Traffic Characterization and Modeling of Wavelet-based VBR Encoded Video*

Yu Kuo[†] Student Member, IEEE **Sohail Zafar**[‡] Member, IEEE
Bijan Jabbari[§] Senior Member, IEEE

Abstract

Wavelet-based video codecs provide a hierarchical structure for the encoded data, which can cater to a wide variety of applications such as multimedia systems. The characteristics of such an encoder and its output, however, have not been well examined. In this paper, we investigate the output characteristics of a wavelet-based video codec and develop a composite model to capture the traffic behavior of its output video data.

Wavelet decomposition transforms the input video in a hierarchical structure with a number of subimages at different resolutions and scales. The top-level wavelet in this structure contains most of the signal energy. We first describe the characteristics of traffic generated by each subimage and the effect of dropping various subimages at the encoder on the signal-to-noise ratio at the receiver.

We then develop an N-state Markov model to describe the traffic behavior of the top wavelet. The behavior of the remaining wavelets are then obtained through estimation, based on the correlations between these subimages at the same level of resolution and those wavelets located at an immediate higher level. In our paper, a three-state Markov model is developed. The resulting traffic behavior described by various statistical properties, such as moments and correlations, etc., are then utilized to validify our model.

*This work was partially supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Computational and Technology Research, U.S. Department of Energy, under Contract W-31-109-Eng-38.

[†]Yu Kuo is with the ECE Dept. at George Mason University, Fairfax, VA.

[‡]Sohail Zafar is with the Mathematics and Computer Science Div. at Argonne National Lab., Argonne, IL.

[§]Bijan Jabbari is with the ECE Department at George Mason University, Fairfax, VA.

1 Introduction

Video applications have emerged as one of the most important user of the upcoming broadband integrated and ATM networks. Due to the nature of video data, a significant amount of bits are required to represent and to transport the signals. A video codec encodes and compresses the otherwise huge amount of data down to a manageable level, which then requires smaller bandwidth to transmit.

It has been noted that the statistical properties of such encoded video are quite different from those of the computer data and voice. In order to better understand the behavior of such video data, it is important to study its characteristics. Based on the study, an analytical model can be developed and represent a video source as part of the multimedia integrated networks. The data modeling then provides the possibilities of designing and implementing effective rate control schemes at the end node of a network, of further studying the behavior and performance of such aggregated video traffic, etc.

In this paper, we investigate the encoded output of a wavelet-based video codec. Such codecs provide a hierarchical structure for the encoded data, which can cater to a wide variety of applications such as multimedia systems. We will try to address the behavior of a hierarchical codec by characterizing the output of the codec when different bands/subimages are dropped. We will also carefully study the statistical characteristics of this data. A composite model is then proposed to model and capture the traffic behavior of its output video data. The composite model is flexible enough such that when certain bands are dropped, only minor modification to the analytical model is required to reflect the effect of dropping. For the subimage containing the most signal energy, we adopt a three-state Markov model to describe its traffic characteristics, while the remaining ones are then obtained through estimation, based on the correlations between subimages at different level of the pyramid structure.

2 Motion-adaptive Wavelet-based Video Codec

In this study, we consider the encoded data, generated by a motion-adaptive wavelet-based (MAW) video codec [4]. The MAW video codec employs wavelet decomposition for signal representation rather than utilizing the traditional Discrete Cosine Transform (DCT) as specified in the Motion Picture Expert Group (MPEG) standards [2]. In addition, multi-resolution motion estimation techniques are incorporated to improve the coding efficiency. It has been shown that the MAW video codec either outperforms the MPEG-based video codec or provides equal performance quality, in terms of complexity, bit rate offered and the signal-to-noise ratio

at that rate [3].

2.1 Encoding Schemes for a Full-motion Video Sequence

Full-motion video implies that there are various motion activities present in the sequence, ranging from high to low, as opposed to only very low to meduim motion in video conferencing setup. High motion activities refer to either relatively large translation of objects between consecutive frames, scene changes, or significant variation in luminance from one frame to the next. Low motion activities refer to small changes of translation or in luminance. Identification of these motion activites leads to an efficient tool for motion estimation and compensation.

Each frame of the incoming video is first hierarchically represented by applying the wavelet transform as opposed to partitioning the picture into small blocks and applying DCT on these blocks as done in the MPEG or any small-block transformed-based video codec. In doing so, the blocky effect occuring at low bit rates after reconstruction is eliminated because wavelet decomposition is a global transformation. The hierarchical representation of this transformation allows us to capitalize on the correlation among different subbands or subimages, which was previously not considered due to independent nature of transformtion of each block in DCT-based schemes.

After wavelet decomposition, multiresolution motion estimation is performed on the hierarchical structure using block-based search scheme [1]. Since in a wavelet decomposed video there is no concept of blocks, we define a *motion block* by partitioning the decomposed subimages into small squares similar to what is done in a DCT-based scheme. Note that these are just motion blocks and will go through a global transformation during the reconstruction process, thus eliminating any blockiness introduced by the block-based motion compensation. Motion estimation may or may not be carried out at all the subimages but all are motion compensated, except for the reference frames (I-frames). All the subimages are individually quantized and entropy coded, regardless what types of motion activities they may contain. That is, motion activities in subimages of a frame will not affect the selection of encoding schemes for that particular frame. The motion block still serves as the basic unit in the motion estimation process and when a matching block is discovered within a designated searching area, associated displacement and residual are quantized and encoded. Nevertheless, when a matching block cannot be found within the searching area, we encounter a *high motion* situation and the motion block itself is quantized and encoded for transmission.

To categorize motion activities and subsequently determine whether a motion block should be motion compensated or not, a minimum criteria difference function

is defined as follows:

$$\Phi(x,y) = \min \left\{ \sum_{k=1}^{p} \sum_{l=1}^{q} |I_i(k,l)|, \right.$$
$$\left. \arg \min_{x \in \Omega_x, y \in \Omega_y} \left\{ \sum_{k=1}^{p} \sum_{l=1}^{q} |I_i(k,l) - I_{i-1}(k+x, l+y)| \right\} \right\}, \quad (1)$$

where

$\Omega_x \in \{1,2, \ldots ,m\}$, $\Omega_y \in \{1,2, \ldots ,n\}$
x, y: displacements and $x \in \{-m, \ldots ,0, \ldots ,m\}$, $y \in \{-n, \ldots ,0, \ldots ,n\}$,
k, l: coordinates within a motion block,
p, q: dimensions of a motion block,
m, n: dimensions of a searching area,
I: Pixel intensity, and
i: frame sequence number.

