MULTI-RESOLUTION MIXTURES OF PRINCIPAL COMPONENTS

MULTI-RESOLUTION MIXTURES OF PRINCIPAL COMPONENTS

by
Christopher Lesner

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree
Master of Science

McMaster University

MASTER OF SCIENCE (1998)
(Computer Science)

McMaster University
Hamilton

TITLE:	Multi-Resolution Mixtures of Principal Components
AUTHOR:	Christopher Lesner, B.Sc. (McMaster University)
SUPERVISOR:	Professor Suzanna Becker
NUMBER OF PAGES:	95

Abstract

The main contribution of this thesis is a new method of image compression based on a recently developed adaptive transform called Mixtures of Principal Components (MPC). Our multi-resolution extension of MPC — called Multi-Resolution Mixtures of Principal Components (MR-MPC) compresses and decompresses images in stages. The first stage processes the original images at very low resolution and is followed by stages that process the encoding errors of the previous stages at incrementally higher resolutions.

To evaluate our multi-resolution extension of MPC we compared it with MPC and with the excellent performing wavelet based scheme called SPIHT. Fifty chest radiographs were compressed and compared to originals in two ways. First, Peak Signal to Noise Ratio (PSNR) and five distortion factors from a perceptual distortion measure called PQS were used to demonstrate that our multi-resolution extension of MPC can achieve rate distortion performance that is 220% to 720% better than MPC and much closer to that of SPIHT. And second, in a study involving 724 radiologists' evaluations of compressed chest radiographs, we found that the impact of MR-MPC and SPIHT at 25:1, 50:1, 75:1 on subjective image quality scores was less than the difference of opinion between four radiologists.

Acknowledgements

I wish to express gratitude to the many people who have helped me to complete this project. The first of these are my two supervisors, Dr. Sue Becker and Dr. Simon Haykin. Both Sue and Simon have displayed patience, good humour, and encouragement, not withstanding my underestimate of the time and effort required for write-up. I also wish to thank Dr. Claude Nahmias for coordinating the medical evaluations and Dr. Nicholas Christoforou, David Palmer, Lianne Tam, and Bev Kenyon for assistance in printing evaluation radiographs. Finally, I am particularly grateful to a number of people for carefully reviewing draft material and for suggestions which in one way or another have influenced this project. They are Dr. Javad Alirezaie, Dr. Nicholas Christoforou, Dr. Jeff Zucker, Jean Lim, Wai Shan Lau, Linda Builder, Dr. Bob Dony, Ken Seergobin, Dr. Ivan Bruha, Dr. Sanzheng Qiao and Dr.Stephen Link.

Contents

List of Figures

List of Tables

Chapter 1

Medical Image Compression

1.1 Teleradiology and PACS

Teleradiology and Picture Archiving and Communication Systems (PACS) are medical technologies that improve health care. Teleradiology systems transmit remotely collected diagnostic images to a central location for expert analysis [21] and allow physicians to consult with specialists in a more timely manner by eliminating the need to physically ship images and experts [12]. For example, teleradiology can enable a busy radiology practice to provide 24 hour coverage to many peripheral hospitals and remote clinics at which patient volumes do not justify retaining a full-time in-house radiologist for around the clock coverage. Picture Archiving and Communication Systems (PACS) organize and archive locally collected images and enable physicians to view radiographic images on computer workstations located whereever clinical care is delivered. The benefits of PACS are decreased turnaround time for both routine and urgent examinations, faster specialist consultation [32] and more efficient communication between emergency physicians [1].

1.2 Medical Image Size

Implementing teleradiology and PACS is a challenge. Teleradiology requires long distance transmission of remotely acquired medical images. PACS requires storage of locally acquired medical images. These requirements are not trivial given the size of most medical images. For example, to fully represent a single chest radiograph with 50-μm pixels — each logarithmically scaled for dynamic range with guard bits added to measure the latitude of exposure and corresponding to a limiting resolution of 10 line pairs per millimeter (lp/mm) — requires 120 Mb [11]. Table 1.1 summarizes the approximate amount of data generated by various imaging procedures. Although commonly used images are smaller than the chest radiograph just described, a digital radiology department may nevertheless generate 50 gigabytes of image data on a busy day [21].

1.3 Image Compression

The solution to dealing with this much data is to represent images in a way that requires less storage space and transmission time. One way to do this is called *loss-*

Procedure	I_{height}	I_{width}	I_{bpp}	I_n	Mb
CT	512	512	12	30	16.0
MRI	256	256	12	50	6.5
US	512	512	6	36	9.5
PET	128	128	16	62	2.0
SPECT-8	128	128	8	50	0.8
SPECT-15	128	128	16	50	1.6
DSA	1024	1024	8	20	20.0
DF	1024	1024	8	15	15.0
CR	2048	2048	16	2	16.0
MAMMOGRAM	4500	3200	12	1	38.0

Table 1.1: This table shows the amount of data generated per exam for various diagnostic procedures [15, 11, 19]. Image dimensions I_{height} and I_{width} and the number of bits used to represent each pixel I_{bpp} all vary between procedures. Image files are larger when high spatial and high contrast resolution is needed. Local conventions determine the number of images collected I_n. The amount of data generated per procedure is $I_{height} * I_{width} * I_{bpp} * I_n$

less compression. Lossless compression works by removing image repetition. On still frame gray scale medical images lossless compression can achive ratios around 4:1. These ratios however are hard to improve becasue the maximum amount of compression lossless compression can achieve is upper bound by the amount of repetition [17].

Another way of representing images so they require less storage space and transmission time is called *lossy compression.* Lossy compression approximates images by leaving out what's least important and can — at the price of worse approximation — attain compression ratios that are orders of magnitude higher than lossless ratios. The following three recent studies — which are discussed further in chapter 5 — give an idea as to how much compression lossy compression can achieve:

1. Savcenko et al. [36] assessed the effect of wavelet compression of 60 posterior-anterior chest radiographs on detection of small uncalcified pulmonary nodules and fibrosis. They found no substantial difference in the overall diagnostic accuracy between uncompressed images and ones compressed at 40:1 and 80:1.

2. Erickson et al. [9] assessed wavelet compression of 40 posterior-anterior chest radiographs on radiologist visibility of anatomic structures. Their findings suggest that lossy compression at 40:1 or more can be used without perceptible loss in the representation of anatomic structures.

3. Goldberg et al. [27] assessed lossy wavelet compression of twelve abnormal radiographs by presenting original and compressed/decompressed images in random

order and asked reviewers to judge whether diagnosticly significant image degradation was present. All seven board-certified radiologists found no degradation below a compression ratio of 30:1.

1.4 Motivation

The best image compression algorithms in existence today are based on the discrete wavelet transform (DWT) which offers good concentration of energy and decorrelation for a wide class of signals in both time and frequency. [19, 26, 37, 34, 33]. In this thesis, we introduce and evaluate a new method of image compression based on a multi-resolution extension to a recently developed adaptive transform called Mixtures of Principal Components (MPC) [15, 16]. Although MPC compression is superior to compression techniques based on and related[1] to the optimal Karhunen-Loève Transform (KLT), MPC does not, like wavelets, exploit repetition present in between multiple resolution of an image. Our goal was to improve MPC compression by extending it to process images at multiple resolutions.

The motivation for our work is fourfold. First, there exists the potential to boost image transmission through put on all present day communication links. This is important for the cost effective deployment of teleradiology today. Second, there is the potential to conserve bandwidth in the crowded wireless communication spectrum.

[1] MPC is superior to the discrete cosine transform (DCT) approximation of KLT used in the common compressed image format JPEG.

This will be important for itinerant teleradiology systems in the future. Third, there is the potential to increase image capacities of all present day storage media. This is important for storage strained PACS today. Fourth, there is the potential to boost all future storage media improvements multiplicatively — for example a mere tenfold improvement in hardware multiplied by 10:1 compression gives a hundredfold improvement. This will be important for PACS in the future.

1.5 Organization

The remainder of this thesis is organized in five chapters. Chapter 2 reviews some conventional lossless and lossy compression techniques along with the recently developed adaptive transform called MPC. Our method, which is a multi-resolution extension of MPC, is introduced in chapter 3. The performance of the new method is investigated with chest radiographs first, numerically in chapter 4 and second, empirically in chapter 5. Finally, chapter 6 concludes the thesis by reviewing the salient points of our method.

Chapter 2

Image Compression

2.1 Lossless Compression

Image compression modifies images so that less storage space and transmission time are required. *Lossless compression* is a reversible form of compression that *removes image repetition.* For example, images have many areas in which pixels have the same value. One way to reduce this kind of repetition is to store the value and number of similar pixels, instead of storing individual pixel values separately.

2.1.1 Run Length Compression

This method — called *run length compression* — first finds the least frequently occurring pixel value. This value is then used to indicate that the next two items in the compressed stream refer to a pixel value and the number of pixels in a row having

```
01 01 00 00 00 00 00 00 00 02 02 02 02 02 01 07 ...
-- -- -------------------- -------------- -- --
|  |     ,-------'               |        |  |
|  |     |                       |        |  |
|  |     |        ,--------------'        |  |
|  |     |        |     ,-----------------'  |
|  |     |        |     |     ,--------------'
|  |     |        |     |     |
-- -- -------- -------- -- --------
01 01 07 00 07 07 02 05 01 07 07 01 ...
```

Figure 2.1: This figure shows run length compression of an image pixel stream. 07 is known to be the least common value so it is used to indicate that the next two values specify a compressed run. The compressed data is stored as 01, 01, run of seven 00's, 01, run of five 02's, 01 and 07. The last 07 must be represented as a run to avoid misinterpreting the two values that follow it in the stream. 16 values are compressed to 12 values achieving 1.33:1 compression ratio.

that value. The scheme is demonstrated in figure 2.1. Three values are required to describe a run, so runs are compressed if they contain more than three pixels. The least frequent pixel is the only exception — it must always be encoded as a run to avoid misinterpreting the two pixels that follow it during decoding.

Run length compression works well when adjacent pixels have the same value often. If repeated pixels are not next to each other, run length compression fails — the compressed image file size is larger then the original image file size due to the overhead of having to represent all occurrences of the least common pixel with three values.

pixel value	00	01	02	03	04	05	06	07
frequency	25	4	6	12	2	0	1	50
fixed length code	000	001	010	011	100	101	110	111
variable length code	01	00001	0001	001	000001	0000000	0000001	1

Table 2.1: This figure demonstrates Huffman compression. A 10x10 image contains pixel values with indicated frequencies. If each pixel is assigned a 3 bit string, the image can be encoded in 300 bits. However, if common pixels are assigned shorter bit strings, the image can be encoded in just 199 bits.

2.1.2 Huffman Compression

An alternative to run length compression that works even when repeated pixels are not next to each other, is *Huffman Compression.* This method reduces the average number of bits necessary to represent each pixel by using the knowledge that some pixel values repeat more than others. For example a 10x10 image has pixel frequencies shown in table 2.1. Pixel value 07 occurs 50 times, value 00 occurs 25 times, and so on. Only eight different pixel values occur altogether. If each pixel is assigned a fixed 3-bit code word, the image can be stored in 300 bits. However, by giving frequently occurring pixel values short codewords and infrequent pixel values long codewords, a variable-length Huffman code can represent images more efficiently.

This is shown in table 2.1. The most common pixel value 07 is assigned the 1-bit string '1' while the least common pixel value 05 is assigned the 7-bit string '0000000'.

Using this code the image can be encoded in just

$$25 * 2 + 4 * 5 + 6 * 4 + 12 * 3 + 2 * 6 + 0 * 7 + 1 * 7 + 50 * 1 = 199 \text{ bits}$$

The variable length code words in table 2.1 are paths from a binary Huffman tree. Huffman trees are built by successively combining the two least frequent symbols to form a new composite symbol that represents the frequency of all nodes beneath it. A code generated in this way has the important property that no bit string is a prefix of any other. This property simplifies decompression to identifying the initial bit string, looking up the original pixel value, and repeating the process on any bits that remain. For example, '111 011 001 000 101 011' compresses to '1 001 00001 01 0000000 001' which decodes to pixel values '07 03 01 00 05 03'.

2.2 Lossy Compression

Although lossless compression schemes such as run length and Huffman compression are completely reversible, the compression ratios they can achieve are limited by the amount of repetition present in the original image. On still–frame gray–scale medical images lossless ratios from 2:1 to 3:1 are hard to improve [17]. To get more compression we must resort to an irreversible form of compression called *lossy compression.* Lossy compression approximates images by *leaving out what's least important.* For example, in most raw medical images the area of interest is surrounded by a useless border. Removing this border reduces the image size. Further compression can also be achieved by reducing the number of bits that represent each pixel. This is called *scalar quantization.* For example, computed tomography (CT) algorithms require 16 bits per pixel calculating accuracy, but in the final image, only 12 bits are significant – 4 bits of every pixel are wasted on recording noise and can be removed[21].

2.2.1 Block Coding

Another way to do lossy compression is to first cut images into small non-overlapping rectangular blocks of pixels. Figure 2.2 shows 1x2 pixel blocks from a typical image. The pixels in each block $P1$ and $P2$ are plotted against each other in D-dimensional ($D = 2$) *block space.* Why do some blocks form linear trends and others appear in clusters? Trends occur because inside each block pixel values are correlated — in images dark pixels tend to have dark neighbors and light pixels tend to have light neighbors. Clusters occur when all pixels in one block have similar values to pixels in other blocks — in images some patches resemble each other.

2.2.2 Karhunen-Loève Transform (KLT)

To optimally compress the information present in trends we can use a linear projection called the *Karhunen-Loève Transform.* KLT is a special case of Principal Component Analysis (PCA) which is defined as follows: For a set of N observed D-dimensional vectors $T(n)$, $n \in 1, \ldots, N$, the Q principal axes $W(q)$, $q \in 1, \ldots, Q$, are those

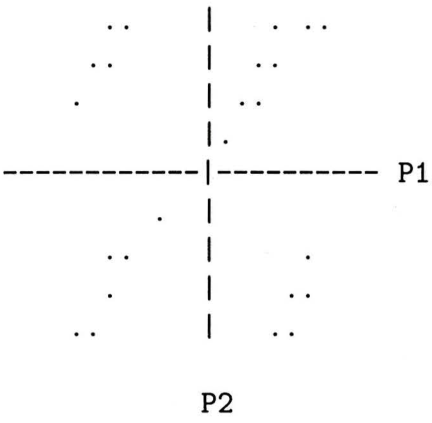


Figure 2.2: Plot of 2x1 pixel blocks from a typical image. Each block's two pixels $P1$ and $P2$ are plotted against each other in D-dimensional ($D = 2$) block space.

orthonormal axes onto which retained variance under projection is maximal. It can be shown that vectors $W(q)$ are the Q dominant eigenvectors (dominant = those with largest associated eigenvalues $Y(q)$) of the sample covariance matrix

$$S = \exp[(T(i) - U)(T(i) - U)^T]$$

such that

$$SW(q) = Y(q)W(j)$$

where U is the mean of $T(n)$.

The Q principal components of observed vector $T(n)$ are given by vector

$$X(n) = W^T(T(n) - U)$$

where

$$W^T = (W(1), W(2), \ldots, W(q))^T$$

The variables $X(j)$ are decorrelated in that the covariance matrix

$$\exp[XX^T]$$

is diagonal with elements $Y(q)$ [20].

PCA is equivalent to KLT when output dimension Q is less than the input dimension D. Here the decorrelating property of PCA can be used for optimal dimensionality reduction. The KLT reconstruction of $T(n)$ is given by

$$T'(n) = W(X(n) + U)$$

and is optimal [31] in the sense that it minimizes the sum of squared reconstruction error

$$\sum_n ((T(n) - T'(n))^2)$$

The effect of KLT is to represent image blocks by their projections onto better axes. Figure 2.3 shows a one component KLT reconstruction. Blocks from figure 2.2 are mapped onto the principal axis along which variability is greatest. 2:1 compression is achieved because after projection each block is represented by one principal axis coordinate instead of two pixel value coordinates. More compression is possible when

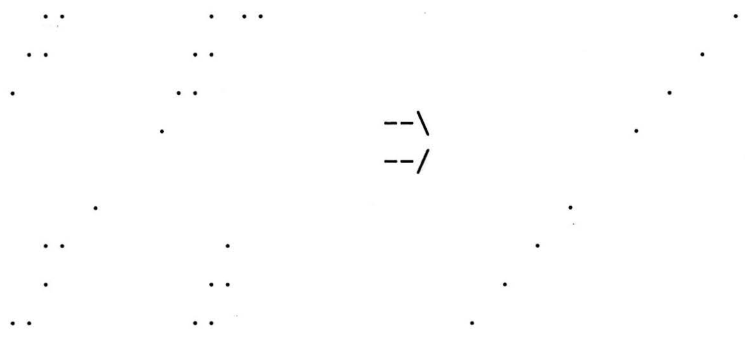


Figure 2.3: A one component MPC reconstruction of blocks from figure 2.2. Blocks are mapped onto the principal axis along which variability is greatest. 2 : 1 compression is achieved because after projection each block is represented by one principal axis coordinate instead of two pixel value coordinates.

the dimensionality of blocks is bigger — in practice as few as 4 principal coordinates can adequately represent 8x8 input blocks, giving 16:1 compression.

Points displaced the least distance by projection have the lowest reconstruction error. Notice that in figure 2.3 blocks that are greatly displaced are initially far from the principal axis — with $P1$ very different from $P2$. Unfortunately, some of these blocks represent image edges — any block that spans an edge must have $P1$ different from $P2$. Edge blocks are rare with respect to the entire image and are poorly reconstructed by KLT. This is unfortunate because edge blocks are extremely important for human observers — hence KLT, although optimal at preserving global information content, is suboptimal at reconstructing edge blocks that are perceptually very important.

2.2.3 Vector Quantization (VQ)

An alternative to KLT relies on the observation that blocks tend to cluster. Clusters can be compressed using a generalization of scalar quantization called vector quantization [30]. Scalar quantization maps a signal $T(n)$ to a series of K discrete messages. For the k'th message, there exist thresholds $H(k)$ and $H(k+1)$, and an output value $U(k)$ such that $H(k) < U(k) < H(k+1)$. VQ is a D-dimensional generalization that uses a set — codebook — of representation vectors — codewords — in place of output levels. Given a codebook of K codewords $\{W(i)\bar{i} = 1, \ldots, n\}$, a vector $T(n)$ is represented by the i'th codeword such that the reconstructed vector $T'(n)$ is given by

$$T'(n) = W(i)$$

where

$$||T(n) - W(i)|| = \min_{i=1}^{K}(||T(n) - W(i)||)$$

VQ achieves compression because all the pixels in each image block are represented by a single cluster index. Figure 2.4 shows a four class VQ reconstruction — blocks from figure 2.2 map to their nearest cluster centers. Blocks with $P1$ very different from $P2$ are displaced less, which means that VQ represents edges much better than KLT does. However many clusters are needed to reconstruct non-edge blocks with fidelity comparable to KLT because VQ does not take advantage of linear trends. The next section shows how this compromise can be avoided.

2.2.4 Mixtures of Principal Components (MPC)

Recently [15, 16] have introduced a generalization of VQ and KLT called Mixtures of Principal Components. MPC partitions data to K non-overlapping regions and represents each region with a Q-dimensional linear subspace. It can be shown that MPC is equivalent to VQ when all components retained ($Q = D$) and KLT when number of classes is one ($K = 1$). Thus, if block X is in class C_i its MPC encoding is

$$Y = W_i X$$

and decoding is

$$X' = {W_i}^T Y$$

A four-class, one-component MPC reconstruction is shown in figure 2.5. Blocks from figure 2.2 are mapped onto the principal axis of their closest cluster. Both clusters and trends are taken into account by this method, thereby improving reconstruction beyond that of either KLT or VQ alone [15, 16].

2.3 Summary

This chapter has distinguished between lossless and lossy compression and reviewed some basic image compression algorithms to provide the necessary background for the lossy image compression scheme we introduce in the next chapter.

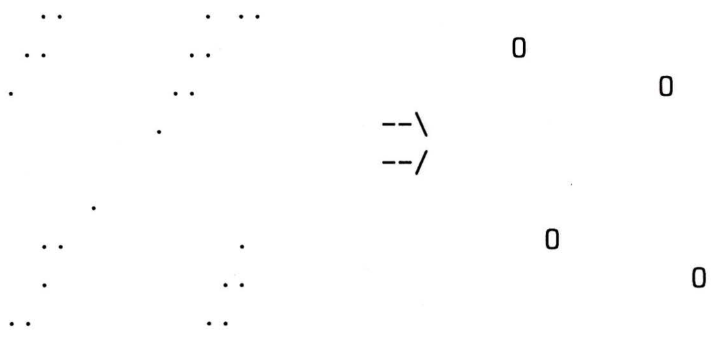


Figure 2.4: A four class VQ reconstruction of blocks from figure 2.2. Each pixel in a given block is mapped to the center of the nearest cluster. Compression is achieved because all pixels in each block are represented by a single cluster index.

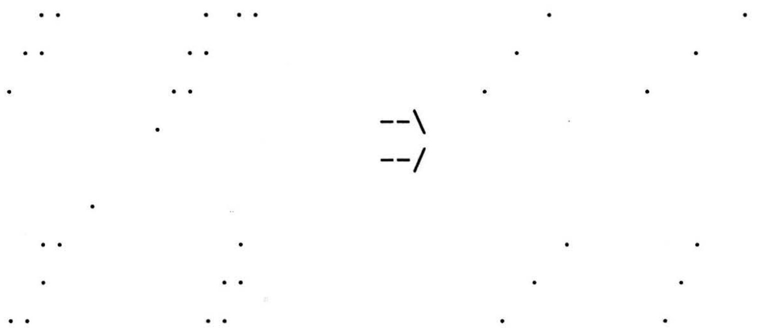


Figure 2.5: A four class one component MPC reconstruction of blocks from figure 2.2. Each pixel in a given block is mapped onto the principal axis of the closest cluster. Use of both clusters and trends improves reconstruction beyond that of either KLT or VQ alone.

Chapter 3

Multi-Resolution MPC (MR-MPC)

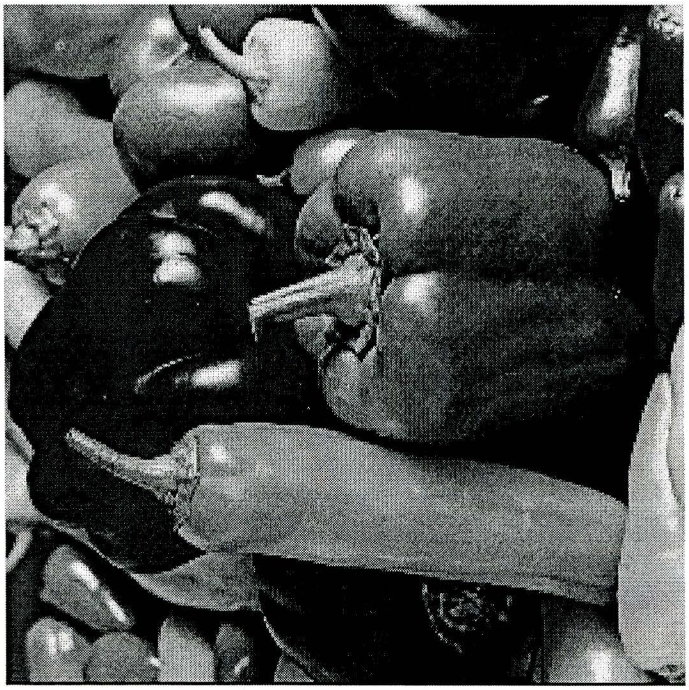
3.1 Introduction

The best image compression algorithms in existence are based on the discrete wavelet transform (DWT) which offers good concentration of energy and decorrelation for a wide class of signals in both time and frequency [19, 26, 37, 34, 33]. This chapter introduces an alternative to wavelet compression based on the MPC transform discussed in section 2.2.4. Although MPC can compress better than the optimal Karhunen-Loève Transform (KLT), MPC does not, like wavelet based algorithms, exploit repetition present in between multiple resolution of an image. Our goal was to improve MPC compression by extending it to process images at multiple resolutions. We begin by reviewing some of the ideas that motivated the development of our algorithm.

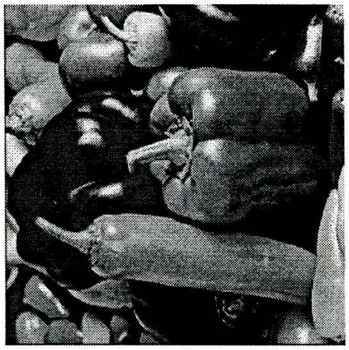
3.1.1 Review

Down-sampling removes image detail and reduces image size. For example, figure 3.1 shows an original image (A) down-sampled to half resolution (B) and quarter resolution (C). *Up-sampling* increases image size and is, in spirit, the opposite of down-sampling. For example, figure 3.2 shows an original image (A) up-sampled to double (B) and quadruple resolution (C). Up-sampling cannot exactly undo down-sampling because down-sampling removes high frequency information — detail is lost. The lost detail can, however, be approximated using interpolation during up-sampling. For example, figure 3.3 shows an image down-sampled to a low resolution and then up-sampled back to its original resolution using interpolation. In this figure, the original image (A) and the interpolated image (C) look similar because the detail lost during down sampling can be interpolated from the low resolution image (B). The detail that interpolation cannot restore is the difference between images (A) and (C) shown in image (D). The error image (D) is easy to compress using methods described in chapter 2 because nearly all its pixels are either zero or close to zero – that is, it's mostly black.

Figure 3.1: This figure demonstrates down-sampling. (A) is the original 512x512 image, (B) is a 256x256 image down-sampled from (A) and (C) is a 128x128 image down-sampled from (A). Details of down-sampling are discussed in section 3.3.1.



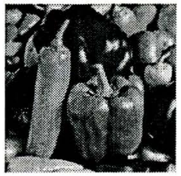
A



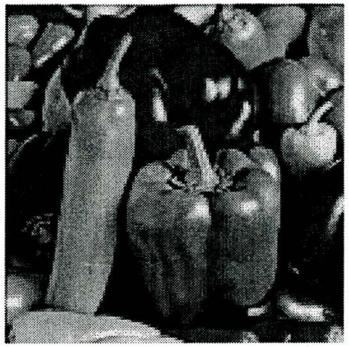
B

C

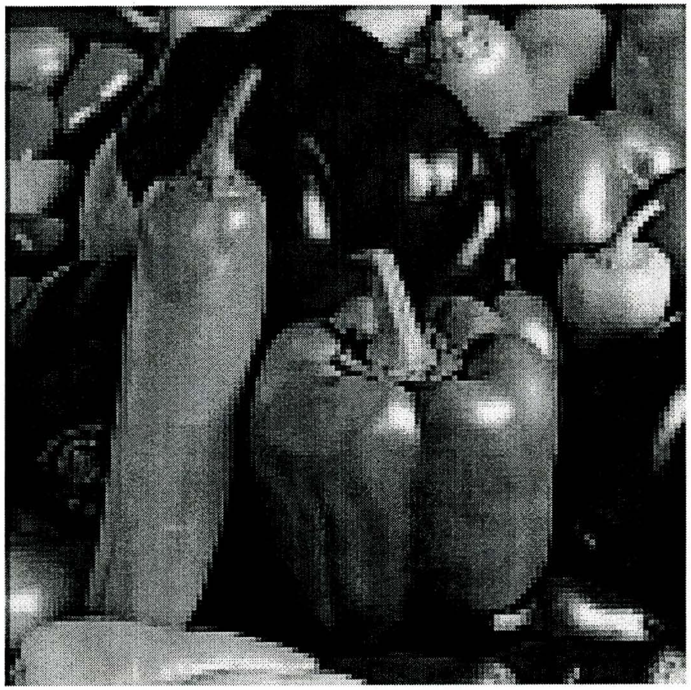
Figure 3.2: This figure demonstrates up sampling. (A) is the original 128x128 image, (B) is a 256x256 image up-sampled from (A) and (C) is a 512x512 image up sampled from (A). Details of up-sampling are discussed in section 3.3.2.



