for air (I_{ij} = 1000) are independent of the X-ray spectrum. However, the CT values of human tissues depend on the X-ray spectrum. In general, lung and fat have negative CT values, muscle has positive I_{ij} and bone has rather large CT values up to 2000 HU. The CT values of different tissues are shown in Table 4.1.

Table 4-1 CT numbers of various tissues [96]

Tissue	CT number (Hounsfield Units)
Bone	300+
Liver	50-80
Muscle	44-59
Blood	42-58
Water	0
Fat	-20 to -100
Lung	-300
Air	-1000

Computed Tomography Angiography (CTA)

Computed Tomography Angiography, also called CT angiography or CTA, is a test that combines the technology of a conventional CT scan with that of traditional angiography to make detailed images of the blood vessels in the body. In a CT scan, x rays and computers create images that show the cross sections of body. Angiography involves the injection of an iodine-rich contrast material (dye) into a large blood vessel, usually in your leg, to help visualize the blood vessels and the blood flow within them **[96]**.

4.1.2. Peripheral Arterial Diseases (PAD)

PAD results from obstruction (stenosis or total occlusion) of vessels feeding the extremities. The major cause of vascular obstructions is arteriosclerosis (fatty deposits in different stage of organization in the vascular wall) inducing reduction of the vascular lumen that results in ischemia of tissues distal to the obstruction. Lower limbs are more frequently involved than the upper extremities. Therapeutic approaches depend on the anatomic segments involved, the degree of the stenosis, and the length of the lesion **[97]** **[98]**.

A normal anatomy of the lower extremity vessels, related to abdomen, pelvis, thighs, and legs, is shown in Figure 4.2. The three red arrows show the cross sections of body.

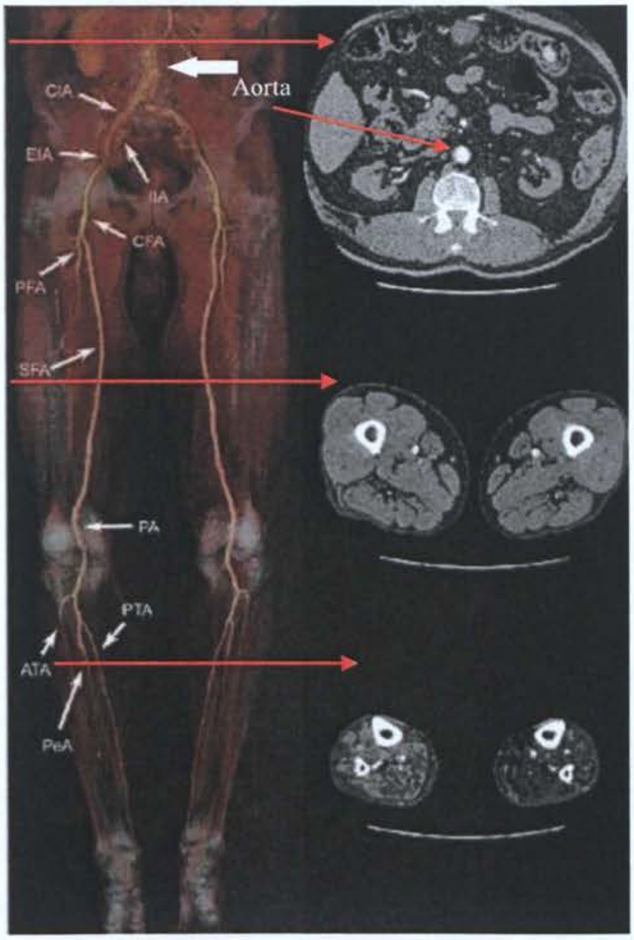


Figure 4-2 The arterial vascular anatomy [97]

4.2. Segmentation Methods

Segmentation is the process of dividing an image into significant regions most frequently to find regions of interest (foreground) from everything else (background). In the simplest cases, there would be only these two classes (foreground and background) and the segmented image would be a binary image. Some practical applications of segmentation include detection of organs such as the arterial tree, lungs, heart, liver or brain in CT images, distinguishing pathological tissue from normal tissue such as tumours, measuring tissue volumes, and computer-guided surgery. Image gray level or brightness is the most basic attribute for defining the regions, although other properties can be used such as intensity and geometrical features. General-purpose methods and techniques were developed over the past years for image segmentation **[99]**. Due to the lack of general solution to the segmentation problem, these methods can be utilized to solve a particular problem space when combined with specific knowledge domain.

According to the features and the technique used in segmentation methods, they can be classified as either non contextual or contextual **[100]** **[101]**. The features include texture, gradient magnitudes, and pixel values. In non contextual methods, the relationships which exist between features in an image are ignored; pixels are grouped together based on some attributes such as grey level. Intensity based thresholding method is a non contextual technique in which each pixel is assigned to a particular region based on the pixel value. In contextual techniques, the relationships between image features have been additionally used. Therefore, the pixels that have similar pixel levels or similar gradient values and are close to one another can be grouped as a region. Some methods in the contextual techniques are region based and boundary based. In region based techniques, connected regions are found based on some similarities of the pixels within them such as region growing methods. However, the boundary based methods are finding pixel differences rather than pixel similarities **[102]** **[103]**.

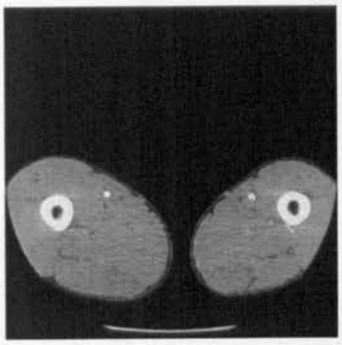
For ROI based segmentation in our approach, a combination of thresholding and region growing methods are used. In the following sections, some thresholding and region growing techniques are described.

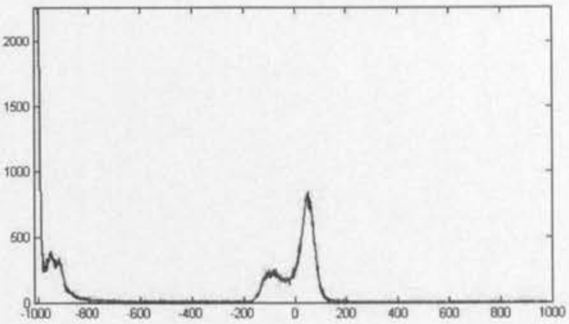
4.2.1. Thresholding Methods

One of the simplest segmentation methods is thresholding. Suppose an image f(x,y) is composed of light objects on a dark background in such a way that object and background pixels have intensity levels grouped into two dominant values. One simple way to segment objects from the background is to select a threshold T that separates these modes. Then any point (x,y) for which $f(x,y) \geq T$ is known as an object point; otherwise, the point is known as a background point **[104]**. The thresholded image g(x,y) is defined as

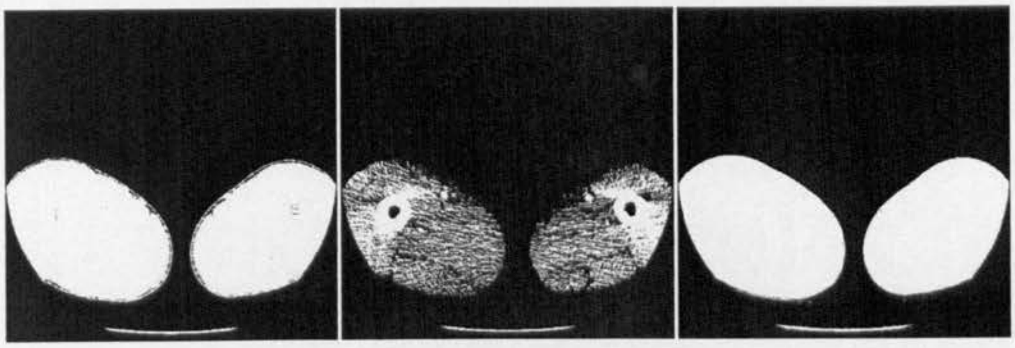
$$g(x,y) = \begin{cases} 1 & if\ f(x,y) \geq T \\ 0 & if\ f(x,y) < T \end{cases} \tag{4.2}$$

The thresholding process can be successful if an appropriate threshold, *T*, is chosen as shown in Figure 4.3.





(i) An image (ii) Image histogram



(iii) Too high (iv) Too low (v) Just right

Figure 4-3 Image thresholding with different threshold values

The gray level histogram ideally includes two separate distributions which represent foreground and background objects without any overlap. Any value between two separate distributions on the valley floor[2] can be chosen as a threshold value, *T*. However, in a real case, there is a bimodal gray level histogram resulting from the overlap of the two underlying distributions. Any value between the peaks or modes at the bottom of the valley, *T1* and *T2*, can be selected as conventional threshold as shown in Figure 4.4 (The pixels in region 'A', above the threshold T1 or T2 comprise the foreground). The sensitivity to small errors in selecting the threshold can be reduced by using this method.

[2] Valley floor is a flat area between two distributions in the gray level histogram. For example, the valley floor in Figure 4.3 (ii) is an area between -800 and -200.

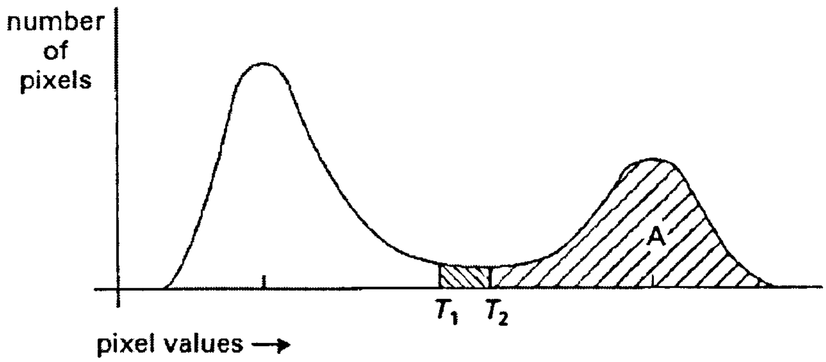


Figure 4-4 A bi-modal histogram [104]

The implementaion of thresholding method can be difficult in a number of conditions such as poor image contrast, varying intensity background, poor spatial resolution, and variable illumination of an object with varying levels of brightness. The histogram in poor image contrast contains overlapping peaks, thus it is difficult to find a threshold value for distinguishing foreground from background. Also, choosing a single threshold for the other conditions can add to the difficulty **[105]**. The variety of conditions under which segmentation is performed, requires different approaches, some of which are explained below. The best parameters for choosing a suitable approach are type of image and experiment.

4.2.1.1. Optimal Thresholding

The optimal threshold is not the bottom of the valley between peaks in the gray level histogram. However, the optimal thresholding considers the histogram of an image to be a weighted sum of two (or more) probability densities. Therefore, the threshold is set as the gray level which results in the smallest number of pixels being misclassified, i.e., background pixels being classified as foreground and vice versa. This means that the intersection of the two normal distributions can be chosen as optimal threshold instead of the bottom of the valley between the two peaks which is a conventional threshold as shown in Figure 4.5.

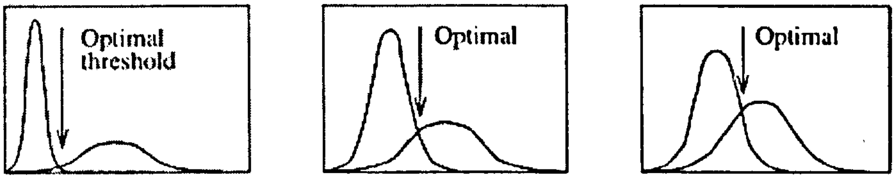


Figure 4-5 Gray level histograms approximated by two normal distributions [104]

There are a number of approaches to implement optimal thresholding (for example see [105]). The general methodology is to consider the pixels as belonging to two classes or clusters; foreground and background. The goal is to pick a threshold such that each pixel on each side of the threshold is closer in value to the mean of the pixels on that side of the threshold than the mean of the pixels on the other side of the threshold.

The *Otsu* [106] method describes the gray level histogram of an image as a probability distribution, so that

$$p_i = {}^{n_i}/_{N}, \qquad \sum_{i=1}^{L} p_i = 1 \tag{4.3}$$

Where L is the number of gray levels (e.g., 256 for an 8 bit image). The number of pixels with gray value i is denoted by n_i and N is the total number of pixels in the image, so p_i is the probability of a pixel having gray value i. Suppose that the pixels are divided into two classes, c_0 and c_1, by a threshold at level k. The class probabilities, ω_0 *and* ω_1, are

$$\omega_0 = p(c_0) = \sum_{i=1}^{k} p_i = \omega(k) \tag{4.4}$$

$$\omega_1 = p(c_1) = \sum_{i=k+1}^{L} p_i = 1 - \omega(k) \tag{4.5}$$

$$\omega_0 + \omega_1 = \sum_{i=1}^{L} p_i = 1 \tag{4.6}$$

and the class mean levels are

$$\mu_0 = \sum_{i=1}^{k} i \ p(i|c_0) = \sum_{i=1}^{k} i \, p_i/\omega_0 = \mu(k)/\omega(k) \tag{4.7}$$

$$\mu_1 = \sum_{i=k+1}^{L} i \ p(i|c_1) = \sum_{i=k+1}^{L} i \, p_i/\omega_1 = \frac{\mu_T - \mu(k)}{1-\omega(k)} \tag{4.8}$$

Where μ_T is the total mean level of the original picture and is calculated using the following equation:

$$\mu_T = \sum_{i=1}^{L} i \, p_i \tag{4.9}$$

The class variances are given by

$$\sigma_0^2 = \sum_{i=1}^{k} (i-\mu_0)^2 \ p(i|c_0) = \sum_{i=1}^{k} (i-\mu_0)^2 \, p_i/\omega_0 \tag{4.10}$$

$$\sigma_1^2 = \sum_{i=k+1}^{L} (i-\mu_1)^2 \ p(i|c_1) = \sum_{i=k+1}^{L} (i-\mu_1)^2 \, p_i/\omega_1 \tag{4.11}$$

We can define the within-class variance, σ_W^2, as the weighted sum of the variance of each class:

$$\sigma_W^2 = \omega_0\sigma_0^2 + \omega_1\sigma_1^2 \tag{4.12}$$

Computing the σ_W^2 for each of the two classes for each possible threshold involves a lot of computation, but there is an easier way.

Subtracting the within-class variance from the total variance gives the between-class variance, σ_B^2, the sum of the weighted squared distances between class means and grand mean, i.e.,

$$\sigma_B^2 = \sigma_T^2 - \sigma_W^2 = \omega_0(\mu_0-\mu_T)^2 + \omega_1(\mu_1-\mu_T)^2 \tag{4.13}$$

$$= \omega_0\omega_1(\mu_1-\mu_0)^2$$

Where the total variance, σ_T^2 , of the combined distribution is given by

$$\sigma_T^2 = \sum_{i=1}^{L} (i - \mu_T)^2 p_i \tag{4.14}$$

The Equation (4.13) can be simplified:

$$\sigma_B^2 = \omega(k)(1 - \omega(k)) \left(\frac{\mu_T - \mu(k)}{1 - \omega(k)} - \frac{\mu(k)}{\omega(k)}\right)^2$$

$$= \omega(k)(1 - \omega(k)) \left(\frac{\omega(k)(\mu_T - \mu(k)) - \mu(k)(1 - \omega(k))}{(1 - \omega(k))\, \omega(k)}\right)^2$$

$$= \frac{(\mu_T\, \omega(k) - \mu(k))^2}{(1 - \omega(k))\, \omega(k)}$$

$$= \frac{(\mu_T\, \omega(k) - \mu(k))^2}{\mu(k)\, \omega(k)} \tag{4.15}$$

Since the total variance, σ_T^2, is constant and independent of k, the effect of changing the threshold is merely to move the contributions between σ_B^2 and σ_W^2. Thus maximizing the between-class variance is equivalent to minimizing the within-class variance. Therefore, the optimal threshold k' is

$$\sigma_B^2(k') = max\ \sigma_B^2(k) \quad 1 \le k < L \tag{4.16}$$

or

$$\sigma_W^2(k') = min\ \sigma_W^2(k) \quad 1 \le k < L \tag{4.17}$$

Thus k' is chosen to maximize the separation of the two classes (foreground and background), or alternatively minimize their spread, so that their overlap is minimized. The method is quite general and can be applied to features other than brightness. The obtained optimal threshold is stable based on the integration of the gray level histogram (a global property) rather than its differentiation (a local property such as the valley). The between-class variance is always smooth and unimodal and makes it easy to find the maximum.