The difference function Φ finds the minimal energy of a particular block within a searching area with associated displacements in the horizontal and vertical directions, x and y. If the minimal energy is generated from the motion block itself, i.e. from the first argument of equation (1), no matching block is actually discovered, and a high-motion situation is declared and x and y are irrelevant. On the other hand, if some other block in the searching area gives the minimal energy, the motion block is classified as having low to medium activity and the resulting motion vector is encoded and subsequently transmitted.

It is possible for the MAW video codec to match video images in a forward, backward, or bi-directional manner. Backward prediction allows the assessment of a video frame based on the knowledge of a previous frame, whereas forward prediction does so by using the next frame as a reference. Bi-direction precidtion combines the previous two schemes. In our motion-adaptive codec we choose the minimum energy obtained by either the block itself or, the minimum of forward, backward and bi-directional compensation. Therefore, the term *motion-adaptive* is used specifically in this context of motion estimation and compensation. It has been shown that motion-adaptive bi-directional motion compensation (B-frames in MPEG terminology) actually results in a lower data rate [3] than one without the bidirectional scheme.

In this study, we only focus on backward prediction for simplicity reasons: a smaller buffer is sufficient for the decoding and the encoded data need not be sent out of order. Therefore, our video sequence consists of a collection of I- and P-type frames, where I and P indicate corresponding encoding scheme. P frames refer to the aforementioned predicted frames which employ interframe coding, whereas I frames refer to refresh frames. The includion of I frames periodically provides a

clean and more appropriate reference for the frames that will follow. It also prevents prediction errors from being propagated beyond one refresh cycle (GOP in MPEG terminology). In order to remain as a reference, an entire picture is intraframe coded as it is without relying on any other frame. That is, the encoded video data along is sufficient to reconstruct that picture. A resulting video sequence is thus formed by a collection of IPPP..PPIPPP..PP.. frames arranged in a repetitive pattern.

2.2 MAW Codec Configuration

A block diagram of the MAW video codec is shown in Figure 1. Its functional components perform wavelet decomposition, quantization/dequantization, frame store, motion estimation, as well as DPCM and entropy codings. An incoming video frame may travel through these functional modules of the video codec via two different routes. When a video frame arrives at the input node of the codec, it is first wavelet decomposed into several subimages of different scales and resolutions. The number of levels and the number of subimages in each level depend on the transform parameters that have been chosen at the time of encoding.

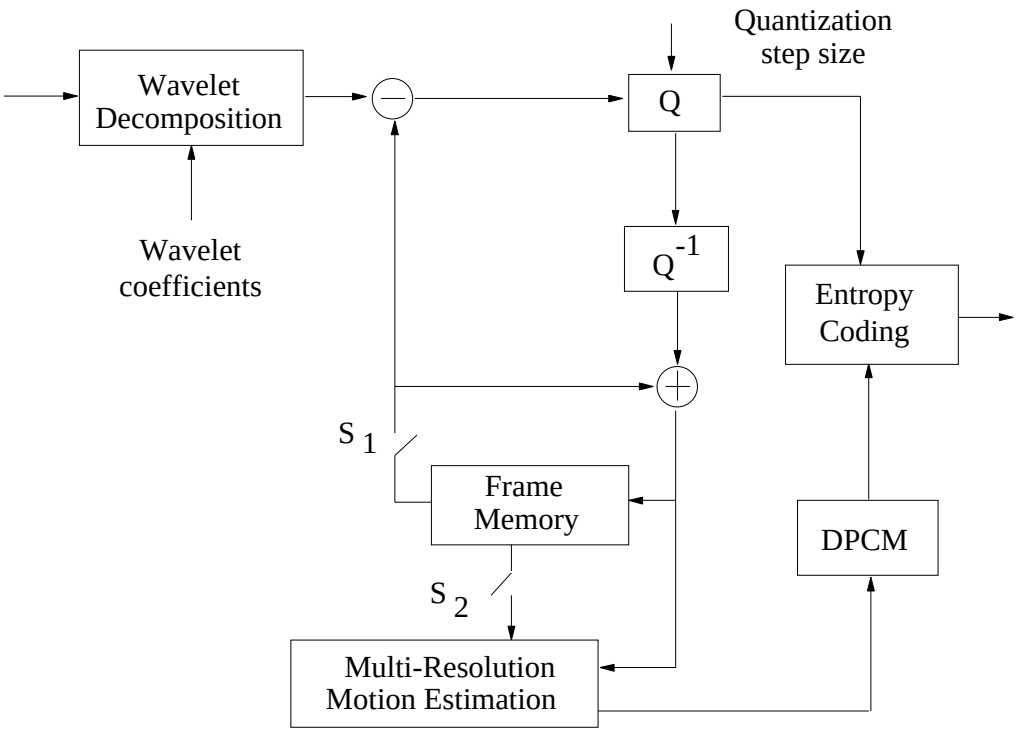


Figure 1: Configuration of the Motion-adaptive Wavelet-based Video Codec

The frames are categorized in a similar fashion as the MPEG standard. The first frame is an I-frame and therefore no motion compensation will be performed. After decomposition, every subimage of an I frame is quantized, entropy coded, and transmited. Meanwhile, the quantized image is dequantized and stored in *frame memory* (FM) for motion prediction of the next frame. Since an I frame is processed without referencing to previous frames, switches S_1 and S_2 remain open and thus disable the interframe functional components.

The next couple of frames are P-frames and their encoding require a slightly different scenario involving all the functional blocks in the MAW video codec. Both of the switches S_1 and S_2 are closed thereby activating the motion estimation in the feedback loop. After an incoming image is wavelet decomposed, each subimage is treated as follows; A current subimage is motion compensated against the corresponding subimage in the frame memory which is actually a reconstructed previous frame. The residual subimages after motion compensation are quantized and transmitted. The motion vectors obtained, are DPCM coded and sent along with the quantized residual frame difference information coming from the upper branch of the video codec. In order to update the information stored in the FM, the quantized residual is dequantized, motion compensated, and combined with the contents of the frame memory, updating it to the current frame again.

2.3 Hierarchical Structure of the Encoded Video

Wavelet decomposition tranforms an image into several lower resolution subimages which actually correspond to different spatial frequency bands. These sunimages can be further decomposed by repeating the transformation process into another set of subimages with even lower resolution, and thus generating a hierarchical representation of the encoded data. The number of resolutions and levels can vary from one application to another.