A



B

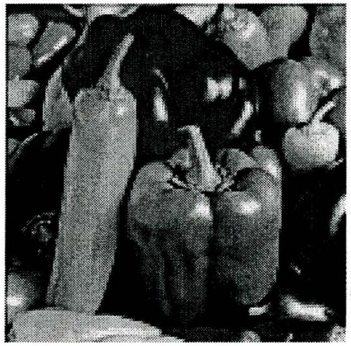


C

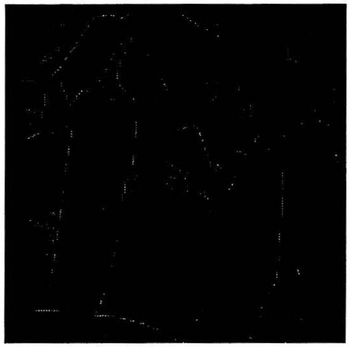
Figure 3.3: This figure demonstrates that most high resolution detail can be approximated by interpolation of low resolution images. The original 512x512 image (A) is down-sampled to 256x256 (B) then up-sampled using interpolation to 512x512 (C). The original image (A) and interpolated image (C) look similar because image detail lost during down-sampling can be interpolated from the low resolution image (B). The detail that interpolation cannot restore is shown in the error image (D) which is the difference between images (A) and (C). The error image (D) is easy to compress because nearly all of its pixels are either zero or close to zero — it's mostly black.

A

B



C



D

22.5

3.2 Design

The ideas in the previous section can be used to extend the MPC compression scheme proposed by [15, 16]. Our method is called called Multi Resolution Mixtures of Principal Components (MR-MPC), and is based on two key observations. The first observation is that *high frequency image detail is predictable from lower resolutions.* The second observation is that the *detail not predictable from lower resolutions is easy to compress.*

The job of MR-MPC is twofold — *compression* and *decompression.* Compression progressively encodes the original image to a compressed stream. Decompression progressively decodes the compressed stream and updates an approximation of the original image. These tasks may run sequentially — compressing image files on a file system and later decompressing them as needed — or concurrently — at opposite ends of a communications link.

MR-MPC compresses and decompresses images in stages. The first stage processes the original image at a very low resolution and is followed by stages that process the encoding errors of the previous stages at incrementally higher resolutions. The number of stages and resolution at each stage is controlled by a resampling scheme parameter explained in section 3.2.4.

3.2.1 Compression Overview

The task of compression is carried out by a pipeline of N stages. Figure 3.4 depicts the five steps that are carried out in each stage of the compression pipeline.

- Step (1) down-samples the full resolution error image Ife(n) to a low resolution error image Ile(n). This is explained in section 3.3.1.
- Step (2) encodes the low resolution error image Ile(n) to a compressed error stream Sle(n) using MPC. This is explained in section 3.3.3.
- Step (3) decodes the compressed error stream Sle(n) to Ilr(n) which is a low resolution reconstruction of Ile(n). This is explained in section 3.3.4. At this point a copy of Sle(n) is also sent to the step (1) of decompression. This is explained in section 3.2.2.
- Step (4) up-samples the low resolution reconstruction image Ilr(n) to a full resolution reconstruction image Ifr(n). This is explained in section 3.3.2.
- Lastly step (5) computes the error image that is processed by the next stage by subtracting Ifr(n) from Ife(n).

Figure 3.5 shows what happens at each step of a three stage compression pipeline. When the compression pipeline starts, Ife(1) is set to the original image. At subsequent stages, Ife(n) is the full-resolution residual error image from the (n-1)'st stage.

And when this pipeline finishes, Ife(N+1) is the residual error image from the last stage.

3.2.2 Decompression Overview

The task of decompression is also carried out by a pipeline of N stages. The n'th stage of decompression has three steps which are shown in figure 3.6.

- Step (1) decodes the compressed error stream Sle(n) which is the output from step (2) of compression (described in section 3.2.1) to a low resolution reconstruction Ilr(n). This is explained in section 3.3.4.
- Step (2) up-samples the low resolution reconstruction image Ilr(n) to a full resolution reconstruction image Ifr(n). This is explained in section 3.3.2.
- Lastly step (3) computes the full resolution cumulative image for the next stage Ifz(n+1) by adding Ifr(n) to Ifz(n).

How stages of compression interact with stages of decompression is shown in figure 3.7. When compression starts, Ife(1) is set to the original image. At subsequent stages, Ife(n) is the full-resolution residual error image from the (n-1)'st stage. When compression stops, Ife(N+1) is the residual error image. When decompression starts, Ifz(1) is set to a zero image. At subsequent stages, Ifz(n) is progressively refined by corrections from the (n-1)'th stage of compression. When decompression stops, Ifz(N+1) is the recovered image. Figure 3.8 shows an image being progressively

Figure 3.4: This figure depicts five steps in the n'th stage of MR-MPC's N stage compression pipeline. When this pipeline starts, Ife(1) is set to the original image. At subsequent stages, Ife(n) is the full-resolution residual error image from the (n-1)'st stage. When pipeline stops, Ife(N+1) is the is residual error image. Step (1) down-samples. Step (2) encodes error. Step (3) decodes error. Step (4) up-samples. Step (5) computes error image for compression in stage n+1.

26.5

MR-MPC COMPRESSION

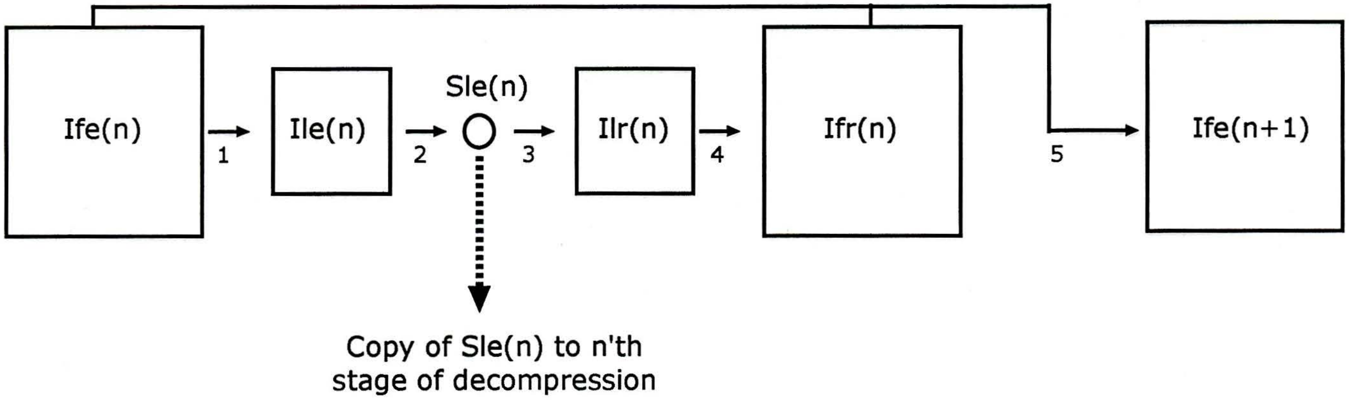
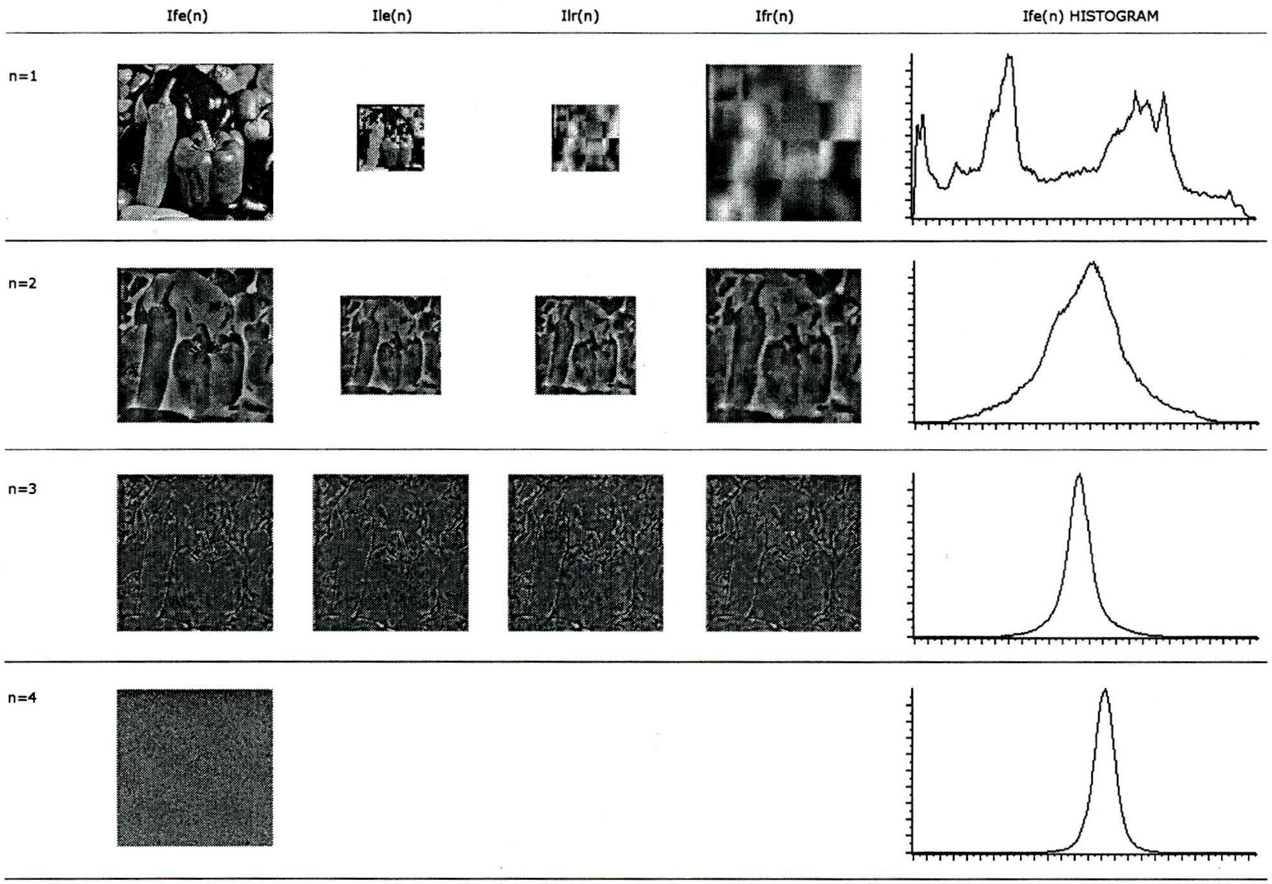


Figure 3.5: This figure shows what happens in a three stage compression pipeline. When this pipeline starts, Ife(1) is set to the original image. At subsequent stages, Ife(n) is the full-resolution residual error image from the (n-1)'st stage. When the pipeline stops, Ife(4) is the residual error image. Progressive decompression is shown in figure 3.8.



27.5

decompressed as it's being received from the three stage compression pipeline in figure 3.5.

3.2.3 Training Overview

The MPC transforms done in step (2) and (3) of compression and step (1) of decompression use K transformation matrices $\{W_1, W_2, \ldots, W_K\}$. These matrices are empirically determined by initial training on a pool of images representative of those that will be compressed in practice. The training is carried out in an N stage pipeline similar to that used for compression. The n'th stage of this pipeline is shown in figure 3.9. This training stage differs from the compression stage shown in figure 3.4 in only three ways.

- Firstly, training does not send a copy of the compressed error stream to decompression.
- Secondly, training works with a 3D data structure that contains all training images whereas compression works with a 2D data structure that contains only one image.
- Lastly, in step (2) of each training stage, MPC weights are trained first (this is explained in section 3.3.8) after which, as in the compression procedure, the weights are used to encode the low resolution error image Ile(n) to a compressed error stream Sle(n).

Figure 3.6: This figure depicts three steps in the n'th stage of MR-MPC's N stage decompression pipeline. When this pipeline starts, Ifz(1) is set to a null or zero image. When this pipeline finishes, Ifz(N+1) is the recovered image. Step (1) decodes error image. Step (2) up-samples. Step (3) computes approximated image for next level.

29.5

MR-MPC DECOMPRESSION

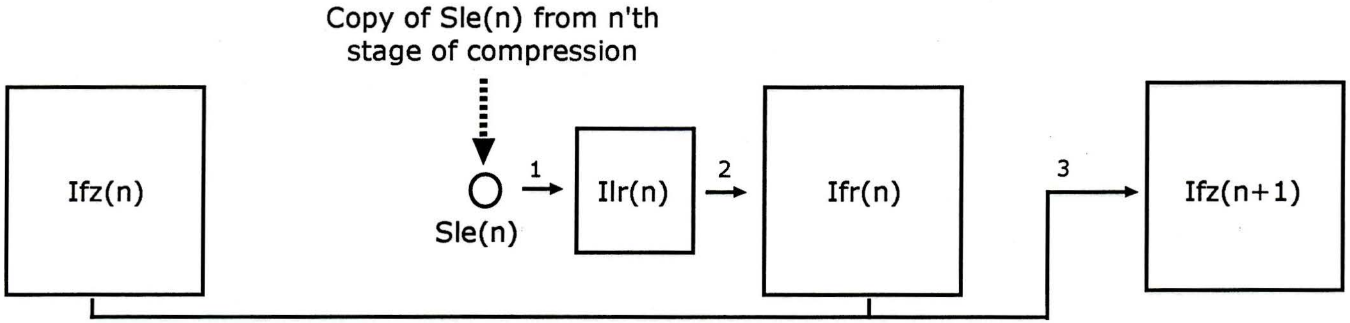
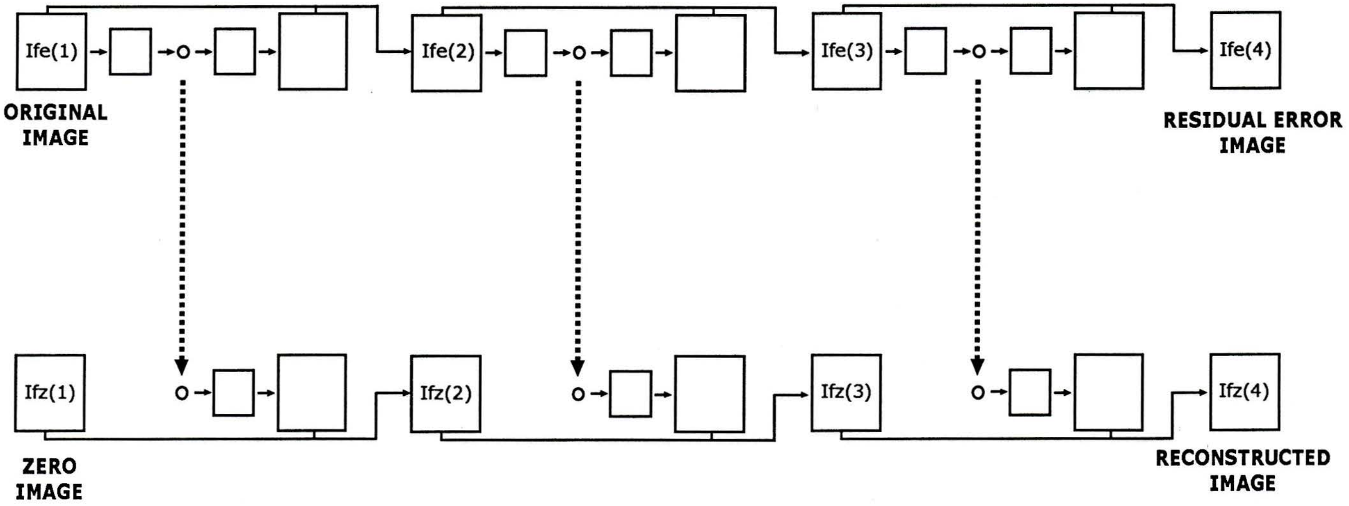


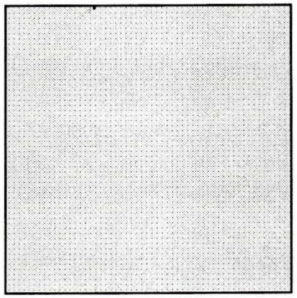
Figure 3.7: This figure shows three stages of interaction between compression (figures 3.4 and 3.5) and decompression (figures 3.6 and 3.8). When compression starts, Ife(1) is set to the original image. At subsequent stages, Ife(n) is the full-resolution residual error image from the (n-1)'st stage. When compression stops, Ife(4) is the residual error image. When decompression starts, Ifz(1) is set to a zero image. At subsequent stages, Ifz(n) is progressively refined by corrections from the (n-1)'th stage of compression. When decompression stops, Ifz(4) is the recovered image.

COMPRESSION

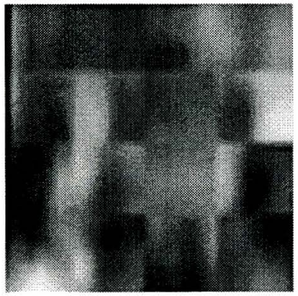


DECOMPRESSION

Figure 3.8: This figure shows an image being progressively decompressed as it's being received from the three stage compression pipeline in figure 3.5. When decompression pipeline starts, Ifz(1) is set to a zero image. At subsequent stages, Ifz(n) is progressively refined by corrections from the (n-1)'th stage of compression. When decompression pipeline stops Ifz(4) is the recovered image.

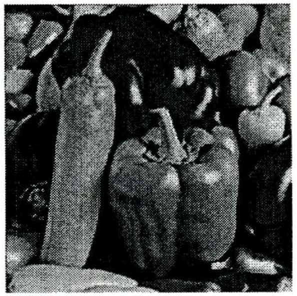


Ifz(1)



Ifz(2)

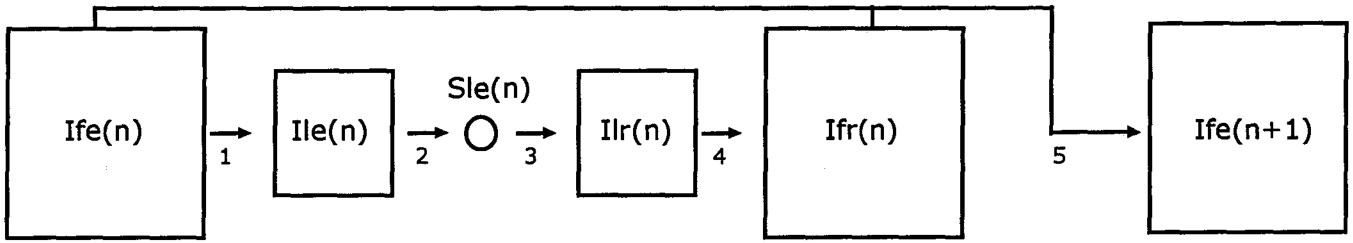
Ifz(3)



Ifz(4)

Figure 3.9: This figure depicts the five steps in the n'th stage of MR-MPC's N stage training pipeline. Step (1) down-samples. Step (2) trains MPC on Ile(n) then uses weights learned to encode Ile(n) to compressed error stream Sle(n). Step (3) decodes error. Step (4) up-samples. Step (5) computes error image for training in stage n+1. Training differs from compression only in that it trains weights in step (2), works with data structures that contain all training images and does not send a copy of the compressed error error stream Sle(n) to decompression.

MR-MPC TRAINING



3.2.4 Resampling Scheme

Compression (section 3.2.1), decompression (section 3.2.2) and training (section 3.2.3) tasks consist of multiple stages. The number and resolution of these stages are defined by an empirically determined N-tuple of integers called the *resampling scheme.* For example a (16,4,1) resampling scheme has three stages: the first stage down-samples by a factor of 16, the second stage by a factor of 4, and the third and last stage processes the image at full resolution.

Resampling schemes with more and finer spaced stages better reduce inter-resolution repetition if inter-resolution repetition is present. If not, resampling schemes with more and finer spaced stages add to overhead and give less compression. To find a good resampling scheme, we tested several different ones on 512x512 pixel natural images and 2048x2048 pixel chest radiographs. Octave spaced resampling schemes worked best. The final decision was between (16,4,1) and (16,8,4,2,1). Both resampling schemes worked about the same, so we chose (16,4,1) for all subsequent experiments because it was simpler.

3.3 Details

3.3.1 Down-Sampling

In step (1) of compression, which is first mentioned in section 3.2.1, a full resolution error image Ife(n) is down-sampled to a low resolution error image Ile(n). Down-sampling is achieved by dividing an image into equal non-overlapping blocks each of which determines the value of one pixel in the down-sampled image. One approach is to use the value of a single pixel in each block — for example, the first pixel. This is called *subsampling* and is demonstrated in figure 3.10. Another approach is to compute and use the block's average pixel value. This is demonstrated in figure 3.11.

We tried both types of down-sampling and found that although subsampling was faster, images down-sampled with averaging were better approximated during subsequent up-sampling. Thus, results reported in this thesis are based on MR-MPC that down-samples using averaging.

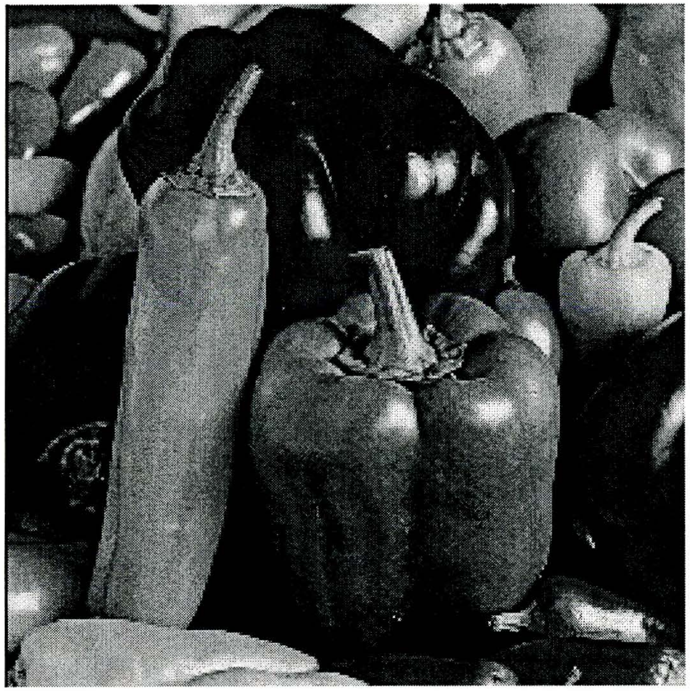
3.3.2 Up-Sampling

In step (4) of compression (section 3.2.1) and step (2) of decompression (section 3.2.2), a low resolution reconstruction image is up-sampled to a full resolution reconstruction image Ifr(n). One method of up-sampling is to generate images so that all the pixels in each non-overlapping block take the value of the corresponding single pixel in the low resolution image. This approach is demonstrated in figure 3.12. The problem with this type of up-sampling is that the images it generates are blocky. If this approach is applied to images that have been down-sampled, the up-sampled images are not good approximations of what the original images looked like before down-sampling. This is shown in figure 3.13.

A better way to up-sample is to interpolate the unknown pixels in high resolution

Figure 3.10: This figure demonstrates down-sampling using subsampling. (A) is the original 512x512 image, (B) is a 256x256 image generated by dividing (A) into non-overlapping 2x2 pixel blocks and using the first pixel in each block. (C) is a 128x128 image similarly generated by dividing (A) into 4x4 pixel blocks.

35.5

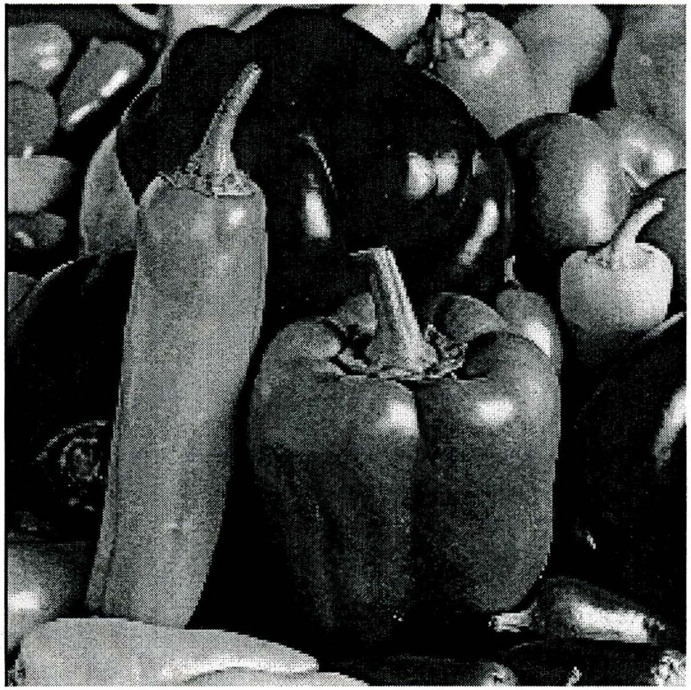


A

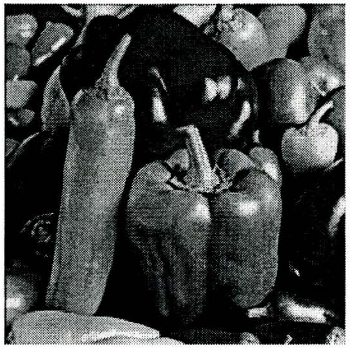
B

C

Figure 3.11: This figure demonstrates down-sampling using averaging. (A) is the original 512x512 image, (B) is a 256x256 image generated by dividing (A) into non-overlapping 2x2 pixel blocks and using the average value of all pixels in each block. (C) is a 128x128 image generated similarly by dividing (A) into 4x4 pixel blocks.



A

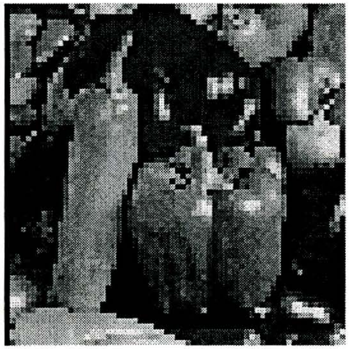


B

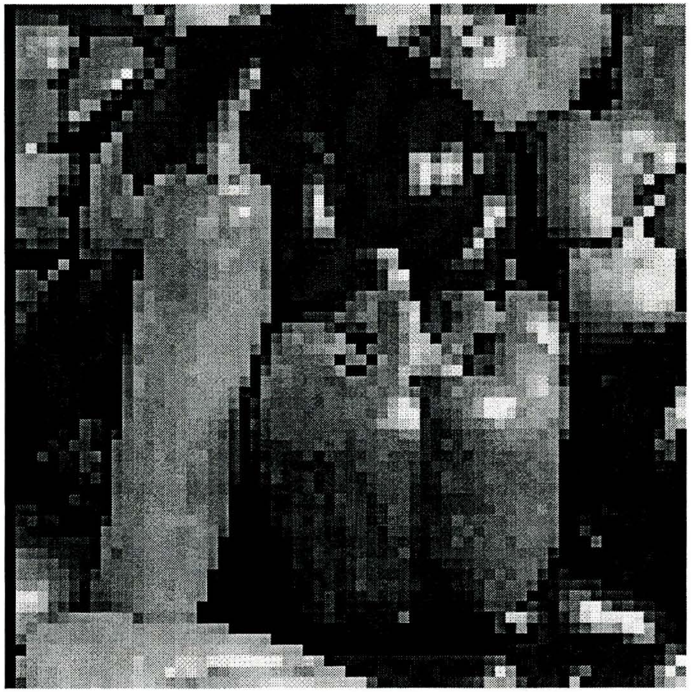
C

Figure 3.12: This figure demonstrates up-sampling without interpolation. (A) is the original 128x128 image. (B) is a 256x256 image generated so all pixels in each non-overlapping 2x2 pixel block take the value of one (A) pixel. (C) is a 512x512 image generated similarly with 4x4 blocks.

A



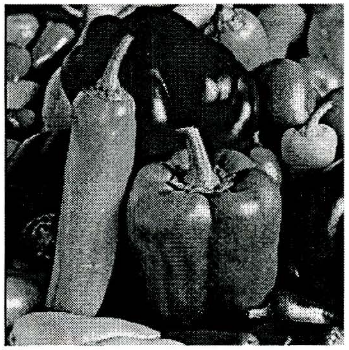
B



C

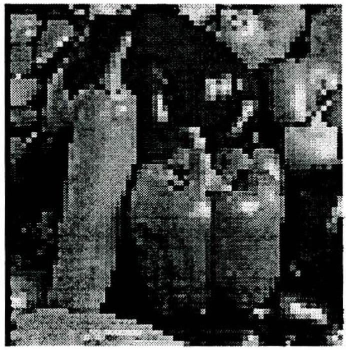
37.5

Figure 3.13: This figure demonstrates the problem with up-sampling without interpolation. The original 512x512 pixel image (A) is down-sampled to 256x256 (B) then up-sampled without interpolation to 512x512 (C). The up-sampled image (C) looks blocky and is not a good approximation of the original image (A). The disparity is shown in the error image (D) which is the difference between images (A) and (C).



