Visual Perception Based Bit Allocation for Low Bitrate Video Coding

by

Rajesh Suryadevara

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Computer Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 1996

Author .....
Department of Electrical Engineering and Computer Science
May 28, 1996

Certified by .....
V. Michael Bove
Associate Professor
Thesis Supervisor

Accepted by .....
Frederic R. Morgenthaler
Chairman, Departmental Committee on Graduate Students

Visual Perception Based Bit Allocation for Low Bitrate Video Coding

by

Rajesh Suryadevara

Submitted to the Department of Electrical Engineering and Computer Science
on May 28, 1996, in partial fulfillment of the
requirements for the degree of
Master of Science in Computer Science and Engineering

Abstract

In this thesis an algorithm which makes spatial bit allocation decisions in low bitrate video coding based on perceptual considerations is developed and implemented. The current state of the art in video compression is pushing against the limits of what is achievable because it employs mainly a statistical method. Coding errors in highly textured areas of an image are relatively less noticeable than errors in areas of little texture. This perceptual effect can be used to reduce errors in areas where they are more visible and increase them where they are relatively less so. The algorithm presented here determines the significance of errors and uses this to modify the quantization scheme. The algorithm is used to code several video sequences at low bitrates. The improvements in perceptual quality are illustrated in still frames taken from the video sequences.

Thesis Supervisor: V. Michael Bove
Title: Associate Professor

Acknowledgments

I would like to thank my colleagues at the David Sarnoff Research Center for their help and support. Ya-Qin Zhang for his encouragement and leadership of the project which supported this work. Steve Hsu for many fruitful discussions and technical help. Iraj Sodagar and Steve Martucci for their help in implementation and evaluation of the results. Michal Irani for leading me to the idea behind this work and P. Anandan for letting me work on several projects to gain a better technical understanding of this area of research.

I would also like to thank Professor V. Michael Bove for his valuable feedback about the overall work and the presentation of the results.

Contents

1 Introduction **9**

1.1 Motivation 9

1.2 Image Format 10

1.3 Significance Map 10

1.4 Quantization 11

2 Visual Perception of Coding Errors **12**

2.1 Visual Perception Based Image Coding 12

2.2 Useful Effects of Visual Perception 13

2.2.1 Contrast Sensitivity 13

2.2.2 Spatial Masking 14

3 Motion Compensated DPCM Coder **16**

3.1 Overlapped Block Motion Estimation and Compensation 17

3.1.1 Motion Vectors 17

3.1.2 Overlapped Blocks 17

3.2 Discrete Wavelet Transform 18

3.3 Zerotree Entropy Coding of Wavelet Coefficients 20

4 Perception Based Significance Map **23**

4.1 Significance Map 23

4.2 Quantization 25

4.2.1 Weighting vs. Modified Deadzone 25

4.2.2 Frequency vs. Spatial Domain 26

5 Results **28**

5.1 Akiyo at 10 kbits/sec 29

5.2 Akiyo at 20 kbits/sec 34

5.3 Container at 20 kbits/sec 39

5.4 Conclusions 44

List of Figures

1-1 Significance map used in coding scheme 10

2-1 Contrast Sensitivity Threshold for static luminance gratings (Y) and isoluminant chrominance gratings (R/Y and B/Y) averaged over seven observers. 14

2-2 Contrast Sensitivity Threshold of a vertically oriented grating with a superimposed 0.25 contrast red/cyan isoluminant mask 15

2-3 Contrast Sensitivity Threshold of a vertically oriented grating with a superimposed 0.25 contrast blue/yellow isoluminant mask 15

3-1 Motion Compensated DPCM Coder 16

3-2 2-D Wavelet decomposition with three levels 19

3-3 Parent-child Dependencies in a Wavelet Tree 20

3-4 Reorganization of Wavelet Trees into Wavelet Blocks 21

4-1 Pixels used in the computation of gradient 24

4-2 Coder with significance map based bit-allocation 27

5-1 Frame 10 of Akiyo at 10 kbits/sec with image from significance map based coder on the left 30

5-2 Error images, frame 10 of Akiyo at 10 kbits/sec. Shown with a gain of 10 and suitably biased 30

5-3 Frame 20 of Akiyo at 10 kbits/sec 31

5-4 Detail from frame 20 of Akiyo at 10 kbits/sec 31

5-5 Error images, frame 20 of Akiyo at 10 kbits/sec 32

5-6 Frame 30 of Akiyo at 10 kbits/sec 32
5-7 Error images, frame 30 of Akiyo at 10 kbits/sec 32
5-8 Frame 40 of Akiyo at 10 kbits/sec 33
5-9 Detail from frame 40 of Akiyo at 10 kbits/sec 33
5-10 Error images, frame 40 of Akiyo at 10 kbits/sec 33
5-11 Significance map for frame 10 and fraction of samples determined to be insignificant for each frame for Akiyo sequence at 10 kbits/sec . . 34
5-12 Frame 10 of Akiyo at 20 kbits/sec with image from significance map based coder on the left 35
5-13 Error images, frame 10 of Akiyo at 20 kbits/sec. Shown with a gain of 10 and suitably biased 35
5-14 Frame 20 of Akiyo at 20 kbits/sec 35
5-15 Detail from frame 20 of Akiyo at 20 kbits/sec 36
5-16 Error images, frame 20 of Akiyo at 20 kbits/sec 36
5-17 Frame 30 of Akiyo at 20 kbits/sec 36
5-18 Error images, frame 30 of Akiyo at 20 kbits/sec 37
5-19 Frame 40 of Akiyo at 20 kbits/sec 37
5-20 Detail from frame 40 of Akiyo at 20 kbits/sec 37
5-21 Error images, frame 40 of Akiyo at 20 kbits/sec 38
5-22 Significance map for frame 10 and fraction of samples determined to be insignificant for each frame for Akiyo sequence at 20 kbits/sec . . 38
5-23 Frame 10 of Container at 20 kbits/sec with image from significance map based coder on the left 39
5-24 Error images, frame 10 of Container at 20 kbits/sec. Shown with a gain of 10 and suitably biased 39
5-25 Frame 40 of Container at 20 kbits/sec 40
5-26 Error images, frame 40 of Container at 20 kbits/sec 40
5-27 Frame 67 of Container at 20 kbits/sec 41
5-28 Wavelet artifacts in righthand image, detail from frame 67 of Container at 20 kbits/sec 41

5-29 Bird, detail from frame 67 of Container at 20 kbits/sec 41
5-30 Error images, frame 67 of Container at 20 kbits/sec 42
5-31 Frame 68 of Container at 20 kbits/sec 42
5-32 Error images, frame 68 of Container at 20 kbits/sec 42
5-33 Frame 74 of Container at 20 kbits/sec 43
5-34 Error images, frame 74 of Container at 20 kbits/sec 43
5-35 Significance map for frame 10 and fraction of samples determined to be insignificant for each frame for Container sequence at 20 kbits/sec 44

Chapter 1

Introduction

1.1 Motivation

The current state of the art in video compression is pushing against the limits of what is achievable because it employs mainly a statistical method. The goal is to maximize the numerical fidelity of the decoded video to the original. A measure such as Signal-to-Noise Ratio (SNR) is often used.

This is unsatisfactory for two reasons. First, the numerical fidelity of current compression schemes is getting harder to improve because they already remove almost all of the statistical dependency in a video sequence. Second and more importantly, numerical fidelity alone has been shown to be a poor measure of image quality [1]. While current algorithms reduce statistical redundancy, they fails to take into account the perceptual importance of errors in image coding. In lossy compression and low bitrate video coding in particular, the final video quality is degraded compared to the original due to bandwidth limitations. In these applications it is necessary to take perceptual factors into account to achieve the best possible image quality. The numerical fidelity of the coded image should be reduced in ways that are the least perceptually significant.

This thesis develops a new method for incorporating visual perception into the bit allocation during the coding of a video sequence. The eye is less sensitive to errors in areas of an image where there is a lot of activity. For example, if there is a highly

textured area in an image, errors in that area are relatively less noticeable that in an area where there is little texture. This effect can be used to decrease the relative number of bits that are allocated to less significant areas of an image [2].

A measure of significance, or perceptual importance, is computed for each pixel in an image. This significance measure is then used to quantize pixels in the residual image. The residual image is defined as the difference between the predicted image after motion compensation and the current image. Figure 1-1 shows how the significance map is computed and used in a standard motion-compensated differential coding scheme.

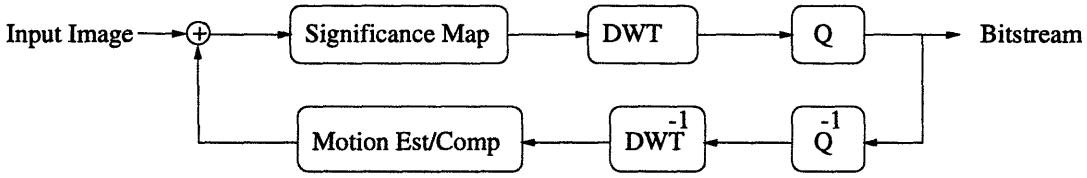

Figure 1-1: Significance map used in coding scheme

1.2 Image Format

The input images for testing this algorithm are the MPEG4 [3] test sequences. They have been subsampled from their original CCIR 601 [4] format to YUV420. This format has a luminance channel and two chrominance channels subsampled by a factor of four relative to the luminance. This is useful for achieving visually good compression since chrominance is less important than luminance [6].

1.3 Significance Map

The first step is to compute the significance of each pixel in the image. The significance is based on the activity in an area of the image and the magnitude of the residual pixels in the same area. If there is high activity in the area around a pixel then errors in that pixel value are less visible and its visual significance is reduced.

Conversely, if a residual pixel has a large magnitude then the significance is increased.

The activity around a particular pixel is found by taking the gradient over a small neighborhood around that pixel. Since the gradient is a very local measure, it is computed over a multiresolution pyramid [5] to make it more global. The gradients computed at each level of the pyramid are upsampled by the appropriate factor, weighted, and added together.

The significance of each pixel is then found by taking the sum of the residual pixels in a small neighborhood around the pixel and dividing by the sum of the gradients in the same neighborhood. A small constant is added to the denominator for numerical stability.