In this study, we have three levels of resolution, and, within each level, four subimages are generated with one of them been subsequently filtered into another four. Figure 2 illustrates a composite filter structure, which produces hierarchical results as such. This figure provides a detailed view of the wavelet-transform component of the codec shown in Figure 1. The decomposition is performed by using 1D filters separately in the horizontal and the vertical directions. An incoming image is first horizontally filtered by a low pass (L) and a high pass (H) filter. The two resultant images, one smoothed and the other detail image, respectively, are downsampled by a factor of two in the horrizontal direction and next filtered by the pair (L and H), this time along the vertical direction. The resulting subimages are now downsampled in the vertical direction. While focusing on one single level of decomposition, the double low-passed subimage is expected to contain most of

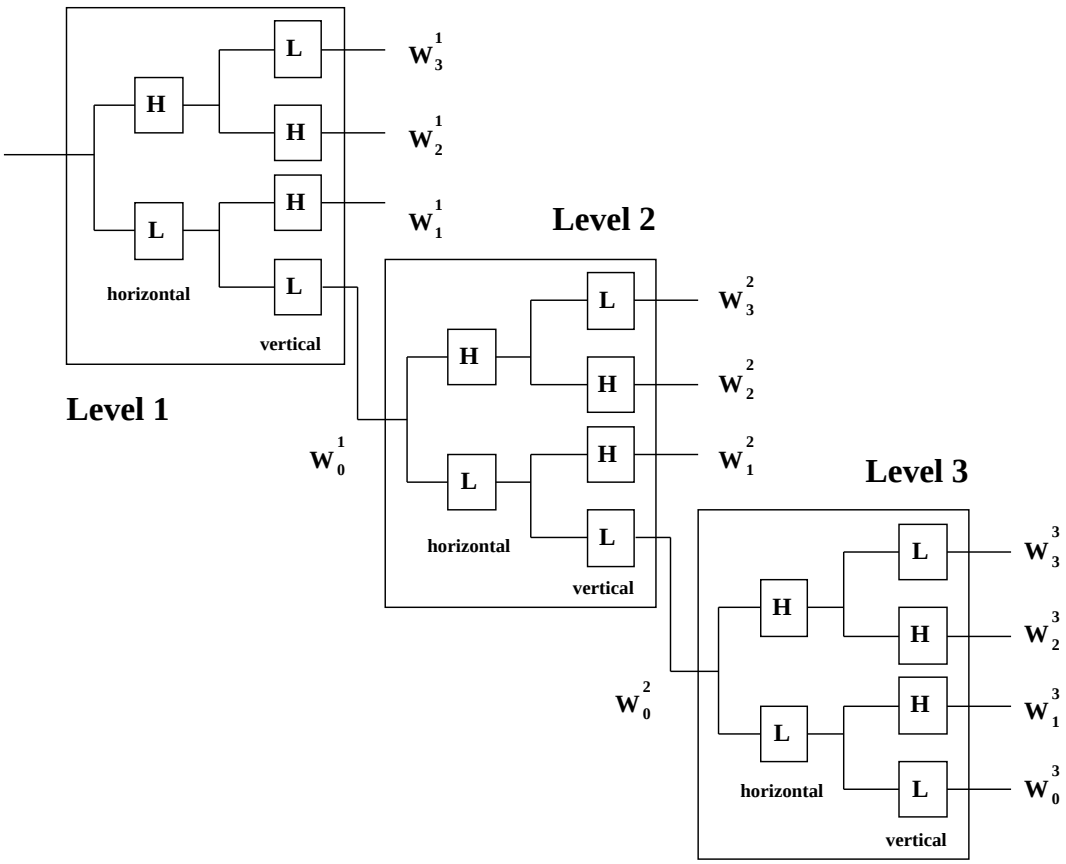


Figure 2: Filter Configuration of Wavelet Decomposition

the original signal energy, whereas the remainder contains high-frequency detail of different spatial orientations.

Three identical horizontal/vertical filter structures (or levels) are cascaded in Figure 2, since there are three iterations of decomposition in our study. Various output and components are denoted by the following notations. W_i^l refers to a transformed subimage and its associated superscript l and subscript i indicate what level and which subimage it is in the pyramid structure of Figure 3.

The term *level* implies how many iterations take place and how many folds the resolution and scale of an original image are reduced to. The subimage that is located at the top of the pyramid is the result of the lowpass of all the levels. Therefore, in our example, level 1 is the first iteration which produces four subimages, each of which is one fourth in resolution and in size of their original. W_3^1 is the third subimage with horizontal detail generated after level 1. Our subimages for one particular frame can then be represented by the set $\{W_i^l,\ i$=0,1,2,3 and l=1,2,3$\}$ and later refered to as $W1$ through $W10$ with W_0^3 as $W1$, W_1^3 as $W2$ and W_3^1 as