A

B



C

D

38.5

images from values of neighboring pixels in low resolution images. In one dimension we can estimate the value of an unknown high resolution pixel $P(x)$ at point x as follows:

$$P(x) = (1-h)P(x_a) + hP(x_b)$$

where $P(x_a)$ and $P(x_b)$ are the two closest known neighboring pixels such that $x_a <= x <= x_b$ and

$$h = (x - x_a)/(x_b - x_a)$$

is the fraction of distance x is between x_a and x_b such that $0 <= h <= 1$.

Similarly in two dimensions we can use bilinear interpolating to estimate the value P(x,y) using the four closest known neighboring pixels $P(x_a, y_a)$, $P(x_b, y_b)$, $P(x_a, y_b)$, $P(x_b, y_a)$ such that $x_a <= x <= x_b$ and $y_a <= y <= y_b$ as follows

$$P(x,y) = (1-h)(1-v)P(x_a,y_a)+(h)(1-v)P(x_b,y_a)+(h)(v)P(x_b,y_b)+(1-h)(v)P(x_a,y_b)$$

where

$$h = (x - x_a)/(x_b - x_a)$$

is the fraction of distance x is between x_a and x_b such that $0 <= h <= 1$ and

$$v = (y - y_a)/(y_b - y_a)$$

is the fraction of distance y is between y_a and y_b such that $0 <= v <= 1$

An image up-sampled using this approach is demonstrated in figure 3.14. The benefit of interpolation during up-sampling is that generated images are smooth. Moreover, if this approach is applied to images that have been down-sampled, the up-sampled images are fairly good approximations of what the original images looked like before down-sampling. This is shown in figure 3.15. All results in this thesis are based on MR-MPC that up-samples using interpolation.

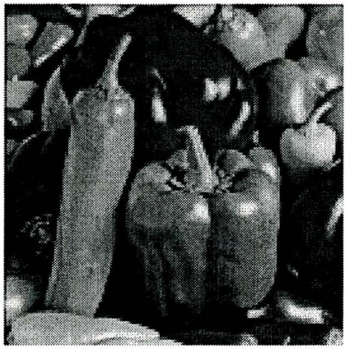
3.3.3 Error Stream Encoding

In step (2) of compression (section 3.2.1), the low resolution error image Ile(n) is encoded to a compressed error stream Sle(n). The encoding is done in three steps shown in figure 3.16.

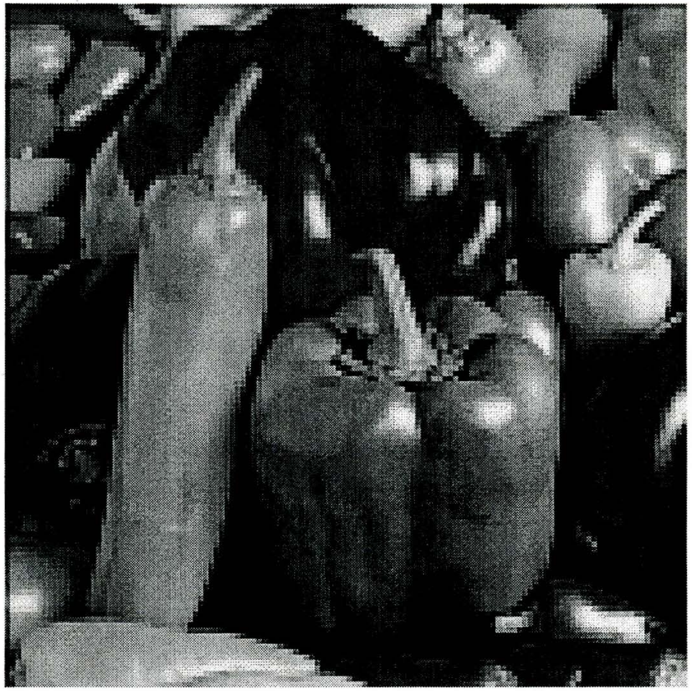
- Step (1), forward MPC transforms the low resolution error image Ile(n) to produce transform coefficients tc(n). This is explained in section 3.3.6.
- Step (2), scalar quantizes transform coefficients tc(n) to produce a symbol stream ss(n). This is explained in section 3.3.5.
- Lastly, in step (3), the symbol stream ss(n) is Huffman coded to produce a compressed error stream Sle(n). Huffman compression was explained and demonstrated in section 2.1.2.

Figure 3.14: This figure demonstrates up-sampling with interpolation. (A) is the original 128x128 image. (B) is a 256x256 image generated so that each pixel's value is a distance weighted average of four immediately neighboring pixel values in (A). (C) is a 512x512 image generated similarly by interpolating (B).

A



B



C

Figure 3.15: This figure demonstrates the benefit of up-sampling with interpolation. The original 512x512 pixel image (A) is down-sampled to 256x256 (B) then up-sampled with interpolation to 512x512 (C). Error image (D) is the difference between images (A) and (C). (D) is mostly black because nearly all it's pixels are either zero or close to zero — the up-sampled image (C) is a good approximation to original image (A).

```
,--------,        ,--------,        ,--------,        ,--------,
| Ile(n) |   ->   | tc(n)  |   ->   | ss(n)  |   ->   | Sle(n) |
`--------'        `--------'        `--------'        `--------'
            (1)                (2)                (3)
```

Figure 3.16: This figure shows the three steps of error stream encoding which is done in step (2) of compression (section 3.2.1). During error stream encoding a low resolution error image Ile(n) is encoded to a compressed error stream Sle(n). Step (1) is the forward MPC transformation. Step (2) is quantization. Step (3) is Huffman coding.

3.3.4 Error Stream Decoding

In step (3) of compression (section 3.2.1) and in step (1) of decompression (section 3.2.2), a compressed error stream Sle(n) is decoded to a low resolution error image Ile(n). The decoding is done in three steps shown in figure 3.17.

- In step (1) Sle(n) is entropy decoded to produce symbols stream ss(n). Entropy decoding is done using Huffman decoding which is explained and demonstrated in section 2.1.2.
- In step (2) the symbol stream ss(n) is scalar dequantized to make approximated transform coefficients tc(n)'. This is explained in section 3.3.5.
- Lastly, in step (3) the approximated coefficients tc(n)' are MPC transformed to produce the low resolution error image Sle(n). This is explained in section 3.3.7.

3.3.5 Scalar Quantization

Step (2) of error stream encoding scalar quantizes transform coefficients tc(n) to produce a symbol stream ss(n) and in step (2) of error stream decoding the symbol stream ss(n) is scalar dequantized to produce approximated transform coefficients tc(n)'. Scalar quantization is introduced in 2.2 and defined in 2.2.2. Its goal is to control the tradeoff between goodness of approximation and higher compression.

```
,--------,        ,--------,        ,--------,        ,--------,
| Sle(n) |   ->   | ss(n)  |   ->   | tc(n)' |   ->   | Ilr(n) |
`________'        `________'        `________'        `________'
             (1)                (2)                (3)
```

Figure 3.17: This figure shows the three steps of error stream decoding. A compressed error stream Sle(n) is decoded to a low resolution error image Ile(n). Step (1) is entropy decoding. Step (2) is dequantization. Step (3) is a backward MPC transformation. Error stream decoding is done in step (3) of compression which is described in section 3.2.1 and in step (1) of decompression which is described in section 3.2.2.

To achieve more compression, quantization bins are made larger. To achieve better approximation, quantization bins are made smaller. All the compression ratios considered in this thesis were achieved by manually adjusting quantization bin size. This has to be done manually because the exact relationship between bin size and compression ratio and quality of image reconstruction is only approximate and is dependent on the particular image being compressed.

3.3.6 Forward MPC Transformation

In step (1) of error stream encoding (section 3.3.3), a low resolution error image Ile(n) is MPC transformed to produce coefficients tc(n). The effect of this transformation is to represent each block x of Ile(n) using the principal coordinates of the cluster for which reconstruction error is least. A simple example of this is described in section 2.2.4 and is shown in figure 2.5. The transformation is carried out in three steps and uses K transformation matrices $\{W_1, W_2, \ldots, W_K\}$ which are explained in section 3.3.8.

- Step (1) cuts the image into non-overlapping blocks.
- Step (2) classifies each block $x \in C_i$ if

$$||P_i x|| = \min_{j=1}^{K} ||P_j x||$$

where

$$P_i = W_i^T W_i$$

- Lastly, step (3) projects each block x

$$y = W_i x$$

using transform matrix W_i of class C_i.

The output of these steps tc(n) is the concatenation of all projection coefficients y's and class membership labels C_i's.

3.3.7 Backward MPC transformation

In step (1) of error stream decoding (section 3.3.4), approximated coefficients tc(n)' are MPC transformed to reconstruct the low resolution error image Sle(n). In tc(n)' each image block x is represented by class label C_i and principal coordinates y. The backward MPC transformation converts each block back into image pixel coordinates as follows

$$x = W_i^T y$$

using transform matrix W_i of class C_i. The origin of W_i's is explained in section 3.2.3.

3.3.8 MPC Training

The goal of MPC training is to find transformation matrices $\{W_1, W_2, \ldots, W_K\}$ for mapping blocks to and from the principal axis of K classes. A simple example of such a mapping is provided in section 2.2.4. The initial W_i's are set to randomly perturbed estimates of the global principal axes which are obtained by applying PCA[1] to randomly sampled image blocks. Training consists of two steps that are repeatedly applied to all image blocks until W_i's converge.

- Step (1) classifies image block $x \in C_i$ if

  $$||P_i x|| = \max_{j=1}^{K} ||P_j x||$$

  where

  $$P_i = W^T{}_i W_i$$

- Step (2) updates transform matrix W_i according to

  $$W_i = W_i + \alpha Z(x, W_i)$$

  where α is a learning rate parameter and $Z(x, W_i)$ is a learning rule that iteratively converges to the M principal components of $x|x \in C_i$.

[1] PCA is explained in section 2.2.2.

MPC training depends on three parameters: block size, number of clusters and number of components. Details of tuning these parameters are considered by [15]. Here we briefly review what each does and how it was set.

- The *block size* is the dimensionality of the MPC input space. Bigger blocks better reduce intra-resolution repetition but need more processing time and exponentially more training examples[2]. At high compression ratios big blocks introduce noticeable block, stair casing and texturing artifacts. To avoid artifacts and to keep MPC training time reasonable, the dimensionality of the input space was set to 64, corresponding to a 8x8 pixel block size.

- The *number of components* retained determines the dimensionality of the MPC output space. More components give better reconstruction in each cluster. Before quantization and Huffman compression are applied, doubling the number of components retained halves the compression ratio. The number of retained components was kept constant at 4.

- The *number of classes* determines how many clusters MPC tries to fit to the data. Doubling the number of classes adds one bit of overhead to each block being coded but improves the reconstruction of image discontinuities such as edges and texture[3]. The number of classes was set by doubling their number until observed performance ceased to improve. For the chest radiographs de-

[2] This known as the *curse of dimensionality.*

[3] Why more clusters improve reconstruction of image discontinuities is explained in section 2.2.2.

scribed in section 4.3 we found that 128 classes worked best for mono-resolution MPC and 16 classes per resolution worked best for MR-MPC.

3.4 Discussion

3.4.1 Similarity to Laplacian Pyramid

The resolution progressive processing of images in MR-MPC is similar to that in Burt and Adelson's Laplacian Pyramid (LP) [10]. In each stage of LP a low pass filtered copy of image is subtracted from the image itself, the low pass filtered image is then down-sampled and the process is repeated. LP achieves compression because error images quantized and stored at each stage have lower variance and entropy. MR-MPC however differs from LP in three significant ways.

- The first difference is that MR-MPC applies the recently developed adaptive transform MPC (see section 2.2.4) to error images before quantizing and lossless coding them. LP on the other hand just quantizes and entropy codes error images.

- The second difference is that MR-MPC passes a residual error image between stages whereas LP passes a down-sampled version of the original image between stages — in LP the residual error is lost. Tracking a residual error image between stages gives MR-MPC lossless reconstruction ability. This is discussed further in section 3.4.2.

- The third difference is that MR-MPC compression and decompression work in the same direction — both process images from low to high resolution — whereas LP compression and decompression work in opposite directions. LP compression processes images from high to low resolution, whereas LP decompression processes images from low to high resolution. The significance of this difference is that unlike LP, MR-MPC can concurrently compress and decompress. Compression need not finish processing an image before progressive decompression can start.

3.4.2 Lossless Compression

The errors introduced by interpolation and lossy MPC coding at lower resolutions in preceding stages propagate and are corrected in successive stages that process images at higher resolutions. This error recovery feature which is missing in LP gives MR-MPC lossless compression ability. If the output of the last compression stage is entropy coded and sent, no information is lost.

3.4.3 Progressive Decompression

Progressive decompression is handy for online interactive browsing since a quickly decoding coarse rendition of an image gives viewers something to look at and the option of early action — for example, to stop transmission if the image is wrong — while detail of progressively finer resolutions is added. Unlike that of MPC, the bit stream generated by MR-MPC is inherently progressive because a coarse rendition of an image is encoded first, followed by *detail corrections* for progressively higher resolutions. As images are decompressed, they initially appear blurry but come to "focus" as more bits arrive. This was illustrated in figure 3.8 from left to right. Unlike that of LP, the MR-MPC compressed bit stream is concurrently progressive. This was mentioned in section 3.4.1 and means that if the compression and decompression tasks are running concurrently — for example at opposite ends of a communications link — progressive decompression of an image can start at the receiving end before progressive compression at the sending end has finished.

3.4.4 Implementation

MR-MPC compression (section 3.2.1), decompression (section 3.2.2) and training (section 3.2.3) logic was implemented in UNIX shell scripts that coordinated the work of 11 program modules written in C. A separate module was dedicated to each of the following functions:

- image down-sampling (section 3.3.1),
- image up-sampling (section 3.3.2),
- MPC training (section 3.2.3),
- MPC forward transform (section 3.2.1),
- MPC inverse transform (section 3.2.2),

- image blocking and deblocking (section 2.2.1),
- scalar quantization and dequantization (section 3.3.5),
- Huffman compression and decompression (section 2.1.2).

Module behavior was controlled using command line arguments and all communication between modules was by means of temporary files on a RAM disk. This implementation allowed reuse of code written by [15] and made debugging easy because interaction between modules was minimized — each module ran sequentially in a separate address space. Another benefit of this design was the potential to pipeline individual MR-MPC stages across multiple CPU's using the UNIX remote procedure call (RPC) subsystem.

3.5 Summary

In this chapter we have discussed the ideas that motivated our multi-resolution extension to MPC, and provided a description of the algorithm, its components and implementation. In the next two chapters we describe how our algorithm was evaluated.

Chapter 4

Numerical Evaluation

4.1 Introduction

This chapter evaluates our multi-resolution extension of MPC by comparing it with MPC and with a multi-resolution wavelet based scheme called Set Partitioning in Hierarchical Trees (SPIHT) [33] which was picked because of its excellent performance [19] and availability of working software[1]. Section 4.2 reviews how lossy compressed images are evaluated. Section 4.3 describes the set of chest radiograph images used and how images were compressed. Section 4.2 presents the results.

[1] A complete description of the SPIHT algorithm along with access to software is available from [33].

4.2 Review

4.2.1 Sum of Squared Error (SSE)

To evaluate lossy image compression we must quantify distortion. The simplest way to do this is to compare compressed images to original images pixel by pixel and add up squared errors — squaring prevents positive and negative errors from canceling. So if Ic is the reconstructed image and Io is the original image, a simple measure of distortion is *sum of squared error*:

$$SSE = \sum_{all\ pixels} (Ic - Io)^2$$

4.2.2 Mean Squared Error (MSE)

The problem with using SSE to measure distortion is that it depends on not only the disparity between the compressed image and original image but also on the number of pixels. For example, the SSE from a 512x512 image will be four times higher than the SSE from a similarly distorted 256x256 image. We can get around this problem by dividing SSE by the number of pixels N to get a measure called *mean squared error*:

$$MSE = E[(Xi - Xr)^2] = SSE/N$$

4.2.3 Peak Signal to Noise Ratio (PSNR)

MSE is better than SSE as a measure of image distortion because it is comparable between images with different number of pixels. MSE however, is not comparable among images represented with a different number of bits per pixel (BPP). For example, if a single pixel image is represented using 2 BPP the maximum pixel error possible is 4 and MSE cannot be higher than 16. In contrast, if this image is represented using 4 BPP, the maximum pixel error possible is 16 and MSE can be as high as 256. The solution to this problem is to normalize MSE by the highest possible pixel error I_{max} as follows:

$$PSNR = 10 * \log_{10}(I_{\text{max}}^2 / MSE)$$

The logarithmically scaled result is measured in decibels (dB) and is called *Peak Signal to Noise Ratio.* I_{max} is the peak signal and MSE is the noise. PSNR is better then SSE and MSE for measuring distortion because it does not depend on the number or representation of pixels.

4.2.4 Picture Quality Scale (PQS)

The measures of distortion considered so far are widely used in the image compression literature. They do not, however, accurately capture the subjective impressions of human observers [18, 8, 25, 35, 22, 25, 23]. Their correlation with perceived image quality is low, about 0.57 [29]. There are two reasons for this. The first is that measures such as PSNR are sensitive to image distortions that humans miss. For example, PSNR will report the adding of a constant to every pixel as a large distortion. For human viewers however, the image has merely been brightened [23]. The second reason is that these distortion measures only account for *random error* which is visible as incremental noise in slow varying image areas and visually masked in active areas. The error prevalent and most visible in lossy compressed images however, is *structured error*, not random error [29]. Human vision is particularly sensitive to errors that cause small edge misalignments (Vernier acuity), errors that look like texture, and errors that are linear such as end of block discontinuities (blocking artifacts) from algorithms that process images in blocks (ie KLT, VQ, MPC, MR-MPC).

Much psychophysical and engineering effort [18, 8, 25, 35, 22, 14, 2, 5] has gone into models of human vision, and recently Picture Quality Scale (PQS) has been proposed and used for objective assessment of image quality [29, 28, 7, 3, 4, 6]. PQS works by quantifying distortion using five factors. Factor one measures the visibility of random error using the CCIR 567-1 noise weighing standard. Factor two measures random error using a model of vision that ignores errors below a threshold and accounts for human luminance and contrast sensitivity using the Weber-Fechner Law and a spatial frequency weighting. Factor three measures blocking artifacts. Factor four measures errors with a high spatial correlation (texture). Factor five measures errors such as edge misalignments in the vicinity of high contrast transitions.

These factors when decorrelated with PCA[2] and combined with regression, mirror human opinion of lossy compressed natural images quite well, with 0.92 correlation [29].

4.3 Method

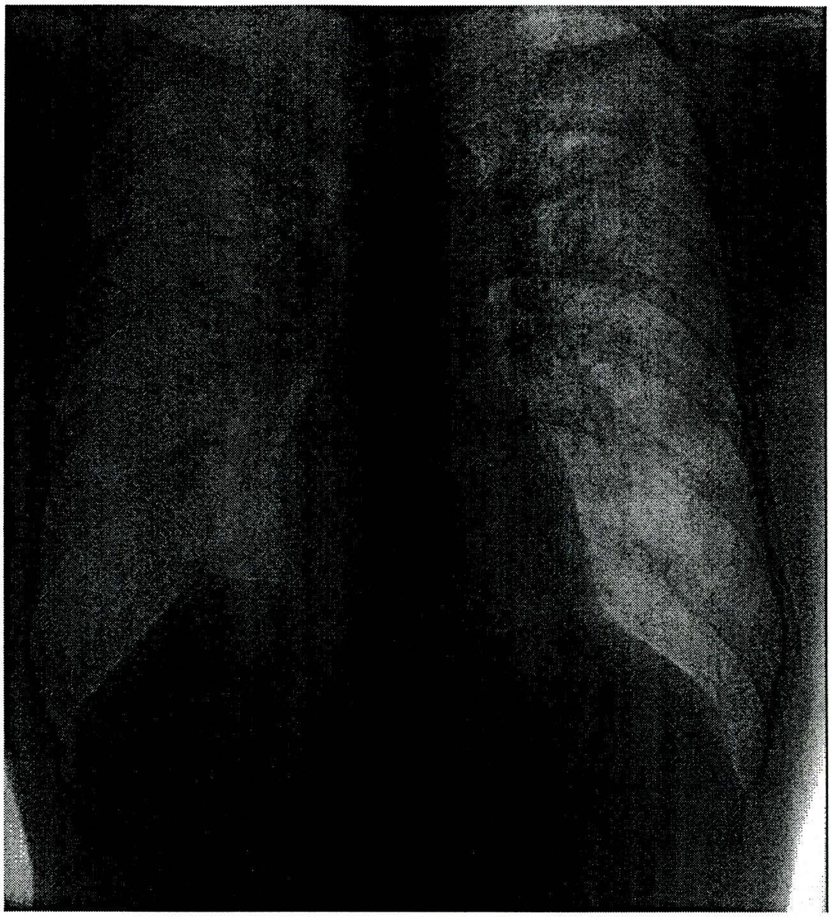
Fifty images were compressed and decompressed using MPC, MR-MPC and SPIHT and then compared to originals. The images were chest radiographs (see figure 4.1) arbitrarily selected from those collected daily in the radiology department of McMaster University Medical Center using a Fuji AC1 computed radiography system [24]. On disk each radiograph was stored as a 2048x2048 pixel image with 16 BPP and occupied 8 Mb of space. However the size of the medically useful image area varied from radiograph to radiograph and only 10 BPP contained significant information. To address these problems images were initially clipped to 10 BPP and all reported compression ratios are calculated relative to 5 Mb per image and represent the mean of all fifty images used for testing.

Prior to compressing the fifty radiograph test set, MR-MPC and MPC were trained, as described in sections 3.2.3 and 3.3.8, on a separate set of nine radiographs. MR-MPC had 16 clusters with 4 components in a three stage (16,4,1) re-sampling scheme for a total of 48 clusters. MPC used 128 clusters with 4 components. Both MPC and MR-MPC used an 8x8 pixel block size.

[2] PCA is explained in section 2.2.2.

Figure 4.1: This figure shows two chest radiograph images collected in the radiology department of McMaster University Medical Center using a Fuji AC1 computed radiography system [24]. On disk each radiograph was stored as a 2048x2048 pixel image with 16 BPP and occupied 8 Mb of space. However the size of the medically useful image area varied from radiograph to radiograph and only 10 BPP contained significant information. For our evaluation fifty such images were compressed and decompressed using MPC, MR-MPC and SPIHT and then compared to originals.

59.2



59.8

4.4 Results

4.4.1 Peak Signal to Noise Ratio (PSNR)

Observed PSNR values for the fifty compressed and decompressed images using the three algorithms are shown in table 4.1 and plotted in figure 4.2. The variability in the observed PSNR values for any given algorithm and bit rate is because some radiographs, for example, those with smaller diagnostic image areas, are easier to compress. Despite this variability however, it's clear from plot in figure 4.2 that, although MPC uses more clusters, use of fewer clusters at multiple resolutions gives MR-MPC a rate-distortion performance that is much closer to that of SPHIT. The difference between all algorithms is greatest at lower bit rates and much less at higher bit rates. For example, to achieve the same average PSNR as MPC at 0.171 BPP, MR-MPC needs only 0.077 BPP. This is 220% better. To achieve the average PSNR of MPC at 0.108 BPP, MR-MPC needs only 0.015 BPP. This is 720% better.

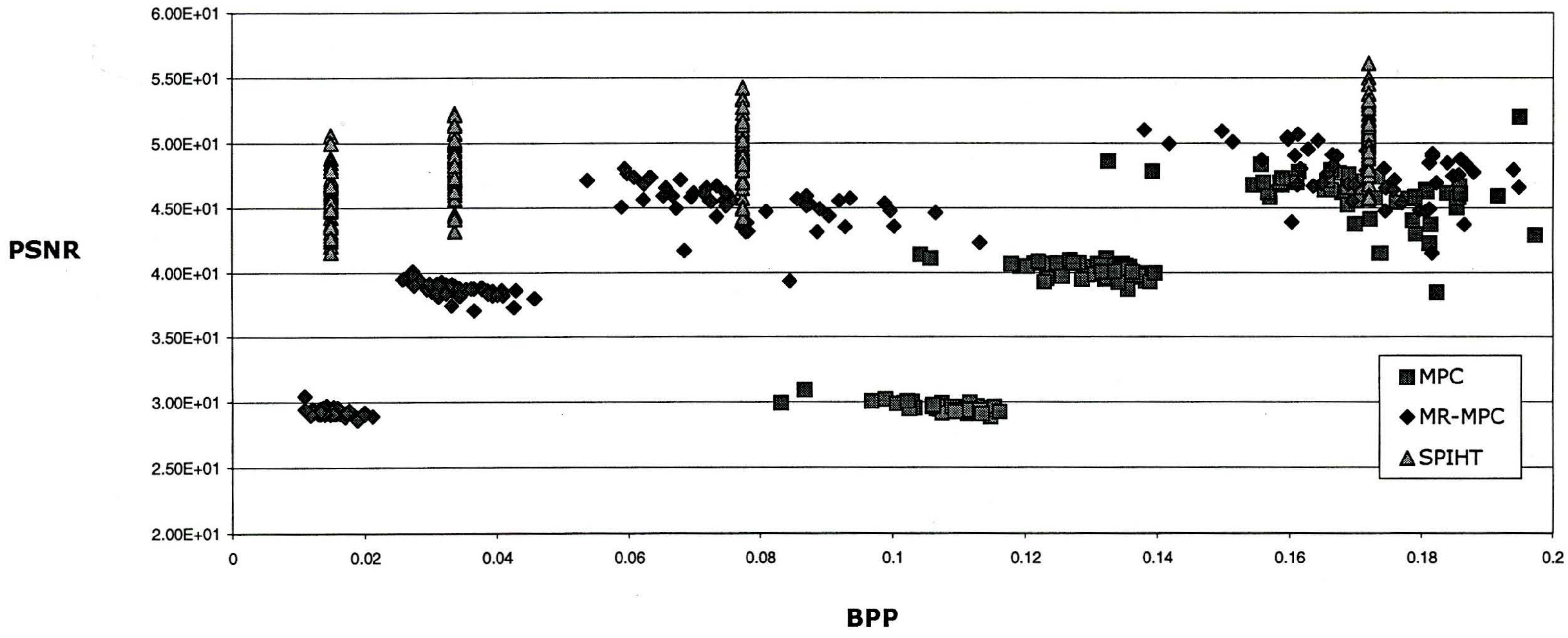
4.4.2 Picture Quality Scale (PQS)

PSNR however, may not be the best performance indicator. As explained in section 4.2.4, PSNR is sensitive to distortions that human viewers miss and is insensitive to structured distortions that human viewers particularly notice. We addressed this in two ways. The first was by asking radiologists for their expert opinion — this is discussed in chapter 5. The second way was to compare the compressed radiographs

Table 4.1: This table shows how observed PSNR values for MPC, MR-MPC, and SPIHT varied with bit rate (BPP). Lower BPP indicates more compression. Higher values of PSNR indicate better reconstruction.