When the distributions are constrained to be normal (i.e., Gaussian) the method is equivalent to *mixture modelling* [107]. In this method, the gray level distributions corresponding to n individual regions are fitted to each local histogram h_{region}, which is modelled as a sum h_{model} [107] of n Gaussian distributions so that the difference between the modelled and the actual histograms is minimized.

$$h_{model}(g) = \sum_{i=1}^{n} a_i \, e^{-(g-\mu_i)^2 / 2\sigma_i^2} \tag{4.18}$$

Where a_i, μ_i, and σ_i represent parameters of the Gaussian distribution for the region i, Variable g shows gray level values from the set G of image gray level and the optimal parameters of the Gaussian distributions are determined by minimizing the F:

$$F = \sum_{g \in G} \left[h_{model}(g) - h_{region}(g)\right]^2 \tag{4.19}$$

Maximum Entropy thresholding [108] is an automatic thresholding technique based on the maximum entropy of the image histogram. It is very similar to Otsu's method except that the former maximising the between-class variance and the latter maximises the between-class entropy. Entropy [109] is a measure of the uncertainty of an event and is defined as:

$$S = -\sum p_i \, . \log_2(p_i) \tag{4.20}$$

Where p is the probability of a pixel gray scale value, i, in the image. The entropy of image is:

$$H_T = -\sum_{i=1}^{L} p_i \, . \log_2(p_i) \tag{4.21}$$

L is the number of gray levels (e.g., 256 for an 8 bit image). Let B, F, and k represent the class of background pixels, the class of foreground pixels, and thresholding level respectively. The entropy of background, $H_B(k)$, and foreground pixels, $H_F(k)$, are given by **[108]**

$$H_B(k) = -\sum_{i=1}^{k} \frac{p_i}{p(B)} \cdot \log_2 \frac{p_i}{p(B)} \tag{4.22}$$

$$H_F(k) = -\sum_{i=k+1}^{L} \frac{p_i}{p(F)} \cdot \log_2 \frac{p_i}{p(F)} \tag{4.23}$$

$$p(B) = \sum_{i=1}^{k} p_i \,, \qquad p(F) = \sum_{i=k+1}^{L} p_i \,, \qquad p(B) + p(F) = 1 \tag{4.24}$$

$p(B)$ and (F) are the class probabilities for background and foreground pixels. Now, let the information between the two classes be:

$$\varphi(k) = H_B(k) + H_F(k) \tag{4.25}$$

$$\varphi(k) = \log[p(B)p(F)] + \frac{H(k)}{p(B)} + \frac{H_T - H(k)}{p(F)} \tag{4.26}$$

Where

$$H(k) = -\sum_{i=1}^{k} p_i \,.\log_2(p_i) \tag{4.27}$$

The Maximum Entropy thresholding selects the thresholding k_{max} in which $\varphi(k_{max})$ is maximum **[108]**.

Optimal thresholding can be implemented iteratively by the *isodata* (iterative self organizing data analysis technique algorithm) method **[110]**. The steps are as follows:

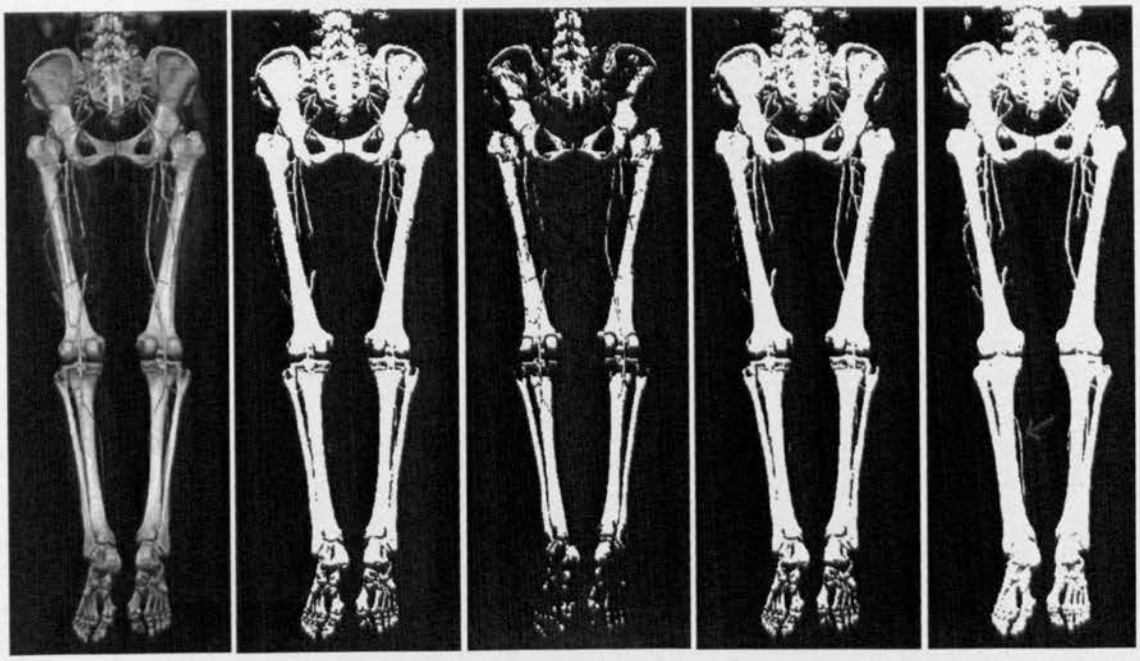
1- Threshold the image using the mean of the two peaks or the mean pixel value, T_0

2- Calculate the mean value of the pixels below this threshold, μ_1, and the mean of the pixels above this threshold, μ_2

3- Threshold the image at a new threshold, $T_i = (\mu_1 + \mu_2)/2$

4- Repeat steps (2)–(3) until $T_i - T_{i-1} \leq \Delta$ (where the change, Δ, can be defined in several different ways, either by measuring the relative change in threshold value or by the percentage of pixels that change sides (foreground to background or vice versa) between iterations).

Figure 4.6 depicts the results of different optimal thresholding algorithms applied on a Maximum Intensity Projection (MIP) **[111]** of a typical peripheral artery computed tomography angiography. Since the results depend on the individual histograms, the choice of the most appropriate algorithm differs with each particular image; however, these results have been assessed by two qualified radiologists based on visual judgment and the accuracy of segmentation by mixture modelling algorithm shows better results in comparison with the other ones as represented in Figure 4.6 (The artery shown in Figure 4.6 (v) with red arrow is not detected by the other three methods).



(i) Original image (ii) Isodata (iii) Maximum entropy (iv) Otsu (v) Mixture modeling

Figure 4-6 Different optimal thresholding algorithms applied on a typical MIP data

4.2.2. Region Growing Methods

Region based methods find connected regions based on the similarity of the pixels within them. The objective is to produce connected regions that are as large as possible (i.e., produce as few regions as possible), allowing for some flexibility within each region. However, if we require the pixels in a region to be too similar, we may over-segment the image, and if we allow too much flexibility, we may merge objects that should be separate. The goal is to find regions that correspond to objects as a person sees them, which is not an easy goal. Region growing is a bottom-up procedure because it starts from "seed" pixels inside the image, and then grows to the whole image by adding neighbouring pixels that have similar properties (e.g. brightness, colour, texture, gradient, geometric properties) to the seed. Connectivity (4 or 8) is used to define neighbouring pixels. We can specify a variance for the property and region growing stops when a pixel is not within the variance. The seeds can be chosen interactively, although automatic segmentation is preferable.

Region growing techniques are generally better in noisy images where edges are difficult to be detected. They are particularly useful with images that have multi-modal histograms.

4.2.3. Boundary Based Methods

Boundary based methods are based on finding pixel differences rather than pixel similarities. The goal is to determine a closed boundary such that an inside (the object or foreground) and an outside (the background) can be defined.

4.2.3.1. Edge Detection and Linking

Edges in an image are detected by using a gradient operator such as the Sobel operator [112] and then thresholding the magnitude of the gradient image. The strongest edges are distinct, but some weaker edges appear broken; noisy images compound the problem, resulting in spurious edges. Smoothing the noisy image reduces the spurious edges, but widens the edges and removes some weak edges completely. Some links of the edges to form a connected boundary are needed. Adjacent edge pixels could be linked if they had similar properties, e.g., a similar gradient magnitude and orientation based on the Sobel results.

Once the links are established, the linked edges become the borders. The linked pixels need to be constrained by, for example, being scanned along the rows or columns; otherwise, clusters of linked pixels are formed rather than long single-pixel thick chains. Edge linking is usually followed by post-processing to find sets of linked pixels separated by small gaps which can then be filled in.

4.2.3.2. Boundary Tracking

Boundary tracking **[107]** may be applied to a gradient image or any other image containing only boundary information. Once a single point on the boundary has been identified, simply by locating a maximum gray level, the analysis proceeds by following or tracking the boundary, assuming it to be a closed shape, with the aim of finding all other pixels on that specific boundary, and ultimately returning to the starting point before investigating other possible boundaries. In one implementation, the search for the highest gray level pixels continues broadly in the same direction as in the previous step, with deviations of one pixel to either side permitted, to accommodate curvature of the boundary.

This simple method is liable to fail under conditions of high noise, when the boundary makes seemingly random and abrupt changes of direction that cannot be successfully tracked in this way. Substantial low-pass filtering is needed beforehand to reduce noise, unless the algorithm used is refined or assisted manually. Simultaneously tracking both sides of a long, thin object, such as a blood vessel, requires each tracking to continuously check the other border and becomes difficult when the object branches or crosses other objects. In sum the extension of boundary tracking to surfaces is complicated **[107]**.

4.3. Pre & Post Processing Techniques

For extracting image components, which are useful in the representation and description of regions shape, some pre and post processing methods are used. Some of the methods are described in following section.

4.3.1. Morphological Methods

Morphology relates to the structure or form of objects. Morphological methods simplify a segmented image to facilitate the search for objects of interest. This is done by smoothing out object outlines, filling small holes, eliminating small projections, and other similar techniques [102].

The two principal morphological operations are dilation and erosion, which are described as bellow.

4.3.1.1. Dilation

Dilation is an operation that allows objects to grow or thicken, thus potentially is filling in small holes and connecting disjointed objects. The specific manner and extent of this thickening is controlled by a shape referred to as a structuring element. Basically, the structuring element is used to probe the image to find how it will fit, or not fit, into the image objects.

The dilation process is performed by laying the structuring element on the image and sliding it across the image in a manner similar to convolution. The difference is in the operation performed. It is best described in a sequence of steps.

1- If the origin of the structuring element coincides with a 0 in the image, there is no change; move to the next pixel.
2- If the origin of the structuring element coincides with a 1 in the image, perform the OR logic operation on all pixels within the structuring elements.

Mathematically, dilation is defined in terms of a set of operations. The dilation of A by B, which denotes $A \oplus B$, is defined as

$$A \oplus B = \{Z | (\hat{B})_z \cap A \neq \emptyset\} \qquad (4.28)$$

Where A, B and $\hat{B}$ are the image, the structuring element, and the reflection of structuring element respectively and $\emptyset$ denotes an empty set. In words, the dilation of A by B is the set consisting of all the structuring element origin locations where the reflected and translated B overlaps at least some portions of A.

4.3.1.2. Erosion

Erosion is an operation that allows objects to shrink or thin. The erosion process is similar to dilation, but we turn pixels to 0, not 1. Slide the structuring element across the image and then the algorithm is described as follows:

1- If the origin of the structuring element coincides with a 0 in the image, there is no change; move to the next pixel.
2- If the origin of the structuring element coincides with a 1 in the image and any of the 1 pixel in the structuring element extend beyond the object (1 pixel) in the image, then change the 1 pixel in the image, whose location corresponds to the origin of the structuring element, to a 0.

The mathematical definition of erosion is similar to that of dilation. The dilation of A by B, which denotes $A \ominus B$, is defined as

$$A \ominus B = \{Z | (B)_z \cap A^c = \emptyset\} \qquad (4.29)$$

Where A^c is the complement of A.

4.3.2. Connected Component Labelling

The method of connected component labelling scans each pixel in an image, generally from top to bottom and left to right, to identify regions containing connected pixels. These regions contain adjacent pixels that share the same membership criteria such as pixel intensity, colour, or texture. Connected component labelling is used on binary or gray level images where different measures of connectivity are possible. Assume we have a binary input image $I = (0, 1)$ and use a 4-connected neighbourhood for our pixels adjacent relationship. The connected components labelling method scans the image along each row until it encounters a pixel p, which denotes a pixel to be labelled and has pixel intensity $I=1$. When this occurs, it examines the four neighbours of p that have already been encountered during the scan. These four neighbours are to the left of pixel p, to the right, above and below it. Based on such information, p is labelled as the follows:

- If all four neighbours are 0, p is assigned with a new label
- If only one neighbour has pixel intensity $I = 1$, p is assigned with its label
- If more than one of the neighbours have pixel intensity $I = 1$, p is assigned with one of the labels and a note is made up of the equivalences

Upon the completion of the scan, pixels with the same label are sorted into the same classes, and a unique label is given to each class.

4.4. Proposed Segmentation Approach

This section utilizes combined and adjusted methods which are described in section 4.2 and 4.3 to develop a fully automatic segmentation algorithm to identify different regions in the vascular imaging and in particular peripheral arteries in 3D CTA images (i.e., background, body, bone and arteries) **[113]** **[114]**.

The proposed segmentation approach contains two parts: (1) Background and body segmentation (This refers to the Secondary Region of Interest (SROI) in Chapter 6) and (2) Peripheral arterial segmentation (This refer to the Primary Region of Interest (PROI) in Chapter 6).

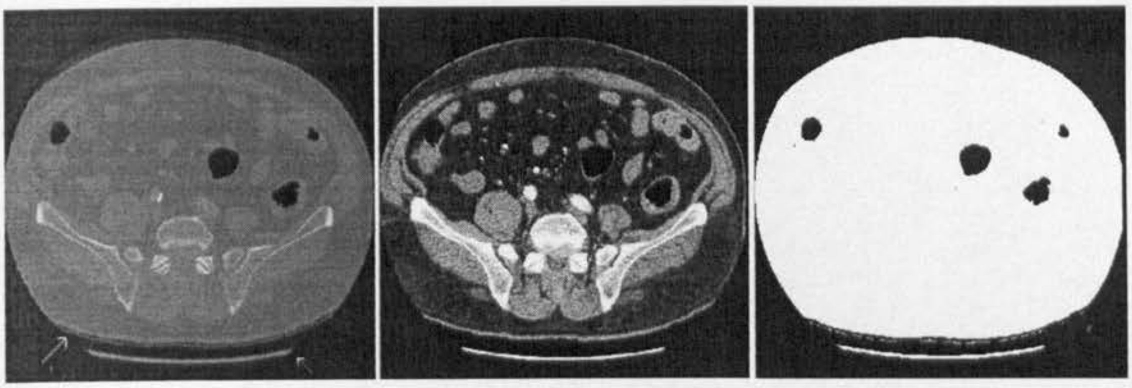
4.4.1. Background and Body Segmentation

The method for body segmentation contains three phases: the first phase is to extract the approximate body, and the second phase is the removal of unnecessary areas and the last phase is to add the inner holes into the body to the final result. All phases run sequentially and are applied in a slice-to-slice over a dataset. This specific strategy has been designed to reduce the overall processing time needed for the pre-processing of a scan when pre-processing is combined into a Picture Archiving and Communication System (PACS) in an on-the-fly mode. In this way, images can be processed sequentially by three phases as they are received from a CT scanner.

4.4.1.1. Approximate Extraction of Body Region

A fixed thresholding value of the body is around -300 Hounsfield units (HU); however, this can vary a little from one CTA to another. Therefore, to optimise the efficiency of thresholding method, an optimal one can be used. Because the contrast between the whole body and background is large, the Otsu method (see section 4.2.1.1) is used to extract the body region. Range of finding optimisation value is restricted to [-300 -200].

The body region extracted from the CTA image contains patient table and patient clothes as illustrated in Figure 4.7a with arrows. The output of this step is a rough image of body region (Figure 4.7c).