1.4 Quantization

The second step is to use this significance map in the quantization of the residual. This can be done by using the significance map to change the deadzone of the quantizer. If the significance of a residual pixel is below a certain threshold then that pixel is set to zero. This reduces the entropy of the residual to be coded and allows either more bits to be allocated to significant pixels, or a lower bitrate. This method has the advantage of not requiring any bitstream overhead.

The next chapter describes visual effects which are useful for reducing evaluating the visual importance of information in a video sequence.

Chapter 2

Visual Perception of Coding Errors

2.1 Visual Perception Based Image Coding

Coding an image sequence at low bitrates results in output that has perceived loss when compared with the original. The goal is to minimize this loss for a given bitrate. Ideally the output sequence has errors that are all below a target level of visual perceptibility.

The image sequence can be thought of as spatial and temporal information, and also luminant and isoluminant color information. If there is information contained in these domains that is invisible then it can be left out without any reduction in the coded quality.

If the errors in the output sequence are to be below a certain threshold of perceptiblity, then the information with a priority lower than this threshold can simply be left uncoded. The remaining information can then be coded using traditional motion-compensated DPCM techniques.

2.2 Useful Effects of Visual Perception

Psychophysical experiments to characterize perception are useful for determining the importance of image information. Results applicable to our model of image information can be found in the literature [6], [7], [8], [9].

2.2.1 Contrast Sensitivity

Contrast is a measure of the range of intensity present in an image. It is defined as follows, where I is intensity.

$$\frac{I_{max} - I_{min}}{I_{max} + I_{min}} \tag{2.1}$$

The Contrast Sensitivity Threshold is the contrast level at which the eye is able to detect the image. This threshold varies with spatial frequency. A grating with fixed spatial frequency is used in an experiment to measure the threshold for that frequency. Figure 2-1 is a plot of contrast sensitivity vs. spatial frequency [6]. Y represents sensitivity to static luminance gratings. Color is measured using isoluminant color gratings that can be expressed as a ratio, R/Y and B/Y, of a primary color to luminance.

Only the threshold for sensitivity is shown in Figure 2-1. Since there will be loss in our compression algorithm, it is useful to know the relative sensitivity vs. frequency above the threshold. Generally the isosensitivity curves above the threshold get flatter as the contrast increases [10].

Visual acuity falls with increasing spatial frequencies and less bits can be allocated to high frequency components of the image sequence to be coded. Visual acuity is also less for chrominance relative to luminance and this will allow the chrominance channels to be subsampled vertically and horizontally.

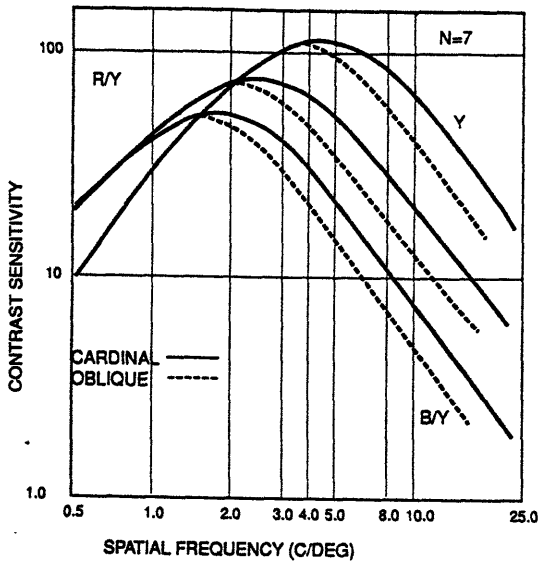

Figure 2-1: Contrast Sensitivity Threshold for static luminance gratings (Y) and isoluminant chrominance gratings (R/Y and B/Y) averaged over seven observers.

2.2.2 Spatial Masking

When two sinusoidal gratings with the same orientation of different spatial frequencies are superimposed, the lower contrast grating becomes less visible [6], [13], [14], [15].

Viewers were presented [6] with a fixed sinusoidal grating of 0.25 contrast. This grating was superimposed on another grating that varied from 0.5 to 12.5 c/deg in spatial frequency and .001 to 0.25 in contrast. The viewer was queried to determine the threshold of perception. Some of the results are shown in Figures 2-2 and 2-3.

The contrast sensitivity threshold of a luminance grating decreases significantly when superimposed with either a luminance or isoluminant chrominance grating with the same orientation. The superimposed grating only has to be within $\pm\frac{1}{2}$ octave in spatial frequency [16], [17].

The next chapter describes the coder whose visually perceived coding performance was improved.

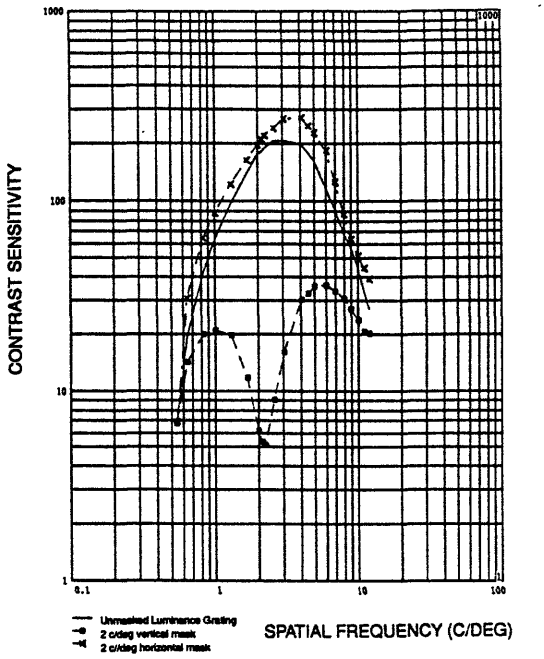

Figure 2-2: Contrast Sensitivity Threshold of a vertically oriented grating with a superimposed 0.25 contrast red/cyan isoluminant mask

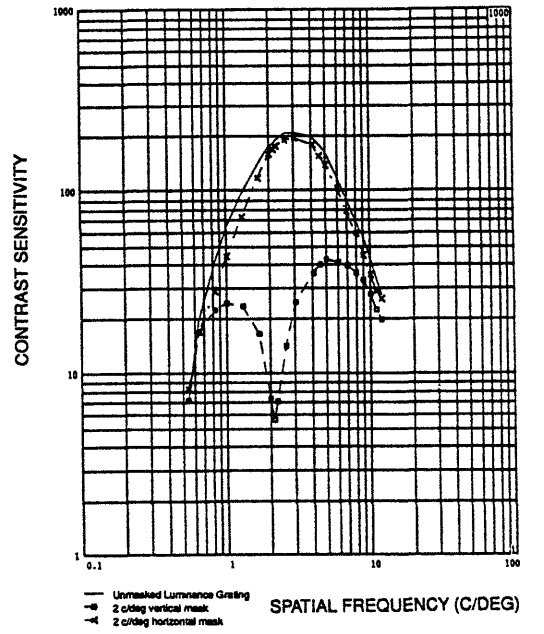

Figure 2-3: Contrast Sensitivity Threshold of a vertically oriented grating with a superimposed 0.25 contrast blue/yellow isoluminant mask

Chapter 3

Motion Compensated DPCM Coder

A block diagram of the video coder which was modified to improve its perceptual performance is shown in Figure 3-1. This coder was submitted in parts to the MPEG4 committee and represents a good starting point for improving low-bitrate video coding performance.

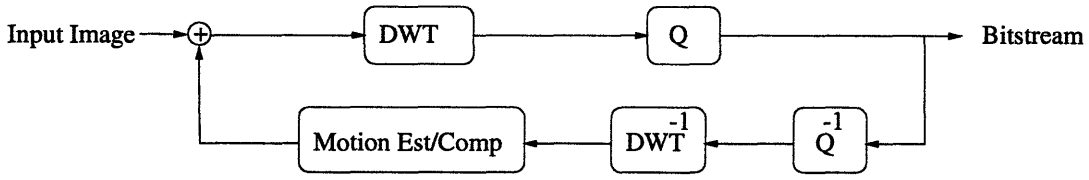

Figure 3-1: Motion Compensated DPCM Coder

The input video is motion compensated, transformed into the frequency domain with the discrete wavelet transform (DWT), and quantized. The coefficients are then represented with zerotrees to reduce the entropy by capturing similarity across scales of the DWT. Each of these steps is described below.

3.1 Overlapped Block Motion Estimation and Compensation

The current frame in the input video sequence is first predicted from the previous frame using motion estimation. This is to take advantage of the fact that much of the information in the current frame is present in the previous frame, possibly in a different location. The motion estimation and compensation schemes are taken from the ITU-T H.263 standard [18] and TMN1.6 [19] which is an implementation of that standard.

3.1.1 Motion Vectors

Each 16x16 macroblock can have either one or four associated motion vectors. If only one vector is transmitted for the current macroblock then all four 8x8 blocks have the same motion vector. If there are four motion vectors then there is one corresponding to each 8x8 block. The estimation algorithm decides whether to code with one or four vectors in order to reduce the number of bits to be transmitted. The motion vectors have half-pel accuracy determined using bilinear interpolation

3.1.2 Overlapped Blocks

This coder uses an Overlapped Block Motion Compensation (OBMC) scheme. OBMC is an improvement over traditional block based motion compensation techniques. Since traditional techniques do not consider the motion of neighboring blocks discontinuities result at block boundaries. These are visible as blocking artifacts. In OBMC the motion of surrounding blocks is used in motion compensation.

Each pixel in an 8x8 luminance block is a weighted sum of three prediction values. The three prediction values are found using the motion vector of the current block and two vectors from the surrounding blocks. These two are either the vectors of the blocks on the left and right of the current block or the vectors of the blocks above and below.