MPC				MR-MPC				SPIHT			
BPP	PSNR	BPP	PSNR	BPP	PSNR	BPP	PSNR	BPP	PSNR	BPP	PSNR
8.323E-02	2.990E+01	1.393E-01	4.784E+01	1.504E-02	2.935E+01	7.797E-02	4.381E+01	1.484E-02	4.608E+01	7.734E-02	5.027E+01
8.675E-02	3.092E+01	1.397E-01	3.992E+01	1.524E-02	2.909E+01	7.821E-02	4.317E+01	1.484E-02	4.604E+01	7.734E-02	5.158E+01
9.690E-02	3.006E+01	1.546E-01	4.675E+01	1.525E-02	2.962E+01	8.091E-02	4.470E+01	1.484E-02	4.587E+01	7.734E-02	4.853E+01
9.897E-02	3.019E+01	1.558E-01	4.833E+01	1.554E-02	2.911E+01	8.448E-02	3.930E+01	1.484E-02	4.627E+01	7.734E-02	5.017E+01
1.007E-01	2.985E+01	1.561E-01	4.694E+01	1.564E-02	2.912E+01	8.564E-02	4.566E+01	1.484E-02	4.206E+01	7.734E-02	4.923E+01
1.018E-01	2.993E+01	1.568E-01	4.620E+01	1.580E-02	2.957E+01	8.696E-02	4.586E+01	1.484E-02	4.686E+01	7.734E-02	5.075E+01
1.024E-01	3.005E+01	1.570E-01	4.579E+01	1.621E-02	2.910E+01	8.706E-02	4.513E+01	1.484E-02	4.548E+01	7.734E-02	4.897E+01
1.026E-01	2.946E+01	1.587E-01	4.695E+01	1.655E-02	2.910E+01	8.708E-02	4.590E+01	1.484E-02	4.285E+01	7.734E-02	5.163E+01
1.030E-01	3.002E+01	1.589E-01	4.677E+01	1.686E-02	2.909E+01	8.752E-02	4.529E+01	1.484E-02	4.581E+01	7.734E-02	4.702E+01
1.031E-01	2.958E+01	1.590E-01	4.728E+01	1.693E-02	2.886E+01	8.870E-02	4.313E+01	1.484E-02	4.243E+01	7.734E-02	5.424E+01
1.035E-01	2.951E+01	1.612E-01	4.696E+01	1.695E-02	2.917E+01	8.905E-02	4.488E+01	1.484E-02	4.707E+01	7.734E-02	4.847E+01
1.043E-01	4.136E+01	1.614E-01	4.777E+01	1.696E-02	2.897E+01	9.049E-02	4.438E+01	1.484E-02	4.509E+01	7.734E-02	4.571E+01
1.059E-01	4.108E+01	1.653E-01	4.638E+01	1.723E-02	2.913E+01	9.199E-02	4.553E+01	1.484E-02	4.620E+01	7.734E-02	5.101E+01
1.061E-01	2.961E+01	1.658E-01	4.710E+01	1.732E-02	2.924E+01	9.290E-02	4.351E+01	1.484E-02	4.703E+01	7.734E-02	4.944E+01
1.062E-01	2.975E+01	1.664E-01	4.792E+01	1.762E-02	2.934E+01	9.374E-02	4.569E+01	1.484E-02	4.150E+01	7.734E-02	5.047E+01
1.063E-01	2.968E+01	1.664E-01	4.639E+01	1.878E-02	2.879E+01	9.887E-02	4.533E+01	1.484E-02	4.360E+01	7.734E-02	4.890E+01
1.067E-01	2.950E+01	1.676E-01	4.769E+01	1.884E-02	2.861E+01	9.972E-02	4.474E+01	1.484E-02	4.428E+01	7.734E-02	4.419E+01
1.068E-01	2.966E+01	1.680E-01	4.620E+01	1.989E-02	2.905E+01	1.003E-01	4.355E+01	1.484E-02	4.312E+01	7.734E-02	5.228E+01
1.071E-01	2.943E+01	1.686E-01	4.523E+01	1.993E-02	2.922E+01	1.066E-01	4.462E+01	1.484E-02	4.851E+01	7.734E-02	4.957E+01
1.075E-01	2.914E+01	1.686E-01	4.666E+01	2.113E-02	2.891E+01	1.132E-01	4.228E+01	1.484E-02	4.580E+01	7.734E-02	4.799E+01
1.075E-01	2.993E+01	1.690E-01	4.757E+01	2.580E-02	3.947E+01	1.381E-01	5.102E+01	1.484E-02	4.557E+01	7.734E-02	4.646E+01
1.090E-01	2.963E+01	1.700E-01	4.374E+01	2.599E-02	3.949E+01	1.420E-01	4.996E+01	1.484E-02	4.474E+01	7.734E-02	4.918E+01
1.095E-01	2.925E+01	1.702E-01	4.552E+01	2.696E-02	3.935E+01	1.498E-01	5.091E+01	1.484E-02	4.668E+01	7.734E-02	4.493E+01
1.095E-01	2.937E+01	1.706E-01	4.628E+01	2.705E-02	3.922E+01	1.515E-01	5.008E+01	1.484E-02	4.790E+01	7.734E-02	4.993E+01
1.096E-01	2.929E+01	1.710E-01	4.627E+01	2.715E-02	3.981E+01	1.559E-01	4.871E+01	1.484E-02	4.357E+01	7.734E-02	5.016E+01
1.097E-01	2.937E+01	1.714E-01	4.708E+01	2.727E-02	3.948E+01	1.598E-01	5.041E+01	1.484E-02	4.652E+01	7.734E-02	4.691E+01
1.097E-01	2.961E+01	1.722E-01	4.413E+01	2.732E-02	4.009E+01	1.598E-01	5.025E+01	1.484E-02	4.347E+01	7.734E-02	4.786E+01
1.099E-01	2.943E+01	1.730E-01	4.574E+01	2.745E-02	3.898E+01	1.604E-01	4.388E+01	1.484E-02	4.264E+01	7.734E-02	4.884E+01
1.101E-01	2.922E+01	1.734E-01	4.739E+01	2.845E-02	3.936E+01	1.609E-01	4.903E+01	1.484E-02	4.722E+01	7.734E-02	5.275E+01
1.112E-01	2.940E+01	1.736E-01	4.147E+01	2.950E-02	3.868E+01	1.613E-01	4.678E+01	1.484E-02	4.677E+01	7.734E-02	4.894E+01
1.113E-01	2.925E+01	1.738E-01	4.147E+01	2.985E-02	3.910E+01	1.613E-01	5.064E+01	1.484E-02	4.816E+01	7.734E-02	4.834E+01
1.113E-01	2.911E+01	1.762E-01	4.541E+01	3.017E-02	3.855E+01	1.617E-01	4.800E+01	1.484E-02	4.665E+01	7.734E-02	4.955E+01
1.114E-01	2.960E+01	1.780E-01	4.568E+01	3.075E-02	3.907E+01	1.628E-01	4.948E+01	1.484E-02	4.788E+01	7.734E-02	4.991E+01
1.116E-01	2.952E+01	1.787E-01	4.403E+01	3.084E-02	3.840E+01	1.636E-01	4.688E+01	1.484E-02	4.524E+01	7.734E-02	5.347E+01
1.117E-01	2.949E+01	1.791E-01	4.585E+01	3.094E-02	3.911E+01	1.644E-01	5.017E+01	1.484E-02	4.603E+01	7.734E-02	4.840E+01
1.117E-01	2.997E+01	1.792E-01	4.295E+01	3.106E-02	3.814E+01	1.649E-01	4.685E+01	1.484E-02	4.491E+01	7.734E-02	5.175E+01
1.119E-01	2.949E+01	1.806E-01	4.636E+01	3.123E-02	3.908E+01	1.660E-01	4.774E+01	1.484E-02	5.057E+01	7.734E-02	4.917E+01
1.121E-01	2.944E+01	1.809E-01	4.622E+01	3.127E-02	3.813E+01	1.666E-01	4.905E+01	1.484E-02	4.684E+01	7.734E-02	4.878E+01
1.126E-01	2.954E+01	1.813E-01	4.222E+01	3.168E-02	3.927E+01	1.666E-01	4.827E+01	1.484E-02	4.704E+01	7.734E-02	5.097E+01
1.128E-01	2.965E+01	1.815E-01	4.367E+01	3.227E-02	3.906E+01	1.672E-01	4.897E+01	1.484E-02	4.580E+01	7.734E-02	4.721E+01
1.129E-01	2.920E+01	1.823E-01	3.844E+01	3.232E-02	3.832E+01	1.690E-01	4.682E+01	1.484E-02	4.545E+01	7.734E-02	4.504E+01
1.132E-01	2.940E+01	1.839E-01	4.613E+01	3.262E-02	3.891E+01	1.696E-01	4.551E+01	1.484E-02	4.473E+01	7.734E-02	5.329E+01
1.135E-01	2.951E+01	1.854E-01	4.500E+01	3.287E-02	3.893E+01	1.699E-01	4.678E+01	1.484E-02	4.380E+01	7.734E-02	5.160E+01
1.135E-01	2.908E+01	1.854E-01	4.574E+01	3.288E-02	3.884E+01	1.716E-01	4.789E+01	1.484E-02	4.350E+01	7.735E-02	5.141E+01
1.138E-01	2.921E+01	1.857E-01	4.579E+01	3.289E-02	3.897E+01	1.717E-01	4.943E+01	1.484E-02	4.369E+01	7.735E-02	4.868E+01
1.141E-01	2.954E+01	1.857E-01	4.588E+01	3.317E-02	3.744E+01	1.719E-01	4.751E+01	1.484E-02	5.000E+01	7.735E-02	4.926E+01
1.141E-01	2.938E+01	1.858E-01	4.667E+01	3.319E-02	3.894E+01	1.743E-01	4.800E+01	1.484E-02	4.512E+01	7.735E-02	4.839E+01
1.142E-01	2.945E+01	1.860E-01	4.608E+01	3.321E-02	3.864E+01	1.746E-01	4.474E+01	1.484E-02	4.548E+01	7.735E-02	4.911E+01
1.146E-01	2.939E+01	1.917E-01	4.588E+01	3.323E-02	3.906E+01	1.746E-01	4.651E+01	1.484E-02	4.485E+01	7.735E-02	5.132E+01
1.148E-01	2.886E+01	1.950E-01	5.202E+01	3.361E-02	3.895E+01	1.758E-01	4.638E+01	1.485E-02	4.476E+01	7.735E-02	4.486E+01
1.154E-01	2.965E+01	1.973E-01	4.288E+01	3.421E-02	3.877E+01	1.759E-01	4.716E+01	3.359E-02	4.798E+01	1.720E-01	5.130E+01
1.162E-01	2.922E+01	2.081E-01	5.105E+01	3.431E-02	3.879E+01	1.771E-01	4.536E+01	3.359E-02	4.858E+01	1.720E-01	5.188E+01
1.179E-01	4.064E+01	2.246E-01	4.890E+01	3.435E-02	3.814E+01	1.797E-01	4.481E+01	3.359E-02	4.956E+01	1.720E-01	5.318E+01
1.193E-01	4.045E+01	2.247E-01	5.012E+01	3.531E-02	3.868E+01	1.807E-01	4.470E+01	3.359E-02	4.721E+01	1.720E-01	5.224E+01
1.204E-01	4.045E+01	2.294E-01	5.107E+01	3.613E-02	3.871E+01	1.812E-01	4.491E+01	3.359E-02	4.872E+01	1.720E-01	5.055E+01
1.215E-01	4.068E+01	2.302E-01	5.245E+01	3.654E-02	3.705E+01	1.813E-01	4.846E+01	3.359E-02	4.800E+01	1.720E-01	5.321E+01
1.221E-01	4.083E+01	2.316E-01	4.901E+01	3.654E-02	3.873E+01	1.816E-01	4.148E+01	3.359E-02	4.864E+01	1.720E-01	4.994E+01
1.230E-01	3.924E+01	2.320E-01	4.982E+01	3.770E-02	3.881E+01	1.817E-01	4.915E+01	3.359E-02	5.223E+01	1.720E-01	5.219E+01
1.234E-01	3.947E+01	2.358E-01	5.033E+01	3.785E-02	3.868E+01	1.817E-01	4.898E+01	3.359E-02	4.598E+01	1.720E-01	5.325E+01
1.234E-01	4.065E+01	2.361E-01	5.013E+01	3.827E-02	3.858E+01	1.822E-01	4.692E+01	3.359E-02	4.562E+01	1.720E-01	5.097E+01
1.249E-01	4.068E+01	2.372E-01	5.075E+01	3.857E-02	3.860E+01	1.840E-01	4.846E+01	3.359E-02	5.147E+01	1.720E-01	5.037E+01
1.258E-01	3.967E+01	2.376E-01	5.017E+01	3.869E-02	3.835E+01	1.848E-01	4.742E+01	3.359E-02	4.736E+01	1.720E-01	4.810E+01
1.269E-01	4.093E+01	2.384E-01	5.128E+01	3.929E-02	3.818E+01	1.857E-01	4.755E+01	3.359E-02	4.450E+01	1.720E-01	5.067E+01
1.274E-01	4.069E+01	2.393E-01	5.158E+01	3.936E-02	3.852E+01	1.860E-01	4.872E+01	3.359E-02	4.906E+01	1.720E-01	4.598E+01
1.282E-01	4.076E+01	2.439E-01	5.034E+01	4.010E-02	3.826E+01	1.865E-01	4.368E+01	3.359E-02	4.787E+01	1.720E-01	4.750E+01
1.286E-01	3.945E+01	2.449E-01	4.591E+01	4.078E-02	3.857E+01	1.868E-01	4.836E+01	3.359E-02	4.876E+01	1.720E-01	5.451E+01
1.289E-01	4.039E+01	2.450E-01	4.615E+01	4.092E-02	3.820E+01	1.880E-01	4.774E+01	3.359E-02	4.760E+01	1.720E-01	4.986E+01
1.295E-01	3.977E+01	2.461E-01	5.237E+01	4.257E-02	3.728E+01	1.940E-01	4.793E+01	3.359E-02	5.000E+01	1.720E-01	4.585E+01
1.306E-01	3.981E+01	2.461E-01	4.928E+01	4.282E-02	3.856E+01	1.949E-01	4.657E+01	3.359E-02	4.717E+01	1.720E-01	4.577E+01
1.310E-01	4.064E+01	2.462E-01	5.020E+01	4.569E-02	3.795E+01	2.006E-01	4.383E+01	3.359E-02	4.314E+01	1.720E-01	5.119E+01
1.315E-01	4.056E+01	2.479E-01	4.865E+01	5.366E-02	4.711E+01	2.033E-01	5.163E+01	3.359E-02	4.649E+01	1.720E-01	5.142E+01
1.317E-01	4.003E+01	2.480E-01	4.714E+01	5.880E-02	4.505E+01	2.240E-01	4.974E+01	3.359E-02	4.938E+01	1.720E-01	5.380E+01
1.322E-01	3.944E+01	2.492E-01	4.533E+01	5.933E-02	4.801E+01	2.248E-01	5.049E+01	3.359E-02	4.651E+01	1.720E-01	5.324E+01
1.323E-01	4.049E+01	2.505E-01	4.636E+01	5.969E-02	4.763E+01	2.333E-01	5.183E+01	3.359E-02	4.447E+01	1.720E-01	5.173E+01
1.323E-01	4.106E+01	2.507E-01	4.898E+01	6.076E-02	4.732E+01	2.349E-01	4.818E+01	3.359E-02	4.671E+01	1.720E-01	5.497E+01
1.324E-01	3.956E+01	2.519E-01	4.749E+01	6.216E-02	4.569E+01	2.364E-01	4.299E+01	3.359E-02	4.561E+01	1.720E-01	4.816E+01
1.326E-01	4.657E+01	2.520E-01	5.063E+01	6.227E-02	4.683E+01	2.369E-01	4.761E+01	3.359E-02	5.217E+01	1.720E-01	5.609E+01
1.336E-01	3.986E+01	2.522E-01	5.134E+01	6.302E-02	4.732E+01	2.372E-01	4.787E+01	3.359E-02	4.835E+01	1.720E-01	4.683E+01
1.336E-01	4.005E+01	2.525E-01	5.070E+01	6.324E-02	4.733E+01	2.378E-01	4.858E+01	3.359E-02	4.435E+01	1.720E-01	5.293E+01
1.337E-01	4.039E+01	2.528E-01	4.799E+01	6.515E-02	4.590E+01	2.386E-01	4.951E+01	3.359E-02	4.413E+01	1.720E-01	5.263E+01
1.341E-01	4.031E+01	2.530E-01	4.541E+01	6.557E-02	4.651E+01	2.388E-01	4.796E+01	3.359E-02	4.694E+01	1.720E-01	5.101E+01
1.341E-01	3.918E+01	2.531E-01	4.864E+01	6.662E-02	4.589E+01	2.399E-01	4.980E+01	3.359E-02	4.623E+01	1.720E-01	5.043E+01
1.344E-01	3.972E+01	2.542E-01	5.152E+01	6.713E-02	4.496E+01	2.405E-01	5.130E+01	3.359E-02	4.709E+01	1.720E-01	5.030E+01
1.345E-01	4.028E+01	2.544E-01	5.270E+01	6.786E-02	4.714E+01	2.416E-01	4.990E+01	3.359E-02	4.791E+01	1.720E-01	4.565E+01
1.347E-01	4.064E+01	2.546E-01	4.955E+01	6.840E-02	4.168E+01	2.418E-01	4.699E+01	3.359E-02	4.729E+01	1.720E-01	5.076E+01
1.353E-01	3.975E+01	2.591E-01	4.864E+01	6.947E-02	4.581E+01	2.439E-01	4.607E+01	3.359E-02	5.023E+01	1.720E-01	5.373E+01
1.353E-01	4.049E+01	2.608E-01	4.895E+01	6.979E-02	4.613E+01	2.450E-01	4.720E+01	3.359E-02	4.939E+01	1.720E-01	5.265E+01
1.356E-01	3.866E+01	2.609E-01	4.948E+01	7.148E-02	4.610E+01	2.454E-01	4.834E+01	3.359E-02	4.760E+01	1.720E-01	4.959E+01
1.357E-01	4.024E+01	2.610E-01	4.679E+01	7.178E-02	4.653E+01	2.462E-01	5.103E+01	3.359E-02	5.069E+01	1.720E-01	4.977E+01
1.359E-01	4.042E+01	2.615E-01	4.867E+01	7.233E-02	4.552E+01	2.481E-01	4.915E+01	3.359E-02	4.984E+01	1.720E-01	4.931E+01
1.360E-01	4.011E+01	2.633E-01	4.722E+01	7.252E-02	4.548E+01	2.492E-01	4.880E+01	3.359E-02	4.712E+01	1.720E-01	4.993E+01
1.361E-01	3.966E+01	2.646E-01	4.296E+01	7.330E-02	4.666E+01	2.510E-01	4.704E+01	3.359E-02	4.741E+01	1.720E-01	5.082E+01
1.363E-01	4.003E+01	2.655E-01	5.036E+01	7.333E-02	4.432E+01	2.517E-01	4.708E+01	3.359E-02	4.697E+01	1.720E-01	5.139E+01
1.367E-01	3.957E+01	2.658E-01	4.527E+01	7.446E-02	4.556E+01	2.523E-01	4.818E+01	3.359E-02	5.083E+01	1.720E-01	5.020E+01
1.370E-01	3.980E+01	2.663E-01	4.934E+01	7.474E-02	4.511E+01	2.524E-01	4.696E+01	3.359E-02	4.776E+01	1.720E-01	5.008E+01
1.372E-01	3.971E+01	2.667E-01	4.870E+01	7.474E-02	4.612E+01	2.532E-01	4.535E+01	3.359E-02	5.015E+01	1.720E-01	5.071E+01
1.374E-01	3.968E+01	2.669E-01	4.937E+01	7.489E-02	4.604E+01	2.538E-01	4.787E+01	3.359E-02	4.904E+01	1.720E-01	5.181E+01
1.380E-01	3.995E+01	2.671E-01	4.894E+01	7.558E-02	4.557E+01	2.554E-01	4.497E+01	3.359E-02	5.133E+01	1.720E-01	4.792E+01
1.384E-01	3.931E+01	2.686E-01	4.877E+01	7.721E-02	4.350E+01	2.562E-01	4.524E+01	3.360E-02	4.825E+01	1.720E-01	4.879E+01
1.389E-01	3.927E+01	2.722E-01	4.911E+01	7.781E-02	4.312E+01	2.577E-01	4.638E+01	3.360E-02	4.659E+01	1.720E-01	4.902E+01

Figure 4.2: This plot shows how observed PSNR values varied with bit rate (BPP) and is based on data in table 4.1. Each point corresponds to a single compressed radiograph. Lower BPP indicates more compression. Higher values of PSNR indicate better reconstruction.


PSNR
6.00E+01
5.50E+01
5.00E+01
4.50E+01
4.00E+01
3.50E+01
3.00E+01
2.50E+01
2.00E+01
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2
BPP
■ MPC
◆ MR-MPC
▲ SPIHT

62.5

to the originals again, this time using the five PQS distortion factors from section 4.2.4. Our PQS results are tabulated in table 4.2 and plotted in figures 4.3, 4.4, 4.5, 4.6, and 4.7.

- The plot in figure 4.3 shows PQS factor one (F1) which measures visibility of random error using the CCIR 567-1 noise weighing standard. F1 closely mirrors PSNR — random error is least visible in SPIHT and MR-MPC compressed images and most visible in MPC compressed images.

- The plot in figure 4.4 shows PQS factor two (F2) which measures random error using a model of vision that ignores errors below a threshold and accounts for human luminance and contrast sensitivity using the Weber-Fechner Law and a spatial frequency weighting. Compared to PSNR and F1, F2 better accounts for human perception of random error. Nevertheless, it mirrors PSNR and F1 — random error is least visible in SPIHT and MR-MPC compressed images, and most visible in MPC compressed images.

- The plot in figure 4.5 shows PQS factor three (F3) which measures blocking artifacts. The F3 results are similar to those of PSNR, F1 and F2 — blocking artifacts are least visible in SPIHT and MR-MPC compressed images, and most visible in MPC compressed images.

- The plot in figure 4.6 shows PQS factor four (F4) which measures error with high spatial correlation (texture). The observed F4 results are similar to those of

PSNR, F1, F2 and F3 — errors with high spatial correlation are least visible in SPIHT and MR-MPC compressed images, and most visible in MPC compressed images.

- The plot in figure 4.7 shows PQS factor five (F5) which measures errors such as edge misalignments in vicinity of high contrast transitions. The observed PQS F5 results are similar to those of PSNR, F1, F2, F3 and F4 — edge misalignment errors are least visible in SPIHT and MR-MPC compressed images, and most visible in MPC compressed images.

4.5 Summary

In this chapter we have evaluated our multi-resolution extension of MPC by comparing it with MPC and SPIHT. Fifty chest radiographs were compressed and compared to originals with PSNR and PQS. Using these images we demonstrated that our multi-resolution extension of MPC can achieve rate distortion performance that is 220% to 720% better than MPC and much closer to that of SPIHT. For all numerical performance measures SPIHT outperformed MPC and MR-MPC by a considerable margin, with one exception. For PQS F5, a measure of visible edge misalignments, the performance of MR-MPC strongly overlapped that of SPIHT at all compression ratios. This reveals a potential advantage of MR-MPC over SPIHT in being able to reconstruct edges better. It remains to be seen whether this will translate to a

Table 4.2: This table shows how observed PQS distortion factors for MPC, MR-MPC, and SPIHT varied with bit rate (BPP). Lower BPP indicates more compression. Higher values of lower PQS distortion factor values indicate better reconstruction. Factors are plotted in figures 4.3 to 4.7.