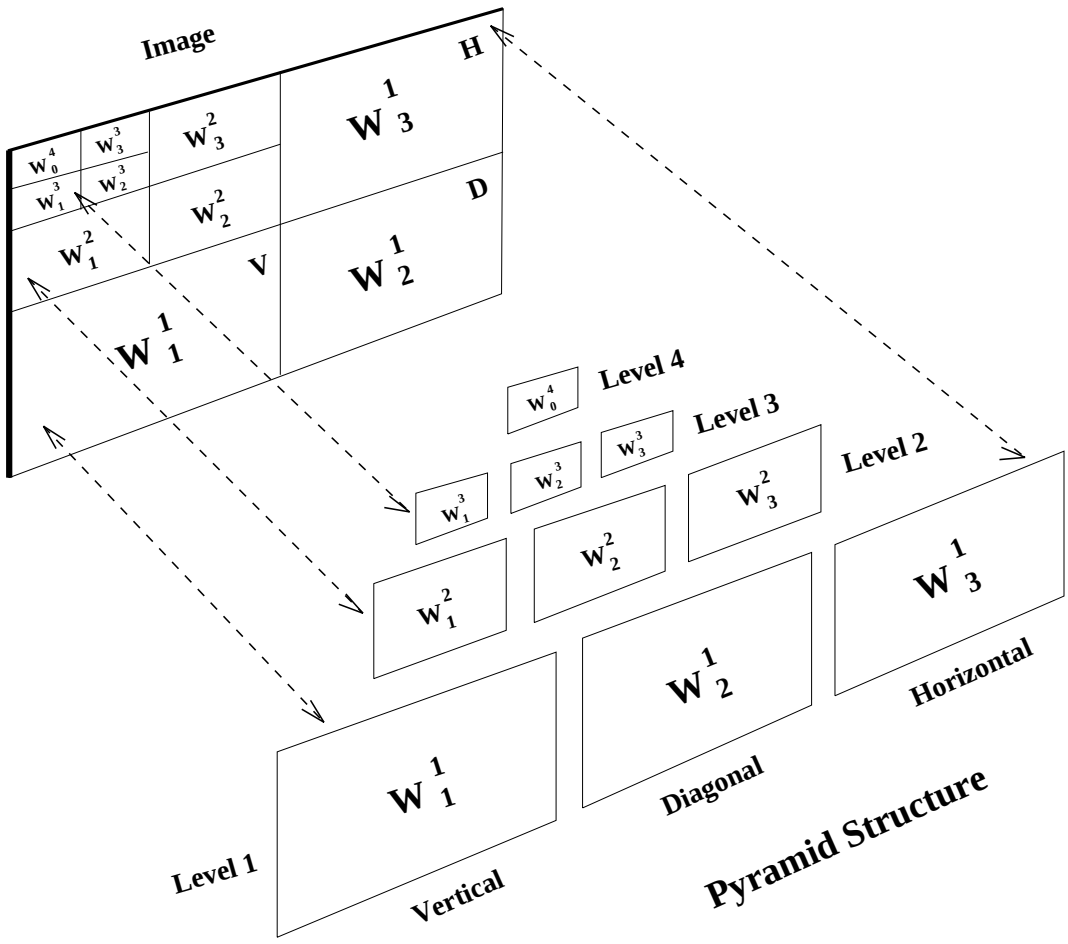


Figure 3: Hierachical Structure of Wavelet-decomposed Data

$W10$, respectively.

Based on the decomposition level and spatial detail orientations, Figure 3 shows the hierarchical arrangement of the wavelet-transformed subimages. Those located at the bottom level have the highest resolution and contain high-frequency detail of the original image in various directions, denoted by V (vertical), D (diagonal) and H (horizontal). After moving up one level in the pyramid structure, corresponding resolution and scale decrease.

Subimages which are decomposed by filters with identical spatial orientations are expected to exhibit relatively large correlation among each other. That is, strong correlation is anticipated to exist among W_i^l's, which locate in the same column of the hierarchy. For instance, Figure 2 illustrates that W_1^1, W_1^2, and W_1^3 are the results of a horizontal lowpass and vertical highpass of an arriving subimage. In

the subsequent sections, our data will support that they are indeed more correlated to each other than to other subimages such as W_2^2 or W_3^3. Moreover, within each column of the hierarchy, subimages of immediate adjacent levels are expected to exhibit stronger correlation, as opposed to those distanced by more than one level. This is due to the fact that further decompositions will simply uncorrelate data more. Therefore, W_1^2 and W_1^3 pair is more correlated than W_1^1 and W_1^3 pair. For the remaining sections, whenever statistics of subimages are to be presented altogether, they will be arranged in this pyramid format.

3 Characteristics of the Hierarchical Codec

The charcteristics of a video codec are very important in determining its robustness in an environment which does not guarantee a perfect channel of transmission. In an ATM environment, the most important factor for any application is the behaviour under cell loss and excessive delay. In the case of video, the latter can be treated as cell loss because delayed video cannot be used in a real-time environment. It is envisioned that there will be some prioritization scheme built in the network and packets will be dropped according to some priority, but rate control at the access point will still be needed to provide a graceful degradation in the quality of service.

In case of video codecs, it is necessary to reduce the output bit rate as the network becomes congested. For a traditional non-hierarchical codec, this can be achieved by increasing the quantization of the DCT coefficients thus decreasing the effective output rate. In a hierarchical codec like the MAW, there is an extra degree of freedom available due to the structure of the codec. Higher bands, which carry mostly details can be dropped at the encoder without changing the quantization.

Another scenario where it is important to know the behavior of the codec without transmiting all the subimages is an example of a hierarchical database of video sequences. A user may first browse through a video sequence at a much lower resolution and when the desired video has been selected, view it at full resolution.

In the following sections we will discuss the performance of the MAW codec when one or more of its subimages are dropped at the encoder to reduce the bit rate. We will first study the effect of dropping individual subimages and then study the SNR and effective bit rate after excluding multiple subbands from transmission.

3.1 Single Dropped Subimages

The signal-to-noise ratio for dropping off a single band for the football sequence is shown in Figure 4 and the corresponding output bit rate of the encoder is shown in

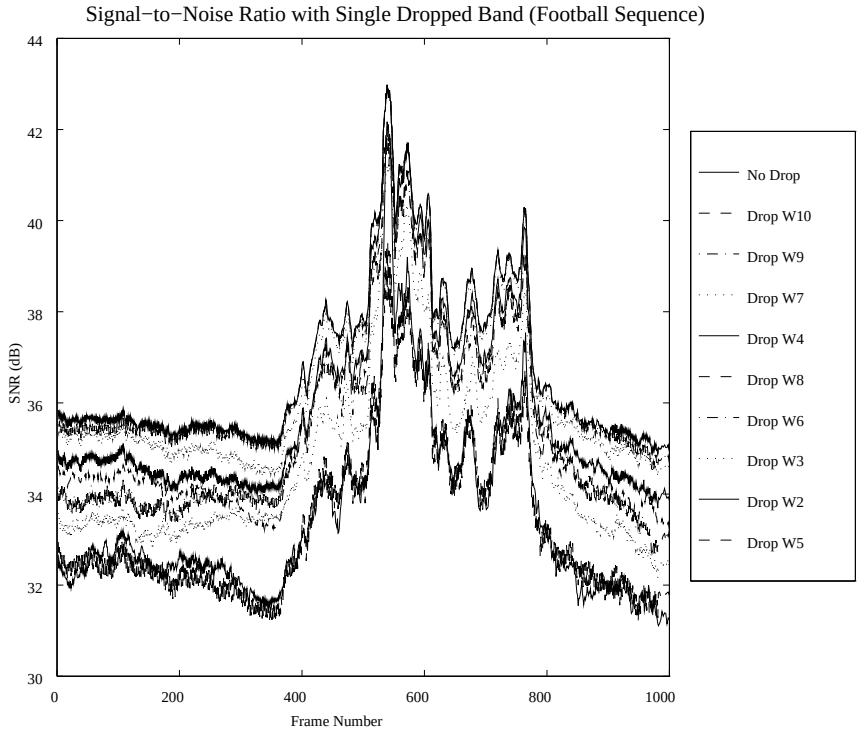


Figure 4: SNR with Single Dropped Band (Football Sequence)