For each pixel, the motion vector of the closest block is used. If the two extra vectors are from above and below then the top half of the block uses the vector for the above block and similarly for the bottom half, the vector from the block below is used. The value for a pixel $p(i,j)$ is defined in Equation 3.1.

$$p(i,j) = (q(i,j) * H_0(i,j) + r(i,j) * H_1(i,j) + s(i,j) * H_2(i,j) + 4)/8 \qquad (3.1)$$

$$q(i,j) = p(i + MV_x^0, j + MV_y^0)$$
$$r(i,j) = p(i + MV_x^1, j + MV_y^1)$$
$$s(i,j) = p(i + MV_x^2, j + MV_y^2)$$

(MV_x^0, MV_y^0) represents the motion vector for the current block. (MV_x^1, MV_y^1) and (MV_x^2, MV_y^2) represent the motion vectors for the blocks above and below respectively, or left and right as defined above. The H_n are weighting matrices defined in [18] and are used to make certain pixels more important than others in motion estimation and compensation.

3.2 Discrete Wavelet Transform

After motion estimation the residual image is transformed using the discrete wavelet transform (DWT) [20]. The DWT is implemented using an octave band tree structure. Each level of the tree is an identical 2-band decomposition with only the low frequency band further divided.

To extend the DWT to two dimensions separable filters are used. Each image is divided as shown in Figure 3-2 into four subimages and then the low-low band is further decomposed. The luminance channel decomposition uses four levels and the chrominance channels use three. In order not to expand the images size and handle the edges of the images properly, symmetric extension is used due to its superior performance in compression applications over circular convolution. A well known 9-tap QMF [21] filter is used for the separable filter.

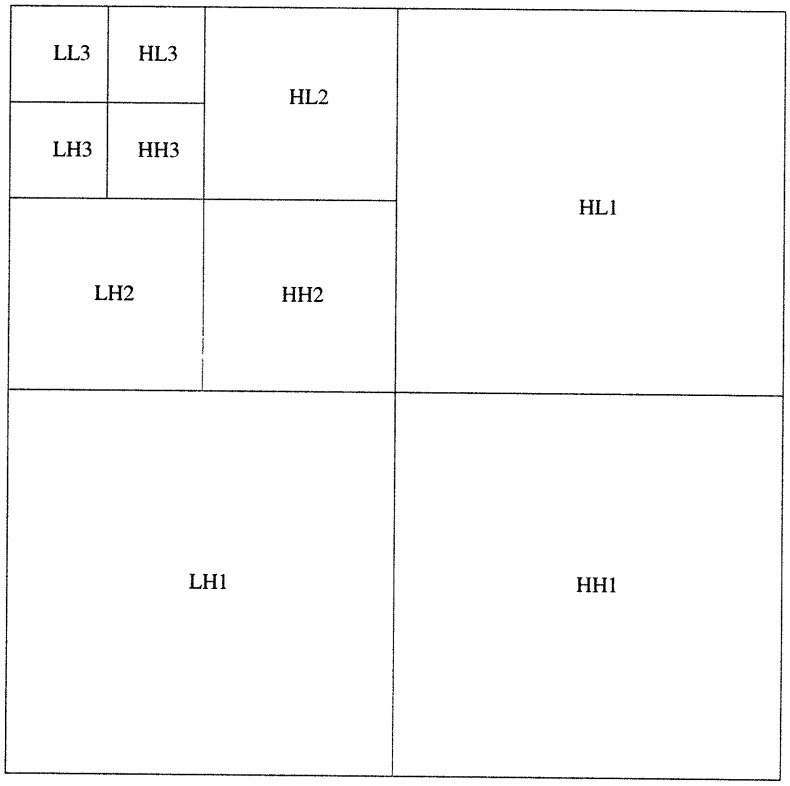

Figure 3-2: 2-D Wavelet decomposition with three levels

3.3 Zerotree Entropy Coding of Wavelet Coefficients

Zerotree Entropy Coding (ZTC) is an efficient technique for coding wavelet transform coefficients of motion-compensated video residuals [22]. Like the EZW algorithm [23], ZTC captures self-similarity inherent in images and video residuals across wavelet scales. ZTC gives up the embedded property of EZW but has improved compression performance for residual images. EZW is used to compress the I-frames and ZTC is used for the P-frames.

Wavelet coefficients can be represented by a *wavelet tree*, in which the nodes of the tree correspond one-to-one with coefficients in the wavelet transform. The coefficient at a coarser scale is called the parent and all coefficients at the same spatial location in the next finer scale are called children of that parent. A *zerotree* is a wavelet tree which has all zeros for its root and nodes.

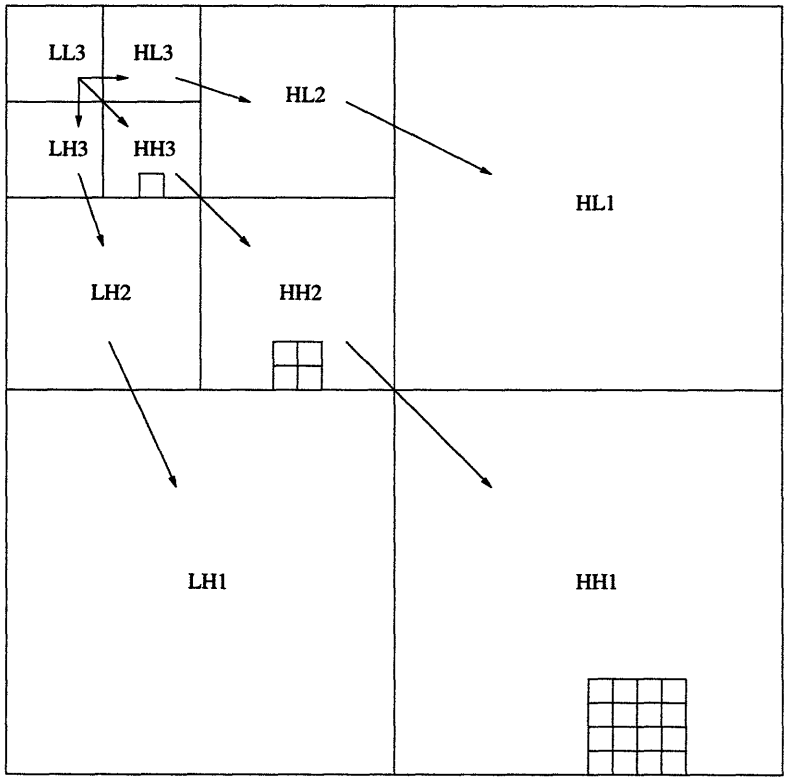

Figure 3-3: Parent-child Dependencies in a Wavelet Tree

A three level wavelet tree is shown in Figure 3-3. Arrows point from the subband of the parents to the subbands of the children. For the lowest frequency band each parent has three children. For the other bands, each parent has four children in the corresponding band at the next scale.

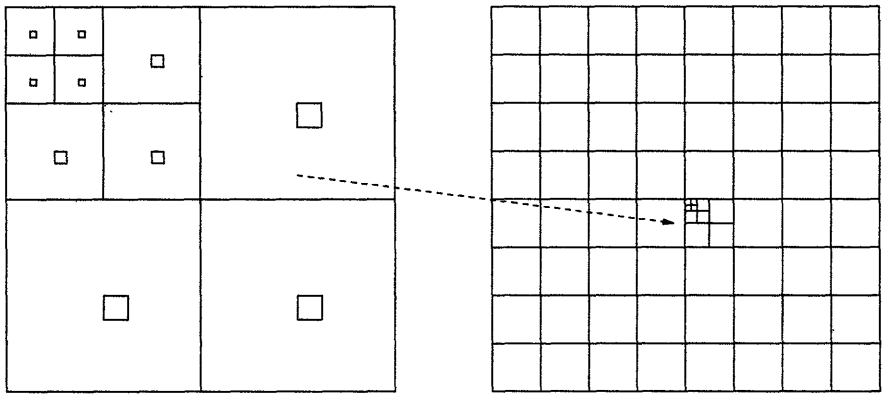

Figure 3-4: Reorganization of Wavelet Trees into Wavelet Blocks

Figure 3-4 shows how the coefficients can be reorganized into wavelet blocks. Each wavelet block comprises the coefficients at all scales and orientations that represent the original frame at that block. This allows quantization specified on a block by block basis in the spatial domain to be applied in the wavelet domain.

Each parent in the wavelet tree is represented by one of three values after quantization: *zerotree root*, *valued zerotree root* and *value*. A zerotree root is used to represent a zero-valued parent which has all zeros in its corresponding wavelet tree. A valued zerotree root represents a non-zero parent which has also has all zeros it its wavelet tree. Value represents a parent with some non-zero descendant. After each parent in the lowest band is scanned, the remaining pixels in the next frequency band are scanned. This process is continued until all the pixels are represented.

This method of scanning is very efficient for wavelet coefficients because of the dependencies that exist across scales. Particularly in a quantized residual image, if a parent coefficient is zero, it is likely that the descendants in higher frequency bands are zero as well. After the coefficients are coded in this manner, they are encoded with an arithmetic coder.

The coding of the wavelet coefficients in ZTC differs from EZW in four ways.

- In ZTC the wavelet trees are built and coded in one pass instead of bit-plane by bit-plane as in EZW.
- The quantization of the coefficients is explicitly defined and can be either independent of or a part of the wavelet tree building process. The quantization can be adjusted based on both spatial location and frequency band.
- Coefficients are scanned tree-by-tree rather than subband by subband. A parent and all its child trees are scanned first before the neighbors of the parent.
- The alphabet of symbols used in arithmetic coding has been optimized for motion-compensated residual images.

The next chapter describes the modifications that were made to this coder to improve its visually perceived coding performance.

Chapter 4

Perception Based Significance Map

This chapter describes the significance map used to improve the performance of the coder described in chapter 3.

4.1 Significance Map

Areas of an image which are highly textured effectively have many luminance and isoluminant color gratings of different spatial frequencies superimposed. While it would be difficult for a coder to decompose the whole picture into all its component gratings, the luminance and chrominance masking effects detailed in chapter 2 can be used to determine which areas of an image are perceptually important. Since the superimposition of many gratings with similar contrast tends to produce masking and appears to the viewer as texture, the coder can allocate relatively less bits to residuals in textured areas of an image.

Ideally a measure of the significance of each pixel would be useful. Once this measure is computed, it can be applied in the quantization pixel-by-pixel. This measure will be related to the visibility of error in a particular pixel. As seen above this visibility is related to the texture in the neighborhood of this pixel.