MPC						MR-MPC						SPIHT					
BPP	F1	F2	F3	F4	F5	BPP	F1	F2	F3	F4	F5	BPP	F1	F2	F3	F4	F5
8.32E-02	8.36E-03	4.19E-03	2.20E+00	7.03E+00	1.26E+02	1.50E-02	2.57E-03	2.02E-03	3.11E+00	9.66E+00	1.41E+02	1.48E-02	1.53E-05	1.96E-04	2.99E-01	5.52E+00	4.27E+01
8.68E-02	5.53E-03	1.10E-03	1.01E+00	5.86E+00	9.02E+01	1.52E-02	2.84E-03	1.96E-03	3.33E+00	1.00E+01	1.81E+02	1.48E-02	1.67E-05	3.53E-04	4.02E-01	5.66E+00	8.20E+01
9.69E-02	4.26E-03	1.48E-03	1.51E+00	6.49E+00	9.88E+01	1.52E-02	3.07E-03	3.74E-03	4.18E+00	9.79E+00	1.78E+02	1.48E-02	1.53E-05	4.08E-04	4.26E-01	5.74E+00	6.51E+01
9.90E-02	3.42E-03	1.05E-03	1.62E+00	6.49E+00	1.08E+02	1.55E-02	3.47E-03	2.90E-03	3.79E+00	1.01E+01	1.67E+02	1.48E-02	2.06E-05	4.08E-04	4.10E-01	5.71E+00	7.38E+01
1.01E-01	2.50E-03	1.23E-03	2.15E+00	8.88E+00	2.00E+02	1.56E-02	3.43E-03	2.58E-03	2.93E+00	1.02E+01	1.60E+02	1.48E-02	4.01E-05	4.32E-04	5.19E-01	6.90E+00	6.01E+01
1.02E-01	3.77E-03	9.89E-04	1.50E+00	6.45E+00	1.31E+02	1.58E-02	2.13E-03	1.45E-03	2.64E+00	9.14E+00	2.13E+02	1.48E-02	1.43E-05	3.48E-04	4.34E-01	5.61E+00	8.07E+01
1.02E-01	2.39E-03	7.91E-04	1.31E+00	7.98E+00	1.04E+02	1.62E-02	3.24E-03	2.81E-03	3.57E+00	1.06E+01	2.01E+02	1.48E-02	1.57E-05	2.19E-04	3.25E-01	5.56E+00	2.25E+01
1.03E-01	4.73E-03	1.02E-03	1.02E+00	6.28E+00	9.41E+01	1.66E-02	3.10E-03	3.16E-03	3.88E+00	1.05E+01	1.78E+02	1.48E-02	3.62E-05	4.17E-04	4.82E-01	6.60E+00	5.11E+01
1.03E-01	3.42E-03	7.81E-04	9.20E-01	6.27E+00	1.04E+02	1.69E-02	2.57E-03	2.28E-03	3.47E+00	1.11E+01	1.68E+02	1.48E-02	1.23E-05	1.31E-04	2.28E-01	5.18E+00	3.03E+01
1.03E-01	3.32E-03	8.87E-04	1.31E+00	7.94E+00	8.85E+01	1.69E-02	3.23E-03	2.71E-03	3.63E+00	1.13E+01	1.86E+02	1.48E-02	2.89E-05	3.67E-04	4.36E-01	6.66E+00	4.99E+01
1.03E-01	5.65E-03	1.33E-03	1.17E+00	6.76E+00	1.08E+02	1.70E-02	2.98E-03	2.02E-03	3.11E+00	1.09E+01	1.74E+02	1.48E-02	1.53E-05	2.46E-04	2.84E-01	5.14E+00	6.91E+01
1.04E-01	2.37E-04	1.30E-04	2.28E-01	4.61E+00	3.97E+01	1.70E-02	3.12E-03	2.36E-03	3.51E+00	1.11E+01	1.80E+02	1.48E-02	2.25E-05	2.74E-04	3.52E-01	5.86E+00	1.01E+02
1.06E-01	2.74E-04	5.52E-04	4.61E-01	5.47E+00	4.29E+01	1.72E-02	3.05E-03	2.49E-03	3.14E+00	1.02E+01	7.64E+01	1.48E-02	1.91E-05	4.52E-04	4.09E-01	5.93E+00	5.04E+01
1.06E-01	7.01E-03	1.30E-03	1.04E+00	5.83E+00	1.10E+02	1.73E-02	2.68E-03	1.53E-03	2.43E+00	9.90E+00	1.11E+02	1.48E-02	1.67E-05	3.90E-04	4.05E-01	5.66E+00	6.36E+01
1.06E-01	3.58E-03	6.30E-04	8.24E-01	5.88E+00	1.02E+02	1.76E-02	2.29E-03	2.17E-03	3.79E+00	1.08E+01	1.56E+02	1.48E-02	3.24E-05	3.25E-04	4.51E-01	6.54E+00	7.24E+01
1.06E-01	4.24E-03	9.24E-04	1.09E+00	6.33E+00	1.05E+02	1.88E-02	2.52E-03	2.34E-03	4.05E+00	1.15E+01	8.52E+01	1.48E-02	3.77E-05	4.20E-04	5.68E-01	7.11E+00	6.67E+01
1.07E-01	3.31E-03	1.02E-03	1.65E+00	6.59E+00	1.31E+02	1.88E-02	2.92E-03	2.61E-03	3.51E+00	1.12E+01	1.21E+02	1.48E-02	2.55E-05	3.40E-04	3.66E-01	5.93E+00	3.59E+01
1.07E-01	3.02E-03	1.27E-03	1.43E+00	7.20E+00	9.66E+01	1.99E-02	2.41E-03	3.12E-03	3.72E+00	1.09E+01	1.32E+02	1.48E-02	3.22E-05	3.75E-04	5.12E-01	6.64E+00	6.14E+01
1.07E-01	3.65E-03	7.77E-04	9.86E-01	6.95E+00	8.66E+01	1.99E-02	2.45E-03	3.36E-03	5.10E+00	1.23E+01	1.71E+02	1.48E-02	8.36E-06	1.44E-04	2.59E-01	4.71E+00	3.42E+01
1.08E-01	4.28E-03	6.92E-04	1.20E+00	6.51E+00	1.63E+02	2.11E-02	2.15E-03	3.07E-03	3.88E+00	1.18E+01	1.52E+02	1.48E-02	1.21E-05	1.50E-04	2.51E-01	5.28E+00	5.08E+01
1.08E-01	2.61E-03	1.04E-03	1.87E+00	8.82E+00	1.08E+02	2.58E-02	1.41E-04	6.99E-04	9.15E-01	7.01E+00	1.60E+02	1.48E-02	2.01E-05	3.91E-04	4.93E-01	6.02E+00	1.03E+02
1.09E-01	3.30E-03	9.12E-04	1.68E+00	6.92E+00	1.19E+02	2.60E-02	2.93E-04	8.38E-04	9.77E-01	7.18E+00	1.43E+02	1.48E-02	1.22E-05	2.64E-04	3.63E-01	5.50E+00	7.04E+01
1.09E-01	3.38E-03	1.11E-03	1.38E+00	6.92E+00	1.14E+02	2.70E-02	1.39E-04	5.66E-04	8.74E-01	6.84E+00	1.01E+02	1.48E-02	1.09E-05	2.65E-04	4.36E-01	5.24E+00	6.18E+01
1.09E-01	3.56E-03	7.08E-04	9.51E-01	5.94E+00	1.04E+02	2.71E-02	1.27E-04	4.63E-04	7.52E-01	6.70E+00	1.46E+02	1.48E-02	8.82E-06	1.98E-04	3.30E-01	5.02E+00	8.47E+01
1.10E-01	2.65E-03	7.85E-04	1.40E+00	8.53E+00	9.47E+01	2.71E-02	3.24E-04	1.85E-03	1.37E+00	7.77E+00	1.26E+02	1.48E-02	1.97E-05	4.00E-04	5.25E-01	6.22E+00	6.98E+01
1.10E-01	3.80E-03	6.38E-04	8.11E-01	6.17E+00	6.78E+01	2.73E-02	1.55E-04	6.96E-04	9.71E-01	7.05E+00	1.73E+02	1.48E-02	1.18E-05	2.59E-04	4.10E-01	5.27E+00	5.95E+01
1.10E-01	4.02E-03	7.06E-04	7.08E-01	6.25E+00	7.17E+01	2.73E-02	2.50E-04	9.87E-04	7.48E-01	6.43E+00	1.25E+02	1.48E-02	1.03E-05	2.14E-04	3.52E-01	5.24E+00	7.61E+01
1.10E-01	3.15E-03	1.54E-03	1.37E+00	7.09E+00	1.52E+02	2.75E-02	1.54E-04	4.47E-04	6.62E-01	6.67E+00	1.33E+02	1.48E-02	2.16E-05	2.91E-04	4.51E-01	6.05E+00	7.32E+01
1.10E-01	3.89E-03	7.83E-04	1.16E+00	5.99E+00	8.38E+01	2.84E-02	1.45E-04	5.84E-04	8.68E-01	6.94E+00	1.33E+02	1.48E-02	8.53E-06	2.17E-04	3.35E-01	5.06E+00	7.82E+01
1.11E-01	3.34E-03	8.45E-04	1.42E+00	7.53E+00	1.19E+02	2.95E-02	1.26E-04	4.51E-04	7.75E-01	6.63E+00	1.33E+02	1.48E-02	1.05E-05	2.15E-04	3.43E-01	5.15E+00	6.52E+01
1.11E-01	3.60E-03	1.53E-03	1.92E+00	8.96E+00	1.17E+02	2.98E-02	1.27E-04	4.58E-04	7.95E-01	6.96E+00	1.19E+02	1.48E-02	2.25E-05	4.38E-04	3.70E-01	5.38E+00	6.34E+01
1.11E-01	3.02E-03	1.19E-03	2.02E+00	9.25E+00	7.96E+01	3.02E-02	1.23E-04	4.58E-04	8.07E-01	6.84E+00	1.63E+02	1.48E-02	8.69E-06	1.99E-04	3.34E-01	4.94E+00	7.46E+01
1.11E-01	3.89E-03	7.83E-04	8.87E-01	7.16E+00	6.23E+01	3.08E-02	1.41E-04	4.18E-04	8.25E-01	6.85E+00	1.18E+02	1.48E-02	2.13E-05	4.82E-04	4.21E-01	5.82E+00	5.28E+01
1.12E-01	3.55E-03	6.65E-04	6.99E-01	7.04E+00	5.65E+01	3.08E-02	1.34E-04	3.31E-04	6.61E-01	6.63E+00	1.31E+02	1.48E-02	2.00E-05	3.78E-04	3.99E-01	5.87E+00	4.80E+01
1.12E-01	3.68E-03	6.60E-04	1.05E+00	6.05E+00	9.63E+01	3.09E-02	1.59E-04	5.40E-04	1.16E+00	7.24E+00	9.26E+01	1.48E-02	1.88E-05	2.65E-04	3.45E-01	5.41E+00	6.22E+01
1.12E-01	2.60E-03	4.40E-04	5.21E-01	5.70E+00	5.40E+01	3.11E-02	1.68E-04	6.40E-04	8.65E-01	6.87E+00	1.84E+02	1.48E-02	2.10E-05	2.56E-04	3.99E-01	5.83E+00	6.86E+01
1.12E-01	3.76E-03	7.10E-04	7.18E-01	6.77E+00	7.15E+01	3.12E-02	1.39E-04	4.68E-04	7.76E-01	6.93E+00	2.02E+02	1.48E-02	3.96E-06	9.43E-05	2.15E-01	3.91E+00	2.67E+01
1.12E-01	4.42E-03	9.41E-04	7.81E-01	6.70E+00	9.25E+01	3.13E-02	1.41E-04	7.88E-04	9.64E-01	7.29E+00	1.39E+02	1.48E-02	6.52E-06	1.24E-04	2.47E-01	4.43E+00	2.77E+01
1.13E-01	4.31E-03	7.52E-04	5.88E-01	6.37E+00	3.54E+01	3.17E-02	2.00E-04	8.89E-04	9.09E-01	7.31E+00	1.10E+02	1.48E-02	1.71E-05	3.67E-04	4.32E-01	5.57E+00	5.49E+01
1.13E-01	3.96E-03	7.56E-04	7.61E-01	6.55E+00	6.74E+01	3.23E-02	1.49E-04	4.88E-04	8.99E-01	7.14E+00	1.29E+02	1.48E-02	1.86E-05	3.49E-04	3.97E-01	5.89E+00	5.41E+01
1.13E-01	4.06E-03	7.36E-04	7.45E-01	6.12E+00	6.64E+01	3.23E-02	1.33E-04	3.40E-04	6.74E-01	6.72E+00	1.66E+02	1.48E-02	1.84E-05	2.64E-04	3.67E-01	5.76E+00	4.70E+01
1.13E-01	3.68E-03	9.02E-04	1.66E+00	6.66E+00	1.17E+02	3.26E-02	1.41E-04	4.45E-04	7.48E-01	6.93E+00	1.17E+02	1.48E-02	2.06E-05	2.48E-04	3.57E-01	5.80E+00	4.09E+01
1.13E-01	3.47E-03	8.27E-04	1.07E+00	8.41E+00	1.06E+02	3.29E-02	1.46E-04	8.71E-04	1.04E+00	7.45E+00	1.13E+02	1.48E-02	2.88E-05	3.47E-04	5.24E-01	6.14E+00	5.81E+01
1.14E-01	4.12E-03	7.60E-04	9.85E-01	6.34E+00	8.92E+01	3.29E-02	1.46E-04	6.25E-04	7.63E-01	7.18E+00	1.15E+02	1.48E-02	1.21E-05	2.98E-04	4.45E-01	5.44E+00	7.80E+01
1.14E-01	3.76E-03	1.19E-03	1.42E+00	8.35E+00	9.52E+01	3.29E-02	1.64E-04	1.07E-03	1.13E+00	7.56E+00	1.84E+02	1.48E-02	2.68E-05	3.58E-04	4.22E-01	6.32E+00	5.23E+01
1.14E-01	2.90E-03	8.89E-04	1.13E+00	8.26E+00	6.47E+01	3.32E-02	1.29E-04	5.04E-04	8.29E-01	7.19E+00	8.66E+01	1.48E-02	9.80E-06	1.60E-04	2.19E-01	4.52E+00	6.25E+01
1.14E-01	3.15E-03	7.86E-04	1.05E+00	8.17E+00	9.70E+01	3.32E-02	1.62E-04	7.63E-04	8.94E-01	7.39E+00	3.42E+02	1.48E-02	1.34E-05	3.05E-04	4.99E-01	5.84E+00	7.85E+01
1.14E-01	3.65E-03	9.69E-04	1.08E+00	7.58E+00	1.56E+02	3.32E-02	1.63E-04	6.84E-04	7.18E-01	7.28E+00	1.44E+02	1.48E-02	1.68E-05	3.08E-04	4.67E-01	5.66E+00	6.39E+01
1.15E-01	3.65E-03	8.58E-04	9.94E-01	8.07E+00	6.68E+01	3.32E-02	1.34E-04	7.36E-04	8.02E-01	6.95E+00	1.46E+02	1.48E-02	1.73E-05	2.47E-04	3.67E-01	5.57E+00	6.84E+01
1.15E-01	3.72E-03	9.58E-04	1.03E+00	8.13E+00	2.99E+02	3.36E-02	1.27E-04	4.90E-04	7.24E-01	6.65E+00	1.28E+02	1.48E-02	1.14E-05	3.07E-04	4.54E-01	5.59E+00	8.67E+01
1.15E-01	3.22E-03	7.96E-04	1.10E+00	6.34E+00	1.61E+02	3.42E-02	1.28E-04	4.55E-04	6.58E-01	7.39E+00	9.71E+01	3.36E-02	4.14E-06	1.30E-04	1.55E-01	4.17E+00	2.99E+01
1.16E-01	3.46E-03	1.02E-03	1.56E+00	6.92E+00	1.20E+02	3.43E-02	1.03E-04	4.29E-04	5.49E-01	6.60E+00	7.57E+01	3.36E-02	2.93E-06	5.80E-05	1.08E-01	3.81E+00	2.53E+01
1.18E-01	1.38E-04	6.22E-05	1.24E-01	4.96E+00	2.37E+01	3.44E-02	1.17E-04	3.76E-04	5.68E-01	6.76E+00	7.53E+01	3.36E-02	3.24E-06	6.38E-05	1.06E-01	3.77E+00	3.24E+01
1.19E-01	1.19E-04	8.26E-05	1.86E-01	4.96E+00	3.16E+01	3.53E-02	1.59E-04	7.49E-04	9.47E-01	7.41E+00	8.75E+01	3.36E-02	4.26E-06	7.65E-05	1.45E-01	4.09E+00	4.31E+01
1.20E-01	1.34E-04	6.84E-05	1.57E-01	5.02E+00	2.64E+01	3.61E-02	1.13E-04	4.37E-04	7.47E-01	7.25E+00	7.48E+01	3.36E-02	2.78E-06	9.38E-05	1.53E-01	3.91E+00	4.34E+01
1.21E-01	1.24E-04	6.66E-05	1.62E-01	4.94E+00	4.17E+01	3.65E-02	1.46E-04	5.60E-04	8.69E-01	7.41E+00	1.03E+02	3.36E-02	3.02E-06	8.24E-05	1.35E-01	3.87E+00	3.65E+01
1.22E-01	1.51E-04	6.02E-05	1.13E-01	4.89E+00	2.49E+01	3.65E-02	1.29E-04	6.28E-04	8.52E-01	7.38E+00	6.80E+01	3.36E-02	3.61E-06	1.14E-04	1.52E-01	4.08E+00	3.82E+01
1.23E-01	6.95E-05	1.52E-04	3.01E-01	5.64E+00	5.67E+01	3.77E-02	1.12E-04	4.36E-04	8.25E-01	7.27E+00	8.65E+01	3.36E-02	1.24E-06	4.22E-05	1.06E-01	2.99E+00	1.96E+01
1.23E-01	9.98E-05	7.85E-05	2.10E-01	5.49E+00	3.21E+01	3.78E-02	1.31E-04	4.39E-04	8.51E-01	7.35E+00	6.56E+01	3.36E-02	6.52E-06	1.60E-04	2.39E-01	4.91E+00	2.97E+01
1.23E-01	1.77E-04	1.27E-04	2.65E-01	5.38E+00	5.14E+01	3.83E-02	1.30E-04	4.08E-04	9.63E-01	7.42E+00	9.57E+01	3.36E-02	6.59E-06	1.95E-04	2.70E-01	4.91E+00	3.32E+01
1.25E-01	1.50E-04	7.09E-05	1.54E-01	5.14E+00	2.96E+01	3.86E-02	1.33E-04	4.63E-04	9.32E-01	7.42E+00	8.10E+01	3.36E-02	2.65E-06	1.07E-04	1.57E-01	4.07E+00	2.67E+01
1.26E-01	8.04E-05	4.81E-05	1.40E-01	5.30E+00	2.79E+01	3.87E-02	1.58E-04	6.19E-04	9.64E-01	7.45E+00	8.62E+01	3.36E-02	3.00E-06	5.44E-05	1.07E-01	3.65E+00	2.27E+01
1.27E-01	2.73E-04	9.55E-05	1.55E-01	5.24E+00	3.46E+01	3.93E-02	1.19E-04	3.85E-04	6.13E-01	6.61E+00	3.72E+01	3.36E-02	6.03E-06	1.82E-04	2.44E-01	4.82E+00	3.07E+01
1.27E-01	1.26E-04	5.08E-05	1.27E-01	4.98E+00	3.37E+01	3.94E-02	1.08E-04	4.55E-04	7.70E-01	7.49E+00	1.05E+02	3.36E-02	3.54E-06	1.05E-04	1.73E-01	4.23E+00	2.86E+01
1.28E-01	1.31E-04	8.48E-05	2.17E-01	5.25E+00	5.04E+01	4.01E-02	1.30E-04	5.81E-04	7.89E-01	7.33E+00	3.24E+01	3.36E-02	4.59E-06	8.17E-05	1.29E-01	4.12E+00	5.66E+01
1.29E-01	1.13E-04	7.94E-05	2.44E-01	6.27E+00	6.22E+01	4.08E-02	1.00E-04	5.14E-04	7.43E-01	6.88E+00	8.30E+01	3.36E-02	3.77E-06	1.28E-04	1.53E-01	4.19E+00	2.65E+01
1.29E-01	1.12E-04	5.00E-05	1.17E-01	5.26E+00	2.45E+01	4.09E-02	1.01E-04	4.28E-04	6.66E-01	6.74E+00	6.85E+01	3.36E-02	3.66E-06	6.46E-05	1.16E-01	3.91E+00	2.93E+01
1.30E-01	7.95E-05	8.12E-05	2.45E-01	5.87E+00	3.54E+01	4.26E-02	1.18E-04	4.92E-04	8.75E-01	7.60E+00	7.23E+01	3.36E-02	2.69E-06	9.40E-05	1.36E-01	3.79E+00	3.10E+01
1.31E-01	1.00E-04	4.95E-05	1.37E-01	5.42E+00	2.45E+01	4.28E-02	1.17E-04	8.18E-04	1.19E+00	8.16E+00	7.62E+01	3.36E-02	5.36E-06	9.59E-05	1.76E-01	4.40E+00	2.58E+01
1.31E-01	1.53E-04	5.16E-05	9.93E-02	5.07E+00	2.21E+01	4.57E-02	9.89E-05	5.87E-04	7.96E-01	7.32E+00	4.10E+01	3.36E-02	6.62E-06	1.49E-04	2.46E-01	5.01E+00	4.00E+01
1.31E-01	1.27E-04	6.96E-05	1.95E-01	5.21E+00	3.62E+01	5.37E-02	1.26E-05	2.00E-04	2.31E-01	4.51E+00	8.98E+01	3.36E-02	5.52E-06	1.61E-04	2.32E-01	4.65E+00	3.70E+01
1.32E-01	1.23E-04	6.79E-05	1.93E-01	5.16E+00	5.61E+01	5.88E-02	8.31E-06	1.26E-04	2.14E-01	4.12E+00	7.60E+01	3.36E-02	2.34E-06	1.13E-04	1.77E-01	4.23E+00	4.78E+01
1.32E-01	8.03E-05	4.44E-05	1.38E-01	5.50E+00	2.48E+01	5.93E-02	1.31E-05	2.76E-04	2.37E-01	3.99E+00	7.09E+01	3.36E-02	2.23E-06	7.27E-05	1.48E-01	4.01E+00	3.88E+01
1.32E-01	1.35E-04	6.81E-05	1.91E-01	5.07E+00	3.32E+01	5.97E-02	6.67E-06	1.04E-04	1.87E-01	4.09E+00	6.77E+01	3.36E-02	2.83E-06	8.02E-05	1.68E-01	4.05E+00	4.04E+01
1.32E-01	8.67E-05	3.65E-05	1.07E-01	4.74E+00	3.03E+01	6.08E-02	7.44E-06	1.90E-04	2.73E-01	4.54E+00	6.85E+01	3.36E-02	4.67E-06	1.66E-04	2.23E-01	4.73E+00	3.21E+01
1.32E-01	1.17E-04	7.23E-05	1.96E-01	5.44E+00	3.26E+01	6.22E-02	7.11E-06	1.49E-04	2.23E-01	4.31E+00	8.73E+01	3.36E-02	3.96E-06	1.48E-04	2.41E-01	4.68E+00	3.78E+01
1.33E-01	9.39E-06	1.73E-05	3.97E-02	2.81E+00	2.18E+01	6.23E-02	6.68E-06	1.75E-04	2.32E-01	4.37E+00	8.26E+01	3.36E-02	2.16E-06	6.98E-05	1.05E-01	3.33E+00	4.07E+01
1.34E-01	1.17E-04	4.50E-05	1.08E-01	5.45E+00	1.86E+01	6.30E-02	1.66E-05	4.98E-04	3.65E-01	5.29E+00	7.56E+01	3.36E-02	2.00E-06	7.23E-05	1.37E-01	3.78E+00	3.51E+01
1.34E-01	1.11E-04	5.99E-05	1.63E-01	5.31E+00	3.52E+01	6.32E-02	7.45E-06	1.22E-04	1.97E-01	4.17E+00	6.26E+01	3.36E-02	2.95E-06	6.24E-05	1.22E-01	3.97E+00	3.87E+01
1.34E-01	1.54E-04	6.50E-05	1.24E-01	5.48E+00	3.26E+01	6.51E-02	7.68E-06	1.45E-04	2.24E-01	4.28E+00	9.57E+01	3.36E-02	4.88E-06	1.45E-04	2.19E-01	4.66E+00	5.12E+01
1.34E-01	1.28E-04	5.50E-05	1.22E-01	5.50E+00	2.68E+01	6.56E-02	7.54E-06	1.73E-04	2.45E-01	4.32E+00	4.58E+01	3.36E-02	1.68E-06	6.73E-05	1.14E-01	3.56E+00	4.41E+01
1.34E-01	1.08E-04	6.75E-05	1.85E-01	5.67E+00	3.28E+01	6.66E-02	6.74E-06	1.38E-04	2.10E-01	4.05E+00	1.08E+02	3.36E-02	2.07E-06	7.75E-05	1.49E-01	3.98E+00	4.64E+01
1.34E-01	1.46E-04	5.83E-05	1.25E-01	5.60E+00	2.11E+01	6.71E-02	6.64E-06	1.76E-04	2.70E-01	4.52E+00	5.70E+01	3.36E-02	2.27E-06	7.75E-05	1.58E-01	4.05E+00	3.59E+01
1.35E-01	1.39E-04	6.00E-05	1.58E-01	5.21E+00	3.37E+01	6.79E-02	7.55E-06	1.06E-04	1.67E-01	4.14E+00	6.06E+01	3.36E-02	2.33E-06	7.18E-05	1.29E-01	3.92E+00	2.99E+01
1.35E-01	1.46E-04	5.82E-05	1.25E-01	5.46E+00	2.24E+01	6.84E-02	1.03E-05	2.58E-04	2.86E-01	4.67E+00	5.74E+01	3.36E-02	3.11E-06	8.09E-05	1.29E-01	4.04E+00	3.70E+01
1.35E-01	1.15E-04	2.88E-04	3.35E-01	5.63E+00	3.65E+01	6.95E-02	7.70E-06	1.56E-04	2.39E-01	4.40E+00	1.46E+02	3.36E-02	3.98E-06	1.31E-04	1.46E-01	3.82E+00	3.06E+01
1.35E-01	1.42E-04	7.01E-05	1.69E-01	5.33E+00	3.29E+01	6.98E-02	8.49E-06	1.60E-04	1.95E-01	4.33E+00	6.62E+01	3.36E-02	2.51E-06	9.12E-05	1.42E-01	3.90E+00	4.24E+01
1.36E-01	1.07E-04	5.06E-05	1.28E-01	5.82E+00	6.59E+01	7.15E-02	1.08E-05	2.88E-04	2.86E-01	4.65E+00	3.78E+01	3.36E-02	3.07E-06	5.71E-05	1.12E-01	3.83E+00	1.16E+01
1.36E-01	1.29E-04	5.57E-05	1.55E-01	5.33E+00	2.80E+01	7.18E-02	1.01E-05	2.31E-04	2.27E-01	4.44E+00	4.91E+01	3.36E-02	1.72E-06	4.72E-05	1.00E-01	3.22E+00	1.42E+01
1.36E-01	1.48E-04	7.34E-05	1.77E-01	5.46E+00	3.35E+01	7.23E-02	8.63E-06	2.32E-04	2.87E-01	4.49E+00	4.19E+01	3.36E-02	2.74E-06	7.43E-05	1.27E-01	3.71E+00	3.80E+01
1.36E-01	1.23E-04	6.76E-05	1.49E-01	5.76E+00	4.57E+01	7.25E-02	6.44E-06	1.51E-04	2.27E-01	4.42E+00	7.29E+01	3.36E-02	6.19E-06	1.81E-04	2.57E-01	5.06E+00	4.08E+01
1.36E-01	1.07E-04	5.08E-05	1.34E-01	5.92E+00	3.17E+01	7.33E-02	9.21E-06	2.07E-04	2.51E-01	4.57E+00	4.07E+01	3.36E-02	6.35E-06	1.71E-04	2.01E-01	4.67E+00	2.42E+01
1.36E-01	1.37E-04	7.21E-05	1.96E-01	5.44E+00	4.36E+01	7.33E-02	7.70E-06	1.21E-04	2.14E-01	4.34E+00	5.27E+01	3.36E-02	5.91E-06	1.46E-04	2.34E-01	4.63E+00	3.55E+01
1.37E-01	1.05E-04	9.65E-05	2.23E-01	6.23E+00	2.92E+01	7.45E-02	8.67E-06	1.76E-04	1.85E-01	4.04E+00	2.90E+01	3.36E-02	1.90E-06	5.20E-05	1.07E-01	3.36E+00	1.73E+01
1.37E-01	1.11E-04	5.12E-05	1.30E-01	5.88E+00	2.19E+01	7.47E-02	6.81E-06	1.05E-04	1.60E-01	4.11E+00	5.50E+01	3.36E-02	2.92E-06	6.47E-05	1.14E-01	3.74E+00	3.61E+01
1.37E-01	9.53E-05	4.70E-05	1.30E-01	5.88E+00	3.40E+01	7.47E-02	9.48E-06	3.33E-04	3.25E-01	4.89E+00	6.29E+01	3.36E-02	1.61E-06	7.15E-05	1.26E-01	3.65E+00	4.80E+01
1.37E-01	1.09E-04	7.48E-05	1.81E-01	6.04E+00	3.26E+01	7.49E-02	8.19E-06	1.63E-04	2.27E-01	4.43E+00	6.75E+01	3.36E-02	1.92E-06	7.11E-05	1.38E-01	3.64E+00	3.58E+01
1.38E-01	8.58E-05	4.32E-05	1.34E-01	5.87E+00	2.11E+01	7.56E-02	8.61E-06	2.64E-04	3.21E-01	4.60E+00	9.94E+01	3.36E-02	3.29E-06	1.44E-04	1.41E-01	3.55E+00	3.34E+01
1.38E-01	9.17E-05	7.40E-05	1.80E-01	6.07E+00	1.85E+01	7.72E-02	6.31E-06	1.46E-04	2.21E-01	4.19E+00	6.06E+01	3.36E-02	1.64E-06	6.47E-05	1.25E-01	3.57E+00	3.89E+01
1.39E-01	1.26E-04	9.47E-05	2.11E-01	5.50E+00	5.29E+01	7.78E-02	7.51E-06	1.16E-04	1.69E-01	4.17E+00	5.82E+01	3.36E-02	1.91E-06	6.84E-05	1.35E-01	3.69E+00	4.03E+01