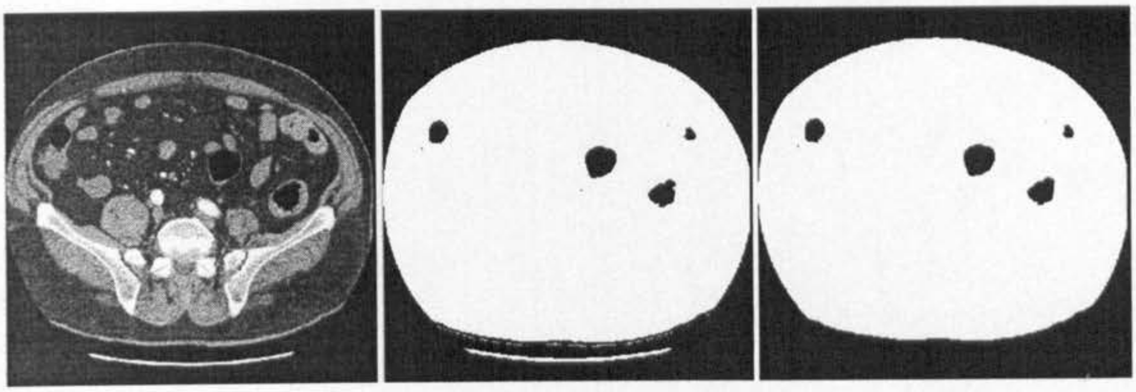


(a) CTA image (normal range) **(b)** CTA image [-150 200] **(c)** Mask of rough body region

Figure 4-7 Approximate extraction of body region

4.4.1.2. Removal of Unnecessary Area in Image

In this step, some unnecessary regions which contain the patient table, patient clothes and noises are removed from output of the previous phase by a morphological operator; in this case, the opening operator is used. The opening operator makes the border of the input region smooth by scanning the inside of the area with a structuring element while removing the areas, which are not covered, by the structuring element. Therefore, the patient table and patient clothes are removed from the body region as illustrated in Figure 4.8. A circle with a radius of three pixels has been used for the structuring element.

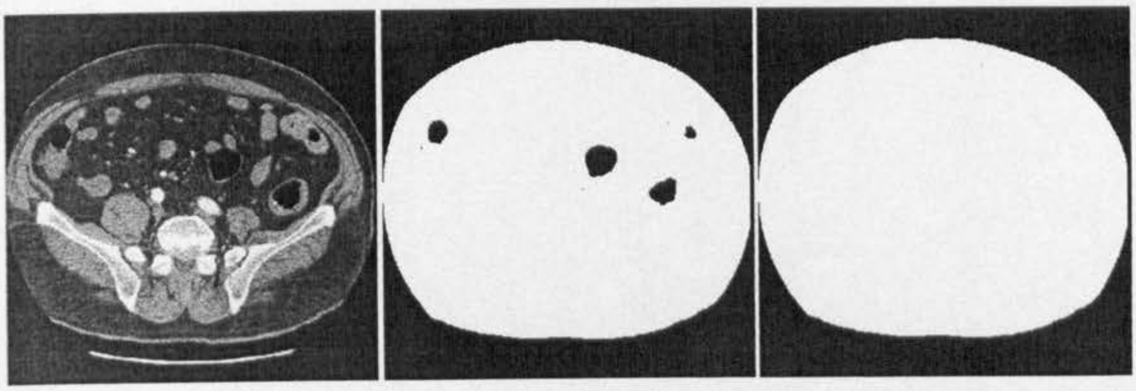


(a) CTA image **(b)** Body region mask from step 1 **(c)** Body region mask without patient clothes and table

Figure 4-8 Removal of unnecessary areas in image

4.4.1.3. Adding the Inner Holes to Body Region

After thresholding in the first step, some organs in the body region that contain air such as the bowel, colon, and stomach appear as holes in the mask as shown in Figure 4.9b. To add such inner holes to the body mask, the connected component analysis (see section 4.3.2) is applied to the result of the previous step as illustrated in Figure 4.9c.



(a) CTA image **(b)** Body region mask from step 2 **(c)** Final body region mask

Figure 4-9 Adding the inner holes to body region

4.4.2. Peripheral Arterial Segmentation

The aim of this phase of the proposed approach is to detect the peripheral arteries which start from the abdominal aorta, as shown in Figure 4.2. The segmentation method contains three steps: (1) Initial segmentation of arteries, (2) Separation of arteries from bone, and (3) Refinement of the arteries.

4.4.2.1. Initial Segmentation of Arteries

In the first step, initial segmentation of the arteries from other tissues is performed by an optimal based segmentation (mixture modelling) which is compared with the other segmentation methods in section 4.2.1.1 and Figure 4.6. It could successfully segment artery and bone from other tissues. To optimise the efficiency of the thresholding method, the range of finding the optimisation value is calibrated by Hounsfield Unit (HU) of artery, restricted between 200 and 300 **[115]**. Then, two morphological operations, the opening and closing operations are applied to separate the connectivity between arteries and bones. A circle with a radius of three pixels has been used for the structuring element.

The resulting images after applying the pre processing step over some slices are shown in Figure 4.14b; it can be observed that the results contain arteries and the boundary of bones.

4.4.2.2. Separation of Arteries from Bone

The 3D region growing algorithm described in section 4.2.2 is employed to extract arteries. The advantages of using this region growing algorithm in comparison with other segmentation methods (described in section 4.2) are to create perfectly thin and fully connected borders of regions (edges). Thus, the shape of regions will not be corrupted (see Figure 4.12). Furthermore, this method is very stable with respect to noise.

In addition, the traditional region growing algorithm usually has two major issues. The first issue is the selection of initial seed point. Incorrect selection of the seed point leads to inaccurate segmentation; therefore, an automatic process is always preferred. The second issue is that even with automated process, the selected seed point may lie on an edge. However, those issues are fixed in our proposed algorithm by choosing the initial seed point automatically.

The 3D region growing algorithm starts from a seed point automatically selected from the top slices and abdominal aorta by the following algorithm.

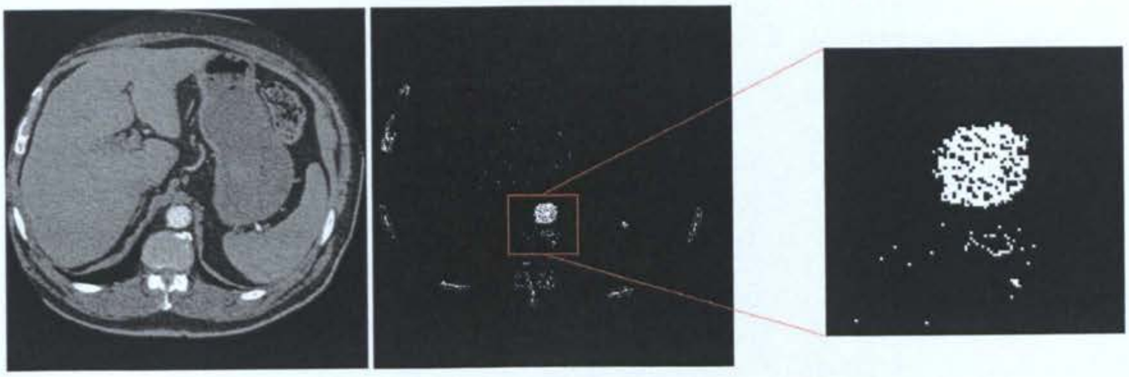
Algorithm 4.1 Seed point selection

1. Extracting prominent objects which include abdominal aorta is based on an optimal based segmentation method (mixture modelling- see section 4.2.1.1). Range of finding optimisation value is calibrated by Hounsfield Unit (HU) of artery, restricted between 200 and 300, as shown in Figure 4.10b.

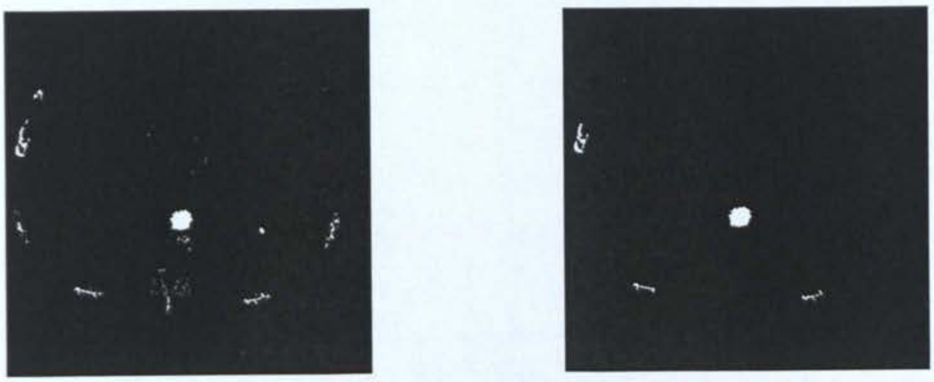
2. Morphological closing operation is used to merge narrow breaks and eliminate small holes in step 1, as shown in Figure 4.10c.

3. The area of all objects, extracted from step 2, are calculated by a 2D connected component labelling. The normal area of an abdominal aorta is around 3 cm2 [116]. The areas less than 0.5 cm2 are removed from the target region, as shown in Figure 4.10d.

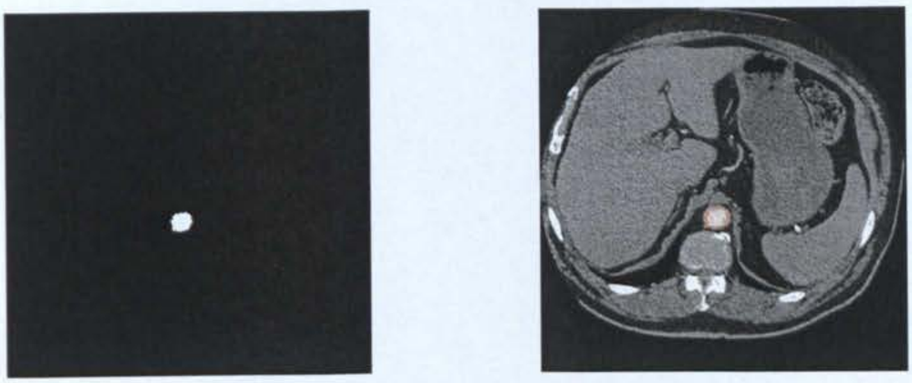
4. The circularity of the remaining objects in step 3 is measured according to the ratio of pixels in an object to total pixels in the smallest rectangles containing the objects. The similarity to a circle increases when the ratio is close to 1. Therefore, the centre of the object which bears the most similarity to a circle is selected as a seed point, as shown in Figure 4.10e.



(a) Original image **(b)** Thresholding result



(c) Merging narrow breaks and eliminating small holes **(d)** Removing small areas



(e) Final mask (seed points inside the aorta) **(f)** The selected seed points over the original image

Figure 4-10 Selecting seed point inside the abdominal aorta

Figure 4.11 shows the final result of the automatic selection of a seed point for three different datasets.

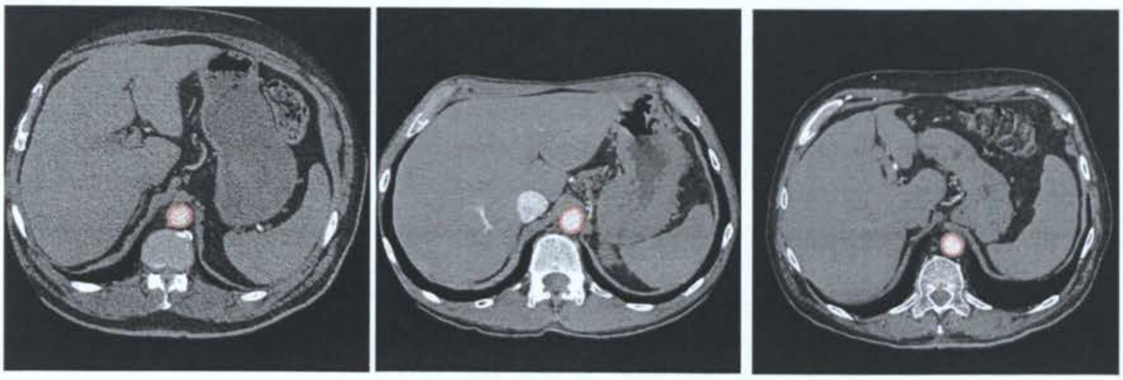


Figure 4-11 Selected seed points inside the abdominal aorta in 3 different datasets

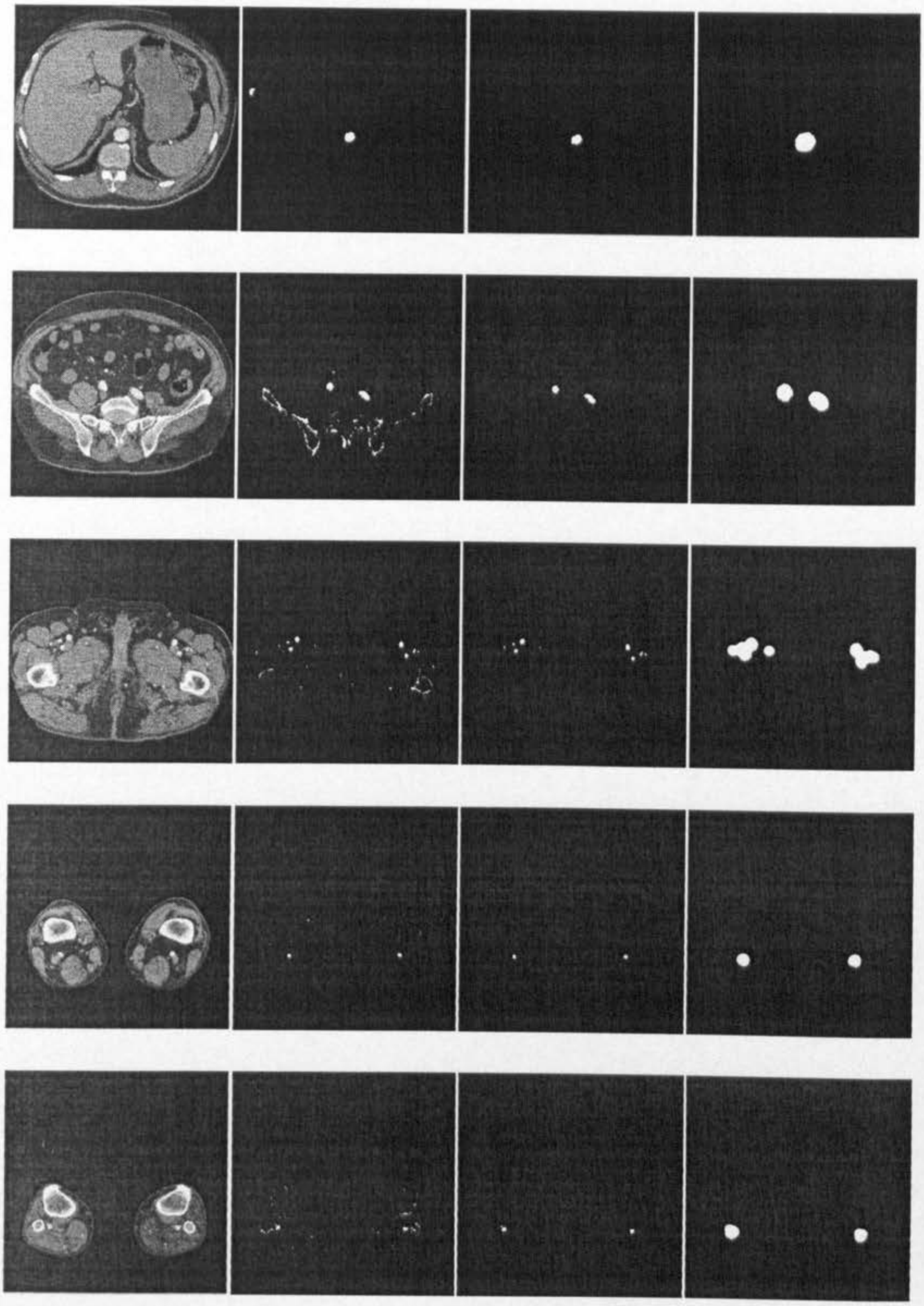
The region is growing from the seed point from top slice to down, based on the similarity of voxel values which vary from 200 HU to 300 HU for arteries [**115**]. The results of arterial segmentation for a dataset are shown in Figure 4. 14c; moreover, the 3D results of arterial segmentation for the three diffrent datasets are presented in Figure 4.12. The results have been assessed by two qualified radiologists based on Diagnostic MOS test and got a satisfactory mark. More details about the assessment are represented in Chapter 6.

Figure 4-12 3D results of arterial segmentation in three different datasets

4.4.2.3. Refinement of the Arteries

The surrounding area of extracted region, arteries, from previous step can be important to diagnose the disease and to increase the precision of segmentation; therefore, this area should be added to the output of the previous step as the final region. Experimentally, the size of the surrounding area is found to be approximately 2.5 times the size of the arteries. Morphological operations, eroding and dilating are applied to do the post processing step. A sample result of arterial segmentation approach for a dataset is shown in Figure 4. 14d; in addition, the 3D results of arterial egmentation for the three diffrent datasets are presented in Figure 4.13.