The gradient of the pixel magnitudes in an image is a good measure of the texture.

If the gradient is large, then errors become less visible. Similarly, large errors are more important than small ones. The significance measure should be designed so that some measure of the gradient is in the denominator and some measure of the residual pixel magnitude is in the numerator.

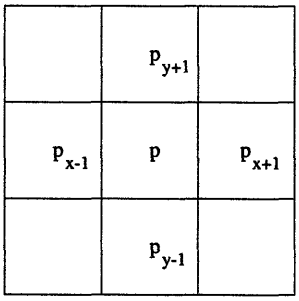

Figure 4-1: Pixels used in the computation of gradient

$$\nabla p = \sqrt{(\frac{|p - p_{x-1}| + |p - p_{x+1}|}{2})^2 + (\frac{|p - p_{y-1}| + |p - p_{y+1}|}{2})^2} \tag{4.1}$$

The gradient measure computes the difference between the current pixel p and the adjacent pixels in the x and y directions.

$$significance(p) = \frac{\sum_{y=-1}^{y=1} \sum_{x=-1}^{x=1} r_{xy}}{(\sum_{y=-1}^{y=1} \sum_{x=-1}^{x=1} \nabla p_{xy}) + c} \tag{4.2}$$

The significance of the residual r at each pixel p is computed by summing the residual pixels in a 3x3 neighborhood and dividing by the sum of the gradients in the same neighborhood plus a small constant for numerical stability.

Since the gradient is a very local measure it is computed over a multiresolution pyramid[5] to make it more global. This helps differentiate between large areas of texture and small ones, such as simple edges. The gradients computed at each level of the pyramid are upsampled by the appropriate factor, weighted, and added together.

4.2 Quantization

The significance map is then used in the quantization of the residual. This can be done in several ways each of which is described below.

4.2.1 Weighting vs. Modified Deadzone

The significance map can be directly applied to modify the quantization in two ways. One way it can be used is to change the deadzone of the quantizer. If the significance of a residual pixel is below a certain threshold then that pixel is set to zero. This reduces the entropy of the residual to be coded and allows either more bits to be allocated to significant pixels, or a lower bitrate. This can be done on a pixel-by-pixel basis and has the advantage of not requiring any bitstream overhead.

The significance map can also be used to weight residual pixels according the their significance. During the quantization step this results in pixels with higher significance being quantized more finely. This technique requires the transmission of the weights along with the residual. The bitstream overhead can be reduced in two ways. The decoder can also try to guess the significance of each pixel by computing the significance map using the previous decoded frame. The encoder then only has to transmit the difference in weights. This works well if there is not too much temporal change in the video sequence. If there are big temporal changes however, the decoder cannot accurately guess the weights.

A second technique is to compute weights only on a block-by-block basis. This can be used along with the decoder trying to guess the weights. Using weights on a block-by-block basis has a significant disadvantage. The significance of pixels that are close by can vary greatly. Using the same weight over a whole block then reduces the effectiveness of using the significance map because it forces the coder to assign the same importance to pixels that might have very different perceptual significances.

In practice the disadvantages of weighting made modifying the deadzone a more effective technique for using the significance map in quantization and this is what is used to modify the coder that is presented.

4.2.2 Frequency vs. Spatial Domain

The significance map represents the significance of each pixel in the residual image. This can be used before the wavelet transform or afterwards to modify the quantization. Since the wavelet transform preserves the spatial relation of coefficients, each wavelet coefficient has a corresponding significance. This significance is simply the significance of the pixel(s) in the spatial domain to which the wavelet coefficient corresponds.

Changing the quantization in the frequency domain is useful if there exists information to determine the relative importance of the various frequency bands. For example, in the coder described in chapter 3, each finer scale of the DWT is quantized with one-fourth the resolution as the previous scale. This takes advantage of the experimental results from chapter 2 which state that low frequency information is more important perceptually.

A similar scheme could be used in either computing the significance map or in applying it to quantization in the frequency domain. In computing the significance map, texture that exists at low frequency can contribute more to significance than that at high frequencies. Similarly, when quantizing coefficients in the frequency domain, the threshold of significance used to modify the deadzone can depend on the scale of the coefficient.

Modifying the quantization in the frequency domain was explored as detailed above, but the results were not as good as in the spatial domain. This seems to be because the technique used for computing the significance map does not inherently take frequency related information into account. Computing it separately for each frequency band also did not improve performance.

The coder presented here is shown in figure 4-2. The significance map is used to set pixels below a certain threshold of significance to zero. This is done on a pixel-by-pixel basis in the spatial domain before the wavelet transform. This results are shown in the next chapter.

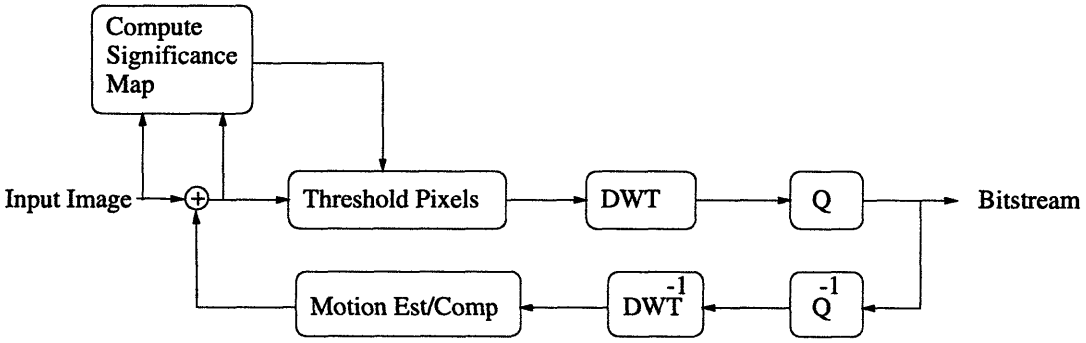

Figure 4-2: Coder with significance map based bit-allocation

Chapter 5

Results

The results of coding the Akiyo and Container sequences [3] with the original coder and the same coder with the significance map based bit allocation are compared to show the improvement in image quality for the same bitrate. Constant quantization is used and equal bitrates means at most a difference of one percent.

The significance map can be used in two ways. It can be used to improve quality for a constant bitrate or it can be used to reduce bitrate while maintaining quality. Since it is harder to judge two sequences to be equal in quality, the results here are for constant bitrate.

The original sequences have been subsampled from CCIR601[4] to 176x144 YUV-420 format at 7.5 frames/sec. 50 frames of the subsampled Akiyo sequence were coded at 10 and 20 kbits/sec and 75 frames of the subsampled Container sequence were coded at 20 kbits/sec. These bitrates are typical for low-bitrate applications such as digital transmission by modem or wireless.

First the original coder was run at the above bitrates. Bits were allocated to the initial I frame so that the SNR was roughly constant from the first frame to the last. Then the same coder with modified bit allocation based on the significance map was run. Using the significance map reduces the entropy of the residual allowing a finer quantization to be used resulting in better quality. This is shown in more detail for each of the coded sequences below.

The error images are included because they show the location of the differences

between the coded images and the originals. In the images that were coded with the modified coder, errors are present in areas of an image where they would be less visible.

When the effect of using the significance map is analyzed, the actual threshold for determining significance is not important. It should be varied based on the image sequence and target bitrate. What is important is the percent of residual pixels that are determined to be unimportant and thus quantized to zero. This percentage gives an idea of how much difference using the significance map makes. A graph of this for each frame and an picture of the significance map with significant areas in white are shown below.

Since color is difficult to show on paper, the improvements in quality shown correspond only to the luminance components of the coded image sequences. Similar results are achieved for the chrominance channels.

5.1 Akiyo at 10 kbits/sec

The Akiyo sequence is coded at 10 kbits/sec using 13 kbits for the initial I frame and quant values of 35 and 100 for the luminance and chrominance channels respectively. Increasing the fineness of the luminance quantization to 30 required all samples with significance greater than 0.31 to be quantized to zero. Several frames of the coded sequences are shown. The frame on the left is with the significance map based bit allocation. The error images are shown with a gain of 10 and are suitably biased.

In the image on the right of Figure 5-1, there are artifacts above the right eyelid, on the left cheek, and around the lips. The first one has been eliminated using the significance map and the other two have been reduced.

The error images bear out this result. In the modified coder, the errors form lines similar to those that would be produced by an edge detector. This reflects the method used to compute the significance of the residual pixels. The pixels next to an edge are less visible and thus have less significance. Therefore using the significance map will concentrate errors in these area and decrease them in areas where they would be

Figure 5-1: Frame 10 of Akiyo at 10 kbits/sec with image from significance map based coder on the left

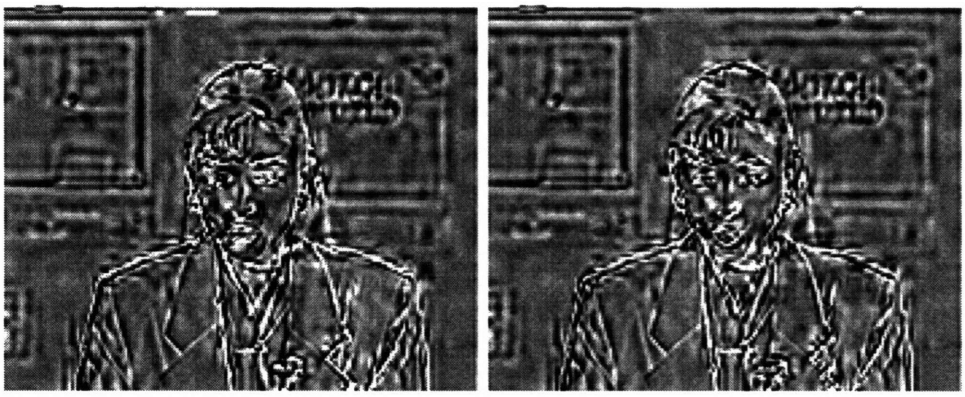

Figure 5-2: Error images, frame 10 of Akiyo at 10 kbits/sec. Shown with a gain of 10 and suitably biased

more visible.