Figure 5. The figures give an insight to the perceptual importance of the individual subimages. The subbands which have the biggest impact on SNR are $W5$ and $W2$ as we can observe that there is a drop of 4 dB with any of these subbands excluded from taking part in the transmission and reconstruction process. The impact on bit rate is not clearly visible from Figure 5, so we plotted the time-averged bit rate over the entire sequence and is shown in Figure 6. It is clear from this figure that dropping $W5$ also has the highest impact on the bit rate but the same is not true for $W2$. Although both the subimages result in equal degradation of the video quality, their impact on the bit rate is different. The average bit rate with all the subbands intact is 3.8986 Mbps and decreases by 800 Kbps for $W5$ but only 445 Kbps when $W2$ is dropped. It will therefore be better to drop $W5$ instead of $W2$ if the 4 dB decrease in video quality is acceptable. Note that the number of samples in $W5$ are four times as those of $W2$ therefore producing such an impact on bit rate while $W2$ is definately perceptually more important because it is higher in the hierarchy.

It can also be observed from the figures that $W10$ and $W9$ are not important at all and there seems to be an insignificant drop in SNR with them. The average bit rate is not affected by dropping $W10$ since there seems to be little or no contribution

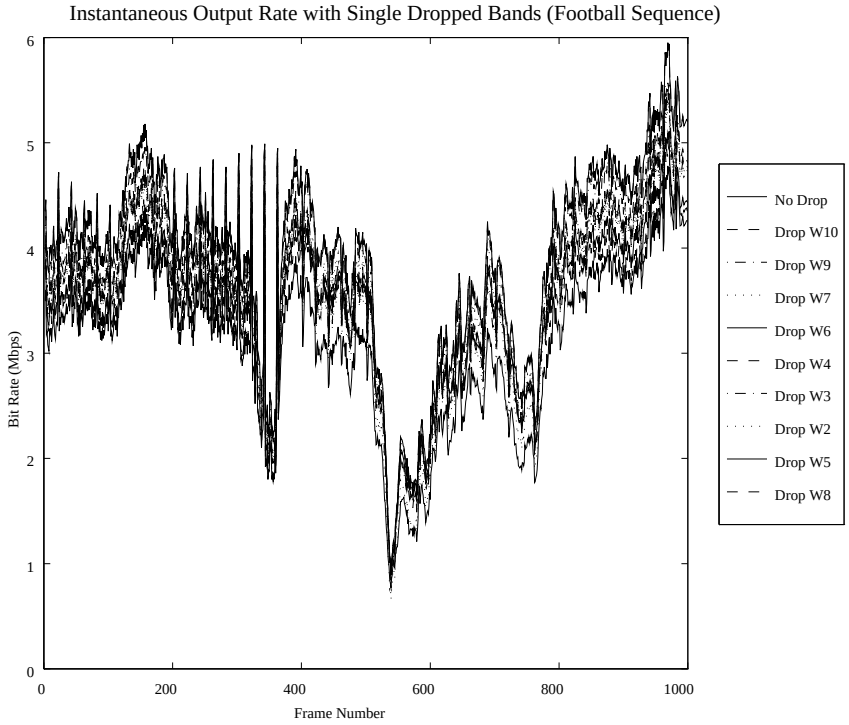


Figure 5: Bit Rate with Single Dropped Band (Football Sequence)

from this subimage in the whole process. Although dropping $W9$ decreases the average bit rate by almost 110 Kbps the insignificant effect on SNR shows that the perceptual contribution of this band is negligible.

Observe that dropping out $W8$ is the best choice when it comes to dropping a single subimage as it achieves a reduction of more than 550 Kbps with a lowering of only 1.5 dB in SNR. In general, we see that the bands which are most visually important are also the ones containing more energy and therefore contribute more to the bit rate. It is well known that the actual distribution of energy in different subimages of the multiresolution motion compensated video depends on many factors including the wavelet filter-bank, the video sequence itself, the motion compensation scheme used, and, the amount and direction of motion present in the video [4].

The legend in the above figures show the correct order in which the signal-to-noise ratio and the bit rate deteriorates as it is a little hard to judge without colored lines. We see that the order of increasing importance in terms of video quality for the football sequence is $W10 \rightarrow W9 \rightarrow W7 \rightarrow W4 \rightarrow W8 \rightarrow W6 \rightarrow W3 \rightarrow W2 \rightarrow W5$, while for the bit rate the order is $W10 \rightarrow W9 \rightarrow W7 \rightarrow W4 \rightarrow W3 \rightarrow W2 \rightarrow$