Figure 4-13 3D results of arterial segmentation after refinement of the arteries



(a) Original images **(b)** pre-processing **(c)** 3D region growing **(d)** post processing

Figure 4-14 The segmentation results for several slices in a dataset

4.5. Summary

A fully automated ROI based segmentation approach is proposed for detection of different Region of Interests (ROIs) over 3D CTA images. The proposed algorithm consists of background & body segmentation (SROI) and peripheral artery segmentation (PROI). SROI is segmented by using an adaptive thresholding method, followed by morphological operations and connected component analysis. The process of PROI segmentation is an initial segmentation of arteries from other tissues followed by a 3D region growing algorithm, in combination with morphological operators to increase the precision of segmentation. The segmentation approach also includes an automatic seed point selection technique.

The results of applying the segmentation approach on several medical datasets have been assessed by two qualified radiologists based on Diagnostic MOS test and got a satisfactory mark. More details about the assessment have been represented in Chapter 6.

5. ROI[1] BASED COMPRESSION APPROACH

In this chapter, the proposed context and ROI based compression approach is presented in peripheral arterial imaging 3D CT Angiography scan and also the proposed rate control algorithm is discussed.

We choose the peripheral arteries as a case study because it imposes a challenge for archiving and proper data transmission (i.e., a dataset of peripheral arteries contains around 2000 medical images with a total size of 1GB).

5.1. Compression Approach

Several ROI based coding techniques have been discussed in Chapter 2; they have two disadvantages regarding their usage in medical image compression. Firstly, the segmentation of ROIs is not fully automated; therefore, it could be extremely time consuming to annotate the ROIs by a radiologist and secondly, for implementing those methods, some definite modifications on the standard image compression are required. This modification necessarily changes, at least in part, the standard of compression

[1] Region of Interest

algorithms approved in DICOM standard. DICOM standard committee is highly conservative about adding new schemes since the incremental benefit of slightly improved performance, or an extra feature, rarely justifies the risk of compromising interoperability. Medical devices are also highly regulated (e.g., by FDA in the US), so they are also expensive to change given the burden of required testing and documentation [27] [28]. Conformance to the standard is mandatory for medical image compressions, which have been integrated into pre-existing PACS.

In order to increase the compression efficiency while properly addressing the above mentioned issues, a fully automated context and ROI based compression approach in compliance with FDA regulation is introduced in vascular medical imaging. Although the method can be applied on any vessel or even expanded on the other anatomy, the application of the proposed method is described for peripheral arteries.

The ROI approach allows for important parts of an image to be coded with better quality than the rest of the image. Images can be split into (1) Primary Region of Interest (PROI), (2) Secondary Region of Interest (SROI), and (3) Background. Each region is described briefly below.

1) Primary Region of Interest (PROI): The PROI is the main region of an image which can be annotated either manually by the radiologist, as illustrated in Figure 5.1, or automatically by a Computer Aided Detection system[2] (CAD).

The PROI contains important information; hence, this region should be kept without any loss of information.

[2] CAD is a sophisticated computer program to provide assistance to radiologists to detect the presence of disease.

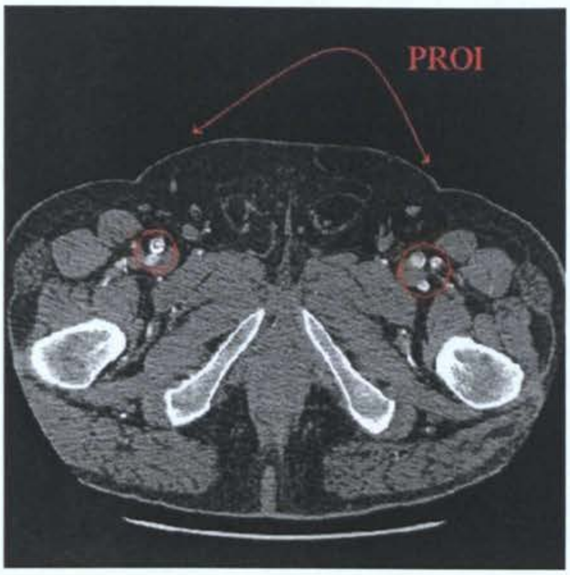


Figure 5-1 The PROI is the arterial tree, highlighted with red circles.

As described in Chapter 4, the arterial tree plays an important role for the diagnosis of PAD. For this reason, in our application, the arterial tree is a good candidate for PROI. For detecting the arterial tree, the proposed method in Chapter 4, section 4.2 is utilized.

This region contains the main clinically important information, thus a lossless region based technique is applied to compress this part of the images.

2) Secondary Region of Interest (SROI): The SROI can be divided into many sub regions with different degrees of interest, but in this scenario, PAD, there is only one region, which contains bones and other tissues in the body, as illustrated in Figure 5.2.

The proposed method in Chapter 4, section 4.1 is used to extract the SROI region, i.e., bones and tissues. For this SROI, lossy region based compression techniques are used to compress it with medium compression rate because there is some diagnostically irrelevant information, i.e., bones and tissues in this part.

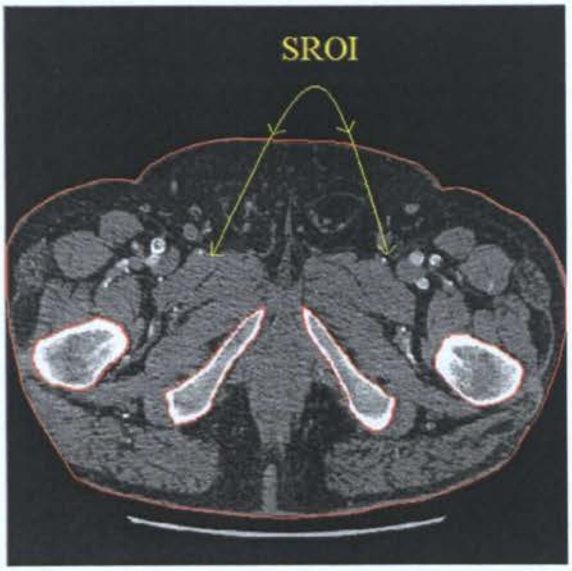


Figure 5-2 The SROI consists of bones and other tissues in the body.

3) Background region: The area outside the human body is background, as shown in Figure 5.3. It does not carry relevant data and thus is compressed with a very high compression rate. The proposed method in Chapter 4, section 4.1 is used to detect the background region.

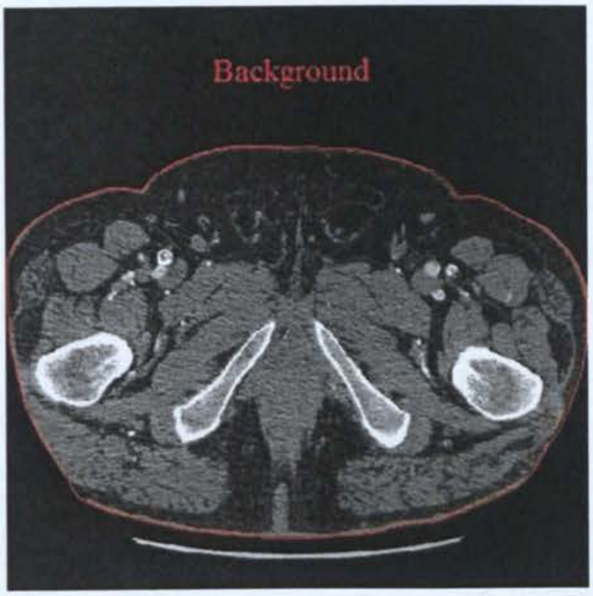


Figure 5-3 The area outside the body is background.

5.2. Description of the Proposed Algorithm

To apply the compression approach on a dataset, an adaptive rate control algorithm is proposed to calculate the bitrate for compressing the slices based on two parameters: (1) The PSNR of SROI and (2) A bitrate constant step. The pseudo-code of this algorithm as depicted in Figure 5.4 is described here, assuming the following definitions:

S_i : Slice Index i=1,..., n_slices

n_slices: Total number of slices

I_{Si}: Original Image

C_{Si} : Compressed image with the proposed method

R_{si} : Bitrate in bits per pixel (bpp)

Δr : Bitrate constant step

Δp : PSNR of SROI

1. Read I_{S1} (First slice of a dataset).
2. Find the minimum bitrate, R_{S1}, for the compression of I_{S1}, such that $PSNR\ (I_{S1}, C_{S1}) \geq \Delta p$
3. For i=2,…, n_slices

a. Calculate the bitrate (R_{Si}) for the next slice (S_i) based on the following algorithm:

$C_{Si} = Compress\ (I_{Si}\ ,\ R_{Si-1} - \Delta r)$

[*Compress the current slice with the rate considered for the previous slice reduced by a constant step*]

If $PSNR\ (I_{Si}, C_{Si}) \geq \Delta p$

$R_{si} = R_{si-1} - \Delta r$

Else

$C_{Si} = Compress\ (I_{Si}\ ,\ R_{Si-1})$

[*Compress the current slice with the rate considered for the previous slice*]

If $PSNR\ (I_{Si}, C_{Si}) \geq \Delta p$

$R_{si} = R_{si-1}$

Else

$R_{si} = R_{si-1} + \Delta r$

End For

Figure 5-4 The proposed rate control algorithm

5.2.1. Selection of Δp

The PSNR requirement for high quality medical image is assumed 40 dB [117] [118]. For any value higher than 40 dB, the original image and the reconstructed image are virtually indistinguishable by human observers. In this algorithm, The PSNR of SROI is denoted by Δp. We set the value of $\Delta p = 40\ dB$. The value of Δp can be properly adjusted according to the application and its requirements in terms of quality.

5.2.2. Selection of Δr

The tissues in the consecutive slices are highly correlated because they are representing the same tissues with a little displacement; the difference between the compression bitrate

Table 5-1 Effect of different values of Δr in two datasets

(a)

Δr	Average bit per pixel	Average file Size (byte)	Average MSE
0.01	2.18	71893	4.39
0.03	2.19	72020	4.09
0.05	2.20	72134	3.87
0.07	2.21	72422	3.64
0.1	2.23	72865	3.29
0.3	2.32	75349	2.23
0.5	2.5	77484	1.57
0.7	2.63	79328	1.31
1	2.76	82023	0.99

(b)

Δr	Average bit per pixel	Average file Size (byte)	Average MSE
0.01	1.58	52266	4.36
0.03	1.59	52393	3.77
0.05	1.60	52680	3.37
0.07	1.61	52971	3.06
0.1	1.63	53488	2.59
0.3	1.74	55800	1.50
0.5	1.82	57576	1.05
0.7	1.93	59292	0.83
1	2.06	61470	0.50

of the two sequence slices is small. In order to select an appropriate value of Δr, the averages of the bitrate in terms of bits per pixel (bpp), file size, and mean square error have been calculated in ten different datasets for different ranges of Δr. The results for two out of these ten datasets have been reported in Table 5.1. The goal is to achieve a low MSE and a small file size. It can be observed that there is a tradeoff between the average MSE and the average file size. As Δr increases, the average MSE decreases, while the average file size increases.

Studying the results for 10 datasets, we observed that decreasing the value Δr lower than 0.1, did not have any considerable effect on reducing the file size. Therefore, we set the value of $\Delta r = 0.1$. The value of this parameter (i.e., Δr) can be properly adjusted according to the application and its requirements in terms of data rate and quality.

5.3. Summary

The context and ROI based compression approach is proposed in this chapter. The proposed approach is utilized to compress 3D CTA scan images that can keep the full skeleton of medical images while the primary regions of interest (PROI), which are critical for medical diagnosis, are maintained intact during the compression process. This means that no information is lost from PROI during the compression process. The secondary regions of interest (SROI) are defined as part of the medical images, which their loss can be tolerated during the diagnosis process. Thus, for the SROI, diagnostically irrelevant information that has no impact on diagnosis decision is lost. The proposed rate control algorithm is also discussed to minimize the interactions and manual interventions with the user. In the automatic rate control algorithm, the variable level of compression in the (x,y) plane as well as in the z axis is automatically calculated based on Δp (PSNR of SROI) and Δr (bitrate constant step).

6. RESULTS AND DISCUSSION:

In this chapter, the experimental results of applying the proposed method and volumetric codecs, JP3D and H.264, on patient datasets are reported. A robust performance evaluation schema (Test-bed) is designed to analyze the results based on the objective and subjective quality metrics. In the evaluation system, two groups and seven scenarios are considered and also a comparison between scenarios in each group is demonstrated in this chapter. The proposed method is implemented in Matlab for which a Graphic User Interface (GUI) is designed to represent the results.

This chapter is organised as follows: first, an evaluation system (Test-bed) for analyzing the proposed method is described. The second section contains the experimental results validated by two qualified radiologists from Lausanne hospital.

6.1. Evaluation System (Test-bed)

The proposed method is applied to the vascular images from 3D CT angiography for peripheral artery images on 10 datasets, which are selected randomly from a group of 100 real datasets without any criteria. Each dataset contains more than 1000 slices/ images, where the average age of patients is 65.4. The datasets are collected from the Lausanne Hospital, Switzerland. The images are stored in DICOM format. The size of the images is 512 by 512, with an original bitrate of 16 bpp. The physical sizes of the pixel and slice thickness are 0.7 mm and 1.25 mm respectively. Based on CT image acquisition protocol, the current of tube varies from 134 mA to 381 mA and the voltage of X-ray tube is 120 kV.

The method is implemented in Matlab and a graphic user interface has been realized for assessing reconstructed images, as illustrated in Figure 6.1.

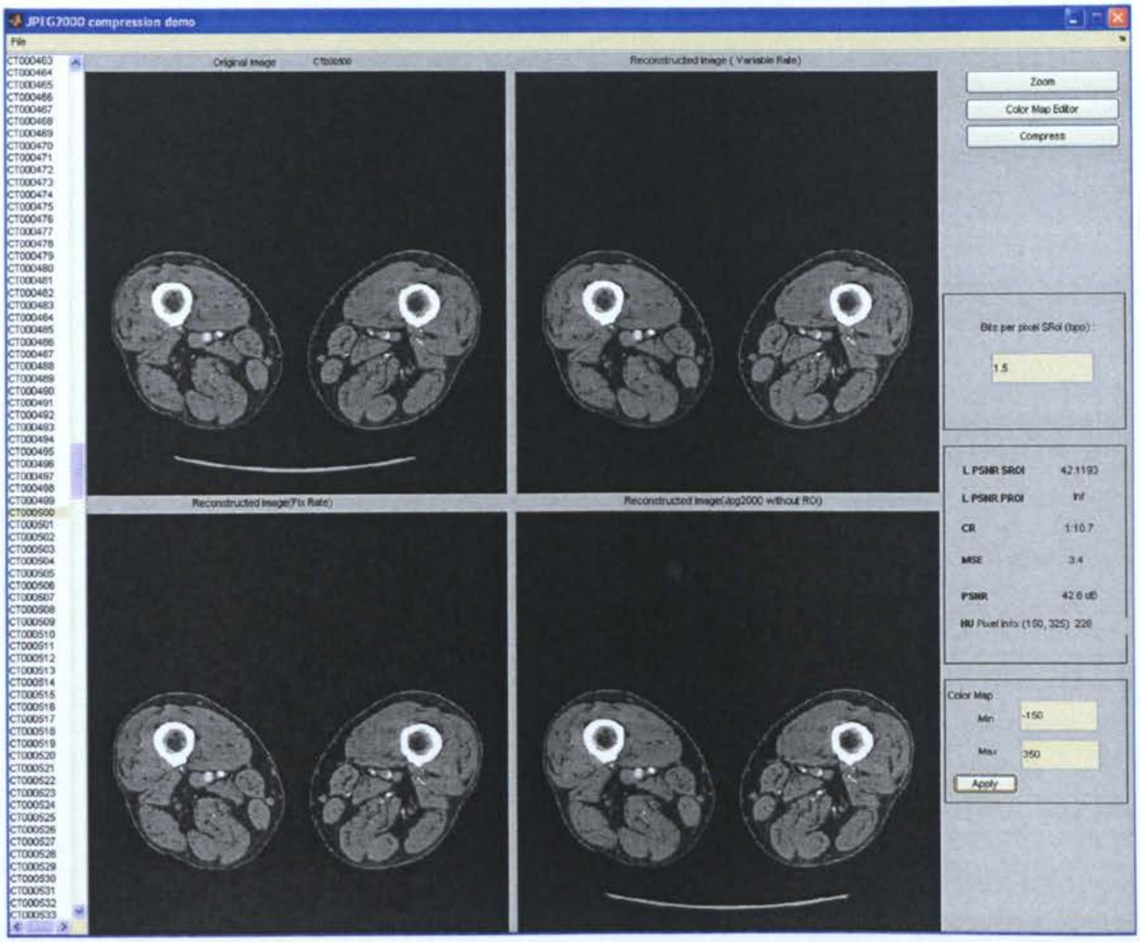


Figure 6-1 The GUI developed for the evaluation of resulting images

Two group scenarios are foreseen to compare the proposed method to lossy and lossless standard methods in this study: Group A consists of our method and lossy scenarios and Group B consists of our method and lossless scenarios.