Figure 5-3: Frame 20 of Akiyo at 10 kbits/sec

Figure 5-4: Detail from frame 20 of Akiyo at 10 kbits/sec

Figures 5-3 and 5-4 show that a severe artifact across the upper lip has been significantly reduced using the significance map.

In Figures 5-6 and 5-8 the whole face looks much smoother in the right hand images. This is because there are less wavelet artifacts due to quantization of the wavelet coefficients.

Looking at the significance map for frame 10 in Figure 5-11, it can bee seen that coding errors in flat areas such as the cheeks, lips, the screens in the back and areas of the jacket are determined to be important. The residual pixels in these areas are preserved and pixels outside these areas are set to zero.

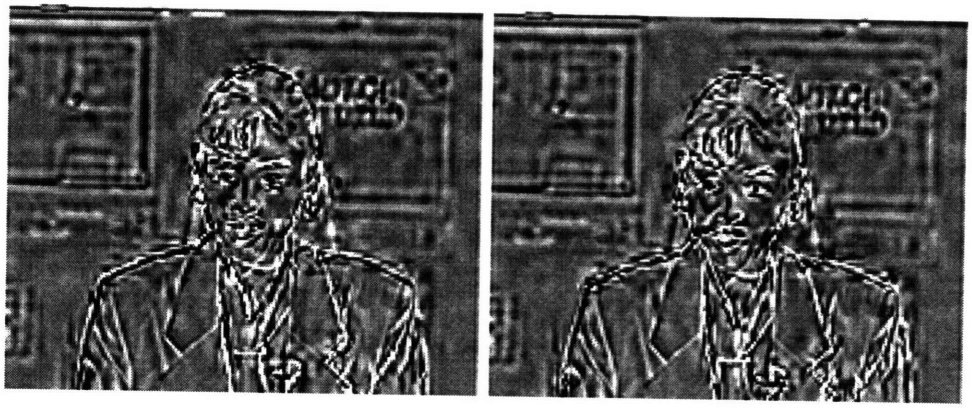

Figure 5-5: Error images, frame 20 of Akiyo at 10 kbits/sec

Figure 5-6: Frame 30 of Akiyo at 10 kbits/sec

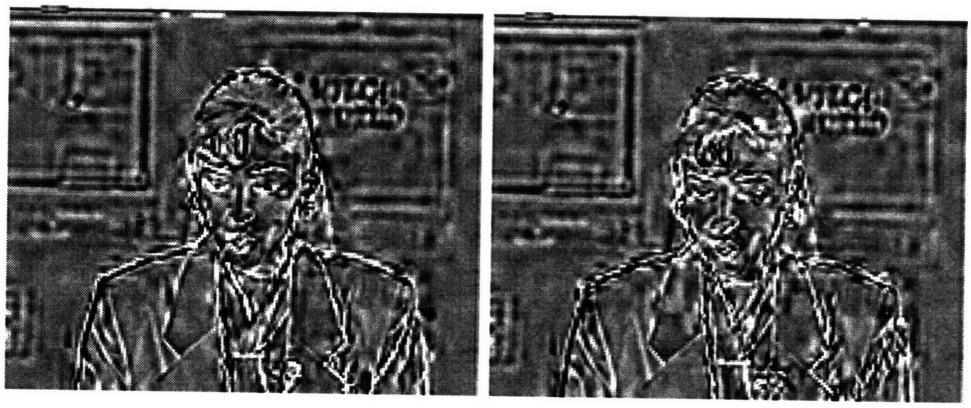

Figure 5-7: Error images, frame 30 of Akiyo at 10 kbits/sec

Figure 5-8: Frame 40 of Akiyo at 10 kbits/sec

Figure 5-9: Detail from frame 40 of Akiyo at 10 kbits/sec

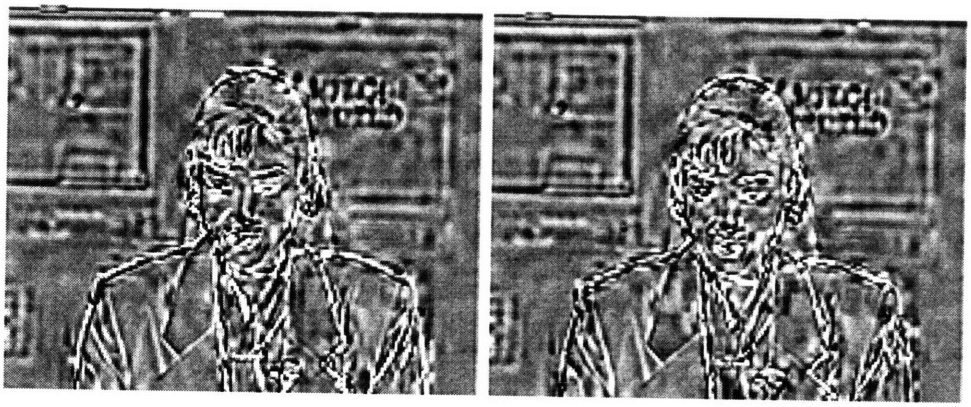

Figure 5-10: Error images, frame 40 of Akiyo at 10 kbits/sec

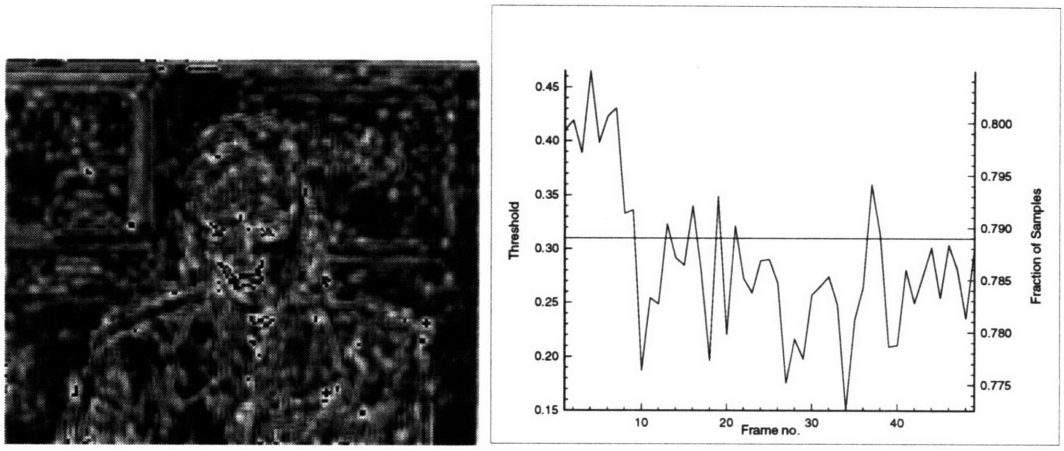

Figure 5-11: Significance map for frame 10 and fraction of samples determined to be insignificant for each frame for Akiyo sequence at 10 kbits/sec

The fraction of samples determined to be insignificant is quite high. The achieved improvement in quality suggests that many of the residual pixels in this motion-compensated wavelet coding scheme are relatively insignificant.

5.2 Akiyo at 20 kbits/sec

The Akiyo sequence is coded at 20 kbits/sec using 17.5 kbits for the initial I frame and quant values of 18 and 60 for the luminance and chrominance channels respectively. Increasing the fineness of the luminance quantization to 13 required all samples with significance greater than 0.31 to be quantized to zero. Several frames of the coded sequences are shown. The images on the left are coded with the significance map based bit allocation.

There is a big difference in the images in Figure 5-14 with the improved bit allocation eliminating artifacts in the right cheek, lip and chin area. The error images show that the significance map made an even bigger difference than at 10kbits/sec. Without the significance map, coding errors are distributed evenly around the image. With the significance map, the errors in flat areas are reduced and the errors in textured areas are increased.

As in the case of 10 kits/sec the improvement in image quality using the significance map provides a clear illustration of the difference in the relative visual signif-

Figure 5-12: Frame 10 of Akiyo at 20 kbits/sec with image from significance map based coder on the left

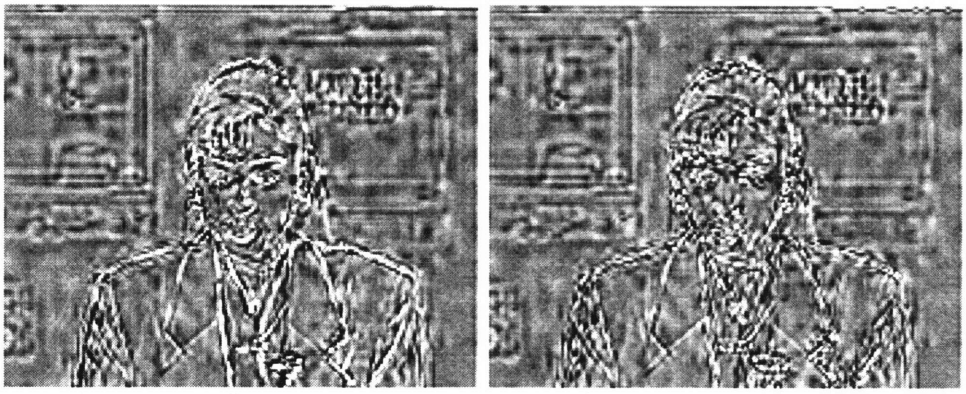

Figure 5-13: Error images, frame 10 of Akiyo at 20 kbits/sec. Shown with a gain of 10 and suitably biased