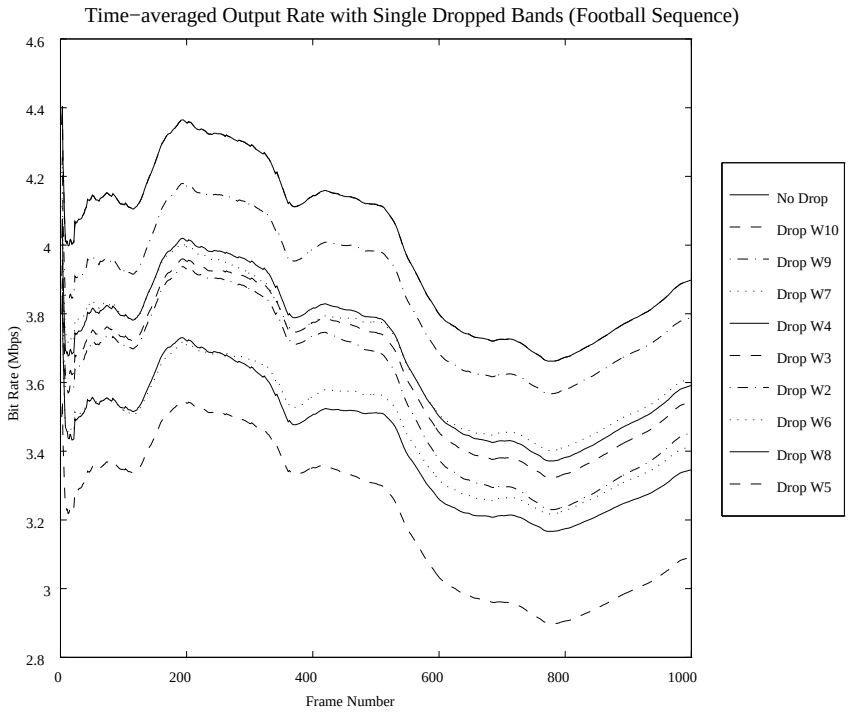
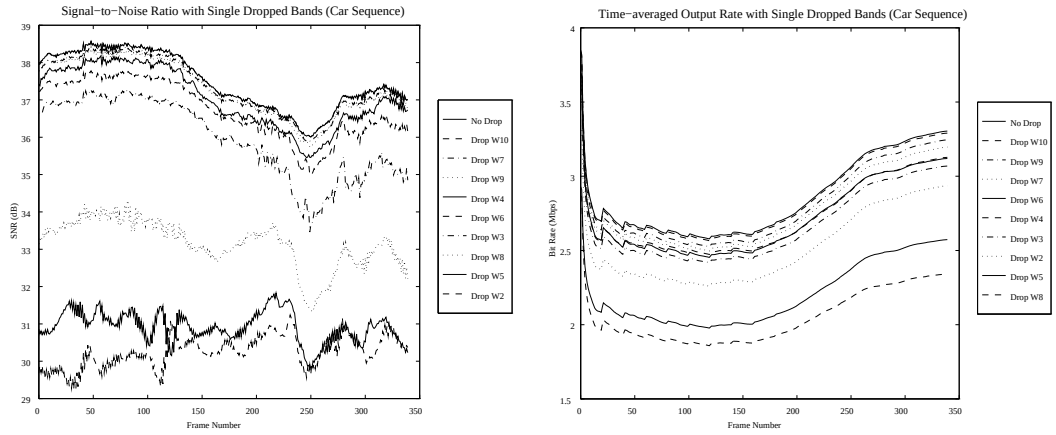


Figure 6: Time-averaged Bit Rate with Single Dropped Band (Football Sequence)

$W6 \rightarrow W8 \rightarrow W5$.

The same codec parameters were used to run another set of simulations on the car sequence. The signal-to-noise ratio and time-averaged bit rate are shown in Figures 7(a) and 7(b), respectively. We have almost similar behavior of the codec with this sequence as in the case of the football sequence. The difference is that the visual importance ordering of the subbands has changed slightly. We now observe that excluding $W10$, $W9$, $W7$, $W4$ and $W6$ has very little effect on the SNR which drops less than 0.5 dB in all these cases. The bit rate reduction is more significant though, dropping from 3.3 Mbps to 3.1 Mbps, a difference of 185 Kbps in case of $W6$ and less for others.

The most significant bands in the car sequence are again $W2$ and $W5$ but thier exclusion has a more serious effect on the SNR which drops by almost 7 dB accompanied by a 830 Kbps drop in bit rate. For this sequence the increasing order in which subimages can be visually prioritised is $W10 \rightarrow W7 \rightarrow W9 \rightarrow W4 \rightarrow W6 \rightarrow W3 \rightarrow W8 \rightarrow W5 \rightarrow W2$ and regarding the bit rate, the ordering is $W10 \rightarrow W9 \rightarrow W7 \rightarrow W6 \rightarrow W4 \rightarrow W3 \rightarrow W2 \rightarrow W5 \rightarrow W8$.



(a) SNR (b) Bit Rate

Figure 7: Performance of MAW Codec with Car Sequence

From the above observations it can be deducted that the actual visual importance of the subimages in the current environment depends largely on the input video, the amount of motion present, the direction of motion and also the prediction scheme used for motion compensation. But, in general, we can say that the lower subimages and the ones with diagonal spatial orientation are the least important and their contribution to the bit rate is also lower.

3.2 Multiple Dropped Subimages

In an integrated services network environment, as the network becomes more and more congested, the output bit rate of the codec has to be further reduced. This can be achieved either by increasing the quantization or by dropping more subbands from transmission. We will evaluate the second approach and study the performance of the codec when multiple subbands are dropped out from transmission at the encoder to reduce the effective bit rate.

Figure 8 shows the SNR when different sets of subbands are dropped from encoding. The corresponding time-averaged bit rate is shown in Figure 9. The figures reveal that if we drop $W10$, $W9$ and $W7$ altogeather, there is very little drop in the signal-to-noise ratio which is about 0.5 dB. This is accompanied by a significant drop in the average bit rate of 400 Kbps as shown by Figure 9. The amount of decrease in bit rate is almost the same if we add the reductions caused by individually dropping each of the subimages from the encoder. The values would have exactly

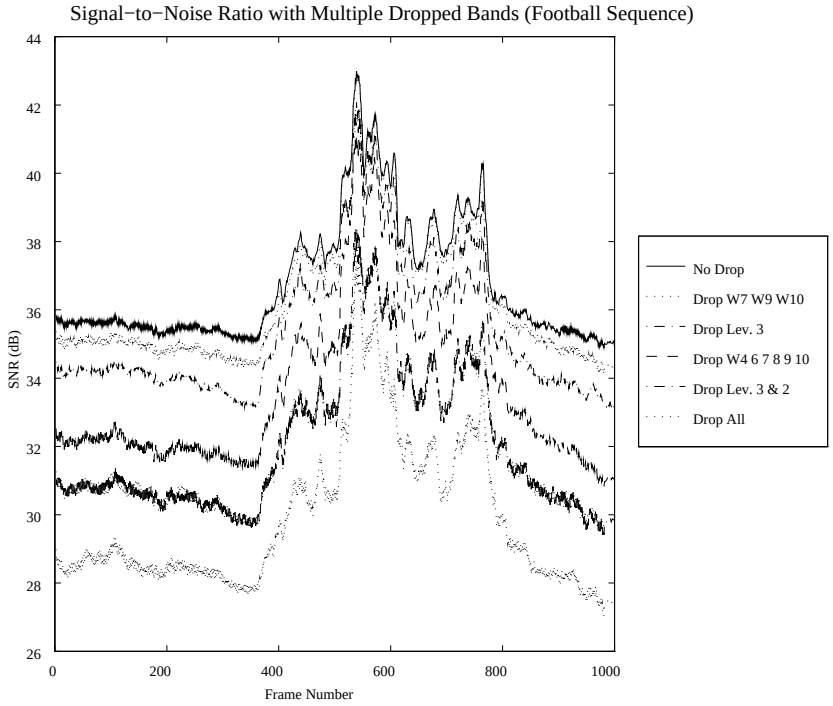


Figure 8: SNR with Multiple Dropped Bands (Football Sequence)

matched if there was no feedback loop in the encoder which tries to correct the degradation caused by dropping the subimages. Note that $W10$, $W9$ and $W7$ are the bands which cause the least amount of degradation in signal quality when they are individually dropped (Figure 4). This clearly shows that these bands are not visually important for this sequence.