MPC						MR-MPC						SPIHT					
BPP	F1	F2	F3	F4	F5	BPP	F1	F2	F3	F4	F5	BPP	F1	F2	F3	F4	F5
1.39E-01	1.06E-05	6.44E-06	9.01E-02	3.43E+00	2.36E+01	7.60E-02	7.28E-06	1.86E-04	2.58E-01	4.66E+00	5.05E+01	7.73E-02	7.57E-07	1.42E-05	3.41E-02	2.60E+00	1.80E+01
1.40E-01	1.19E-04	4.50E-06	1.24E-01	5.42E+00	3.10E+01	7.62E-02	1.84E-05	2.77E-04	3.18E-01	4.62E+00	6.31E+01	7.73E-02	6.59E-07	2.34E-05	4.24E-02	2.52E+00	2.25E+01
1.55E-01	5.57E-06	1.27E-05	3.56E-02	3.07E+00	2.20E+01	8.09E-02	6.80E-06	1.38E-04	1.77E-01	4.17E+00	4.68E+01	7.73E-02	1.81E-06	4.68E-05	9.16E-02	3.21E+00	3.70E+01
1.56E-01	7.63E-06	1.56E-05	3.42E-02	3.39E+00	2.91E+01	8.45E-02	1.04E-05	1.62E-04	2.27E-01	4.48E+00	2.18E+01	7.73E-02	7.97E-07	2.99E-05	5.20E-02	2.74E+00	2.91E+01
1.56E-01	6.62E-06	1.68E-05	3.79E-02	3.15E+00	2.55E+01	8.56E-02	6.76E-06	1.33E-04	1.88E-01	4.33E+00	7.38E+01	7.73E-02	1.61E-06	7.67E-05	1.27E-01	3.41E+00	3.60E+01
1.57E-01	5.10E-06	9.02E-06	3.60E-02	3.02E+00	1.77E+01	8.70E-02	5.88E-06	1.02E-04	1.71E-01	4.12E+00	4.33E+01	7.73E-02	6.63E-07	3.67E-05	6.48E-02	2.77E+00	2.75E+01
1.57E-01	5.00E-06	2.11E-05	6.41E-02	3.09E+00	2.51E+01	8.71E-02	9.57E-06	1.87E-04	2.22E-01	4.45E+00	6.12E+01	7.73E-02	1.49E-06	4.28E-05	8.37E-02	3.16E+00	1.02E+01
1.59E-01	6.74E-06	1.73E-05	4.78E-02	3.35E+00	2.88E+01	8.71E-02	5.86E-06	1.16E-04	1.78E-01	4.18E+00	4.03E+01	7.73E-02	5.44E-07	1.16E-05	2.81E-02	2.14E+00	8.72E+00
1.59E-01	4.86E-06	1.21E-05	4.30E-02	3.10E+00	3.17E+01	8.75E-02	6.78E-06	1.12E-04	1.87E-01	4.28E+00	1.65E+01	7.73E-02	1.75E-06	7.64E-05	1.20E-01	3.44E+00	2.13E+01
1.59E-01	5.19E-06	9.02E-06	3.86E-02	3.14E+00	1.89E+01	8.87E-02	7.49E-06	1.54E-04	2.31E-01	4.44E+00	6.00E+01	7.73E-02	5.13E-07	2.64E-05	4.87E-02	2.54E+00	1.67E+01
1.61E-01	4.07E-06	9.91E-06	3.81E-02	3.13E+00	2.92E+01	8.91E-02	8.20E-06	1.68E-04	2.14E-01	4.28E+00	5.08E+01	7.73E-02	1.75E-06	3.61E-05	7.22E-02	3.08E+00	1.79E+01
1.61E-01	4.38E-06	8.63E-06	3.22E-02	3.15E+00	2.49E+01	9.05E-02	6.67E-06	1.18E-04	1.92E-01	4.26E+00	7.57E+01	7.73E-02	1.66E-06	7.11E-05	1.20E-01	3.39E+00	2.10E+01
1.65E-01	4.47E-06	8.15E-06	3.10E-02	2.94E+00	2.50E+01	9.20E-02	6.99E-06	1.08E-04	1.74E-01	4.21E+00	6.47E+01	7.73E-02	7.01E-07	2.68E-05	5.09E-02	2.69E+00	1.96E+01
1.66E-01	4.28E-06	8.27E-06	3.36E-02	3.11E+00	1.52E+01	9.29E-02	1.06E-05	1.54E-04	2.07E-01	3.88E+00	2.14E+01	7.73E-02	1.59E-06	4.20E-05	8.13E-02	3.13E+00	4.65E+01
1.66E-01	5.07E-06	6.14E-06	2.29E-02	3.20E+00	1.43E+01	9.37E-02	5.76E-06	1.18E-04	1.71E-01	4.27E+00	4.92E+01	7.73E-02	1.02E-06	3.74E-05	5.73E-02	2.71E+00	1.95E+01
1.66E-01	4.12E-06	8.54E-06	3.41E-02	3.18E+00	1.75E+01	9.89E-02	7.33E-06	2.14E-04	2.47E-01	4.35E+00	6.10E+01	7.73E-02	1.76E-06	4.81E-05	8.98E-02	3.18E+00	2.55E+01
1.68E-01	4.51E-06	5.61E-06	2.58E-02	3.16E+00	1.47E+01	9.97E-02	8.78E-06	2.67E-04	3.43E-01	4.91E+00	5.18E+01	7.73E-02	2.34E-06	7.20E-05	1.49E-01	3.74E+00	2.78E+01
1.68E-01	4.86E-06	1.05E-05	3.72E-02	3.20E+00	2.07E+01	1.00E-01	9.89E-06	1.95E-04	2.68E-01	4.05E+00	5.54E+01	7.73E-02	5.39E-07	1.27E-05	3.21E-02	2.20E+00	1.27E+01
1.69E-01	3.93E-06	9.37E-06	3.72E-02	3.11E+00	2.15E+01	1.07E-01	7.46E-06	2.41E-04	3.48E-01	4.83E+00	5.28E+01	7.73E-02	5.66E-07	2.28E-05	4.93E-02	2.52E+00	2.69E+01
1.69E-01	4.27E-06	1.32E-05	4.96E-02	3.28E+00	2.36E+01	1.13E-01	7.38E-06	1.56E-04	2.09E-01	4.44E+00	6.02E+01	7.73E-02	1.42E-06	7.30E-05	1.16E-01	3.57E+00	2.46E+01
1.69E-01	5.25E-06	4.45E-06	1.99E-02	3.24E+00	1.04E+01	1.38E-01	2.26E-06	9.08E-05	1.02E-01	2.64E+00	4.83E+01	7.73E-02	1.43E-06	8.12E-05	1.39E-01	3.74E+00	2.94E+01
1.70E-01	4.10E-06	1.13E-05	4.46E-02	3.10E+00	2.05E+01	1.42E-01	1.35E-06	7.05E-05	8.50E-02	2.83E+00	3.51E+01	7.73E-02	5.02E-07	2.59E-05	5.12E-02	2.53E+00	2.40E+01
1.70E-01	4.41E-06	1.12E-05	4.33E-02	3.33E+00	2.55E+01	1.50E-01	8.48E-07	3.50E-05	5.84E-02	2.50E+00	6.22E+01	7.73E-02	1.41E-06	5.56E-05	1.10E-01	3.43E+00	3.63E+01
1.71E-01	5.05E-06	4.91E-06	1.94E-02	3.19E+00	1.38E+01	1.51E-01	2.78E-06	1.51E-04	1.62E-01	3.67E+00	6.00E+01	7.73E-02	5.70E-07	2.67E-05	5.21E-02	2.62E+00	3.12E+01
1.71E-01	4.18E-06	1.43E-05	5.56E-02	3.33E+00	2.33E+01	1.56E-01	1.81E-06	5.56E-05	9.39E-02	2.61E+00	5.69E+01	7.73E-02	5.80E-07	2.70E-05	5.52E-02	2.58E+00	2.48E+01
1.71E-01	5.60E-06	6.03E-06	2.21E-02	3.34E+00	1.84E+01	1.60E-01	9.50E-07	6.90E-05	9.66E-02	2.96E+00	4.15E+01	7.73E-02	1.30E-06	7.35E-05	1.42E-01	3.63E+00	4.58E+01
1.72E-01	5.60E-06	1.69E-05	5.52E-02	3.22E+00	2.17E+01	1.60E-01	1.27E-06	6.80E-05	8.15E-02	2.90E+00	4.65E+01	7.73E-02	1.08E-06	4.28E-05	9.05E-02	3.30E+00	3.06E+01
1.73E-01	4.14E-06	1.13E-05	4.24E-02	3.22E+00	1.69E+01	1.60E-01	2.46E-06	1.12E-04	1.57E-01	3.12E+00	3.94E+01	7.73E-02	5.79E-07	2.88E-05	5.82E-02	2.75E+00	2.14E+01
1.73E-01	5.32E-06	4.58E-06	1.97E-02	3.29E+00	1.29E+01	1.61E-01	1.59E-06	9.11E-05	1.22E-01	3.38E+00	5.91E+01	7.73E-02	7.93E-07	3.72E-05	4.47E-02	2.19E+00	2.24E+01
1.74E-01	4.98E-06	4.50E-06	2.04E-02	3.06E+00	1.03E+01	1.61E-01	1.16E-06	5.41E-05	9.78E-02	2.56E+00	6.54E+01	7.73E-02	6.06E-07	2.41E-05	5.11E-02	2.63E+00	2.80E+01
1.74E-01	4.69E-06	9.35E-06	3.41E-02	3.30E+00	1.71E+01	1.61E-01	7.32E-07	3.18E-05	5.31E-02	2.49E+00	4.34E+01	7.73E-02	9.36E-07	3.88E-05	7.12E-02	3.10E+00	2.58E+01
1.76E-01	4.72E-06	1.51E-05	5.47E-02	3.07E+00	1.97E+01	1.62E-01	1.54E-06	9.54E-05	1.33E-01	3.29E+00	5.77E+01	7.73E-02	1.62E-06	7.66E-05	1.02E-01	3.23E+00	2.51E+01
1.78E-01	5.95E-06	6.68E-06	2.39E-02	3.37E+00	1.16E+01	1.63E-01	9.34E-07	7.46E-05	1.10E-01	3.04E+00	3.49E+01	7.73E-02	1.15E-06	3.52E-05	5.14E-02	2.81E+00	2.56E+01
1.79E-01	4.90E-06	7.07E-05	9.48E-02	3.32E+00	2.68E+01	1.64E-01	7.88E-06	1.68E-04	3.03E-01	2.73E+00	8.93E+01	7.73E-02	3.86E-07	1.05E-05	3.09E-02	2.06E+00	1.21E+01
1.79E-01	4.95E-06	1.38E-06	9.98E-03	3.11E+00	6.08E+00	1.64E-01	1.35E-06	4.11E-05	6.36E-02	2.57E+00	2.08E+01	7.73E-02	1.46E-06	7.40E-05	1.18E-01	3.47E+00	2.14E+01
1.79E-01	4.34E-06	5.63E-06	2.41E-02	3.28E+00	9.49E+00	1.65E-01	2.74E-06	5.67E-05	9.47E-02	2.81E+00	2.75E+01	7.73E-02	6.47E-07	2.33E-05	4.04E-02	2.41E+00	2.16E+01
1.81E-01	5.45E-06	1.85E-05	4.52E-02	3.45E+00	4.52E+01	1.66E-01	1.76E-06	8.25E-05	1.18E-01	3.11E+00	2.53E+01	7.73E-02	1.89E-06	7.43E-05	1.09E-01	3.48E+00	1.78E+01
1.81E-01	5.08E-06	1.59E-05	6.68E-02	3.23E+00	1.79E+01	1.67E-01	2.49E-06	8.80E-05	1.03E-01	2.96E+00	3.24E+01	7.73E-02	1.66E-06	6.15E-05	1.07E-01	3.34E+00	2.54E+01
1.81E-01	4.66E-06	3.51E-06	1.84E-02	3.26E+00	1.42E+01	1.67E-01	3.46E-06	6.54E-05	1.07E-01	2.78E+00	3.04E+01	7.73E-02	3.63E-07	2.68E-05	4.90E-02	2.48E+00	3.45E+01
1.81E-01	4.15E-06	9.86E-06	4.27E-02	3.35E+00	2.39E+01	1.67E-01	1.24E-06	4.55E-05	7.06E-02	2.73E+00	5.50E+01	7.73E-02	1.26E-06	6.01E-05	1.15E-01	3.50E+00	3.79E+01
1.82E-01	6.37E-06	9.36E-06	3.36E-02	3.27E+00	1.76E+01	1.69E-01	6.81E-06	1.29E-04	2.79E-01	2.82E+00	3.80E+01	7.73E-02	1.36E-06	6.37E-05	1.24E-01	3.38E+00	3.94E+01
1.84E-01	5.32E-06	1.86E-06	1.16E-02	3.26E+00	9.39E+00	1.70E-01	2.11E-05	2.54E-04	4.45E-01	3.35E+00	2.97E+01	7.73E-02	6.67E-07	2.20E-05	3.52E-02	2.32E+00	2.86E+01
1.85E-01	4.39E-06	1.33E-05	4.67E-02	3.27E+00	1.90E+01	1.70E-01	9.12E-07	6.18E-05	1.09E-01	2.86E+00	3.65E+01	7.73E-02	1.17E-06	5.12E-05	6.61E-02	2.73E+00	2.16E+01
1.85E-01	5.45E-06	5.52E-06	1.85E-02	3.30E+00	8.92E+00	1.72E-01	1.50E-06	5.78E-05	9.68E-02	2.72E+00	4.19E+01	7.73E-02	6.74E-07	2.08E-05	3.56E-02	2.42E+00	2.70E+01
1.86E-01	5.57E-06	1.32E-05	2.51E-02	3.38E+00	9.35E+00	1.72E-01	1.29E-06	5.44E-05	7.32E-02	2.69E+00	1.25E+01	7.73E-02	1.76E-06	4.06E-05	8.87E-02	3.13E+00	2.10E+01
1.86E-01	5.61E-06	1.97E-06	1.18E-02	3.27E+00	1.34E+01	1.72E-01	1.29E-06	7.53E-05	1.52E-01	2.85E+00	3.82E+01	7.73E-02	1.67E-06	8.45E-05	1.30E-01	3.65E+00	2.82E+01
1.86E-01	4.73E-06	1.70E-06	1.10E-02	3.18E+00	8.25E+00	1.74E-01	3.66E-06	1.11E-04	1.33E-01	3.16E+00	4.76E+01	7.73E-02	1.50E-06	6.39E-05	1.09E-01	3.25E+00	2.60E+01
1.86E-01	4.83E-06	1.68E-06	1.11E-02	3.22E+00	1.18E+01	1.75E-01	3.06E-06	1.02E-04	1.74E-01	2.99E+00	4.02E+01	7.73E-02	1.52E-06	3.83E-05	8.20E-02	3.07E+00	3.13E+01
1.92E-01	5.14E-06	1.52E-05	2.29E-02	3.34E+00	6.13E+00	1.75E-01	1.38E-06	5.33E-05	8.69E-02	2.67E+00	3.47E+01	7.73E-02	4.97E-07	2.31E-05	4.44E-02	2.49E+00	3.44E+01
1.95E-01	6.61E-07	4.60E-06	1.04E-02	1.59E+00	1.06E+01	1.76E-01	1.67E-06	5.44E-05	7.97E-02	2.69E+00	4.42E+01	7.73E-02	1.62E-06	7.28E-05	1.30E-01	3.64E+00	3.92E+01
1.97E-01	5.26E-06	4.57E-06	2.02E-02	3.39E+00	2.10E+01	1.76E-01	1.12E-06	4.55E-05	6.93E-02	2.50E+00	3.32E+01	1.72E-01	4.54E-07	2.71E-05	4.03E-02	2.25E+00	1.74E+01
2.08E-01	8.97E-07	1.03E-05	1.89E-02	1.94E+00	1.03E+01	1.77E-01	1.64E-06	6.31E-05	1.02E-01	3.01E+00	4.25E+01	1.72E-01	4.12E-07	1.03E-05	2.48E-02	2.10E+00	1.39E+01
2.25E-01	7.40E-07	5.02E-06	1.61E-02	1.68E+00	9.93E+00	1.80E-01	5.25E-06	9.14E-05	1.51E-01	2.50E+00	1.74E+01	1.72E-01	3.98E-07	1.50E-05	2.62E-02	2.16E+00	1.97E+01
2.25E-01	5.52E-07	2.20E-06	6.05E-03	1.66E+00	8.40E+00	1.81E-01	5.91E-06	1.16E-04	1.97E-01	2.70E+00	3.18E+01	1.72E-01	4.75E-07	2.81E-05	4.91E-02	2.46E+00	2.73E+01
2.29E-01	4.93E-07	2.72E-06	6.25E-03	1.69E+00	9.40E+00	1.81E-01	1.38E-06	5.27E-05	8.66E-02	2.60E+00	4.83E+01	1.72E-01	4.29E-07	1.27E-05	2.98E-02	2.17E+00	6.54E+00
2.30E-01	6.12E-07	2.61E-06	5.89E-03	1.77E+00	8.44E+00	1.81E-01	7.53E-07	3.28E-05	4.49E-02	2.60E+00	4.75E+01	1.72E-01	2.19E-07	3.66E-06	1.11E-02	1.67E+00	6.64E+00
2.32E-01	4.85E-07	2.76E-06	9.87E-03	1.71E+00	1.03E+01	1.82E-01	4.18E-06	1.22E-04	1.81E-01	2.95E+00	2.25E+01	1.72E-01	5.22E-07	1.37E-05	2.89E-02	2.20E+00	1.47E+01
2.32E-01	4.53E-07	3.45E-06	1.15E-02	1.75E+00	1.61E+01	1.82E-01	1.25E-06	3.64E-05	5.32E-02	2.52E+00	4.49E+01	1.72E-01	5.17E-07	2.93E-05	4.33E-02	2.30E+00	2.04E+01
2.36E-01	4.82E-07	1.45E-06	5.62E-03	1.59E+00	1.02E+01	1.82E-01	6.93E-07	3.12E-05	4.37E-02	2.52E+00	2.34E+01	1.72E-01	3.47E-07	1.26E-05	2.26E-02	2.04E+00	1.67E+01
2.36E-01	6.01E-07	4.73E-06	1.26E-02	1.91E+00	1.61E+01	1.82E-01	5.92E-06	1.47E-04	3.14E-01	2.91E+00	4.69E+01	1.72E-01	4.71E-07	2.53E-05	4.60E-02	2.49E+00	1.94E+01
2.37E-01	4.42E-07	2.43E-06	9.28E-03	1.71E+00	9.00E+00	1.84E-01	9.83E-07	4.06E-05	5.66E-02	2.61E+00	9.65E+00	1.72E-01	4.83E-07	2.08E-05	3.88E-02	2.28E+00	1.68E+01
2.38E-01	3.56E-07	1.82E-06	7.42E-03	1.68E+00	1.07E+01	1.85E-01	1.11E-06	4.09E-05	5.90E-02	2.63E+00	2.30E+01	1.72E-01	3.83E-07	2.03E-05	4.14E-02	2.45E+00	2.58E+01
2.38E-01	3.63E-07	2.16E-06	7.02E-03	1.79E+00	1.43E+01	1.86E-01	1.50E-06	5.77E-05	8.84E-02	2.81E+00	4.41E+01	1.72E-01	3.90E-07	2.03E-05	4.33E-02	2.27E+00	2.67E+01
2.39E-01	4.79E-07	2.06E-06	6.94E-03	1.81E+00	9.02E+00	1.86E-01	6.51E-07	3.09E-05	4.48E-02	2.60E+00	7.29E+01	1.72E-01	7.07E-07	2.50E-05	5.84E-02	2.73E+00	2.87E+01
2.44E-01	4.63E-07	2.42E-06	7.96E-03	1.77E+00	9.61E+00	1.87E-01	5.71E-06	8.76E-05	1.62E-01	2.89E+00	1.93E+01	1.72E-01	5.20E-07	3.26E-05	6.36E-02	2.78E+00	2.19E+01
2.45E-01	7.85E-07	3.27E-05	3.35E-02	1.99E+00	1.18E+01	1.87E-01	9.06E-07	3.56E-05	4.88E-02	2.58E+00	3.46E+01	1.72E-01	2.12E-07	7.01E-06	1.19E-02	1.66E+00	1.95E+01
2.45E-01	6.71E-07	3.60E-06	1.09E-02	1.80E+00	1.00E+01	1.88E-01	2.05E-06	6.44E-05	7.30E-02	2.81E+00	3.00E+01	1.72E-01	4.07E-07	2.57E-05	5.06E-02	2.43E+00	2.38E+01
2.46E-01	4.12E-07	3.28E-06	1.42E-02	1.76E+00	9.93E+00	1.94E-01	2.10E-06	9.57E-05	1.14E-01	3.16E+00	3.60E+01	1.72E-01	7.90E-07	2.65E-05	5.61E-02	2.86E+00	2.78E+01
2.46E-01	4.04E-07	2.70E-06	1.07E-02	1.81E+00	1.13E+01	1.95E-01	5.66E-06	1.72E-04	3.44E-01	3.31E+00	3.87E+01	1.72E-01	9.77E-07	3.07E-05	6.14E-02	3.01E+00	2.79E+01
2.46E-01	3.74E-07	2.66E-06	9.75E-03	1.74E+00	8.93E+00	2.01E-01	1.93E-06	6.61E-05	9.20E-02	2.96E+00	2.74E+01	1.72E-01	3.87E-07	2.24E-05	4.32E-02	2.34E+00	3.23E+01
2.48E-01	4.17E-07	1.32E-06	5.70E-03	1.58E+00	4.49E+00	2.03E-01	1.05E-06	3.98E-05	6.17E-02	1.70E+00	5.93E+01	1.72E-01	4.04E-07	2.38E-05	4.99E-02	2.34E+00	2.46E+01
2.48E-01	3.56E-07	2.52E-06	9.11E-03	1.73E+00	1.20E+01	2.24E-01	1.08E-06	5.35E-05	7.00E-02	1.89E+00	6.60E+01	1.72E-01	2.37E-07	1.18E-05	1.60E-02	1.54E+00	1.64E+01
2.49E-01	4.36E-07	3.09E-06	1.15E-02	1.74E+00	1.06E+01	2.25E-01	1.53E-06	5.49E-05	6.41E-02	2.19E+00	4.96E+01	1.72E-01	9.01E-07	4.92E-05	6.18E-02	2.60E+00	2.13E+01
2.51E-01	5.05E-07	1.59E-06	5.56E-03	1.69E+00	5.87E+00	2.33E-01	3.91E-07	1.17E-05	2.10E-02	1.59E+00	3.27E+01	1.72E-01	4.24E-07	2.45E-05	4.10E-02	2.28E+00	2.90E+01
2.51E-01	3.86E-07	3.75E-06	1.38E-02	1.86E+00	1.33E+01	2.35E-01	2.21E-06	6.38E-05	9.84E-02	1.94E+00	2.54E+01	1.72E-01	1.23E-07	2.92E-06	9.84E-03	1.47E+00	8.40E+00
2.52E-01	4.85E-07	3.35E-06	1.26E-02	1.86E+00	1.36E+01	2.36E-01	4.47E-06	1.22E-04	2.02E-01	2.23E+00	4.45E+01	1.72E-01	5.47E-07	2.37E-05	4.37E-02	2.34E+00	1.43E+01
2.52E-01	4.67E-07	1.10E-06	3.69E-03	1.75E+00	6.52E+00	2.37E-01	3.48E-06	5.80E-05	9.59E-02	2.08E+00	3.48E+01	1.72E-01	1.37E-07	8.39E-06	1.46E-02	1.71E+00	1.32E+01
2.52E-01	4.49E-07	1.20E-06	5.35E-03	1.77E+00	5.36E+00	2.37E-01	1.45E-06	6.32E-05	7.59E-02	2.15E+00	5.13E+01	1.72E-01	6.50E-07	3.04E-05	5.73E-02	2.57E+00	1.29E+01
2.52E-01	4.97E-07	2.07E-06	6.19E-03	1.65E+00	1.19E+01	2.38E-01	1.93E-06	3.09E-05	3.95E-02	2.11E+00	4.38E+01	1.72E-01	4.88E-07	2.05E-05	3.48E-02	2.26E+00	2.69E+01
2.53E-01	3.48E-07	2.52E-06	9.28E-03	1.77E+00	9.53E+00	2.39E-01	1.43E-06	3.61E-05	4.45E-02	2.03E+00	2.73E+01	1.72E-01	4.26E-07	2.45E-05	4.34E-02	2.37E+00	1.91E+01
2.53E-01	6.05E-07	4.75E-06	1.53E-02	1.93E+00	1.25E+01	2.39E-01	1.48E-06	5.32E-05	7.70E-02	2.13E+00	4.33E+01	1.72E-01	5.03E-07	1.69E-05	3.72E-02	2.27E+00	3.32E+01
2.53E-01	3.90E-07	4.17E-06	1.45E-02	1.89E+00	1.28E+01	2.40E-01	5.39E-07	4.37E-05	5.78E-02	2.12E+00	3.24E+01	1.72E-01	5.02E-07	1.40E-05	3.11E-02	2.19E+00	1.59E+01
2.54E-01	4.65E-07	1.77E-06	7.03E-03	1.78E+00	7.39E+00	2.40E-01	3.22E-07	7.40E-06	1.35E-02	1.59E+00	3.32E+01	1.72E-01	5.13E-07	1.26E-05	2.87E-02	2.19E+00	1.31E+01
2.54E-01	1.57E-07	1.35E-06	2.80E-03	9.81E-01	5.41E+00	2.42E-01	1.90E-06	4.70E-05	6.03E-02	2.11E+00	3.05E+01	1.72E-01	1.64E-06	5.38E-05	1.02E-01	3.31E+00	2.77E+01
2.55E-01	4.52E-07	2.37E-06	8.63E-03	1.76E+00	7.72E+00	2.42E-01	1.02E-06	5.72E-05	1.19E-01	2.03E+00	4.97E+01	1.72E-01	5.44E-07	2.21E-05	3.76E-02	2.38E+00	1.17E+01
2.59E-01	4.04E-07	2.39E-06	9.04E-03	1.73E+00	6.41E+00	2.44E-01	2.31E-06	3.34E-05	5.45E-02	2.05E+00	1.33E+01	1.72E-01	1.80E-07	3.60E-06	1.15E-02	1.60E+00	8.99E+00
2.61E-01	4.81E-07	6.92E-07	2.54E-03	1.72E+00	4.12E+00	2.45E-01	4.66E-06	7.28E-05	1.29E-01	2.19E+00	1.05E+02	1.72E-01	3.92E-07	2.28E-05	4.37E-02	2.39E+00	3.41E+01
2.61E-01	5.65E-07	5.27E-06	1.22E-02	1.91E+00	2.29E+01	2.45E-01	2.11E-06	1.85E-05	2.55E-02	1.73E+00	2.23E+01	1.72E-01	4.06E-07	2.67E-05	5.24E-02	2.52E+00	2.10E+01
2.61E-01	4.15E-07	1.09E-06	4.49E-03	1.79E+00	5.62E+00	2.46E-01	5.21E-07	1.37E-05	2.14E-02	1.66E+00	2.36E+01	1.72E-01	4.12E-07	2.22E-05	4.62E-02	2.40E+00	2.86E+01
2.61E-01	6.33E-07	3.05E-06	9.14E-03	1.92E+00	7.75E+00	2.48E-01	3.25E-06	4.52E-05	3.50E-02	1.96E+00	3.94E+01	1.72E-01	3.99E-07	2.33E-05	4.07E-02	2.36E+00	2.59E+01
2.63E-01	4.33E-07	1.54E-06	5.97E-03	1.80E+00	9.42E+00	2.49E-01	1.45E-06	5.94E-05	1.09E-01	2.12E+00	2.60E+01	1.72E-01	5.29E-07	1.56E-05	3.63E-02	2.24E+00	2.40E+01
2.65E-01	7.48E-07	3.25E-06	9.07E-03	1.82E+00	9.63E+00	2.51E-01	5.56E-07	3.95E-05	9.13E-02	2.18E+00	2.37E+01	1.72E-01	4.65E-07	2.54E-05	4.31E-02	2.29E+00	2.47E+01
2.66E-01	4.70E-07	4.02E-07	1.76E-03	1.72E+00	5.37E+00	2.52E-01	4.96E-06	7.30E-05	9.27E-02	2.09E+00	2.14E+01	1.72E-01	6.31E-07	3.00E-05	4.18E-02	2.42E+00	2.63E+01
2.66E-01	4.31E-07	2.47E-06	9.35E-03	1.93E+00	1.35E+01	2.52E-01	9.95E-07	4.63E-05	8.28E-02	2.14E+00	7.78E+00	1.72E-01	4.02E-07	2.45E-05	4.63E-02	2.37E+00	1.44E+01
2.66E-01	4.95E-07	3.90E-07	1.65E-03	1.77E+00	5.95E+00	2.52E-01	5.45E-06	1.24E-04	2.49E-01	2.22E+00	4.56E+01	1.72E-01	4.35E-07	2.36E-05	4.17E-02	2.26E+00	1.69E+01
2.67E-01	4.97E-07	6.28E-07	2.47E-03	1.81E+00	5.47E+00	2.53E-01	2.23E-06	6.68E-05	8.49E-02	2.06E+00	3.01E+01	1.72E-01	4.35E-07	1.19E-05	2.92E-02	2.13E+00	2.18E+01
2.67E-01	4.18E-07	3.28E-07	1.59E-03	1.74E+00	6.93E+00	2.54E-01	5.21E-07	3.94E-05	8.63E-02	2.12E+00	1.61E+01	1.72E-01	3.16E-07	2.87E-05	4.87E-02	2.42E+00	3.45E+01
2.67E-01	5.55E-07	1.85E-07	1.12E-03	1.73E+00	4.88E+00	2.55E-01	1.29E-06	4.50E-05	7.15E-02	2.12E+00	3.30E+01	1.72E-01	4.28E-07	2.98E-05	5.97E-02	2.60E+00	3.35E+01
2.69E-01	5.08E-07	1.27E-06	3.87E-03	1.84E+00	4.61E+00	2.56E-01	2.34E-06	6.53E-05	8.42E-02	2.17E+00	3.83E+01	1.72E-01	3.90E-07	2.37E-05	5.05E-02	2.49E+00	2.38E+01
2.72E-01	4.33E-07	8.50E-06	9.62E-03	1.86E+00	3.04E+00	2.58E-01	2.33E-06	4.20E-05	5.35E-02	2.09E+00	2.72E+01	1.72E-01	4.23E-07	2.79E-05	4.76E-02	2.48E+00	1.68E+01

Figure 4.3: This figure compares MPC, MR-MPC and SPIHT using PQS factor one (F1) which measures visibility of random error using CCIR 567-1 noise weighing standard. This plot shows how observed PQS F1 varied with bit rate (BPP). Each point corresponds to a single compressed radiograph. Lower values of PQS F1 (which occur near the top of the graph) indicate that less random error is visible.