Figure 5-14: Frame 20 of Akiyo at 20 kbits/sec

Figure 5-15: Detail from frame 20 of Akiyo at 20 kbits/sec

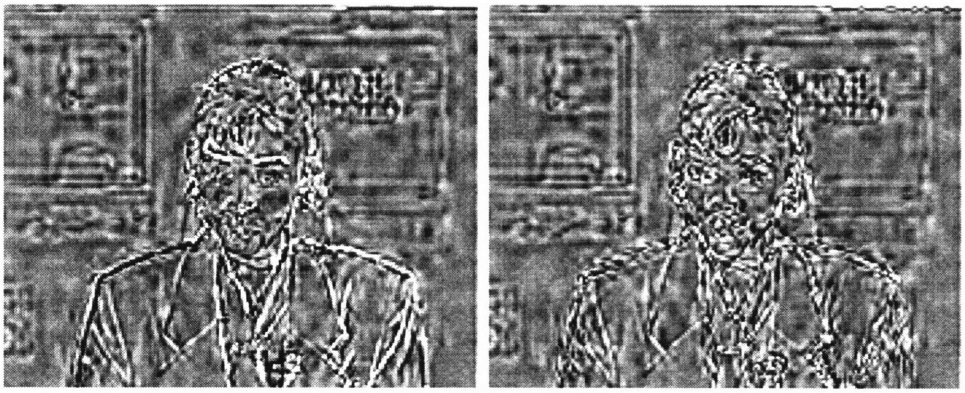

Figure 5-16: Error images, frame 20 of Akiyo at 20 kbits/sec

Figure 5-17: Frame 30 of Akiyo at 20 kbits/sec

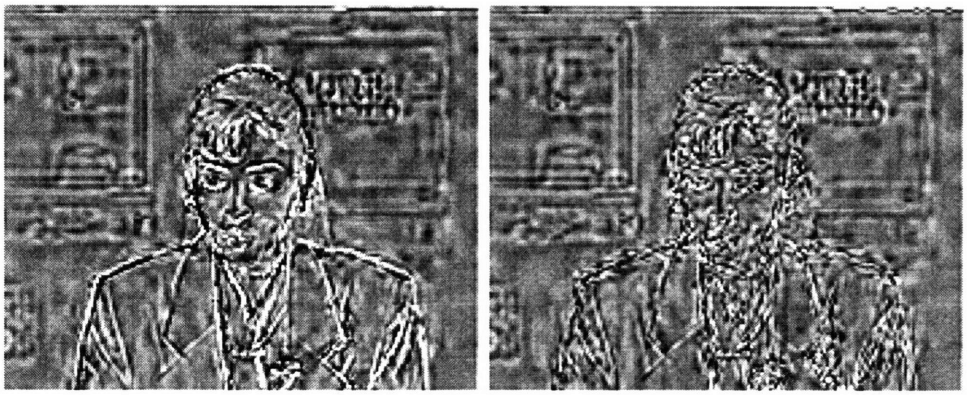

Figure 5-18: Error images, frame 30 of Akiyo at 20 kbits/sec

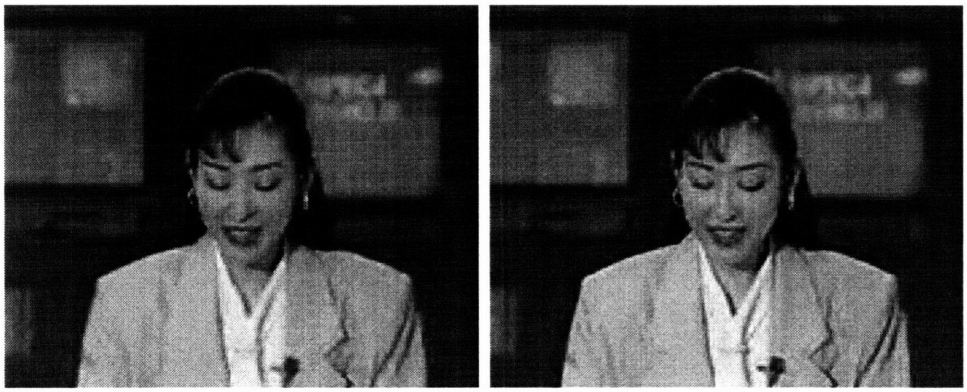

Figure 5-19: Frame 40 of Akiyo at 20 kbits/sec

Figure 5-20: Detail from frame 40 of Akiyo at 20 kbits/sec

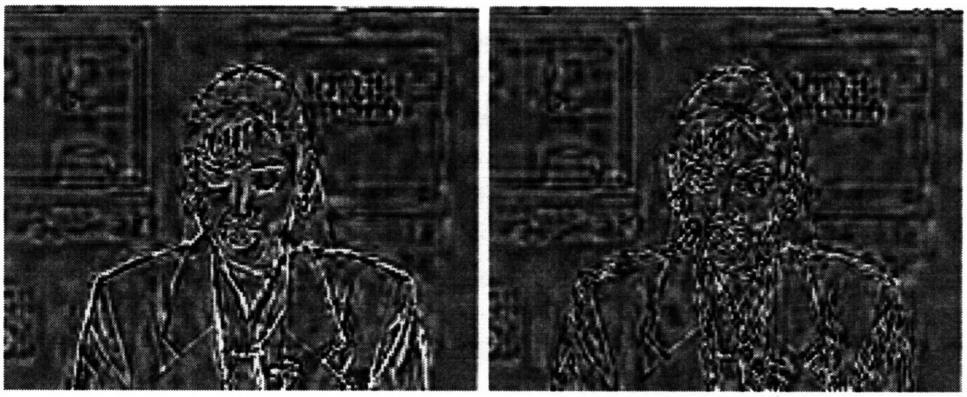

Figure 5-21: Error images, frame 40 of Akiyo at 20 kbits/sec

icance of pixels in an image. The number of errors due to quantization of wavelet coefficients corresponding to visually important pixels is reduced.

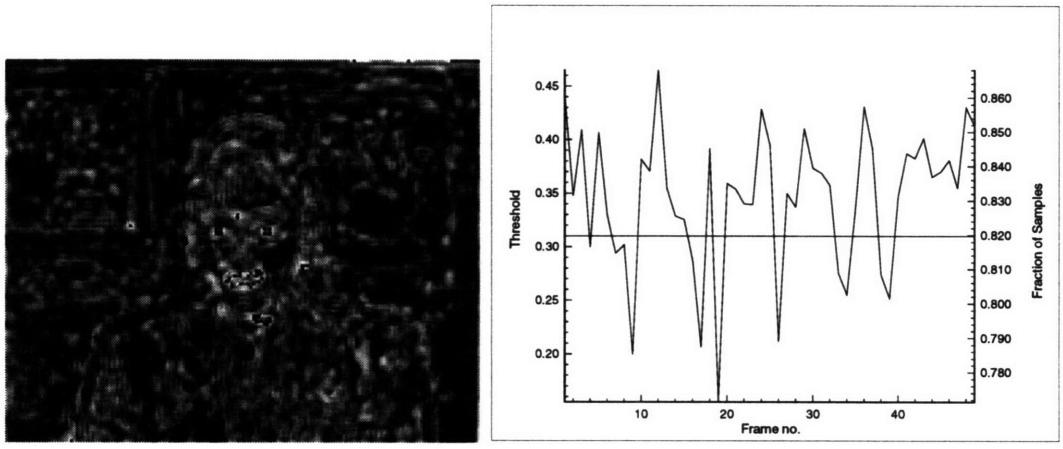

Figure 5-22: Significance map for frame 10 and fraction of samples determined to be insignificant for each frame for Akiyo sequence at 20 kbits/sec

Looking at the significance map for frame 10 in Figure 5-22, it can bee seen that areas in the Akiyo sequence where there are errors in flat areas such as parts of the face, the screens in the back and areas of the jacket are determined to be important. The significance map is similar to that for Akiyo at 10 kbits/sec but the since the image is of higher quality, the flat areas in the background are coded better. This leads to the areas in the face and jacket being relatively more significant. As before, the residual pixels in these are preserved and those outside these areas are set to zero.

5.3 Container at 20 kbits/sec

The Container sequence was coded at 20 kbits/sec using 19 kbits for the initial I frame and quant values of 27 and 100 for the luminance and chrominance channels respectively. Increasing the fineness of the luminance quantization to 22 required all samples with significance greater than 0.312 to be quantized to zero. Several frames of the coded sequences are shown with the images coded with the significance map based coder on the left.

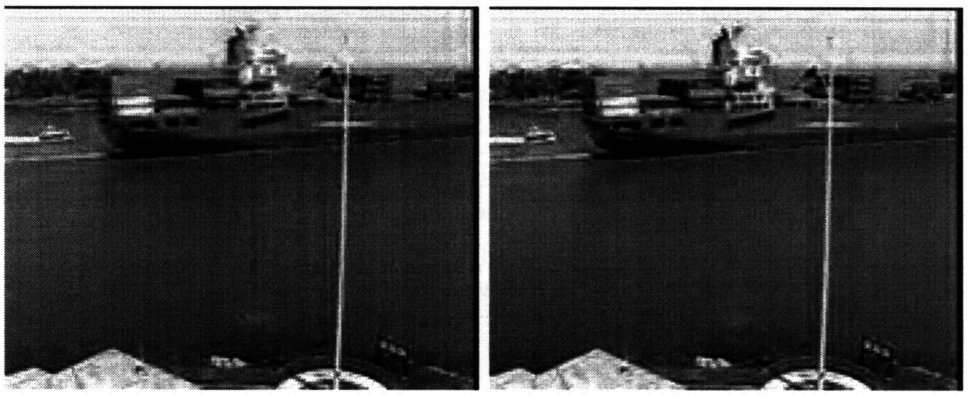

Figure 5-23: Frame 10 of Container at 20 kbits/sec with image from significance map based coder on the left

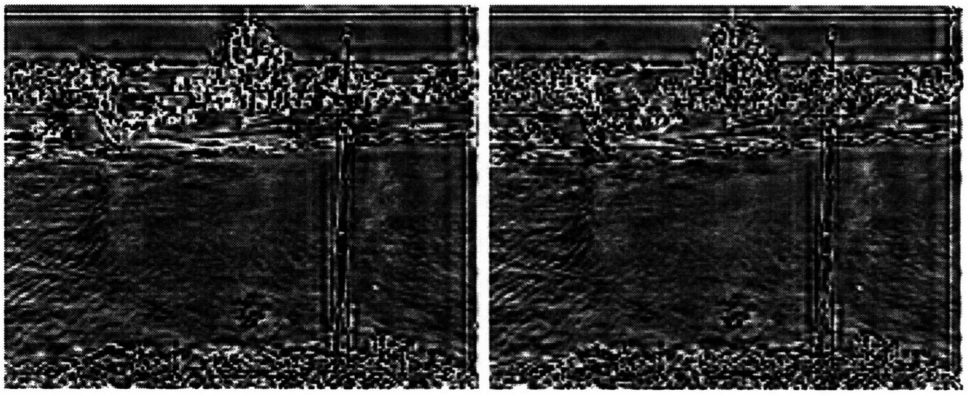

Figure 5-24: Error images, frame 10 of Container at 20 kbits/sec. Shown with a gain of 10 and suitably biased