We now add the next three least visually important subimages to our list of dropped subbands, and drop $W4$, $W6$, $W7$, $W8$, $W9$ and $W10$. This almost reduces the bit rate to half the original value of 3.9 Mbps to 2.16 Mbps, with only 4 dB drop in SNR from the original figure of 36 dB.

Compare the above results to the case where we drop the whole of level 3 instead of being selective. We see that the SNR drops by almost 2 dB and the bit rate falls by 660 Kbps, with respect to the value when all the bands are kept, as compared to values of 0.5 dB and 400 Kbps. This shows that inclusion of $W8$ was responsible in greater degradation in SNR which is supported by our observation from previous section of single dropped subimages.

Observe also that if we drop both levels 2 and 3, the SNR drop is 5 dB and the bit rate drops to 1.66 Mbps. This is a reduction of 500 Kbps from the former case of selective dropping ($W4$, $W6$, $W7$, $W8$, $W9$ and $W10$) with only a difference of

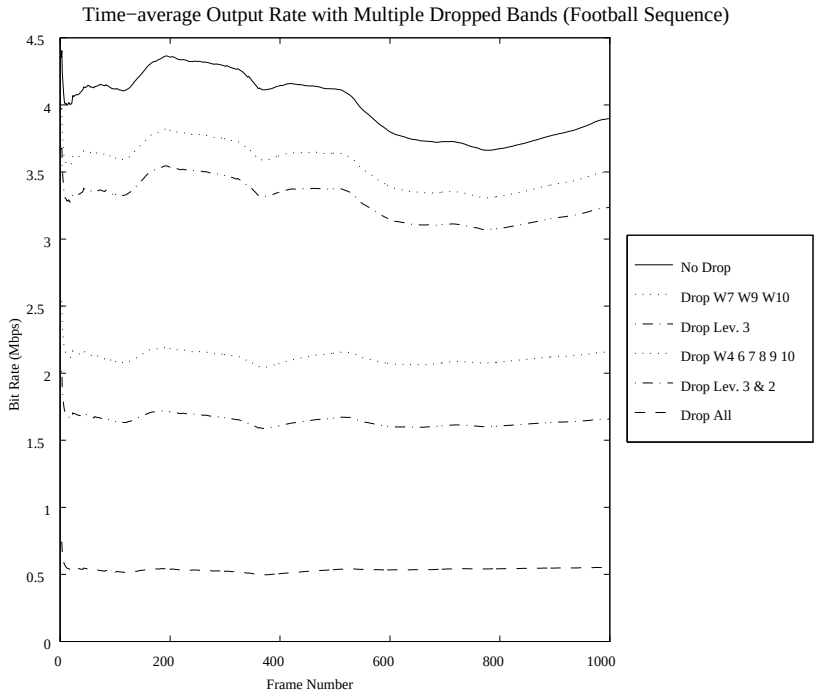


Figure 9: Time-averaged Bit Rate with Multiple Dropped Bands (Football Sequence)

1 dB. These differences in bit rates and signal-to-noise ratios remain almost constant over the entire length of the sequence except during the interval where there is very little motion and the bit rate drops and the signal-to-noise ratio increases for all the cases.

We also considered the worst case in which all the subimages are dropped except for $W1$. We observe that the codec still performs reasonably well giving an SNR of above 28 dB at an average bit rate of only 0.55 Mbps. Note that the SNR value is calculated by reconstructing the image to the original resolution of the input video. Also note that when we say that a subband is dropped, it is true for all the color components Y, U and V. For example, when $W10$ is dropped for Y component, it is also dropped for U and V as well. Results obtained for the car sequence are very similar to those of the football sequence. The differnces are similar to the differnces obtained in the singly dropped band case.

In this section we have tried to address the performance of a hierarchical codec by characterizing the behavior of the MAW codec when different bands/subimages are dropped from transmission. The study also helps us identify a general strategy that can be adopted to selectively drop bands as the network gets congested in order

to provide a graceful degradation in the quality of the video at the recieving end. It also helps to identify the selection of subimages for transmission in cases where the full bandwidth is not available and is known prior to transmission.

4 Characteristics of Multiscale Encoded Video

Each frame is a combination of 10 subimages. In this section, we'll start by discussing the characteristics of the overall frame, followed by those of the subimages since both important to our study due to the following reasons. Those observed and measured from the overall image represent part of the composite attributes of the video data and are one of the many factors to be considered while performing codec evaluation/design, network design, performance optimization, rate control, etc. On the other hand, the characteristics associated with each subimage can provide us insights as to how the composite results are achieved and therefore benefit the modeling of this type of data. By examining the properties of subimages, we can further understand the qualities and performance of the wavelet-based codec. Going one step further, we have seen in section 3 that the hierarchical structure of the subimages can be exploited to provide a flexible rate control scheme. The excercise carried out at a subimage level can be used to develop strategy to control the output rates at an end node, without compromising too much of the picture quality. Therefore, it is important to study the properties of the data at this level as well.

The statistics of the output video that are of interest include 1) composite bit rates of Y, U, and V components of the individual subimages and also of the overall frame, 2) histograms of the above bit rates, 3) correlations between consecutive bit rates within each subimage and the overall frame, and 4) the cross correlations among various subimages.

4.1 Characteristics of the Output Video

For the overall frame, we are interested in its bit rate profile, histogram, and correlations between consecutive frames.