Three scenarios are considered in Group A and four scenarios are considered in Group B, as follows:

Group A: 1- Proposed method with variable bitrate; 2- Proposed method with fixed bitrate; 3- Lossy JPEG2000 without ROI.

Group B: 1- Lossless JP2K; 2- Lossless JP3D (JP2K: Part 10); 3- Proposed method with variable bitrate; 4- Lossless H.264.

In the first and second scenarios of group A, the proposed method is used to compress each slice in a dataset but the main difference between them is the compression bitrate in each slice. In the first scenario, the compression bitrate is calculated based on the bitrate constant, Δr, and PSNR of SROI, Δp between consequent slices. However, in the second scenario, the compression bitrate of each slice is estimated as a mean of the compression bitrates, which is obtained from the first scenario. Finally, in the last scenario, the method of compression is lossy JPEG2000 without ROI. Moreover, the compression bitrate is the same as the second scenario.

In group B, the proposed method has been compared with three well-known lossless compression methods, JP2K, JP3D, and H.264 in terms of the average size and compression ratio.

As discussed in Chapter 3, JP3D or JP2000 Part 10 is the 3D extension of JP2000 for compression of volumetric datasets. JP3D uses 3D discrete wavelets transform (3D DWT). All of functionality in JP2K is supported in JP3D (see section 3.2.1 for more details). The JP3D VM version 1.1 has been used for evaluation test **[119]**. An overview of the JP3D VM configuration for this research is shown in Table 6.1.

Table 6-1 The JP3D VM configuration

Parameter	Value
Number of Tile	1
Size of Code block	64, 64, 64
Transform method	Reversible 3D DWT(5,3)
Decomposition Level	3
Entropy encoding	3D EBCOT
Lossless Mode	On

H.264 is one of the newest video coding standards. The core of H.264 is similar to that of all digital video coding standards but some new techniques have been used in this standard, for instance motion compensation with variable block size, content adaptive binary arithmetic coding (CABAC), and multiple reference pictures (see section 3.2.2 for more details). The compression performance in this standard has been significantly improved in comparison with the other video codec standard. The H.264 joint-reference model (JM) version 18.2 is utilized for the creation of CTA video streams [120]. An overview of the JM configuration parameters for this research is depicted in Table 6.2.

Table 6-2 The JM configuration

Parameter	Value
Profile	FREX: 244 High 4:4:4
Source Width(pixel)	512
Source Height(pixel)	512
Input Format	YUV
IntraPeriod(Period of IFrame)	1-10
RDOptimization	Enable for FREX (Lossless)
YUVFormat	4:0:0
SourceBitDepthLuma	12
SourceBitDepthChroma	12
OutputBitDepthLuma	12
OutputBitDepthChroma	12
LosslessCoding	Enable
QPISlice	Disable (-24)
QPPSlice	Disable (-24)

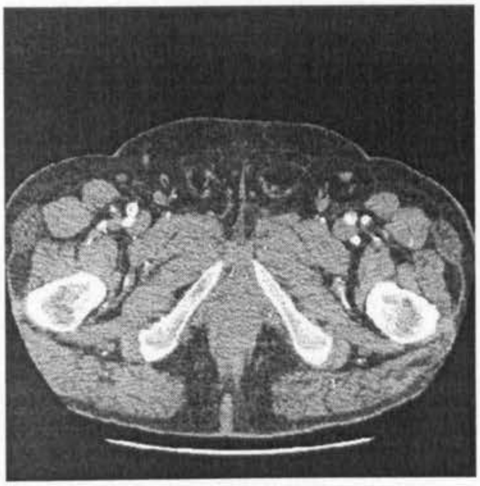
The CTA slices are converted into YUV format. The FRExt Profile, 244 High 4:4:4 (H444P) is used to support a 12 bit depth (see section 3.2.2.2-The FRExt Amendment); moreover, the quantisation parameter is disabled for lossless encoding. The periods of I-Frames are changed between 1 and 10 in this test, but they have not affected the size of compression file. Many parameters in the JM reference software are not used in our test.

6.2. Experimental Results

6.2.1. Group A (Lossy Scenarios)

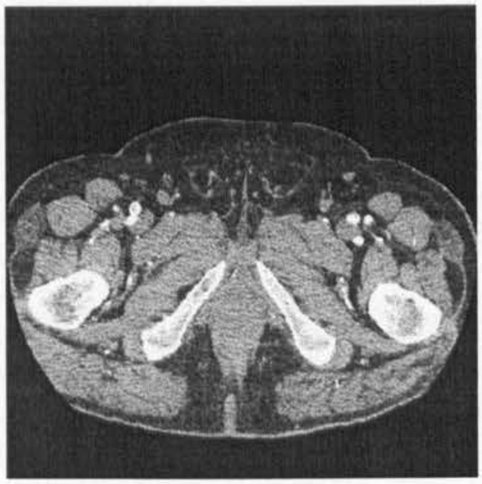
For group A, the three scenarios are applied on peripheral artery images and the experimental results are compared. The obtained results are reported in Table 6.3, Table 6.4, Table 6.5, Table 6.6, Table 6.7, and Table 6.8. The effects of the three methods are demonstrated in Figure 7.2-7.7. The details of tables and figures are discussed below.

The reconstructed images for a typical slice are shown in Figure 6.2. Figure 6.2a shows the original DICOM image (slice 300, size: 512x512, 16 bits). Figure 6.2b, Figure 6.2c, and Figure 6.2d present a reconstructed version of Figure 6.2a in different scenarios.

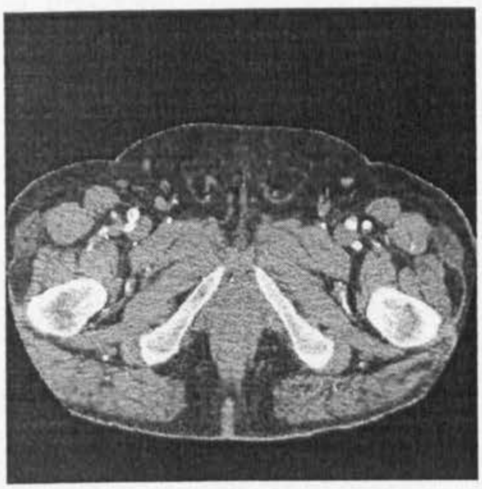


(a) Original Image; slice 300; 512x512

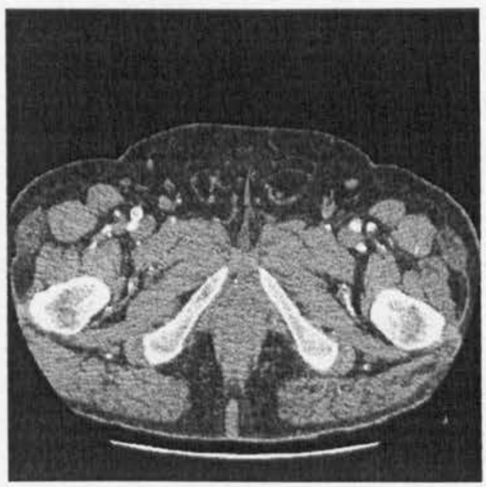
(DICOM-16 bits)



(b) Proposed method with variable bitrate: Reconstructed image with lossless compression in PROI and lossy compression with 3.5 bpp for SROI+ background



(c) Proposed method with fixed bitrate: Reconstructed image with lossless compression in PROI and lossy compression with 2.3 bpp for SROI+ background



(d) Lossy JP2k without ROI support: Reconstructed image with lossy compression with 2.3 bpp for the whole image

Figure 6-2 Comparison of three scenarios

In Figure 6.2b and Figure 6.2c, the PROI is compressed without any losses, while a lossy method is applied to compress SROI and background. For reducing the side effects of lossy compression in the proposed method with a variable bitrate, Figure 6.2b, a minimum PSNR value for loss of information is considered (PSNR $\geq$ 40 dB).

A summary of Compression bitrate, Entropy, PSNR, and MSE values for Figure 6.2 and several slices of a typical dataset[1] are presented in Table 6.3.

Table 6-3 Objective measurements over several slices of a typical dataset

Slice No.	Bitrate1 (bpp)	Entropy (bbp)	PSNR1 (db)	PSNR2 (db)	PSNR3 (db)	MSE1	MSE2	MSE3
1	4.3	5.3	40.70	19.31	31.00	5.52	761.83	51.59
100	4.0	5.3	40.25	23.96	31.43	6.13	260.86	46.71
200	4.1	5.5	41.78	21.43	32.92	4.31	466.91	33.14
300	3.5	5.4	42.10	23.46	35.47	4.00	292.95	18.44
400	2.3	5.1	40.46	42.51	39.96	5.83	3.64	6.55
500	1.6	4.8	40.98	78	41.28	5.18	0.001	4.83
600	1.7	4.9	54.02	78	40.38	0.25	0.001	5.95
700	1.3	4.5	43.29	78	42.43	3.04	0.001	3.71
800	1.1	4.3	54.89	78	44.13	0.21	0.001	2.50
900	1.2	4.5	57.85	78	42.79	0.10	0.001	3.42
1000	0.7	4.2	56.29	78	45.01	0.15	0.001	2.05

[1] One of the 10 datasets is selected randomly as a typical dataset for showing objective and subjective results in this section. Also, the final results for all of the 10 datasets are reported as well.

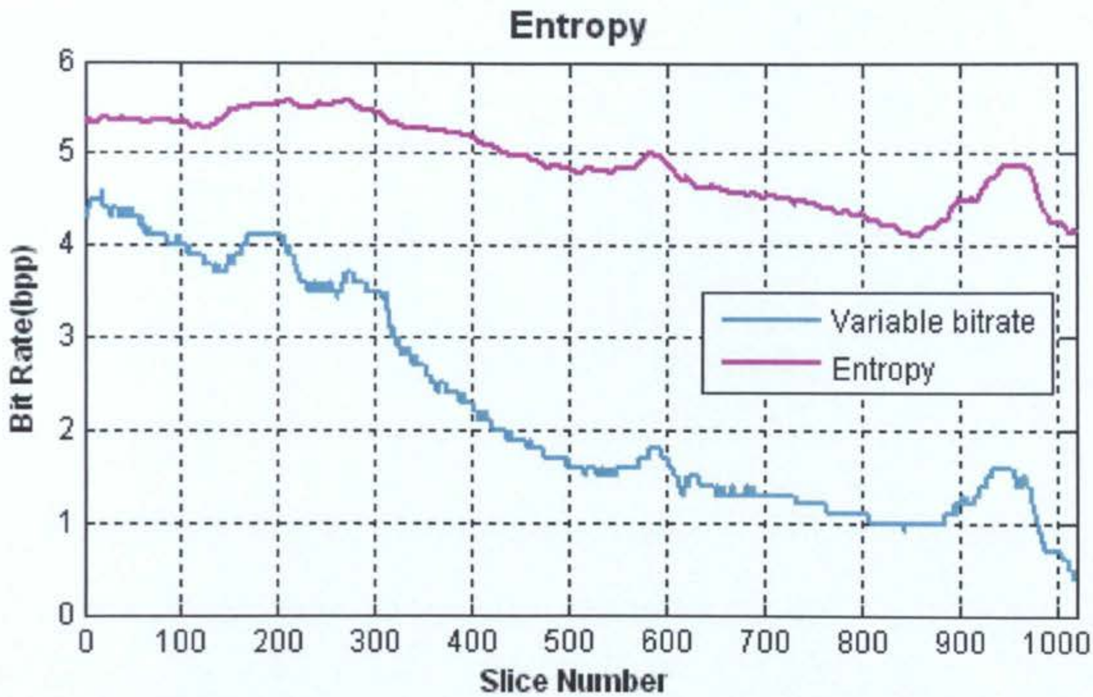
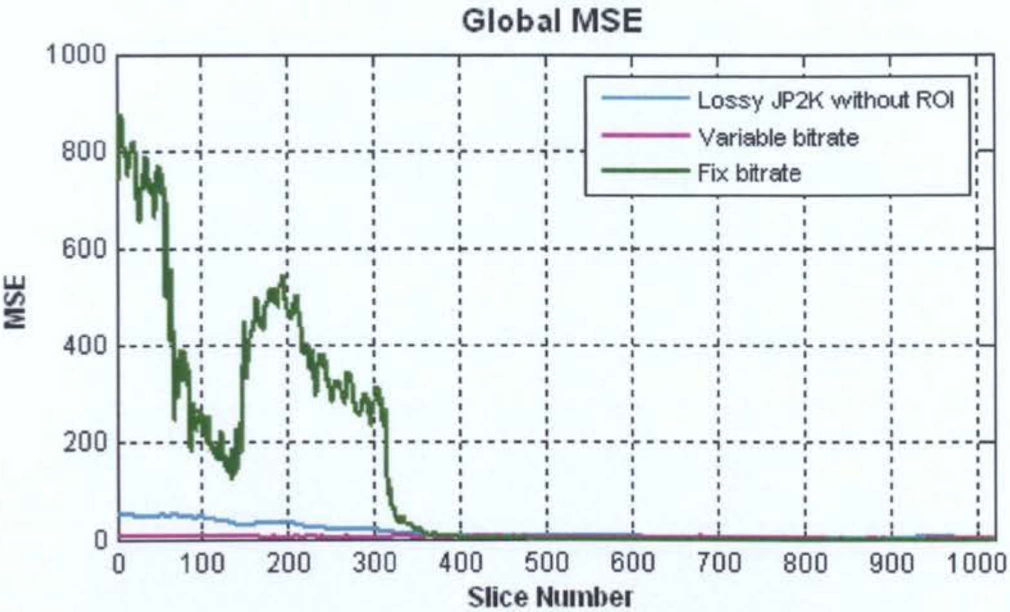


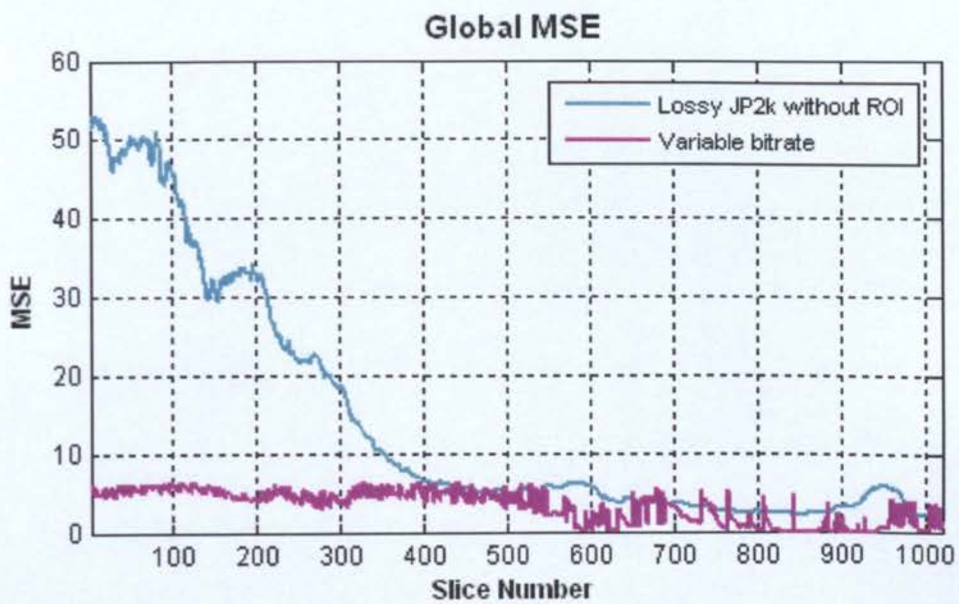
Figure 6-3 Results in terms of entropy of DICOM slices and compression bitrates for the proposed compression method with variable bitrates in the typical dataset.