The images in Figure 5-23 are quite similar but by frame 40, shown in Figure 5-25, visible differences start to appear. The significance map based coder has eliminated a wavelet ringing artifact in the lefthand portion of the water. Generally, the water looks

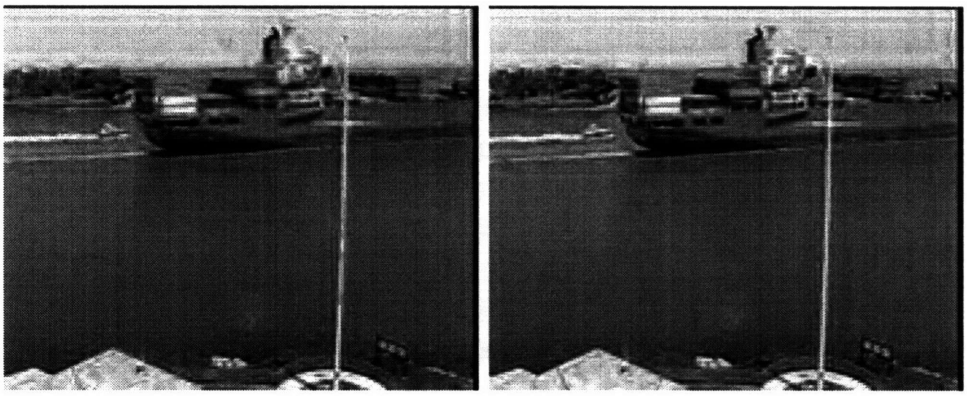

Figure 5-25: Frame 40 of Container at 20 kbits/sec

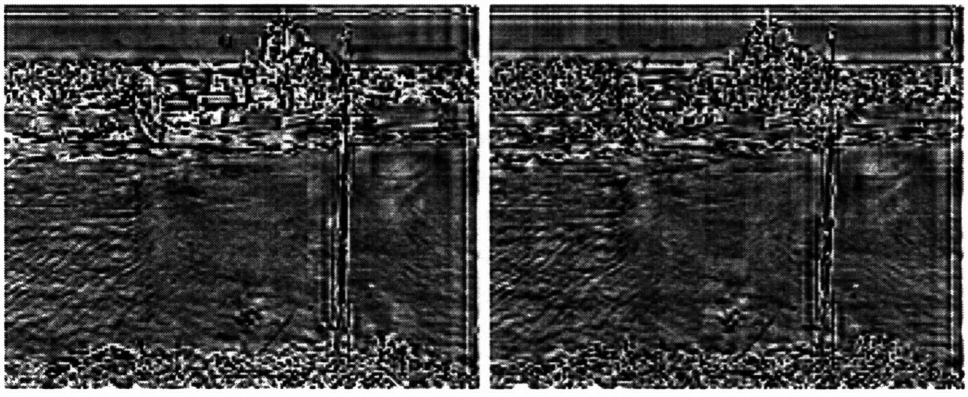

Figure 5-26: Error images, frame 40 of Container at 20 kbits/sec

more blocky in the righthand image because of coarser quantization. Quantization is done at the macroblock level and the macroblocks are more visible with coarser quantization.

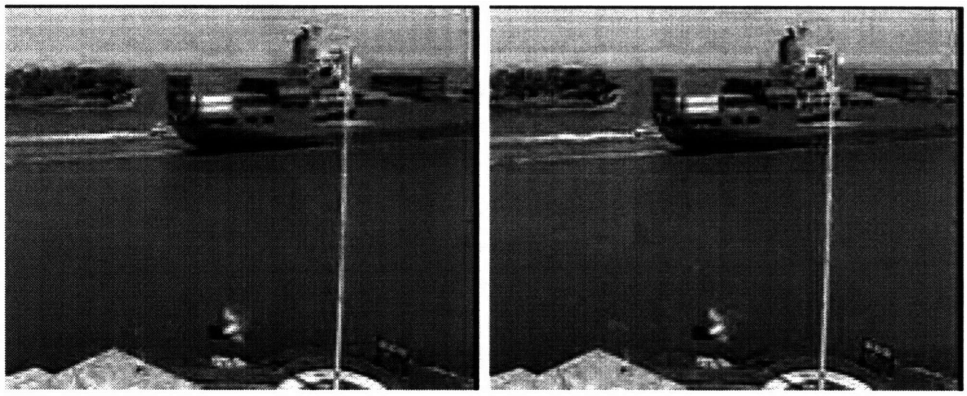

Figure 5-27: Frame 67 of Container at 20 kbits/sec

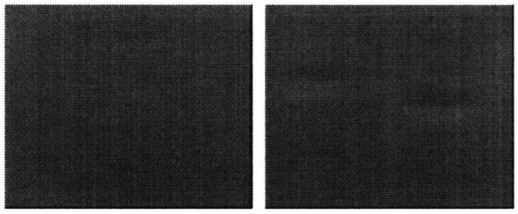

Figure 5-28: Wavelet artifacts in righthand image, detail from frame 67 of Container at 20 kbits/sec

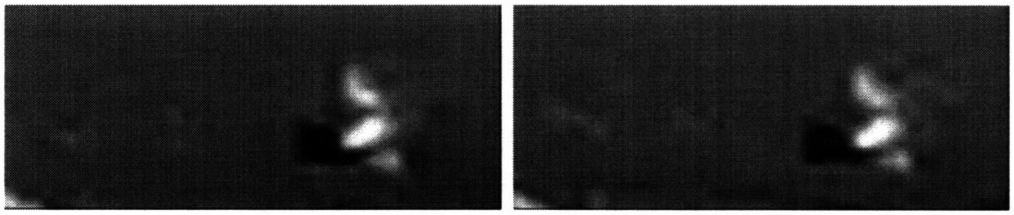

Figure 5-29: Bird, detail from frame 67 of Container at 20 kbits/sec

Frames 67 and 68 are interesting because a bird has flown into the lower portion of the image in Figure 5-27 and temporarily becomes invisible behind the flagpole in Figure 5-31. The bird is coded similarly by both coders, but there is noticeable less ringing just to the right of the bird in the lefthand image. The ringing artifacts that appeared in frame 40 have multiplied in the right hand image along with an increase in the visibility of the macroblock level quantization.