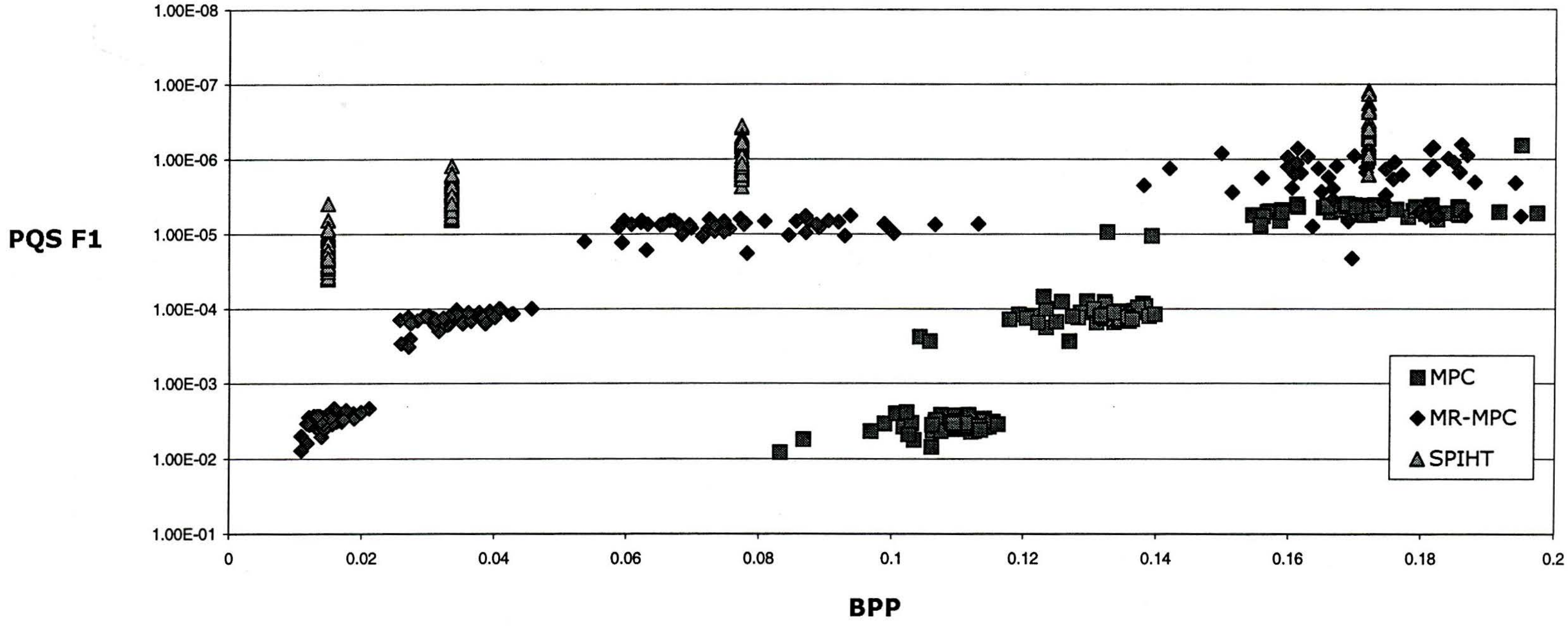
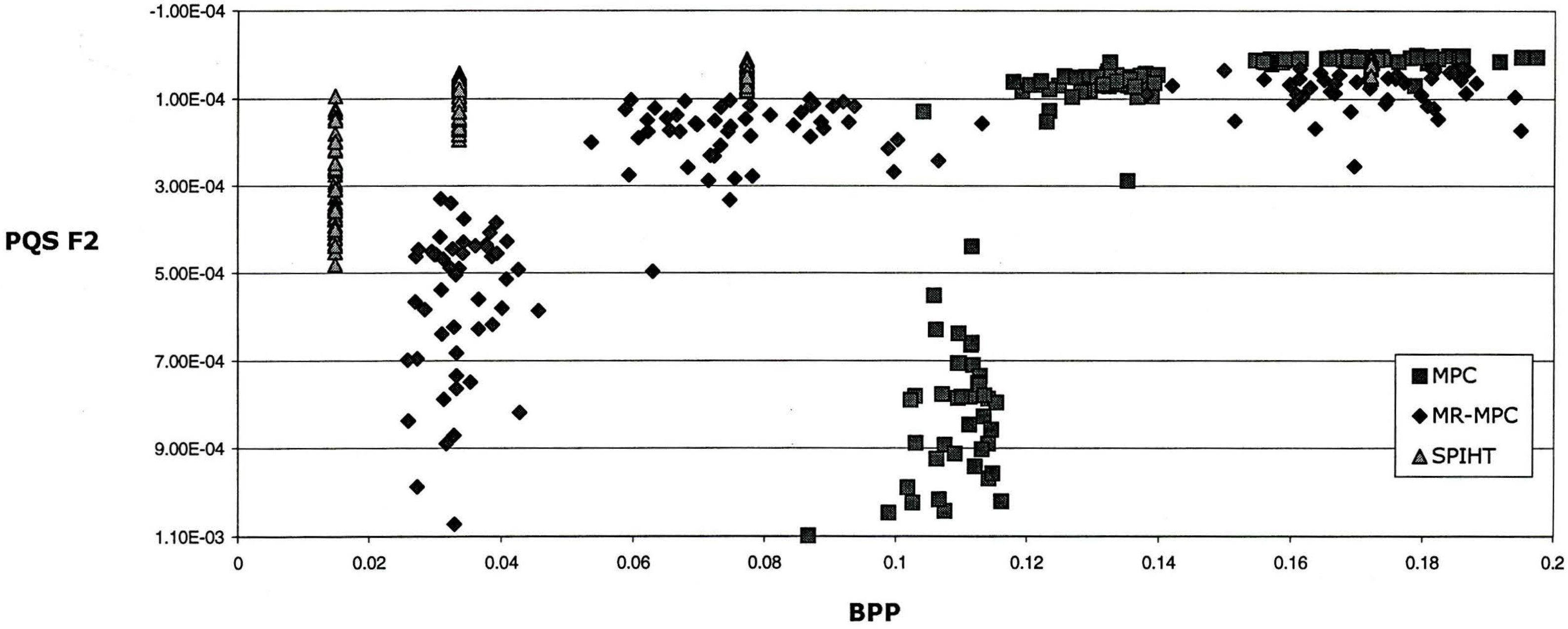

PQS F1
1.00E-08
1.00E-07
1.00E-06
1.00E-05
1.00E-04
1.00E-03
1.00E-02
1.00E-01
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2
BPP
■ MPC
◆ MR-MPC
▲ SPIHT

Figure 4.4: This plot compares MPC, MR-MPC and SPIHT using PQS factor two (F2) which measures visibility of random error using a model of human vision that ignores errors below threshold and accounts for human luminance and contrast sensitivity using Weber-Fechner Law and spatial frequency weighting. The plot shows how observed PQS F2 varied with bit rate (BPP). Each point corresponds to a single compressed radiograph. Lower values of PQS F2 (which occur near the top of the graph) indicate that less random error is visible.


PQS F2
-1.00E-04
1.00E-04
3.00E-04
5.00E-04
7.00E-04
9.00E-04
1.10E-03
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2
BPP
■ MPC
◆ MR-MPC
▲ SPIHT

67.5

Figure 4.5: This plot compares MPC, MR-MPC and SPIHT using PQS factor three (F3) which measures visibility of blocking artifacts. This plot shows how observed F3 varied with bit rate (BPP). Each point corresponds to a single compressed radiograph. Lower values of PQS F3 (which occur near the top of the graph) indicate that fewer blocking artifacts are visible.

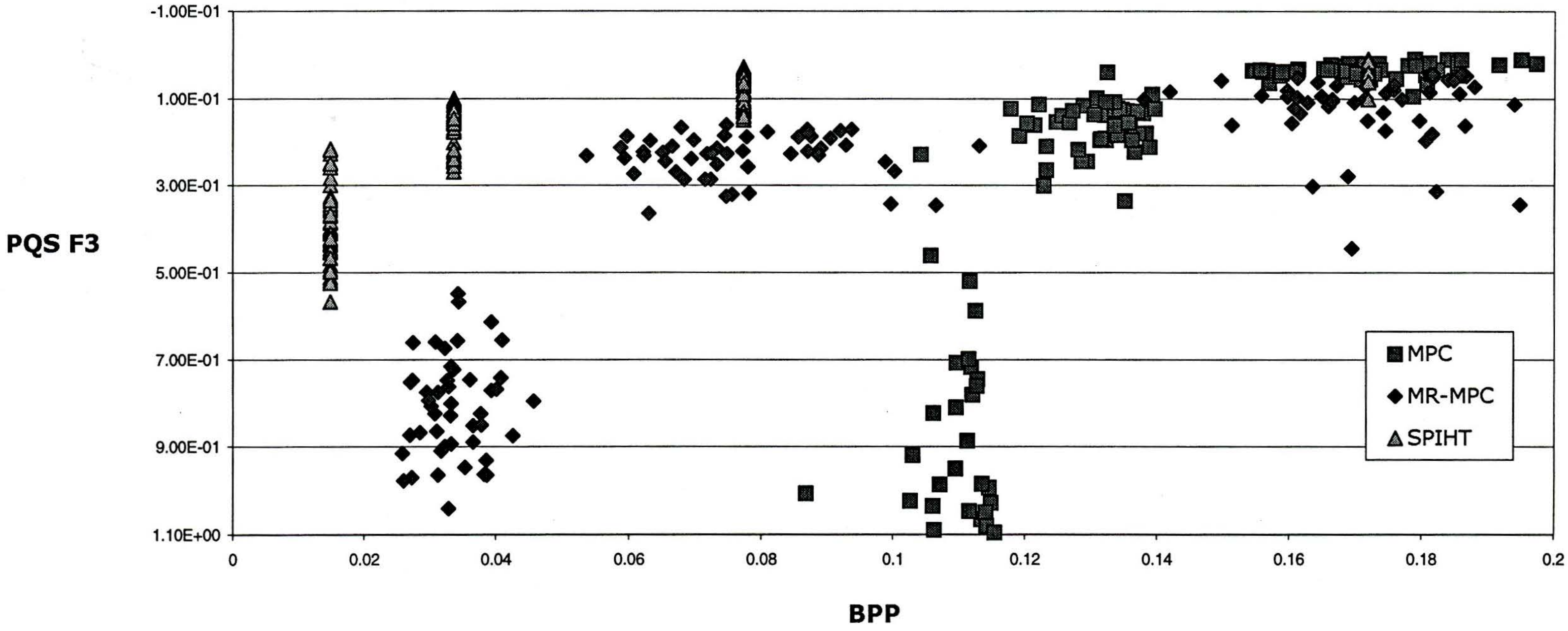

PQS F3
-1.00E-01
1.00E-01
3.00E-01
5.00E-01
7.00E-01
9.00E-01
1.10E+00
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2
BPP
■ MPC
◆ MR-MPC
▲ SPIHT

Figure 4.6: This plot compares MPC, MR-MPC and SPIHT using PQS factor four (F4) which measures visibility of error with high spatial correlation such as texture. This plot shows how observed F4 varied with bit rate (BPP). Each point corresponds to a single compressed radiograph. Lower values of PQS F4 (which occur near the top of the graph) indicate that less error with high spatial correlation is visible.

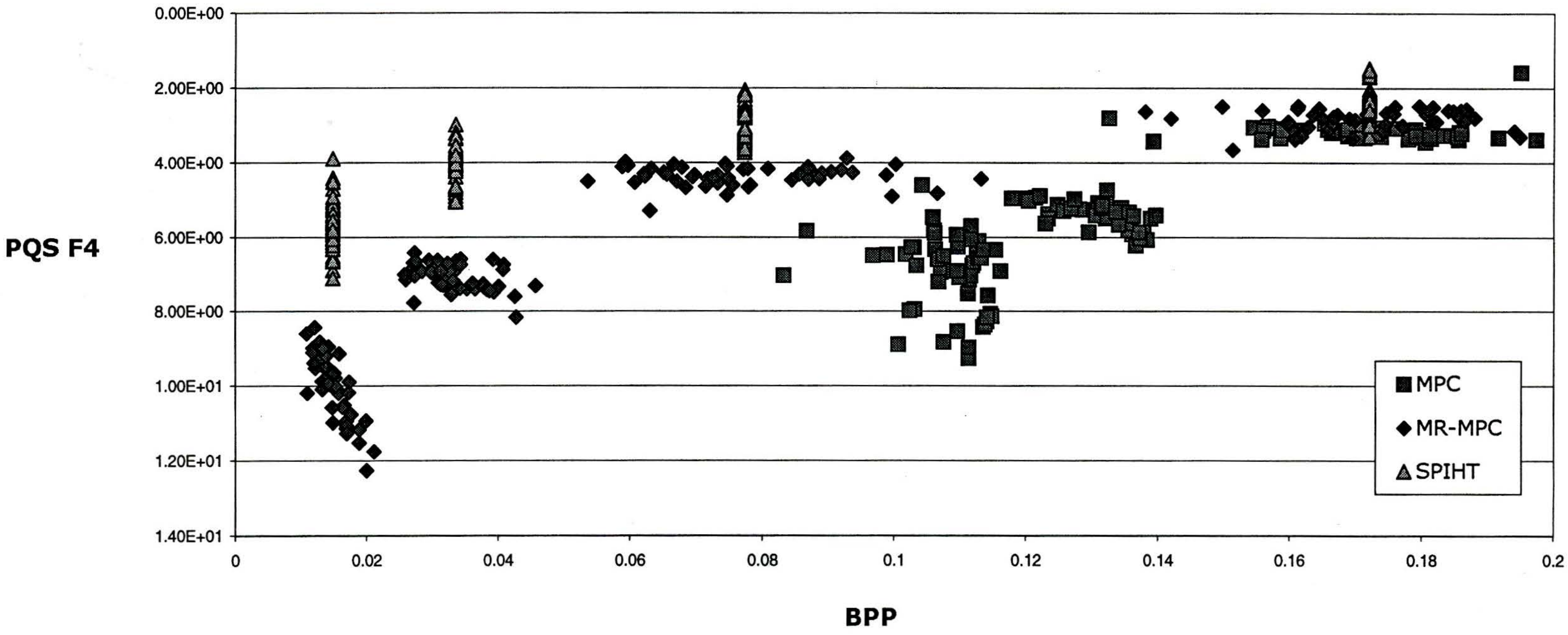
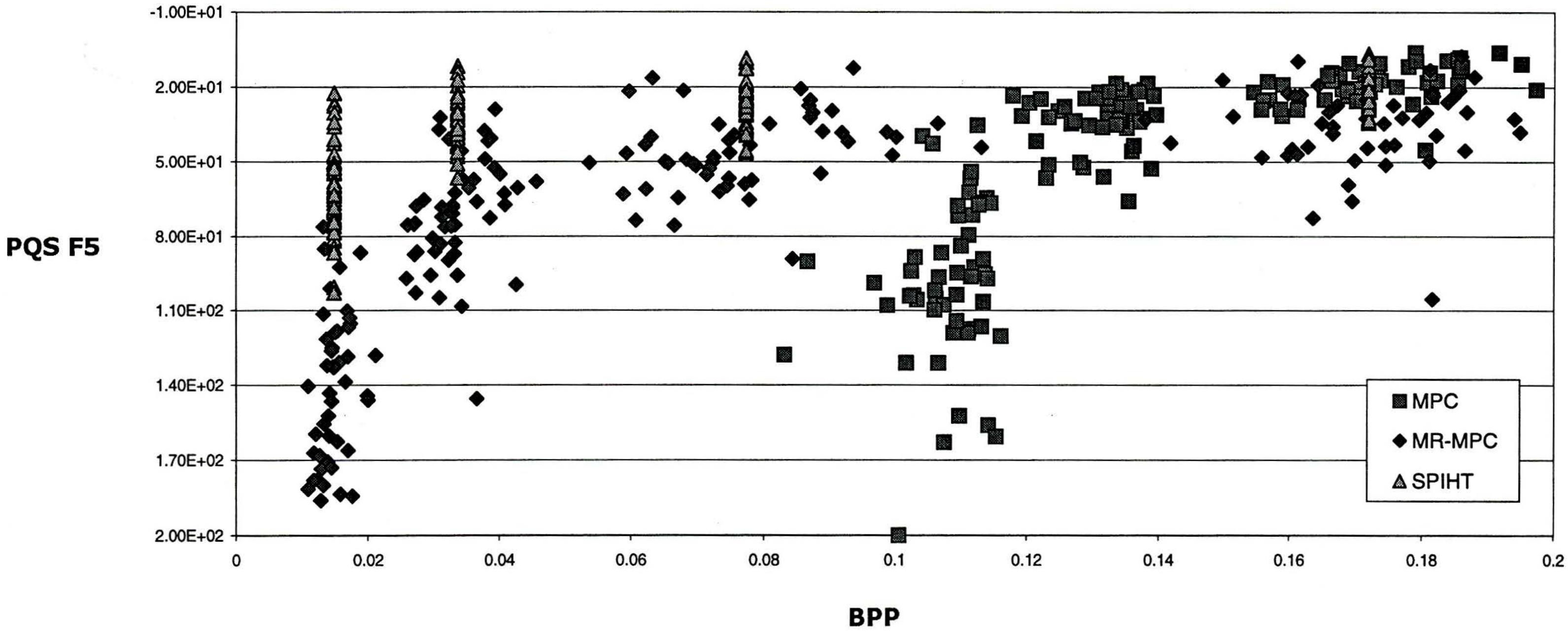

PQS F4
0.00E+00
2.00E+00
4.00E+00
6.00E+00
8.00E+00
1.00E+01
1.20E+01
1.40E+01
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2
BPP
■ MPC
◆ MR-MPC
▲ SPIHT

Figure 4.7: This plot compares MPC, MR-MPC and SPIHT using PQS factor five (F5) which measures errors such as edge misalignments in vicinity of high contrast transitions. This plot shows how observed PQS F5 varied with bit rate (BPP). Each point corresponds to a single compressed radiograph. Lower values of PQS F5 (which occur near the top of the graph) indicate that fewer edge misalignments are visible.


-1.00E+01
2.00E+01
5.00E+01
8.00E+01
1.10E+02
1.40E+02
1.70E+02
2.00E+02
PQS F5
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2
BPP
■ MPC
◆ MR-MPC
▲ SPIHT

performance advantage for real applications.

Chapter 5

Empirical Evaluation

The medical evaluations of lossy compression summarized in section 1.3 are all based on the same class of compression algorithm — wavelets — applied to the same type of images — radiographs. Why does each study report a different 'acceptable' compression ratio? There are at least two reasons why useful lossy compression ratios can vary.

- *The format of images being compressed can make a big difference.* For example, some images have uniform backgrounds or borders that are highly compressible. On disk images may be represented with more bits per pixel than are really significant. Image sampling density may be so high that images are easy to compress.

- *The method of assessment can also make a big difference.* Asking experts to judge whether 'diagnostically significant image degradation is present' [27] is not the same as asking experts to 'judge visibility of anatomic structures' [9] which is not the same as comparing experts ability to 'detect small uncalcified pulmonary nodules and fibrosis' [36].

Clearly, the best empirical assessment is one that most closely simulates the end use of compressed images [19]. Educational images meant to demonstrate a known pathology can be compressed more aggressively than diagnostic images containing pathology which we want to diagnose. The hard part however, is establishing what's acceptable. Verification of diagnostic accuracy by clinical testing is time consuming and expensive [13]. Moreover, *to get sufficient statistical power an estimated minimum of 12 radiologists and 520 patient studies are necessary* [19]. The empirical evaluation study we describe in this chapter is much less ambitious in its scope. We wanted to demonstrate that our method of image compression is as good as the excellent performing wavelet class of compression algorithms talked about in section 1.3 and 1.4.

5.1 Method

Volunteer radiologists scored radiographs compressed to three target compression ratios 25:1, 50:1, 75:1 using MR-MPC and SPIHT as described in section 4.3. MPC was not evaluated due to the clearly superior performance of SPIHT and MR-MPC demonstrated in sections 4.4.1 and 4.4.2. Four versions of each image — original, 25:1, 50:1, 75:1 — were printed in random order on radiograph film. Scores were collected on paper forms — figure 5.1 has sample — that specified the image quality scale: 1 – poor, 2 – below average, 3 – average, 4 – above average, 5 – excellent. The radiologists were not specially trained for the judging task nor were their responses calibrated to any scale. The viewing time, distance and lighting conditions were not constrained. Judges were simply asked:

> "Please score each radiograph independently from 1 to 5 according to image quality. Your assessment should be made based on your own professional standards in terms of image resolution, edge sharpness, noise, artifact, ability to visualize anatomical structures, and whether the image is of acceptable quality. Do not include image size or centering in your evaluation."

5.2 Results

Our goal was to compare MR-MPC compression to SPIHT compression using the quality scores assigned by judges to fifty images, each compressed to one of three target compression ratios — 25:1, 50:1, and 75:1. Using the data collected however, this could not be done directly because no single judge evaluated both types of compression. Our analysis is instead based on the frequency of scores and the frequency of score changes. It was carried out in six steps:

Figure 5.1: This is one of the nine identical evaluation forms each radiologist was asked to fill out by writing down the film number and check marking the quality of each image using a five point scale while viewing four versions of each image printed in random order on radiograph film. When evaluations were complete quality rankings were assigned back to images by looking film numbers up in a randomization table.

Data Compression Study Instructions

Please score each radiograph independently from 1 to 5 according to image quality
Your assessment should be made based on your own professional standards in terms of image resolution, edge sharpness, noise, artifact, ability to visualize anatomical structures, and whether the image is of acceptable quality
Do not include image size or centering in your evaluation

Film #	Image A B C D	1 poor	2 below average	3 average	4 above average	5 excellent
	A					
	B					
	C					
	D					
	A					
	B					
	C					
	D					
	A					
	B					
	C					
	D					
	A					
	B					
	C					
	D					
	A					
	B					
	C					
	D					
	A					
	B					
	C					
	D					

Radiologist:

- Our first step was to assign collected scores back to the known compression levels for each image. This is shown in table 5.1.
- Our second step was to use table 5.1 to tabulate compressed scores against uncompressed score frequencies. This is shown in table 5.2.
- Our third step was to compute and plot the marginal frequencies of scores in table 5.2. This is shown in figure 5.2.
- Our fourth step was to go back to the collected data in table 5.1 and compute the changes in scores after compression. This is shown in table 5.3.
- Our fifth step was to tabulate score changes from table 5.3 against uncompressed scores in table 5.1. This is shown in table 5.4.
- Our sixth step was to compute and plot marginal frequencies of score changes in table 5.4. This is shown in figure 5.3.

5.2.1 Step 1. Collected Scores

One radiologist (judge A) had reviewed MR-MPC images whereas three radiologists (judges B, C, and D) had reviewed SPIHT images. Although some radiologists missed some images, a total of 724 image quality scores were collected. Table 5.1 shows collected data for all four judges.

Table 5.1: This table shows quality scores for fifty chest radiographs compressed to three target compression levels — 25:1, 50:1, 75:1. One radiologist (judge A) evaluated MR-MPC images. Three radiologists (judges B, C, and D) evaluated SPIHT images. A total of 724 scores were collected. The blanks indicate missed images.

	JUDGE A				JUDGE B				JUDGE C				JUDGE D			
IMAGE	1:1	25:1	50:1	75:1	1:1	25:1	50:1	75:1	1:1	25:1	50:1	75:1	1:1	25:1	50:1	75:1
10086_1	1	1	1	1	1	1	1	1	2	2	1	1				
10812_1	2	2	2	2	3	2	1	1	3	2	2	2	3	3	3	3
11295_1	4	4	4	4	5	4	4	4	3	3	3	3	2	2	2	2
11719_1	2	2	2	2	3	2	1	1	2	2	2	2	2	2	2	2
11813_1	3	3	3	3	3	3	3	2	3	3	3	4				
12108_1	1	1	1	1	2	1	1	1	1	1	1	1	2	2	2	2
12108_3	3	3	3	3	2	2	3	1	4	3	3	3	3	3	3	3
12638_1	2	2	3	2	2	1	1	1	3	1	1	1				
12638_2	1	1	1	1	2	2	2	2	3	3	3	3				
13162_1	3	3	3	3	3	3	2	2	3	3	3	2				
13503_1	2	2	2	2	2	1	1	1	1	2	1	1				
13905_1	3	3	3	3	3	2	1	1	2	2	2	2	3	3	3	3
1411_1	3	2	2	2	2	1	1	1	3	2	2	1	2	2	2	2
15615_1	2	2	2	2	2	1	1	1	3	2	1	1				
15767_1	3	3	2	2	2	1	1	1	4	2	2	2	3	3	3	3
16631_1	4	4	4	4	4	5	3	3	3	3	3	3	3	3	3	3
16689_1	3	3	3	3	2	1	1	1	3	1	1	1	3	3	3	3
17169_1	3	3	3	3	3	4	2	1	4	3	3	3				
17877_1	2	2	3	3	2	1	1	1	3	2	1	1				
18559_1	3	3	3	3	4	4	4	4	3	3	3	3				
18564_1	4	4	4	4	2	1	1	1	4	3	3	3	2	2	2	2
18835_1					3	1	2	1	2	2	3	2	3	3	3	3
19158_1	1	1	1	1	3	3	2	2	3	3	2	2				
20067_1	3	3	4	4	3	2	2	2	3	2	2	2	5	5	5	5
2019_1	4	4	3	3	2	1	1	1	4	2	2	2	3	3	3	3
2151_1	3	3	3	4	5	4	4	4	4	3	4	3	6	6	6	6
21715_1	2	2	2	2	2	1	1	1	4	2	2	2	3	3	3	3
22098_1	3	2	2	3	2	2	1	1	2	2	2	2				
22199_1	3	2	2	3	2	1	1	1	3	2	1	1	3	3	3	3
22682_1	3	3	3	3	5	4	4	4	3	4	3	3	2	2	2	2
238_1	3	2	3	2	1	1	1	2	4	3	3	3	3	3	3	3
24074_1	4	4	4	4	4	3	3	3	3	3	3	3				
24248_1	3	3	3	3	5	4	1	1	3	3	3	3	3	2	3	3
24924_1	3	3	3	3	2	1	1	1	3	3	4	4	2	2	2	2
25519_1	3	3	3	3	4	1	1	1	3	2	2	1	2	2	2	2
25576_1	3	4	4	3	2	1	1	1	3	3	2	2	3	3	3	3
25584_1	2	2	3	2	2	1	1	1	2	1	1	1				
25584_2	3	3	3	3	2	1	1	1	2	2	2	1				
25609_1	4	3	3	4	2	1	1	1	2	1	1	1	3	3	3	3
25609_2	3	3	2	2	2	1	1	1	3	1	1	2	3	3	3	3
2919_1	2	2	2	2	2	1	1	1	2	1	1	1				
2944_1	3	2	3	2	3	2	2	2	2	2	1	1	3	3	3	3
29855_1	2	2	2	2	2	1	1	1	2	3	1	2	2	2	2	2
31961_1	2	2	2	2	3	2	1	1	3	2	2	2	2	2	2	2
31961_2	4	4	4	4	3	2	1	1	4	3	2	3				
5418_1					4	3	3	3	4	4	4	4				
8028_1					3	2	2	2	4	3	3	3	2	2	2	2
8144_1	3	3	3	3	4	3	3	3	4	4	4	4	3	3	3	3
8338_1	3	2	2	2	2	1	1	1	3	2	2	2	2	2	2	2
9420_1	2	3	2	3	4	3	3	3	4	4	4	4	3	3	3	3

5.2.2 Step 2. Compressed vs. Uncompressed Scores

The goal of this step was to compare for each image and judge compressed scores with uncompressed scores. Table 5.2 shows frequency of compressed scores in rows against uncompressed scores in columns for all compression levels (25:1, 50:1, 75:1), for each judge (A, B, C, D), and pooled for MR-MPC, SPIHT and entire STUDY.

In each subtable of table 5.2:

- The above-diagonal entries represent scores that dropped after compression. For example, at 25:1 for judge B, four images originally 'excellent' (score 5) dropped to 'above average' (score 4).
- The diagonal-entries represent scores that remained same after compression. For example, at 50:1 for judge A, 16 images retained a score of 'acceptable' (score 3);
- The below-diagonal entries represent scores that went up after compression. For example, at 75:1 for judge C, 2 images originally 'acceptable' (score 3) improved to 'above average' (score 4).
- The entries with zero frequency are blank.

From table 5.2 we made two observations:

- The score frequency pattern of judge D is different from that of judges A, B and C. It shows less spread and is consistent across compression levels. We took this to mean that judge D was insensitive to all levels of compression.

- Most score frequencies for judge A, B and C are either on or above the diagonal. This means that for these judges scores after compression either remained the same or dropped.

5.2.3 Step 3. Score Distributions

Both MR-MPC and SPIHT were expected to reduce scores after compression and indeed this is what we saw — in table 5.2 far more entries were above the diagonal — indicating a drop in score, than below the diagonal — indicating an increase in score. To see this effect more clearly we computed and plotted the marginal frequencies — row and column sums. This is shown in figure 5.2 for uncompressed scores, all compression levels (25:1, 50:1, 75:1), for each judge (A, B, C, D), and pooled for MR-MPC, SPIHT and entire STUDY. The first row of histograms is based on column totals of subtables in table 5.2, the remaining rows of histograms are based on row totals of subtables in table 5.2.