The comparison between entropy and compression bitrates over the whole images of the same dataset (Table 6.3) is graphically shown in Figure 6.3. It shows the average information content of DICOM slices, Entropy, and the obtained bitrates, which resulted from applying the proposed method with variable bitrates based on the proposed adaptive algorithm in the dataset. It can be observed that the average number of bits needed for each DICOM slice, Entropy diagram, falls from slice 1 towards slice 1000. The reason for this is human anatomy (As seen in Figure 4.2, the details of cross sections of the human body reduces from abdominal aorta to legs). Furthermore, the bitrate is considerably decreased as a result of applying the proposed method with variable bitrate over all images of the dataset (Variable bitrate diagram).

The graphs in Figure 6.4 represent the advantage of the proposed method with variable bitrate in comparison with the proposed method with fixed bitrate and lossy JP2K without ROI, in terms of global MSE. The horizontal and vertical axis are slice number and MSE respectively. As it can be seen from Figure 6.4a, the values of the mean square error for the proposed method with fixed bitrate are more noticeable than the two other methods for the slices less than 400. In addition to this result, according to Figure 6.4b, the ranges of MSE for the lossy JP2K without ROI stand out significantly from the proposed method with variable rate. It can be observed that there are no remarkable errors, MSE, for slice numbers after 400 in the three methods.



(a) For the three methods



(b) For the two methods in large scale

Figure 6-4 Results in terms of MSE

Because the average number of bits required to compress the slices in the dataset has been reduced from slice 1 towards slice 1000 (Figure 6.3) and also the compression bitrate for the two scenarios (Fix bitrate and Lossy JP2K without ROI) is estimated based on the fixed bitrate (mean of bitrates in the Variable bitrate method), compression of the slices with the fixed bitrate does not create a remarkable error for slice numbers after 400.

In Figure 6.5, the global PSNR for the three methods is reported. Regarding the minimum acceptable PSNR (40 dB), the PSNR values in our proposed method (Variable bitrate) meet the constraint, whereas the two other methods do not, in particular for slice numbers less than 400.

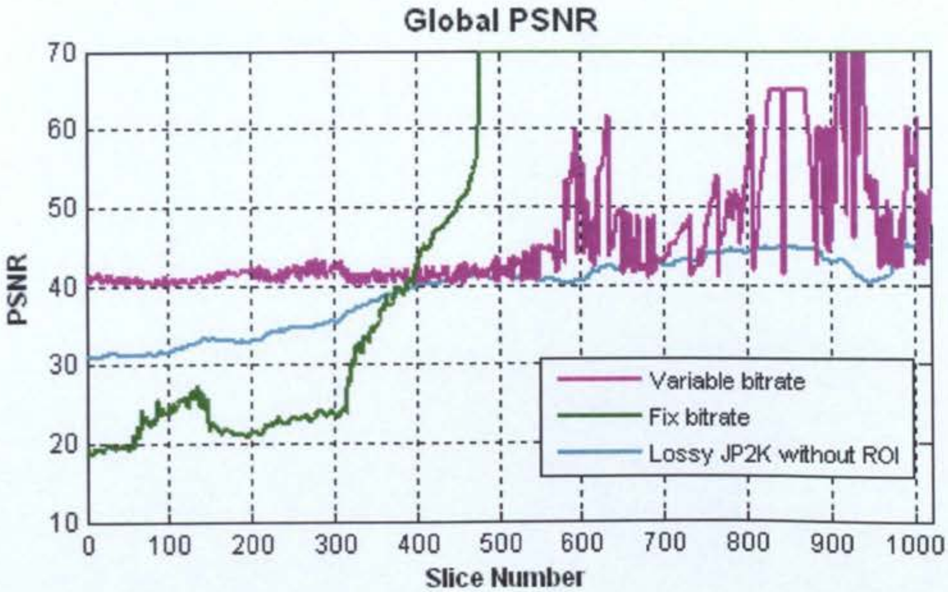


Figure 6-5 Results in terms of PSNR for the three methods in the typical dataset

The average file size and MSE for the typical dataset and 10 datasets are reported in Table 6.4 and Table 6.5 respectively. It can be seen that MSE (mean square error) in the proposed method with variable bitrate is lower than MSE for the two other methods and also the error for the second approach is remarkable in both tables.

Table 6-4 Average file size and global MSE considered in the typical dataset

Method	Average file size (KB)	Average MSE
Proposed method with variable bitrate	71.15	3.29
Proposed method with fixed bitrate	61.59	127.16
Lossy JP2K without ROI	73.46	13.86

Table 6-5 Average file size and global MSE for 10 datasets

Method	Average file size (KB)	Average MSE
Proposed method with variable bitrate	56.44	3.66
Proposed method with fixed bitrate	50.03	139.46
Lossy JP2K without ROI	58.31	15.13

Results in terms of PSNR are reported since this is a well known quality metric in the image processing community. However, the goal of this method is to improve the perceptual quality, rather than a metric such as PSNR. Diagnostic Mean Opinion Score (MOS) results are more significant in this scenario. The Diagnostic MOS scale is shown in Table 6.6. An image is acceptable based on Diagnostic MOS ranking if it gets a satisfactory mark, i.e., equal to or greater than 4.

Table 6-6 Diagnostic Mean Opinion Score (MOS)

Rank	Comment
5	Excellent (No difference)
4	Good (No significant difference)
3	Fair (Significant difference/less affected diagnosis)
2	Poor (Significant difference/affected diagnosis)
1	Bad (Not acceptable)

The quality of the compressed images has been assessed by two qualified radiologists based on Diagnostic MOS test and the results are represented in Figure 6.6, Table 6.7, and Fig 6.7.

As shown by the Diagnostic MOS results for the dataset D1 in Figure 6.6 (Table A.1); for the fixed bitrate scenario the reconstructed PAD images for slice numbers lower than 400 are assessed as *Bad* (not acceptable), whereas for slice numbers above 400 as *Good* (no significant difference). In lossy JP2K method, the result is specified as *Good.* It can be seen that the Diagnostic MOS results for our proposed method, variable rate, is characterized as *Excellent* (no difference) in the whole dataset.

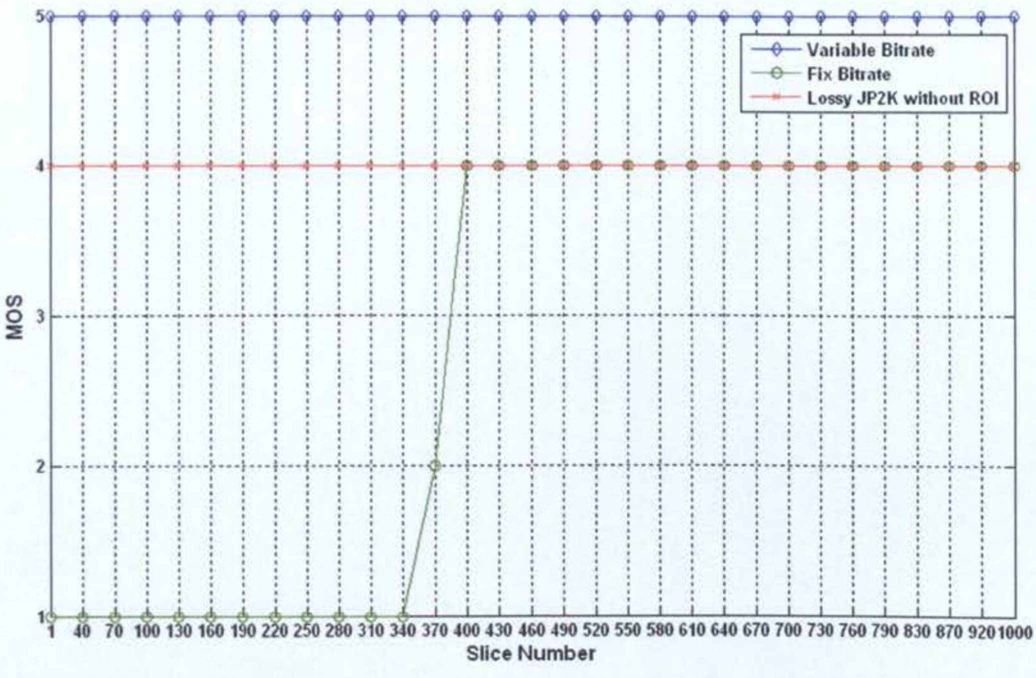


Figure 6-6 Subjective measurement for the three methods based on MOS test

The average subjective measurements for the 10 datasets, D1-D10, are shown in Figure 6.7 (Table A.2). The results show that the average Diagnostic MOS for the proposed method is Excellent in all 10 datasets without any diagnostic effect.

The performed tests successfully confirmed that, from the medical perspective, the images compressed with the proposed methodology are almost indistinguishable from the original ones according to Diagnostic MOS tests.

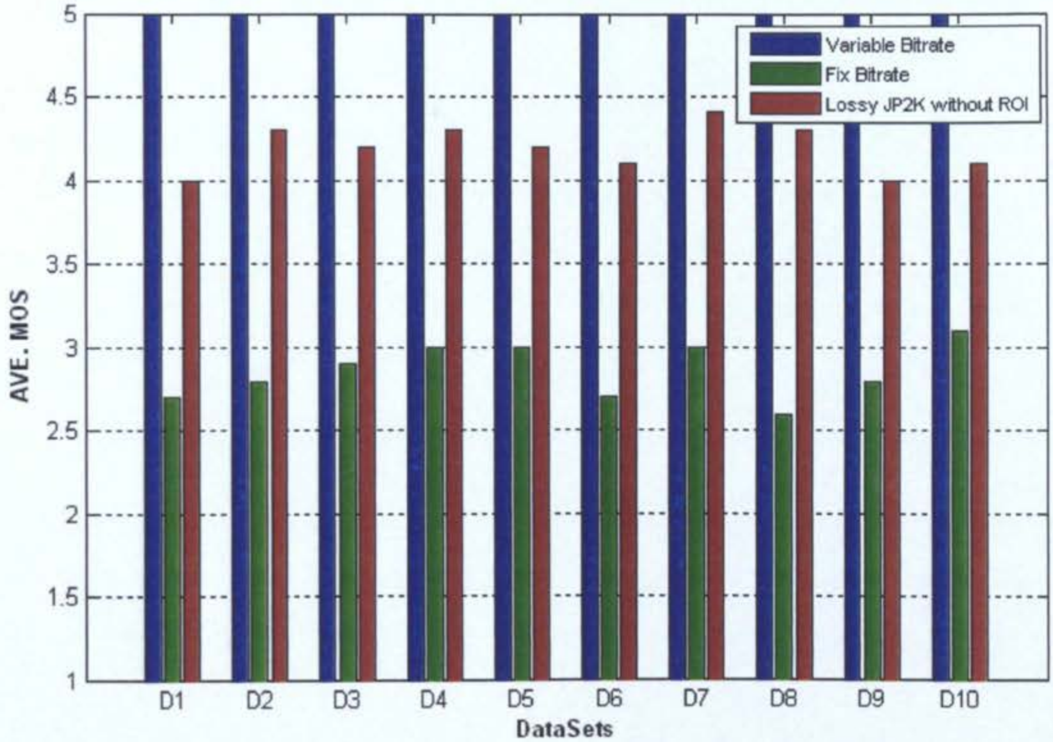


Figure 6-7 Average diagnostic MOS for the three methods for the 10 datasets

6.2.2. Group B (Lossless Scenarios)

For the second set of results (Group B) a comparison between Lossless JP2K, Lossless JP3D, Lossless H.264, and the proposed approach is reported in Table 6.8 and Figure 6.8 in terms of the average size of the DICOM images and the compression ratio for the 10 datasets, D1-D10. A summary of the result in Table 6.7 is shown in Table 6.8 (average over the results over the ten datasets). The average size of the images is reduced from 145.95, 142.78 KB, and 170.3 to 56.44 KB with respect to the image compressed with lossless JP2K standard, lossless JP3D, and lossless H.264 respectively. It can be seen in Figure 6.8 that the average CRs in the proposed method is much higher than the other strategies (Lossless JP2K, Lossless JP3D, and H.264) in all of the 10 datasets.

Table 6-7 Average DICOM size and CR. for the 10 datasets with four compression strategies

Dataset Number of slices	Lossless JP2K [Ave. size] [Ave. CR] KB	Lossless JP3D [Ave. size][Ave. CR] KB	Lossless H.264 [Ave. size] [Ave. CR] KB	Proposed Method [Ave. size] [Ave. CR] KB
D1(1038)	[157.5]-[3.2:1]	[154.18]-[3.3:1]	[188.1]-[2.7:1]	[71.15]-[7.2:1]
D2(0985)	[145.4]-[3.5:1]	[142.45]-[3.6:1]	[169.2]-[3.0:1]	[56.30]-[9.0:1]
D3(1046)	[150.8]-[3.4:1]	[147.08]-[3.4:1]	[176.3]-[2.9:1]	[60.94]-[8.4:1]
D4(1000)	[139.2]-[3.6:1]	[136.13]-[3.7:1]	[160.0]-[3.2:1]	[43.1]-[11.8:1]
D5(1029)	[153.5]-[3.3:1]	[150.15]-[3.4:1]	[181.5]-[2.8:1]	[67.90]-[7.5:1]
D6(1034)	[145.8]-[3.5:1]	[142.51]-[3.6:1]	[170.1]-[3.0:1]	[57.88]-[8.8:1]
D7(0990)	[139.6]-[3.6:1]	[136.55]-[3.7:1]	[158.8]-[3.2:1]	[45.2]-[11.3:1]
D8(1009)	[136.0]-[3.7:1]	[133.05]-[3.8:1]	[161.0]-[3.2:1]	[51.51]-[9.9:1]
D9(1000)	[147.3]-[3.4:1]	[144.25]-[3.5:1]	[171.1]-[3.0:1]	[55.35]-[9.2:1]
D10(0986)	[144.4]-[3.5:1]	[141.51]-[3.6:1]	[166.8]-[3.1:1]	[55.07]-[9.3:1]

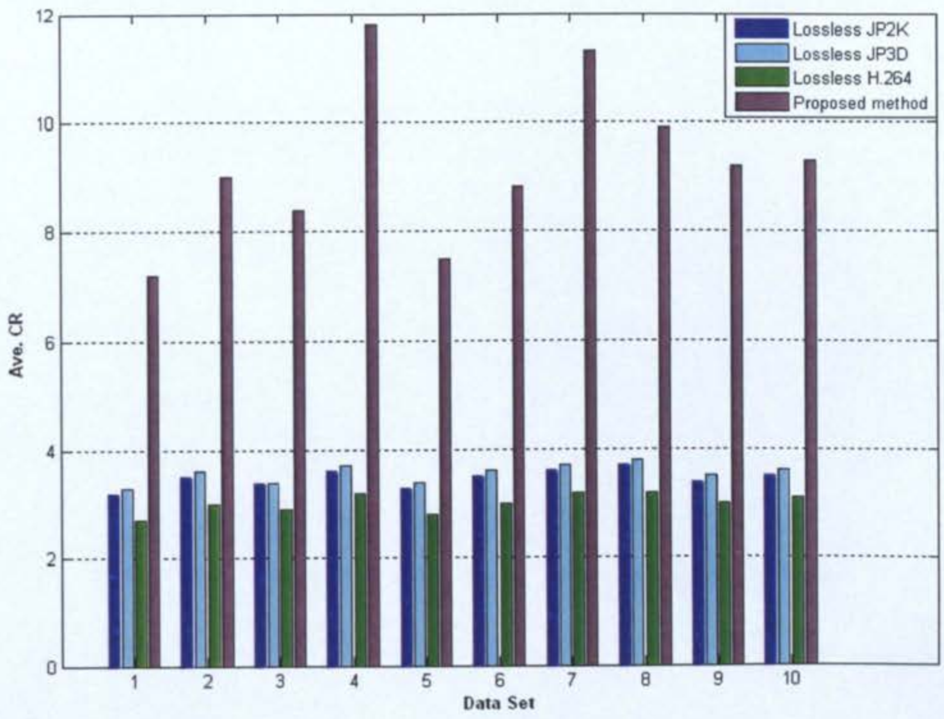


Figure 6-8 Average CR. for the four compression strategies for the 10 datasets

Our results from CTA images clearly highlight that Lossless JP3D, although computationally more intensive, is slightly better than Lossless JP2K 2D in the considered scenario and H.264 cannot be appropriate for compression of CTA images because of many changes in consecutive slices. In our proposed method, the average compression ratio is considerably increased as shown in Table 6.8.