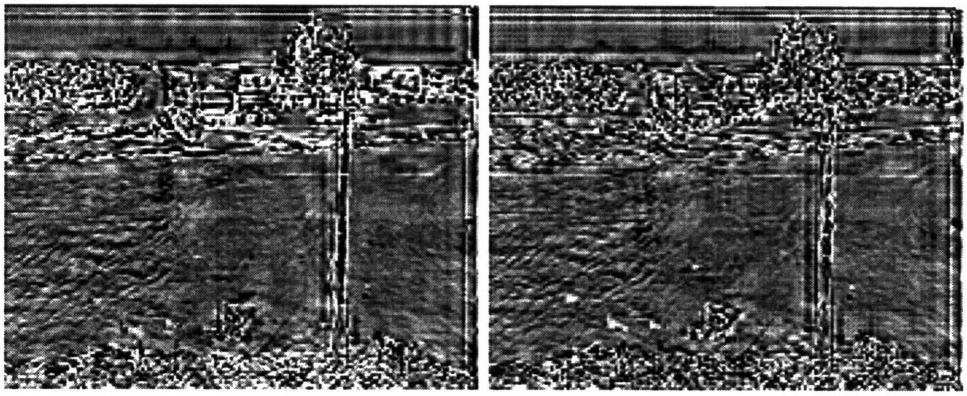

Figure 5-30: Error images, frame 67 of Container at 20 kbits/sec

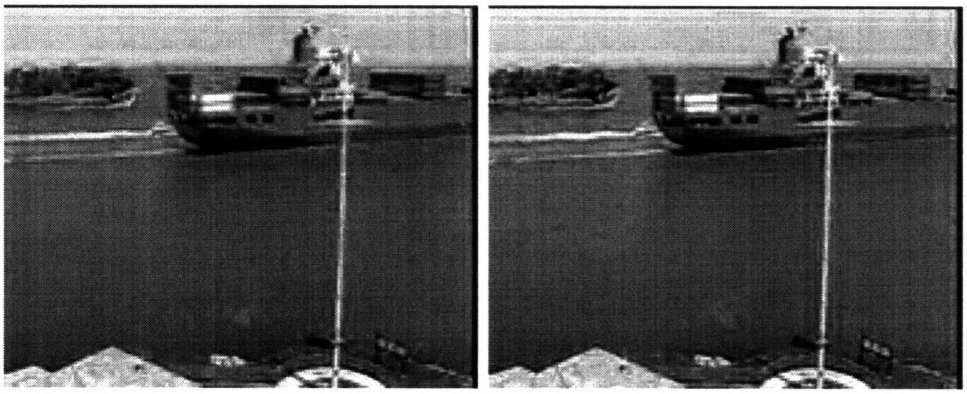

Figure 5-31: Frame 68 of Container at 20 kbits/sec

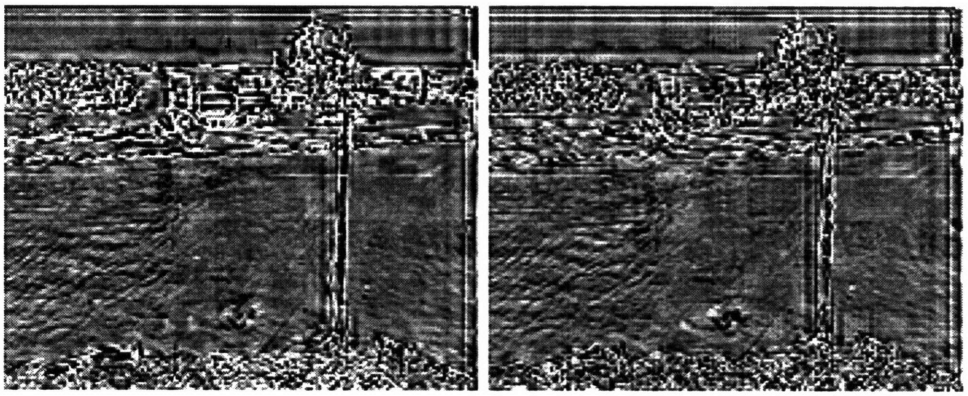

Figure 5-32: Error images, frame 68 of Container at 20 kbits/sec

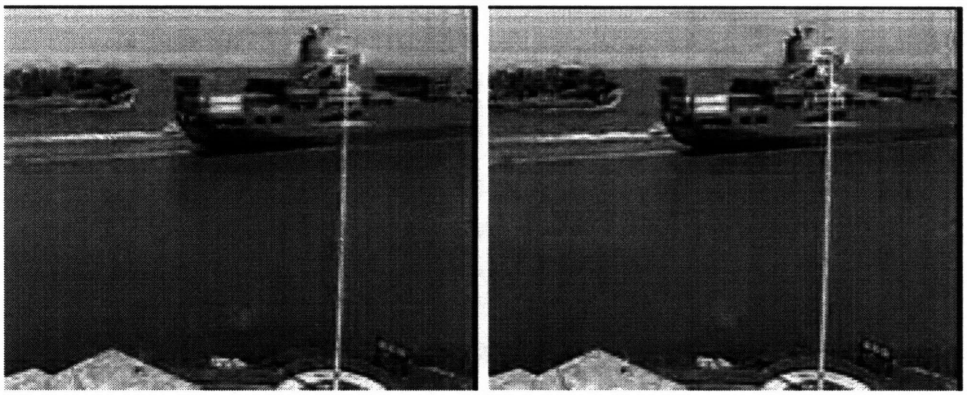

Figure 5-33: Frame 74 of Container at 20 kbits/sec

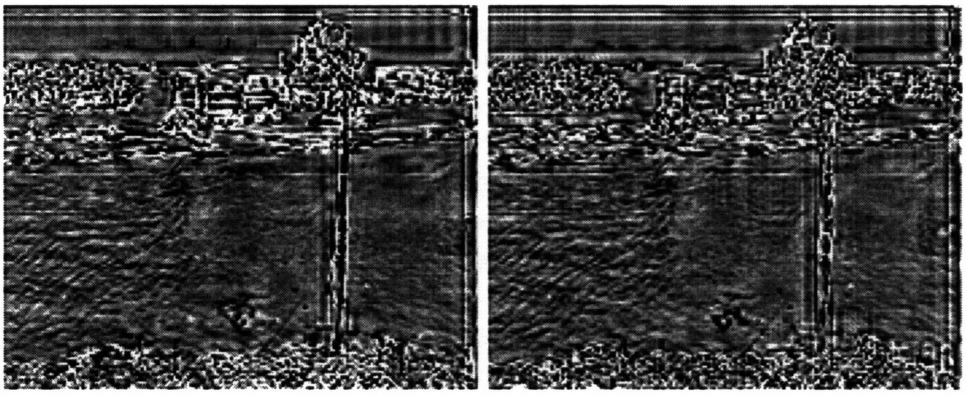

Figure 5-34: Error images, frame 74 of Container at 20 kbits/sec

The final frame of the sequence has been shown in Figure 5-33 to illustrate the reduced visibility of marcroblock edges in the lefthand image. This is due to the finer quantization used in the significance map based coder.

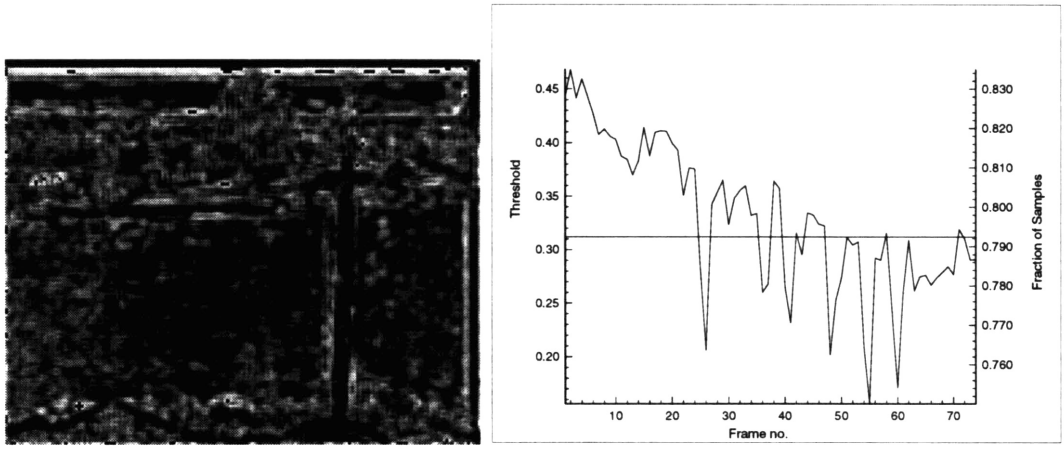

Figure 5-35: Significance map for frame 10 and fraction of samples determined to be insignificant for each frame for Container sequence at 20 kbits/sec

Looking at the significance map for frame 10 in Figure 5-35, it can bee seen that errors in flat areas, such as the borders of the water and ship and flagpole, are determined to be important. The error images also bear this out. The difference have been concentrated in the textured area around the ship and the edges near the flag pole. Errors in the flat areas of water would be very significant but since the residual pixel values are small, there is correspondingly little significance in these areas.

5.4 Conclusions

The visual perception based significance map works well in modifying the bit allocation in a low bitrate video coder. By quantizing residual pixels that are below a set threshold of significance the entropy of the residual image is reduced. The bitrate is maintained using finer quantization resulting in improved coded image quality. The error images show this effect, especially for the Akiyo sequence at 20kbits/sec. Errors that are distributed throughout the whole image become more concentrated in textured areas where they are less visible.

Bibliography

[1] B. Girod, "What's Wrong with Mean-Squared Error?," *Digital Images and Human Vision*, Chap. 15, MIT Press, 1993

[2] M. Irani, S. Hsu, P. Anandan, "Video Compression Using Mosaic Representations," to be published 1996

[3] "MPEG4 Proposal Package Description (PPD) Revision 2," ISO/IEC JTC1/SC29/WG11 N0937, Mar. 1995

[4] "CCIR Rec. 601-2, Encoding Parameters of digital television for studios," International Telecommunication Union, 1990

[5] P. J. Burt and E. H. Adelson, "The Laplacian Pyramid as a compact image code," *IEEE Transactions on Communications*, vol. COM-31, pp. 532-540, 1983

[6] W. E. Glenn, "Digital Image Compression Based on Visual Perception," *Digital Images and Human Vision*, Chap. 6, MIT Press, 1993

[7] E. H. Adelson and J. R. Bergen, "Spatiotemporal Energy Models for the Perception of Motion," *Journal of the Optical Society of America*, vol. 2, no. 2, pp. 284-299, Feb. 1985

[8] K. K. De Valois and E. Switkes, "Simultaneous Masking Interactions Between Chromatic and Luminance Gratings," *Journal of the Optical Society of America*, col. 73, no. 1, pp. 11-18, Jan. 1983

[9] J. Lubin, "A Visual Discrimination Model for Imaging System Design and Evaluation," *Visual Models for Target Detection and Recognition*, 1995

[10] M. A. Georgeson and G. D. Sullivan, "Contrast constancy: Deblurring in human vision by spatial frequency channels," *Journal of Physiology*, vol. 252, pp. 627-657, 1975

[11] J. Lee and B. W. Dickinson, "Temporally Adaptive Motion Interpolation Exploiting Temporal Masking in Visual Perception," *IEEE Transactions on Image Processing*, vol. 3, No. 5, pp. 513-526, Sep. 1994

[12] W. E. Glenn and K. G. Glenn, "The Design of Systems that Display Moving Images Based on Spatiotemporal Vision Data," *Journal of Imaging Technology*, vol. 15, no. 2, pp. 64-69, Apr. 1989

[13] F. W. Campbell and J. J. Kulikowski, "Orientation Selectivity of the Human Visual System," *Journal of Physiology*, vol. 187, pp. 437-445, 1996

[14] G. C. Phillips and H. R. Wilson, "Orientation Bandwiths of Spatial Mechanisms Measured by Masking," *Journal of the Optical Society of America*, vol. A1(2), pp. 226-232, 1984

[15] C. Blakemore and F. W. Campbell, "On the Existence of Neurons in the Human Visual System Selectively Sensitive to the Orientation and Size of Retinal Images," *Journal of Physiology*, vol. 203, pp. 237-260, 1969

[16] A. P. Ginsburg, "Visual Information Processing based on Spatial Filters Constrained by Biological Data," *Air Force Aerospace Medical Research Lab Tech Report*, AMRL-TR-78-129, 1978

[17] R. L. De Valois, Albrecht, and Thorell, *Vision Research*, vol. 22, pp. 545-559, 1982

[18] ITU-T Recommendation H.263: "Video Coding for Low Bitrate Communication", 1995

[19] "Video
Codec Test Model TMN5," *http://www.fou.telenor.no/brukere/DVC/tmn5/*, Telenor Research (TR), Jan. 1995

[20] MPEG95/N0437, "A Flexible Wavelet Transform Package for Image and Video Representation," Sodagar, Chiang, Lee, Martucci, Peterson, Suryadevara, Wine, and Zhang, 1995

[21] I. M. Daubiches, "Orthonormal Bases of Compactly Supported Wavelets," *Communications Pure and Applied Math*, vol. 41, pp. 909-996, 1988

[22] MPEG95/N0441, "A Zero-Tree Entropy Coding of Wavelet Compression of Video," Martucci, Chiang, Lee, Peterson, Sodagar, Suryadevara, Wine, and Zhang, 1995

[23] J. M. Shapiro, "Embedded Image Coding Using Zerotrees of Wavelet Coefficients," *IEEE Transaction on Signal Processing*, vol. 41, pp. 3445-3462, Dec. 1993