The histograms in figure 5.2 revealed what score distributions looked like and allowed us to make two more observations:

- The first observation was that *images differed in quality even before compression was applied.* For example, instead of 50 images untouched by compression

Table 5.2: This table shows compressed vs. uncompressed score frequency for all compression levels (25:1, 50:1, 75:1), for each judge (A, B, C, D), and pooled for MR-MPC, SPIHT and entire STUDY.

		JUDGE A					JUDGE B					JUDGE C					JUDGE D					MR-MPC					WAVELET					STUDY				
		1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
25:1	1	4					2	21	1	1		1	3	3								4					3	24	4	1		7	24	4	1	
	2		11	6				3	8			1	7	9	3			13	1				11	6			1	23	18	3		1	34	24	3	
	3		1	17	1				3	4			1	11	7				20				1	17	1			1	34	11			2	51	12	
	4			1	6				1	1	4			1	3									1	6				2	4	4			3	10	4
	5									1																				1					1	
50:1	1	4					2	22	5	1	1	2	6	6								4					4	28	11	1	1	8	28	11	1	1
	2		9	6				1	7				4	8	4			13					9	6				18	15	4			27	21	4	
	3		3	16	2			1	1	5			1	9	5				21				3	16	2			2	31	10			5	47	12	
	4			2	5					1	3			1	4									2	5				1	5	3			3	10	3
	5																																			
75:1	1	4					1	23	7	1	1	2	6	7								4					3	29	14	1	1	7	29	14	1	1
	2		10	6			1	1	6				5	8	3			13					10	6			1	19	14	3		1	29	20	3	
	3		2	16	1					5				7	7				21				2	16	1				28	12			2	44	13	
	4			2	6					1	3			2	3									2	6				2	4	3			4	10	3
	5																																			

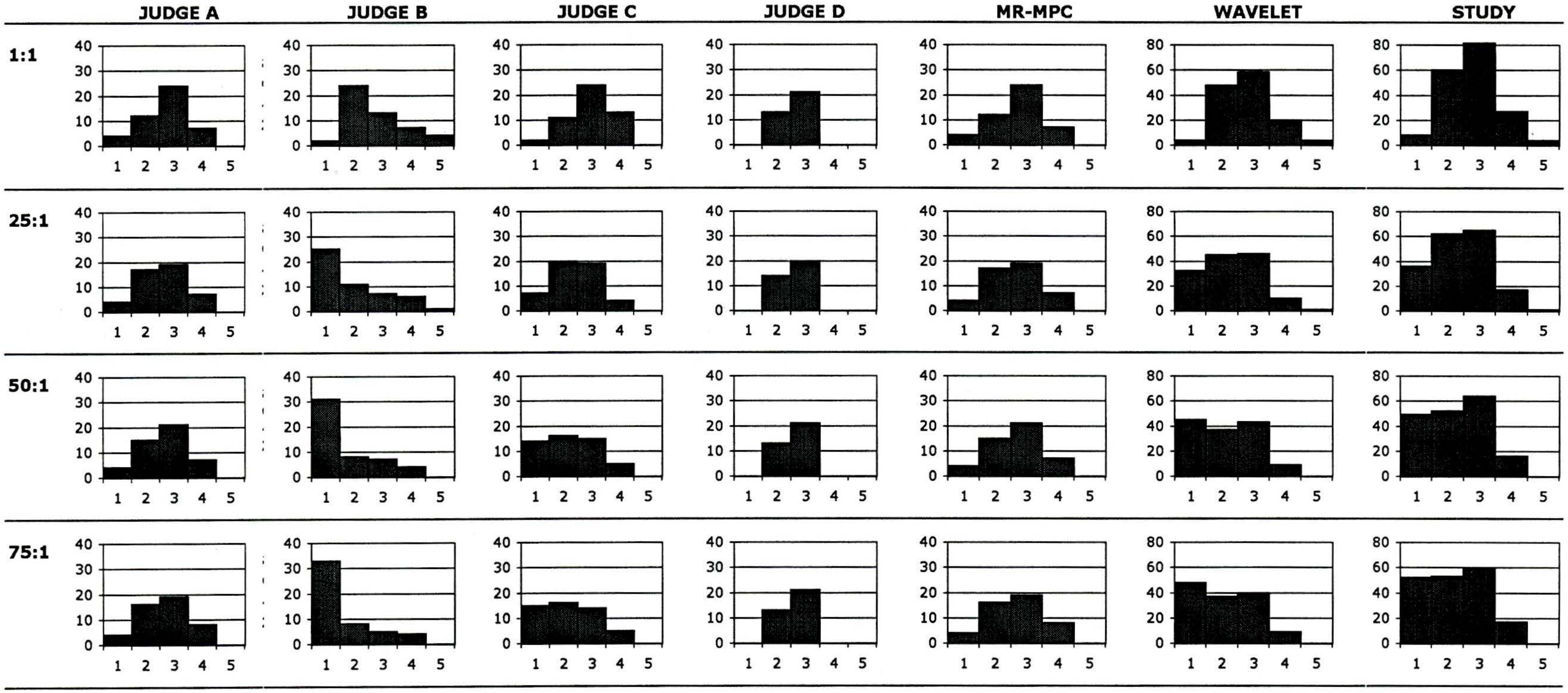
having the same score, judge C considered 13 'above average', 24 'average', 11 'below average', and 2 'poor'.

- The second observation was that *score distributions varied among judges.* For example, uncompressed image median score for judges A, C, and D is 'average' but 'below average' for judge B. Moreover, uncompressed score frequency distributions of judge A, C and D are roughly symmetric but distributions of judge B are negatively skewed. This variability suggests that radiologists use different criteria to judge image quality, and as a consequence, scores between judges may not mean the same thing.

5.2.4 Step 4. Compressed Score Differences

Score distribution histograms in figure 5.2 revealed two types of variability. One type of variability was in score distributions among judges. This variability suggested that radiologists use different criteria to judge image quality and that scores between judges may not mean the same thing. If this is true we must take caution in interpreting scores pooled across judges — ie. those of SPIHT and STUDY. The other type of variability was among uncompressed images. This variability indicated that images differed in quality even before compression was applied. We can compensate in part for this type of variability by looking at the difference in scores of compressed images relative to their uncompressed counterparts — this is shown in table 5.3 which is based on table 5.1.

Figure 5.2: These histograms show score frequency distributions for uncompressed images, all compression levels (25:1, 50:1, 75:1), for each judge (A, B, C, D), and pooled for MR-MPC, SPIHT and entire STUDY. The first row of histograms is based on column totals of subtables in table 5.2, the remaining rows of histograms are based on row totals of subtables in table 5.2.



JUDGE A
JUDGE B
JUDGE C
JUDGE D
MR-MPC
WAVELET
STUDY

1:1
25:1
50:1
75:1

Table 5.3: This table shows differences between compressed and uncompressed scores and is based on table 5.1. Looking at difference in scores instead of absolute scores compensates, in part, for the fact that images differed in quality even before compression was applied.

IMAGE	JUDGE A			JUDGE B			JUDGE C			JUDGE D		
	25:1	50:1	75:1	25:1	50:1	75:1	25:1	50:1	75:1	25:1	50:1	75:1
10086_1	0	0	0	0	0	0	0	1	1			
10812_1	0	0	0	1	2	2	1	1	1	0	0	0
11295_1	0	0	0	1	1	1	0	0	0	0	0	0
11719_1	0	0	0	1	2	2	0	0	0	0	0	0
11813_1	0	0	0	0	0	1	0	0	-1			
12108_1	0	0	0	1	1	1	0	0	0	0	0	0
12108_3	0	0	0	0	-1	1	1	1	1	0	0	0
12638_1	0	-1	0	1	1	1	2	2	2			
12638_2	0	0	0	0	0	0	0	0	0			
13162_1	0	0	0	0	1	1	0	0	1			
13503_1	0	0	0	1	1	1	-1	0	0			
13905_1	0	0	0	1	2	2	0	0	0	0	0	0
1411_1	1	1	1	1	1	1	1	1	2	0	0	0
15615_1	0	0	0	1	1	1	1	2	2			
15767_1	0	1	1	1	1	1	2	2	2	0	0	0
16631_1	0	0	0	-1	1	1	0	0	0	0	0	0
16689_1	0	0	0	1	1	1	2	2	2	0	0	0
17169_1	0	0	0	-1	1	2	1	1	1			
17877_1	0	-1	-1	1	1	1	1	2	2			
18559_1	0	0	0	0	0	0	0	0	0			
18564_1	0	0	0	1	1	1	1	1	1	0	0	0
18835_1				2	1	2	0	-1	0	0	0	0
19158_1	0	0	0	0	1	1	0	1	1			
20067_1	0	-1	-1	1	1	1	1	1	1	0	0	0
2019_1	0	1	1	1	1	1	2	2	2	0	0	0
2151_1	0	0	-1	1	1	1	1	0	1	0	0	0
21715_1	0	0	0	1	1	1	2	2	2	0	0	0
22098_1	1	1	0	0	1	1	0	0	0			
22199_1	1	1	0	1	1	1	1	2	2	0	0	0
22682_1	0	0	0	1	1	1	-1	0	0	0	0	0
238_1	1	0	1	0	0	-1	1	1	1	0	0	0
24074_1	0	0	0	1	1	1	0	0	0			
24248_1	0	0	0	1	4	4	0	0	0	1	0	0
24924_1	0	0	0	1	1	1	0	-1	-1	0	0	0
25519_1	0	0	0	3	3	3	1	1	2	0	0	0
25576_1	-1	-1	0	1	1	1	0	1	1	0	0	0
25584_1	0	-1	0	1	1	1	1	1	1			
25584_2	0	0	0	1	1	1	0	0	1			
25609_1	1	1	0	1	1	1	1	1	1	0	0	0
25609_2	0	1	1	1	1	1	2	2	1	0	0	0
2919_1	0	0	0	1	1	1	1	1	1			
2944_1	1	0	1	1	1	1	0	1	1	0	0	0
29855_1	0	0	0	1	1	1	-1	1	0	0	0	0
31961_1	0	0	0	1	2	2	1	1	1	0	0	0
31961_2	0	0	0	1	2	2	1	2	1			
5418_1				1	1	1	0	0	0			
8028_1				1	1	1	1	1	1	0	0	0
8144_1	0	0	0	1	1	1	0	0	0	0	0	0
8338_1	1	1	1	1	1	1	1	1	1	0	0	0
9420_1	-1	0	-1	1	1	1	0	0	0	0	0	0

5.2.5 Step 5. Differences vs. Uncompressed Scores

The differences computed in the previous step can, for a given image and judge, be attributed to two sources. One source is observer error which, because in our study images were printed on film in scrambled order and radiologists were not told which images were which, is just as likely to improve as drop scores. The other source is the effect of compression which at low levels may improve scores[1], but will almost always make scores drop at higher levels. We suspected that the value of uncompressed image score also might play a role. To find out we tabulated score differences from table 5.3 against uncompressed scores from table 5.1. This is shown in table 5.2.

5.2.6 Step 6. Difference Distributions

Score changes are subject to clipping at scale borders. For example, images originally rated 'poor' cannot drop in score, images originally rated 'below average' can only drop to 'poor', and so on. Similarly, images originally rated 'excellent' cannot go up in score, images rated 'above average' can only go up to 'excellent' after compression, and so on. For this reason the changes in scores after compression must be interpreted with caution. For example, the frequency distributions of judge B, which appear in the second column of histograms in figure 5.2 are sharply peaked at the lowest end of

[1] At low ratios lossy compression low pass filters the high frequency noise that is inherent to process of image acquisition — this can improve perceived image quality. For example, two of the three studies summarized in section 1.3 found that "at low compression (10:1) there was a *slight preference* for [lossy] compressed radiographs" [9] and "readers performed *better* on images [lossy] compressed at 40:1 compared with uncompressed images" [36].

Table 5.4: This table shows score differences (from table 5.3) in rows against uncompressed scores (from table 5.1) in columns. For example, 11 uncompressed scores of 'below average' (score = 2) for judge A at 25:1 compression, had a score changes of 0 (ie, did not change) whereas 6 uncompressed scores of 'average' (score = 3) for judge B had a score changes of -1 (ie, dropped to 'below average').

		JUDGE A					JUDGE B					JUDGE C					JUDGE D					MR-MPC					WAVELET					STUDY				
		1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
25:1	-4																																			
	-3									1																				1					1	
	-2								1					3	3														4	3				4	3	
	-1			6	1			21	8	4	4		3	9	7				1					6	1			24	18	11	4		24	24	12	4
	0	4	11	17	6		2	3	3	1		1	7	11	3			13	20			4	11	17	6		3	23	34	4		7	34	51	10	
	1		1	1					1	1		1	1	1									1	1			1	1	2	1		1	2	3	1	
50:1	-4										1																				1					1
	-3									1																				1					1	
	-2								5					6	4														11	4				11	4	
	-1			6	2			22	7	5	3		6	8	5									6	2			28	15	10	3		28	21	12	3
	0	4	9	16	5		2	1	1	1		2	4	9	4			13	21			4	9	16	5		4	18	31	5		8	27	47	10	
	1		3	2				1					1	1									3	2				2	1				5	3		
75:1	-4										1																				1					1
	-3									1																				1					1	
	-2								7					7	3														14	3				14	3	
	-1			6	1			23	6	5	3		6	8	7									6	1			29	14	12	3		29	20	13	3
	0	4	10	16	6		1	1		1		2	5	7	3			13	21			4	10	16	6		3	19	28	4		7	29	44	10	
	1		2	2			1							2									2	2			1		2			1	2	4		

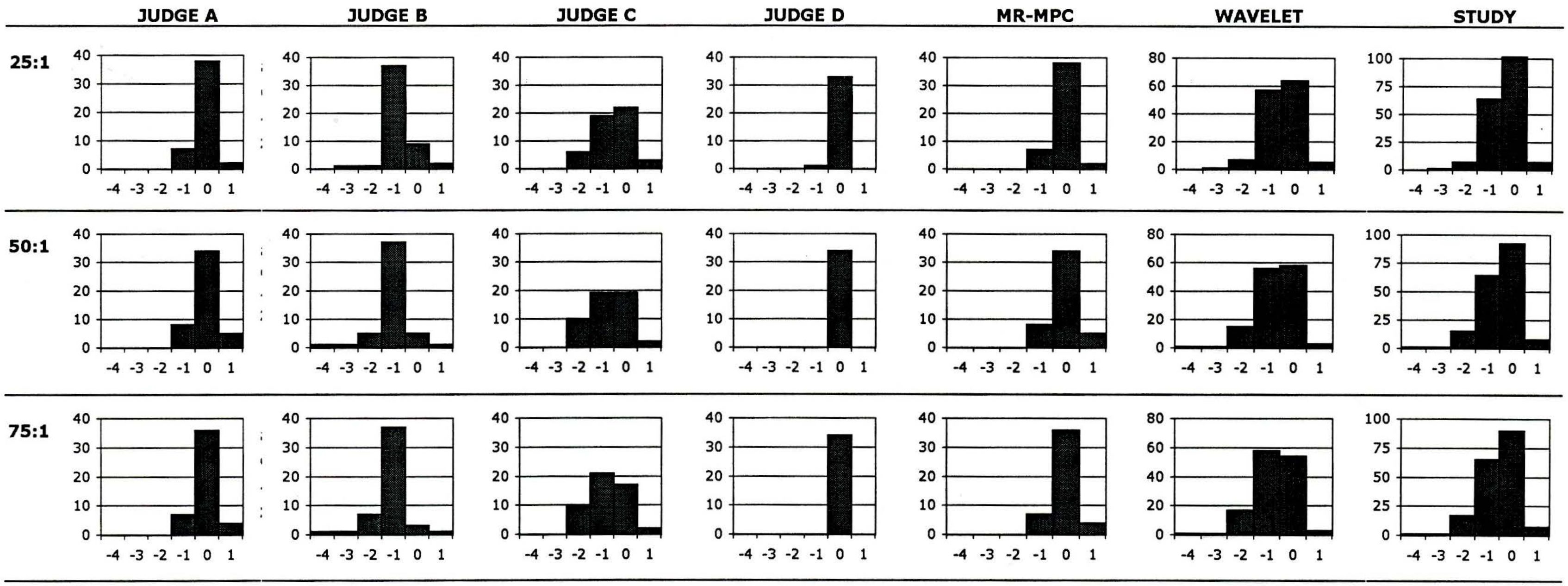
the scale. Judge B probably ranked most compressed images 'poor' because he could not rank them any lower.

From histograms in figure 5.3 we made two final observations. The first observation was that regardless of amount of compression, judge A (MR-MPC) and judge D (SPIHT) always saw little or no difference between compressed and uncompressed images. The second observation was that regardless of compression, judges B and C (SPIHT) consistently saw a difference of 1-2 points which, in most cases for judge B and in some cases for judge C, was the most scores could drop due to compression. To see this, refer also to figure 5.2 which shows somewhat for judge C and mostly for judge B, scores collecting in the gutter at lowest end of the image quality scale.

5.3 Summary

The effect of lossy compression at all levels was too small on scores of judges A and D and too large on scores of judges B and C to meaningfully compare MR-MPC and SPIHT. The only conclusion we can make is that the difference between of MR-MPC and SPIHT compression is less than the difference of opinion between four radiologists.

Figure 5.3: These histograms show frequencies of score differences and are based on marginals from the tables in figure 5.4. From these histograms we made two final observations. The first observation was that regardless of amount of compression, judge A (MR-MPC) and D (SPIHT) always saw little or no difference between compressed and uncompressed images. The second observation was that regardless of compression, judges B and C consistently saw a difference of 1-2 points which, in most cases for judge B and in some cases for judge C, was the most scores could drop due to compression. To see this, refer also to figure 5.2 which shows somewhat for judge A and mostly for judge B, scores collecting in the gutter at lowest end of the image quality scale.


JUDGE A
JUDGE B
JUDGE C
JUDGE D
MR-MPC
WAVELET
STUDY
25:1
50:1
75:1
0
10
20
30
40
60
80
25
50
75
100
-4
-3
-2
-1
0
1

Chapter 6

Summary

6.1 What

The main contribution of this thesis is a new method of image compression based on a multi-resolution extension of a recently developed adaptive transform called Mixtures of Principal Components [1] (MPC) [15, 16]. Although MPC compression is superior to compression techniques based on and related to[2] the optimal Karhunen-Loève Transform[3] (KLT), MPC does not however, like the best image compression algorithms in existence [19, 34, 33], exploit repetition present inbetween multiple resolution of an image. Our goal was to improve MPC compression by extending it to process images at multiple resolutions.

[1] How MPC works is explained in sections 2.2.4, 3.3.6, 3.3.7, and 3.3.8

[2] For example, the discrete cosine transform (DCT) used in JPEG

[3] How KLT works is explained in section 2.2.2.

6.2 Why

The motivation for our work is fourfold. First, there exists the potential to boost image transmission throughput on all present day communication links. This is important for the cost effective deployment of teleradiology today. Second, there is the potential to conserve bandwidth in the crowded wireless communication spectrum. This will be important for itinerant teleradiology systems in the future. Third, there is the potential to increase image capacities of all present day storage media. This is important for storage strained PACS today. Fourth, there is the potential to boost all future storage media improvements multiplicatively — for example a mere tenfold improvement in hardware multiplied by 10:1 compression gives a hundredfold improvement. This will be important for PACS in the future.

6.3 Method

Our multi-resolution extension of MPC — called Multi-Resolution Mixtures of Principal Components (MR-MPC) — is based on two key observations[4]. The first observation is that *high frequency image detail is predictable from lower resolutions.* The second observation is that the *detail not predictable from lower resolutions is easy to compress.* MR-MPC applies these observations by compressing and decompressing images in stages. The first stage processes the original images at very low resolution

[4] Both observations are developed in section 3.1.1 and illustrated in figure 3.3.

and is followed by stages that process the encoding errors of the previous stages at incrementally higher resolutions.

The job of MR-MPC is twofold — *compression* and *decompression*. Compression progressively encodes the original image to a compressed stream. This is explained in section 3.2.1 and figure 3.4. Decompression progressively decodes the compressed stream and updates an approximation of the original image. Decompression is explained in section 3.2.2 and figure 3.6. Figure 3.7 shows how compression and decompression work together and figures 3.5 and 3.8 show what happens to an image as it is progressively compressed and decompressed.

6.4 Results

To evaluate our multi-resolution extension of MPC, we compared it with MPC and with the excellent performing wavelet based scheme called SPIHT. Fifty chest radiographs were compressed and compared to originals in two ways. In chapter 4 we used PSNR[5] and five distortion factors from a perceptual distortion measure called PQS[6]. These numerical measures of distortion demonstrate — in figures 4.2 and 4.3 to 4.7 — that our multi-resolution extension of MPC can achieve rate distortion performance that is 220% to 720% better than MPC and much closer to that of SPIHT. In chapter 5, the performance of our new method was also investigated empirically in a study

[5] PSNR is explanation in section 4.2.3. PSNR results are in section 4.4.1 and figure 4.2.

[6] PQS is discussed in section 4.2.4. PQS results are in section 4.4.2 and figures 4.3 to 4.4.

involving 724 radiologists' evaluations of compressed chest radiographs. We found the effect of lossy compression at all levels was too small on scores of two judges and too large on scores of the other two judges to meaningfully compare MR-MPC and SPIHT. The only conclusion we could make was that the difference between MR-MPC and SPIHT compression is less than the difference of opinion between four radiologists.

Bibliography

[1] Parasyn A., Hanson RM., Peat JK., and De Silva M. A comparison between digital images viewed on a picture archiving and communication system diagnostic workstation and on a pc-based remote viewing system by emergency physicians. *Journal of Digital Imaging*, 11(1), February 1998.

[2] V. Ralph Algazi, Yoshiaki Kato, Mikato Miyahara, and Kazanori Kotani. Distortion factors and quality metrics in still image coding. In *PCS*, 1993.

[3] V. Ralph Algazi, Mikato Miyahara, Kazanori Kotani, Hideo Ohira, and Yoshiaki Kato. Comparison of image coding techniques with a picture quality scale. *Proceedings of the SPIE*, 1771, 1992.

[4] V. Ralph Algazi, Mikato Miyahara, Kazanori Kotani, Hideo Ohira, and Yoshiaki Kato. Comparison of image coding techniques with a picture quality scale. *Proceedings of the SPIE*, 1771, 1992.

[5] V. Ralph Algazi, Mikato Miyahara, Kazanori Kotani, Hideo Ohira, and Yoshiaki Kato. Important distortion factors in the encoding of very high quality images. In *PCS*, 1994.

[6] V. Ralph Algazi, Mikato Miyahara, Kazanori Kotani, Hideo Ohira, and Yoshiaki Kato. Comparison of wavelet image coders using the picture quality scale. *Proceedings of the SPIE*, 2491:1119–1130, April 1995.

[7] V. Ralph Algazi, T. R. Reed, G. E. Ford, and Yoshiaki Kato. Comparison of image coding techniques for high quality. In *SID*, Seattle, Washington, May 1993.

[8] R.A. Bednarek. On evaluation of impaired television pictures by subjective measurements. *IEEE Transactions on Broadcasting*, BC-25:41–46, 1979.

[9] Erickson BJ., Manduca A., Persons KR., Earnest F 4th., Hartman TE., Harms GF., and Brown LR. Evaluation of irreversible compression of digitized posterior-

anterior chest radiographs. *Journal of Digital Imaging*, 10(3):97–102, August 1997.

[10] P. J. Burt and E. H. Adelson. The laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31:532–540, 1983.

[11] Kuni Christopher C. *Introduction to Computers and Digital Processing in Medical Imaging.* Year Book Medical Publishers Inc., Chicago, 1988.

[12] Ruggiero C. Teleradiology: a review. *Journal of Telemedicine and Telecare*, 4(1):25–35, 1998.

[13] P. C. Cosman, C. Tseng, R. M. Gray, et al. Tree-structured vector quantization of ct chest scans: Image quality and diagnostic accuracy. *IEEE Trans Medical Imaging*, 12:727–739, December 1993.

[14] D.J.Sakrison. On the role of the observer and a distortion measure in image transmission. *IEEE Transactions on Communications*, COM-25, November 1977.

[15] Robert Dony. *Adaptive Transform Coding of Images Using a Mixture of Principal Components.* PhD thesis, McMaster University, 1995.

[16] Robert Dony and Simon Haykin. Optimally adaptive transform coding. *IEEE Transactions on Image Processing*, 4:1358–1370, October 1995.

[17] Aberle DR, Glesson F, Sayre JW, et al. The effect of irreversable image compression of diagnostic accuracy in thoracic imaging. *Invest. Radiol.*, 28:398–403, 1993.

[18] A. M. Eskicioglu and P. S. Fisher. Image quality measures and their performance. *IEEE Transactions on Communications*, 43:2959–2965, December 1995.

[19] Robert M. Gray, Richard Olshen, S. M. Perlmutter, Pamela C. Cosman, Debra Ikeda, Robin Birdwell, et al. Compression and classification of digital mammograms for storage, transmission and computer aided screening. *http://www-isl.stanford.edu/ gray/army.html*, March 1997.

[20] I. T. Jolliffe. *Principal Component Analysis.* Springer-Verlag, New York, 1996.

[21] John D. Newell Jr. and Charles A. Kelsey. *Digital imaging in diagnostic radiology.* Churchhill Livingstone, New York, 1990.

[22] J.O. Limb. Distortion criteria of the human viewer. *IEEE Transactions of System, Man, and Cybernetics*, SMC-9:778–793, December 1979.

[23] Jean loup Gailly. *FAQ from the internet newsgroup comp.compression.* jloup@gzip.org, http://www.faqs.org/faqs/compression-faq, 1998.

[24] Fuji Photo Film Co. Ltd. *Fuji Computed Radiography System FCR AC-1 Product Specification*, volume 2nd Edition. Fuji Photo Film Co. Ltd., Tokyo, Japan, 1990.

[25] F.X.J. Lukas and Z.L. Budricks. Picture quality prediction based on a visual model. *IEEE Transactions on Communications*, COM-30:1679–1692, July 1982.

[26] Antonini M, Barlaud M, Mathieu P, et al. Image coding using wavelet transform. *IEEE Trans Image Proc*, 1:205–220, April 1992.

[27] Goldberg MA., Pivovarov M., Mayo-Smith WW., Bhalla MP., Blickman JG., Bramson RT., Boland GW., Llewellyn HJ., and Halpern E. Application of wavelet compression to digitized radiographs. *American Journal of Roentgenology*, 163(2):463–468, August 1994.

[28] Makato Miyara, K Kazunori Kotani, and V. Ralph Algazi. Objective picture quality scale (pqs) for image coding. In *Society for Information Display*, Boston, Mass., 1992.

[29] Makato Miyara, K Kazunori Kotani, and V. Ralph Algazi. Objective picture quality scale (pqs) for image coding. *IEEE Transactions on Communications*, preprint.

[30] N. M. Nasrabadi and R. A. King. Image coding using vector quantization: A review. *IEEE Transactions on Communications*, 36:957–971, August 1988.

[31] A. N. Netravali and B. G. Haskell. *Digital Pictures: Representation and Compression.* Plenum, New York, 1998.

[32] Hangiandreou N.J., King B.F., Swenson A.R., and Webbles W.E. Picture archive and communication systems implementation in a community medicine practice. *Journal of Digital Imaging*, 10(3 Suppl 1):36–37, August 1997.

[33] Amir Said and William Pearlman. A new fast and efficient image codec based on set partitioning in hierarchical trees. *IEEE Trans. Circuits and Systems for Video Tech*, 6:243–250, 1996.

[34] Jerome M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Trans. Signal Processing*, 41:3445–3462, December 1993.

[35] Patric C. Teo and David J. Heeger. Perceptual image distortion. *First International Conference on Image Processing*, 2:982–986, November 1994.

[36] Savcenko V., Erickson BJ., Palisson PM., Persons KR., Manduca A., Hartman TE., Harms GF., and Brown LR. Detection of subtle abnormalities on chest radiographs after irreversible compression. *Radiology*, 206(3):609–616, March 1998.

[37] J. Villasenor, B. Belzer, and J Liao. Wavelet filter evaluation for image compression. *IEEE Transations on Image Processsing*, 1:1053–1060, August 1995.