Table 6-8 Performance evaluation of the four methods (average over the 10 datasets)

Method	Average file size (KB)	Average compression ratio
Original file	512	1:1
Lossless JP2K	145.95	3.5:1
Lossless JP3D	142.78	3.6:1
Lossless H.264	170.3	3:1
Proposed method	56.44	9:1

These results show that, for the reconstructed images, the target of respecting the lossless constraint in the PROI and keeping an acceptable appearance in the SROIs and background can be reached by this method at a high compression ratio. The average size of images can be reduced to 61, 60, 66, and 89 percent with respect to lossless JP2K, Lossless JP3D, Lossless H.264, and original DICOM image respectively, with no impairment for the diagnosis.

There are three well-defined advantages over other region of interest based compression methods [17]-[24]. Firstly, the proposed method is independent of the standard compression engine. This means that the method could be used in any ROI based compression method, while some definite modifications are required to implement the previous methods. Secondly, in our approach, ROIs are segmented in a fully automated fashion. Finally, the PROIs are compressed losslessly and SROIs are compressed near losslessly (diagnostically losslessly according to our Diagnostic MOS tests).

6.3. Summary

A robust performance evaluation schema (test-bed) was designed to analyze the results based on the objective and subjective quality metrics. Using a GUI designed to represent the compressed images, two groups (Lossy and Lossless) including seven scenarios were evaluated over 10 datasets, concerning the image quality and the potential impact on diagnosis accuracy. The performance evaluation results of compressing more than 10,000 medical images showed a remarkable achievement in compression rate compared to the existing techniques (i.e., JP2K, JP3D, and H.264) in both groups.

7. CONCLUSIONS AND FUTURE WORK

An innovative context based[1] and region of interest (ROI) based approach is proposed in this research. The proposed approach was utilized to compress 3D digital medical images that can keep the full skeleton of medical images while the primary regions of interest (PROI) critical for medical diagnosis, are maintained intact during the compression process. This means that no information is lost from the PROI during the compression process. The secondary regions of interest (SROI) are defined as part of the medical images, which their loss can be tolerated during the diagnosis process. Therefore, for the secondary region of interest, diagnostically irrelevant information that has no impact on diagnosis decision is lost.

Without loss of generality, the proposed approach was designed to be applied to the vascular images from CT angiography for peripheral arteries.

[1] In the proposed method, the ROI are areas corresponding to a higher degree of clinical interest which will be identified based on contextual information given by a radiologist or anatomy.

The interactions and manual interventions with the user were minimized using the following two proposed algorithms: (1) In the automatic rate control algorithm, the bitrate for each slice was automatically calculated with the rate considered for the previous slice based on the PSNR of SROI and bitrate constant step and (2) The automatic ROIs segmentation algorithm which consisted of background & body segmentation (SROI) and peripheral artery segmentation (PROI). SROI was segmented using an adaptive thresholding method, followed by morphological operations and connected component analysis. The process of PROI segmentation was an initial segmentation of arteries from other tissues followed by a 3D region growing algorithm, in combination with morphological operators to increase the precision of segmentation. The segmentation approach includes an automatic seed point selection technique.

A robust performance evaluation schema (test-bed) was designed to analyze the results based on the objective and subjective quality metrics. Using the GUI designed to represent the compressed images, two groups (Lossy and Lossless) including seven scenarios were evaluated over 10 datasets, concerning the image quality and the potential impact on diagnosis accuracy. In Lossy group, the experimental results showed that the average Diagnostic Mean Opinion Score (MOS) for the proposed method was characterized as Excellent (5 of 5) in all 10 datasets (the Diagnostic MOS test was designed based on visual judgement of two radiologists). Furthermore, the average mean square error (MSE) in the proposed method was very smaller than the other methods in this group. In lossless group, the average compression ratio in the proposed method was considerably increased. The average size of images was reduced to 61%, 60%, 66%, and 89% with respect to lossless JP2K, Lossless JP3D, Lossless H.264 standard, and original image respectively, without considerable impacts on the diagnostic accuracy using visual judgement of two radiologists.

The proposed compression strategy can hence reduce the cost of storage, transmission time, and therefore faster diagnostic analysis without losing any diagnostically relevant information. This method can be extended to be used on other medical images.

In spite of ongoing development towards increasing the data transmission speed and storage capacities, the proposed approach can be utilized in web based application design and developed for use in mobile devices (Smartphone and/or handheld PC) in 3G mobile networks.

Future Work

The proposed approach and techniques can be extended and complemented in several different directions.

- The method applied on 10 datasets (each dataset contains more than 1000 slices/images) and the experimental results showed significant improvement in compression ratio; however, in order to have a robust method which can be used in practices; more datasets should be fed to our system to further validate our compression gain.

- The method was applied on the peripheral artery as a case study; it can also be applied on vascular system and then the whole body since the characteristics of peripheral artery is very similar to other vascular systems. With a little adaptation of the segmentation technique, we are able to apply the method on different vascular imaging (i.e., pulmonary artery, coronary artery, and abdominal aortic). However, to apply this method on the other anatomy (i.e., brain), further work should be done to define and segment the PROI and SROI but the compression part of the method will not be changed in this scenario.

- The method was applied on Computed Tomography Angiography (CTA); it can be applied on different types of image modalities, i.e., Magnetic Resonance Angiogram (MRA) to be used as a standard tool for vessel compression. That means the new ROI segmentation is required to be designed for each Modality (for instance, the proposed segmentation is working based on the range of Hounsfield Unit (HU) which is fixed in CT images but this range varies from one image to another in MRA).

- The conservative region *of interest method was designed to make* sure that it includes ROIs and the obtained results demonstrate that the compression ratio is very good with respect to standard medical image codecs and subjective expectations; however, a further improvement in the compression ratio is possible with optimisation in the ROI segmentation mechanism.

- The method was compared with three standard medical image codecs (i.e., JP2K, JP3D, and H.264). The obtained results have been remarkable; however, the method can be compared with other new video standard codecs i.e., High Efficiency Video Coding (HEVC) to be used in practical applications.

- The compression part of the proposed method can be extended for transmission over error prone wireless networks in Telemedicine applications (Telemedicine applications need that medical imaging data to be efficiently accessed and transmitted over error-prone wireless networks of various bandwidth capacities). The PROI and SROI may be compressed separately and then transmitted via different transport layer protocols (i.e., PROI can be sent via connection oriented channel such as TCP and SROI can be sent via connectionless channel such as UDP). This can guarantee the reconstruction of images even in the presence of transmission error.

REFERENCES

[1] X. Cao and H.K. Huang, "Current Status and Future Advances of Digital Radiography and PACS," *IEEE Engineering in Medicine and Biology Magazine*, vol. 9, pp. 80-88, 2000.

[2] E. Samei, J.A. Seibert, K Andriole, A. Badano, J. Crawford, B. Reiner, M.J. Flynn, and P. Chang, "AAPM/RSNA tutorial on equipment selection: PACS equipment overview *general guidelines for purchasing and acceptance testing of* PACS equipment," *Radiographics*, vol. 24, pp. 313–334, 2004.

[3] R. Smith-Bindman, D.L. Miglioretti, E. Johnson, C. Lee, H.S. Feigelson, M. Flynn, R.T. Greenlee, R.L. Kruger, M.C. Hornbrook, D. Roblin, L.I. Solberg, N. Vanneman, S. Weinmann, and A.E. Williams, "Use of diagnostic imaging studies and associated radiation exposure for patients enrolled in large integrated health care systems, 1996-2010," *Journal of the American Medical Association*, vol. 307, no. 22, pp. 2400-2409, 2012.

[4] M.G. Martini, "Wireless broadband multimedia health services: Current status and emerging concepts," in *IEEE Personal Indoor and Mobile Radio Communications*, 2008, pp. 1-6.

[5] M.G. Martini, R.S.H. Istepanian, M. Mazzotti, and N.Y. Philip, "Robust Multilayer Control for Enhanced Wireless Telemedical Video Streaming," *IEEE Transactions on Mobile Computing*, vol. 9, no. 1, pp. 5–16, Jan 2010.

[6] K. Krishnan, M.W. Marcellin, A. Bilgin, and M.S. Nadar, "Efficient transmission of compressed data for remote volume visualization," *IEEE Transactions on Medical Imaging*, vol. 25, no. 9, pp. 1189 – 1199, Sept 2006.

[7] M.G. Martini and M. Mazzotti, "Quality driven wireless video transmission for medical applications.," in *Engineering in Medicine and Biology Society*, 2006, pp. 3254 - 3257.

[8] M.A. Ansari and R.S. Anand, "Context based medical image compression for ultrasound images with contextual set partitioning in hierarchical trees algorithm," *Advances in Engineering Software*, vol. 40, no. 7, pp. 487-496, July 2009.

[9] M.J. Zukoski, T. Boult, and T. Lyriboz, "A novel approach to medical image compression," *IJBRA*, vol. 2, no. 1, pp. 89-103, 2006.

[10] G.K. Anastassopoulos, and A.N. Skodras, "JPEG 2000 ROI Coding in Medical Imaging Applications," , *Visualization, Imaging, and Image Processing*, Sep 2002.

[11] J. Bartrina-Rapesta, J. Serra-Sagrista, and F. Auli-Llinas, "JPEG2000 ROI Coding Through Component Priority for Digital Mammography," *ELSEVIER Computer Vision and Image Understanding*, vol. 115, no. 1, pp. 59-68, January 2011.

[12] H.Y. Chan, H. Sari-Sarraf, B.I. Grinstead, and S.Sl Gleason, "Content-based compression of mammograms with fractal-based segmentation and a modified JPEG2000," *SPIE Optical Engineering*, vol. 43, no. 12, pp. 2986-2993, December 2004.

[13] K.J. Kim, T. Richter, H.S. Kang, S.Y. Kim, and Y.H. Kim, "JPEG2000 3D compression vs 2D compression: An assessment of artifact amount and computing time in compressing thin-section abdomen CT images," *Medical Physics*, vol. 36, no. 3, 2009.

[14] H. Ringl, R. Schernthaner, E. Sala, K. El-Rabadi, M. Weber, W. Schima, C.J.

Herold, and A.K. Dixon, "Lossy 3D JPEG2000 Compression of Abdominal CT Images in Patients with Acute Abdominal Complaints: Effect of Compression Ratio on Diagnostic Confidence and Accuracy," *Radiology*, vol. 284, pp. 476-484, Augest 2008.

[15] K.J. Kim, K.H. Lee, B. Kim, T. Richter, Il.D. Yun, S.Uk. Lee, K.T. Bae, and H. Shim, "Improved Compressibility in JPEG2000 2D and 3D Reversible Compressions of Thin-section Chest CT Images by Increasing the Data Redundancy outside the body Region," *IEEE COMSOC MMTC E-Letter*, vol. 6, no. 7, July 2011.

[16] D. Taubman and M. Marcellin, "JPEG2000: image compression fundamentals, standards and practice," , Norwell, MA: Kluwer, 2002.

[17] H. Yang, M. Long, and H.M. Tai, "Region-of-interest image coding based on EBCOT," *IEEE Proceedings Vision, Image and Signal Processing*, vol. 152, no. 5, pp. 590-596, Oct. 2005.

[18] S. Agili, V. Balasubramanian , and A. Morales, "Semi-automatic region of interest identification algorithm using wavelets," *Optical Engineering*, vol. 46, no. 3, March 2007.

[19] L. Zhang and X. Han, "New Region of Interest Medical Image Coding for Jpeg2000: Compensation-Base Partial Bitplane Alternating Shift," *International Conference on Machine Learning and Cybernetice*, pp. 3924-3928, Aug 2006.

[20] P. Akhtar, M.I. Bhatti, T.J. Ali, and M.A. Muqeet, "Significance of ROI Coding using MAXSHIFT Scaling applied on MRI Images in Teleradiology-Telemedicine," *Journal of Biomedical Science and Engineering, ISSN: 1937-6871*, vol. 1, no. 2, pp. 110-115, 2008.

[21] P. Akhtar, M.I. Bhatti, T.J. Ali, and M.A. Muqeet, "Significance of Region of Interest applied on MRI and CT Images in Teleradiology-Telemedicine," *Springer-Link LNCS 4987, ISSN: 0302-9743*, pp. 152-160, 2008.

[22] L. Liu and G. Fan, "A new JPEG2000 region-of-interest image coding method: partial significant bitplanes shift," *IEEE Signal Processing Letters*, vol. 10, no. 2, pp. 35-38, Feb 2003.

[23] P.G. Tahoces, J.R. Varela, M.J. Lado, and M. Souto, "Image compression: Maxshift ROI encoding options in JPEG2000," *Computer Vision and Image Understanding*, vol. 109, no. 2, pp. 139-145, February 2008.

[24] Z. Xiong, X. Wu, S. Cheng, and J. Hua, "Lossy-to-lossless compression of medical volumetric images using three-dimensional integer wavelet transforms," *IEEE Transaction on Medical Imaging*, vol. 22, no. 3, pp. 459-470, March 2003.

[25] E. Seeram, "Irreversible compression in digital radiology. A literature review," *Radiography*, vol. 12, no. 1, pp. 45-59, February 2006.

[26] K.H. Lee, Y.H. Kim, B.H. Kim, K.J. Kim, T.J. Kim, H.J. Kim, and S. Hahn, "Irreversible JPEG 2000 compression of abdominal CT for primary interpretation: assessment of visually lossless threshold," *European Radiology*, vol. 17, no. 6, 2007.

[27] US Food and Drug Administration. (2000, July) FDA. [Online]. http://www.fda.gov/MedicalDevices/DeviceRegulationandGuidance/GuidanceDocuments/ucm073720.htm

[28] US Food and Drug Administration. (2000, July 27) Guidance for the Submission Of Premarket Notifications for Medical Image Management Devices. http://www.fda.gov/MedicalDevices/DeviceRegulationandGuidance/GuidanceDocuments/ucm073720.htm.

[29] C.A. Christopoulos, A.N. Skodras, and T. Ebrahimi, "The JPEG 2000 still image coding system: An overview," *IEEE Transaction on Consumer Electronics*, vol. 46, no. 4, pp. 1103-1127, Nov 2000.

[30] M. Rabbani and R. Joshi, "An overview of the JPEG 2000 still image compression standard," *Signal Processing Image Communication*, vol. 17, no. 1, pp. 3-48,

2002.

[31] C. Herley and M. Vetterli, "Wavelets and filter banks: Theory and design," *IEEE Transactions on Acoustics and Speech Signal Processing*, vol. 40, no. 9, pp. 2207-2232, September 1992.

[32] WG10, JPEG: ISO/IEC JTC1/SC29, "Information Technology - Digital ComCompressionand Coding of Continuous- tone Still Images - Requirement and Guide lines," Final Draft International Standard ISO/IEC 10918-1, 1993.

[33] "Information Technology - JPEG 2000 Image Coding System:Part 1 - Core Coding System," , ISO/IEC JTC1/SC29/WG1, IS 15444-1, 2000.

[34] MPEG-2: ISO/IEC, ISO/IEC 13818-2:2000, Information Technology - Generic coding of moving pictures and associated audio information: Video, 2000.

[35] G. Sullivan, P. Topiwala, and A. Luthra, "The H.264/AVC advanced video coding standard: overview and introduction to the fidelity range extensions," *Proc. SPIE Int. Soc. Opt. Eng*, vol. 5558, pp. 454-474, August 2004.

[36] J.M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transaction on Signal Processing*, vol. 41, no. 12, pp. 3445-3462, 1993.

[37] A. Said and W.A. Peralman, "A new fast and efficient image coded based on set partitioning in hierarchical trees," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243-250, 1996.

[38] D. Stranneby and W. walker, *Digital Signal Processing and Applications*, 2nd ed., 2004.

[39] D. Salomon, *Data Compression: The Complete Refrence*, Forth ed., 2007.

[40] M. Brani, *Document and Image Compression.*, 2007.

[41] A. Bilgin, P.J. Sementilli, F. Sheng, and M.W. Marcellin, "Scalable image coding using reversible integer wavelet transforms," *IEEE Transactions on Image*

Processing, vol. 9, no. 11, pp. 1972 - 1977, December 2000.

[42] A.W. Liew and N. Law, "Image Compression Concepts Overview," in *Encyclopedia of Information Science and Technology.*, 2009, pp. 1805-1811.

[43] M.J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," *IEEE Transaction on Image Processing*, vol. 9, no. 8, pp. 1309-1324, 2000.

[44] G. Seroussi and M.J. Weinberger, "On adaptive strategies for an extended family of Golomb-type codes," in *Data Compression Conference*, 1997, pp. 131-140.

[45] D. Santa-Cruz, R. Grosbois, and T. Ebrahimi, "JPEG2000 performance evaluation and assessment," *Signal Processing: Image Communication*, vol. 17, no. 1, pp. 113–130, 2002.

[46] L. Shizhong and A.C Bovik, "Efficient DCT-domain blind measurement and reduction of blocking artifacts," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 12, no. 12, pp. 1139 - 1149, 2002.

[47] J.D. Villasensor, B. Belzer, and J. Liao, "Wavelet filter evaluation for image compression," *IEEE Transaction on Image Processing*, vol. 4, no. 8, pp. 1053-1060, 1995.

[48] "Information Technology—JPEG 2000 Image Coding System—Part 10: Extensions," , ISO/IEC 15444-10, 2008.

[49] P. Schelkens, A. Skodras, and T. Ebrahimi, *The JPEG 2000 Suite.*, 2009.

[50] "Information Technology - JPEG 2000 Image Coding System:Part 2 -Extensions," , ISO/IEC JTC1/SC29/WGI, IS 15444-2, 2002.

[51] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, 2003.

[52] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammar, and T. Wedi, "Video Coding with H.264/AVC: Tools, Performance and Complexity," *IEEE Circuit and system Magazine*, vol. 4, 2004.

[53] G.V.D. Auwera, P.T. David, and M. Reisslein, "Traffic and Quality Characterization of Single-Layer Video Streams Encoded with the H.264/MPEG-4 Advanced Video Coding Standard and Scalable Video Coding Extension," *IEEE Transactions on Broadcasting*, vol. 54, no. 3, 2008.

[54] B.J. Erickson, "Irreversible Compression of Medical Images," *Journal of Digital Imaging*, vol. 15, no. 1, pp. 5-14, 2002.

[55] P. Felkel, R. Wegenkittl, and A. Kanitsar, "Vessel tracking in peripheral CTA datasets - an overview," in *IEEE Spring Conference on Computer Graphics*, 2001, pp. 232-239.

[56] G. Gerig, O. Kubler, R.Kinkinis, and F.A Jolez, "Nonlinear anisotropic filtering of MRI data," *IEEE Transactions on Medical Imaging*, vol. 11, no. 2, pp. 221-232, 1995.

[57] A.F. Frangi, W.J. Niessen, K.L. Vincken, and M.A. Viergever, "Multicale vessel enhancement filtering," in *Medical Image Computing and Computer-Assisted Intervention*, 1998, pp. 130-137.

[58] T.M. Lehmann, C. Gonner, and K.Spitzer, "Surrvey: interpolation methods in medical image processing," *IEEE Transactions on Medical Imaging*, vol. 18, no. 11, pp. 1049-1075, 1999.

[59] D.L. Wilson and J.A. Noble, "An adaptive segmentation algorithm for time-of-flight MRA data.," *IEEE Transactions on Medical Imaging*, vol. 18, no. 10, pp. 938-945, 1999.

[60] W.E. Higgins, W.J.T. Spyra, and E.L. Ritman, "Automatic extraction of the arterial tree from 3D angiograms.," in *IEEE Conference of Engineering in Medicine and Biology*, vol. 2, 1989, pp. 563-564.

[61] H. Schmitt, M. Grass, V. Rasche, O. Schramm, S. Haehnel, and K. Sartor, "An X-ray-based method for the determination of the contrast agent propagation in 3-D vessel structures," *IEEE Transaction on Medical Imaging*, vol. 21, no. 3, pp. 251-262, 2002.

[62] Y. Masutani, T. Schiemann, and K.H Hohne, "Vascular shape segmentaion and structure extraction using a shape-based region-growing model.," in *International Conference on Medical Computing and Computer Assisted Intervention*, 1998, pp. 1242-1249.

[63] V. Juhan, B. Nazarian, K. Malkani, R. Bulot, J.M. Bartoli, and J. Sequeria, "Geometrical modelling of abdominal aortic aneurysms," in *Computer Vision, Virtual Reality and Robotics in Medicine*, 1997, pp. 243-252.

[64] C. Pellot, A. Herment, and M. Sigelle, "A 3D reconstruction of vascular structures from two X-ray angiograms using an adapted simulated annealing algorithm," *IEEE Transaction on Medical Imaging*, vol. 13, no. 1, pp. 48-60, 1994.

[65] R.R. Petrocelli, K.M. Manbeck, and J.L Elion, "Three dimentional structure recognition in digital angiograms using Gauss-Markov methods.," in *IEEE Computers in Cardiology*, 1993, pp. 101-104.

[66] C. Kirbas and F.K.H Quek, "A review of vessel extraction techniques and algorithms.," *Association for computing Machinery and Computing Surveys*, vol. 36, no. 2, pp. 81-121, 2004.

[67] A. Kanistar, R. Wegenkittl, P. Felkel, D.Fleichmann, D.Sander, and E. Groller, "Computed Tomography Angiography: A Case Study of Peripheral Vessel Investigation," in *IEEE Visualization*, 2001, pp. 477-480.

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

[69] H.R. Sheikh and A.C. Bovik, "Image information and visual quality," *IEEE*

Transaction on Image Processing, vol. 15, no. 2, pp. 430–444, Feb 2006.

[70] A.D. Angelo, L. Zhaoping, and M. Barni, "A Full-Reference Quality Metric for Geometrically Distorted Images," *IEEE Transaction on Image Processing*, vol. 19, no. 4, pp. 867-881, 2010.

[71] S. Grgic, M. Grgic, and M. Mrak, "Reliability of Objective Picture Quality Measurement," *Journal of Electrical Engineering*, vol. 55, no. 12, pp. 3-10, 2004.

[72] R. Sakuldee and S. Udomhunsakul, "Objective Performance of Compressed Image Quality Assessments," , vol. 26, 2007, pp. 434-443.

[73] C. Chrysafis, "Wavelet image compression rate distortion optimizations and complexity reductions," *PhD dissertation, University of Southern California*, 2000.

[74] C.S Varnan, A. Jagan, J. Kaur, D. Jyoti, and D.S. Rao, "Image Quality Assessment Techniques pn Spatial Domain," *International Journal of Computer Science and Telecommunications*, vol. 2, no. 3, 2011.

[75] Z. Wang and A.C. Bovik, "Mean Squared Error: Love It or Leave It?," *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 98-117, JANUARY 2009.

[76] R. Dosselmann and X.D. Yang, "A Formal Assessment of the Structural Similarity Index," University of Regina, 2008.

[77] R. Istepanian, N. Philip, M. Martini, N. Amso, and P. Shorvon, "Subjective and objective quality assessment in wireless teleultrasonography imaging," in *IEEE Engineering in Medicine and Biology Society*, 2008, pp. 5346 -5349.

[78] Yu. Hongtao, Lin. Zhiping, and P. Feng, "Applications and improvement of H.264 in medical video compression," *IEEE Transactions on Circuits and Systems* , vol. 52, no. 12, pp. 2707 - 2716, 2005.

[79] T.H. Oh, H.S. Lim, and S.Y. Pang, "Medical Image Processing: From Lossless to

Lossy Compression," in *IEEE TENCON*, 2005, pp. 1 - 6.

[80] A.S. Al-Fahoum and A.M. Reza, "Perceptually Tuned JPEG Coder for Echocardiac Image Compression," *IEEE Transaction on Information Technology in Biomedicine*, vol. 8, no. 3, pp. 313-320, 2004.

[81] I. Maglogiannis and G. Kormentzas, "Wavelet-Based Compression With ROI Coding," *IEEE Transactions on Information Technology In Biomedicine*, vol. 13, no. 4, pp. 458-466, 2009.

[82] A.M. Eskicioglu, "Quality Measurement for Monochrome Compressed Images in the Past 25 Years," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Istanbul, Turkey, 2000, pp. 1907-1910.

[83] R. Grosbois, D.S. Cruz, and T. Ebrahimi, "New approach to JPEG 2000 compliant Region Of Interest coding," in *SPIE's 46th annual meeting, Applications of Digital Image Processing XXIV*, vol. 4472, San Diego, CA, USA, Jul. 29- Aug. 3, 2001, pp. 267–275.

[84] A.N. Skodras, C.A. Christopoulos, and T. Ebrahimi, "*JPEG2000: the upcoming* still image compression standard," *Pattern Recognition Letters*, vol. 22, pp. 1337–1345, 2001.

[85] C.A. Christospoulos, J. Askelof, and M. Larsson, "Efficient method for encoding regions of interest in the upcoming Jpeg2000 still image coding standard," *IEEE Signal Processing Letters*, vol. 7, no. 9, pp. 247-249, 2000.

[86] P.J. Sementilli, A. Bilgin, J.H. Kasner, and M.W. Marcellin, "Wavelet TCQ: submission to JPEG 2000," in *SPIE, Applications of Digital Processing*, 1998, pp. 2-12.

[87] B.E. Usevitch, "A tutorial on modern lossy wavelet image compression:Foundations of JPEG 2000," *IEEE Signal Processing Magazine*, vol. 18, pp. 22-35, 2001.

[88] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 Still Image Compression Standard," *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 36 - 58, 2001.

[89] R. Choras , "JPEG2000 IMAGE CODING STANDARD-A REVIEW AND APPLICATIONS," *Advances in Information Processing and Protection*, 2007.

[90] P. Schelkens, A. Munteanu, A. Tzannes, and C. Brislawn, "JPEG2000. Part 10. Volumetric data encoding," in *IEEE International Symposium on Circuits and Systems*, 2006, pp. 3874-3877.

[91] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264 / AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103 - 1120, September 2007.

[92] G.J. Sullivan and T. Wiegand, "Video Compression - From Concepts to the H.264/AVC Standard," in *Proceedings of the IEEE*, vol. 93, 2005, pp. 18-31.

[93] D. Marpe, T. Wiegand, and S. Gordon, "H.264/MPEG4-AVC fidelity range extensions: tools, profiles, performance, and application areas," in *IEEE International Conference on Image Processing*, vol. 1, 2005, pp. 593-596.

[94] S. Kwona, A. Tamhankarb, and K.R. Raoc, "Overview of H.264/MPEG-4 part 10," *Journal of Visual Communication and Image Representation*, vol. 17, no. 2, pp. 186–216, 2006.

[95] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 1, no. 7, pp. 620-623, July 2003.

[96] D.D. Feng, *Biomedical Information Technology.*, 2008.

[97] D. Mukherjee and S. Rajagopalan, *CT and MR Angiography of the Peripheral Circulation, Practical Approach with Clinical Protocols.*, 2007.

[98] G.D. Rubin and N.M. Rofsky, "CT and MR Angiography Comprehensive Vascular Assessment," *1st Edition, Lippincott Williams & Wilkins*, 2008.

[99] S.K. Somasundaram and P.Alli, "A Review on Recent Research and Implementation Methodologies on Medical Image Segmentation," *Journal of Computer Science*, vol. 8, no. 1, pp. 170-174, 2012.

[100] S. Crasteni, "Microstructural imaging algorithms – Design and implementation," in *5th Int. Conference Structural Integrity of Welded Structures (ISCS2007)*, Timisoara, Romania, 2007.

[101] N.S. Zulpe, C. Latur, V.P. Pawar, and N. SRTMU, "Level set and Thresholding for Brain Tumor Segmentation," *International Journal of Computer and Electrical Engineering*, vol. 4, no. 1, pp. 82-86, February 2012.

[102] R. Gonzalez, R. Woods, and S. Eddins, *Digital Image Processing Using MATLAB.*, 2004.

[103] N.R. Pal and S.K. Pal, "A review on image segmentation techniques," *Pattern Recognition*, vol. 26, pp. 1277-1294, 1993.

[104] G. Dougherty, *Digital Image Processin for Medical Applications.*, 2009.

[105] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 146-168, January 2004.

[106] C. Zheng, D.W. Sun, and L. Zheng, "Segmentation of Beef Joint Images Using Histogram Thresholding," *Journal of Food Process Engineering*, vol. 29, pp. 574-591, 2006.

[107] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, Analysis, and Machine Vision*, Third Edition ed., 2008.

[108] X. Jiang, X. Huang, M. Jie, and H. Yin, "Rock Detection Based on 2D Maximum Entropy Thresholding Segmentation and Ellipse Fitting," in *IEEE International*

Conference on Robotics and Biomimetics, China, 2007, pp. 1143-1147.

[109] T. Cover and J.Thomas, "Elements of Information Theory," *John Wiley & Sons*, 1991.

[110] H. Lee and R.H. Park, "Comments on an optimal threshold scheme for image segmentation," *IEEE Transaction on System Man Cybernetics*, pp. 741–742, 1990.

[111] H. Kye, B.S Sohn, and J. Lee, "Interactive GPU-based maximum intensity projection of large medical data sets using visibility culling based on the initial occluder and the visible block classification," *Computerized Medical Imaging and Graphics*, vol. 36, no. 5, pp. 366–374, 2012.

[112] R.C. Gonzalez and R.E. Woods, *Digital Image Processing.*, 2007.

[113] A. Ion, J. Dehmeshki, H. Amin, A. Jouannic, and S.D. Qanadli, "Automated detection of peripheral arteries in CTA datasets," *ICDS, St. Maarten, Netherlands Antilles*, Feb 2010.

[114] J. Dehmeshki, H. Amin, M. Valdivieso, and X. Ye, "Segmentation of pulmonary nodules in thoracic CT scans: A region growing approach," *IEEE Trans. Medical Imaging*, vol. 27, no. 4, pp. 467-480, April 2008.

[115] M. Macari, G.M. Israel, P. Berman, M. Lisi, A.J. Tolia, M. Adelman, and A.J. Megibow, "Infrarenal Abdominal Aortic Aneurysms at Multi–Detector Row CT Angiography: Intravascular Enhancement without a Timing Acquisition," *Radiology*, vol. 220, pp. 519-523, 2001.

[116] A.T. Hirsch, Z.J. Haskal, N.R Hertzer, C.W. Bakal, M.A. Creager, J.L Halperin, L.F Hiratzka, W.R. Murphy, J.W. Olin, J.B. Puschett, K.A. Rosenfield, D. Sacks, J.C. Stanley, L.M. Taylor, C.J. White, J. White, R.A White, E.M Antman, S.C Smith, C.D. Adams, "ACC/AHA Guidelines for the Management of Patients with Peripheral Arterial Disease (lower extremity, renal, mesenteric, and abdominal aortic): a collaborative report from the American Associations for Vascular Surgery/Society for Vascular Surgery," *Journal of Vacular and Interventional*

Radiology, vol. 17, pp. 1383–1398, 2006.

[117] M. Li, R. Poovendran and S. Narayanan, "Protecting patient privacy against unauthorized release of medical images in a group communication environment," *Computerized Medical Imaging and Graphics*, vol. 29, no. 5, pp. 367-383, 2005.

[118] Y.Y. Chen, "Medical images compression for remote diagnosis using modified SPIHT data organization and fidelity enhancement filter," *IJIST*, no. 2, pp. 49-61, 2007.

[119] http://www.openjpeg.org/.

[120] http://iphome.hhi.de/suehring/tml/.

[121] National Electrical Manufacturers Association, "Digital Imaging and Communications in Medicine (DICOM), Part 5: Data Structures and Encoding," , Rosslyn, 2011.

[122] V. Singh, "Recent Patents on Image Compression – A Survey," *Recent Patents on Signal Processing*, vol. 2, pp. 47-62, 2010.

APPENDIX A

Table A.1 Subjective measurements over several slices of the typical dataset

Slice Number	MOS Variable bitrate	MOS Fixed bitrate	MOS Lossy Jp2k
1	5	1	4
40	5	1	4
100	5	1	4
160	5	1	4
220	5	1	4
280	5	1	4
340	5	1	4
400	5	4	4
460	5	4	4
520	5	4	4
580	5	4	4
640	5	4	4
700	5	4	4
760	5	4	4
820	5	4	4
880	5	4	4
940	5	4	4
1000	5	4	4

Table A.2 Average subjective measurements for the 10 datasets

Slice Number	Ave. Diagnostic MOS Variable bitrate	Ave. Diagnostic MOS Fixed bitrate	Ave. Diagnostic MOS Lossy JP2K
D1(1038)	5	2.7	4
D2(0985)	5	2.8	4.3
D3(1046)	5	2.9	4.2
D4(1000)	5	3.0	4.3
D5(1029)	5	3.0	4.2
D6(1034)	5	2.7	4.1
D7(0990)	5	3.0	4.4
D8(1009)	5	2.6	4.3
D9(1000)	5	2.8	4
D10(0986)	5	3.1	4.1