4.1.1 Bit Rate Profile

The composite bit rate profile contains periodic and significant jumps in magnitude at every refresh points. Recall that an I frame is necessary to provide a clean reference. Since an I frame is always intraframe encoded, data compression is only provided by quantization and entropy coding, and thus the relatively high resultant

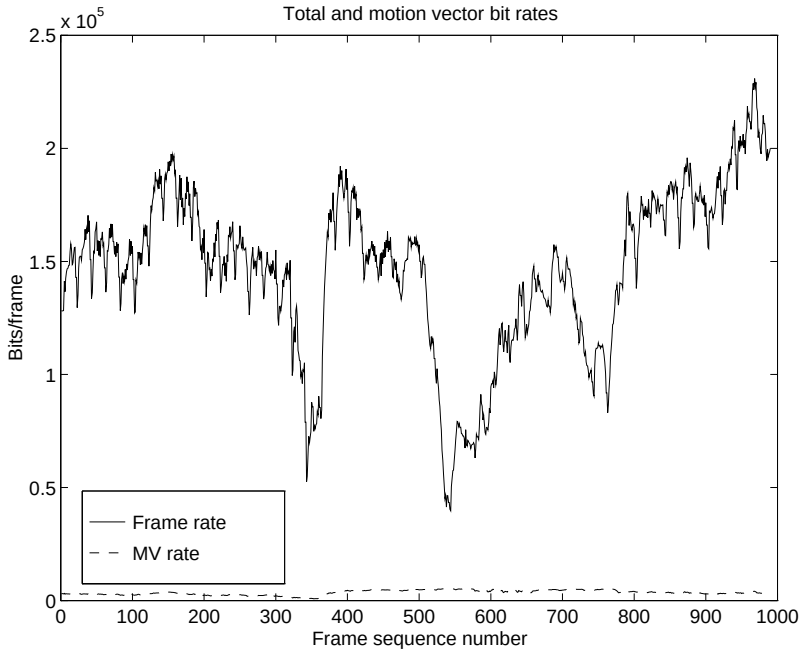


Figure 10: Bit Rate Profile of the Overall Frame

rates. Compared to P frames, in addition to interframe coding, motion compensation is also introduced to further reduce the bit rates. It is therefore obvious that the bit rate of an I frame will be greater than that of a P frame, and thus the large magnitudes at the refreshes in a bit rate profile. Since such instances at which some relatively large magnitudes occur are predictable, one possible approach to characterize and analyze the encoded video data is to smooth out the profile by filtering out the spikes and consider the *slow varying data* and the refreshes separately. That is, output periodically generated at frames adapting only intraframe coding can be independently examined. Intuitively, it is advantageous to do so because different encoding schemes achieve different level of compression efficiency on the output rates, which will most likely be reflected in their statistical characteristics as well.

Two sets of data can be subsequently extracted from one composite bit rate sequence: one has a magnitude centering around 150 kbits/frame as shown in Figure 10, whereas the refreshes generally have large rates, mostly ranging from 350 to 450 kbits/frame.

Although the refreshes have been removed, the total bit rates still exhibit a periodicity property, with a cycle of every 20 frames. This is no surprise since the refresh cycle for this data set is 20 frames. Assume there are no scene changes

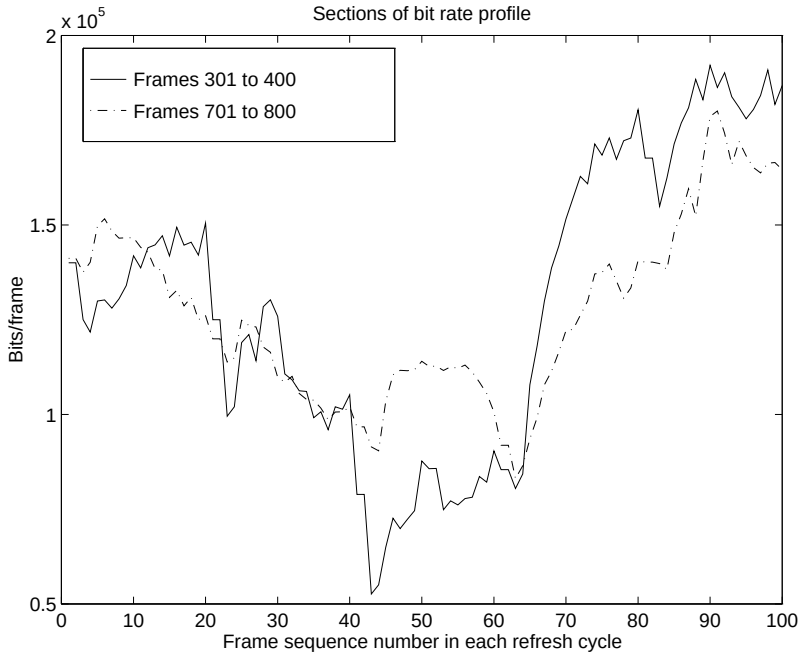


Figure 11: Sections of Bit Rate Profile with Scene Changes

within the refresh cycle for a moment, the first several frames immediately following a refresh are encoded with lower rates, due to the fact that less errors have been propagated. As more frame differences are computed and motion estimated, errors accumulate and the frame rates are expected to show an increase for the next several frames. Depending on the scene content in the next few frames, the rates can go up or down, but gradually, until either the next refresh takes place or a scene change occurs. This trend of periodic rate variation is generally anticipated to repeat itself in each refresh cycle. When a scene change occurs, our codec switches to intraframe coding and the aforementioned cycle is interrupted. Once the frame rate is brought to another magnitude level, the cycle supposedly starts again until the next scene change or refresh.

Two sections of the bit rate profile have been enlarged in Figure 11 to illustrate the effect of a scene change on bit rates and the otherwise slow variations. Within a 20-frame refresh cycle, both sections exhibit a relatively large increase right after a refresh between the 60th and the 70th frames. An approximate 80 Kbits jump occured for the first section, while a 50 Kbits jump did for the second section. Considering the total bit rates are in the vicinity of 150 Kbits, these changes are significant. Otherwise, the variations are gradual, with a magnitude significantly small compared to the absolute bit rate itself at each frame.

Our data is, in fact, collected separately in terms of its Y, U, and V components

of the encoded pixel differences as well as the Y, U, and V components of the motion vectors. In the same token, it is a possible scenario to consider pixel and motion information respectively. However, for this particular video data and the MAW codec, the motion information constitutes a very small portion of the total bit rate per frame, at an average ratio of 2 to 100. Thus, to investigate them individually does not appear efficient. Figure 10 also illustrates the motion vector information.

4.1.2 Histogram and Correlations of Bit Rates

Figure 12 presents the histogram and correlation coefficients for the total bit rates. The histogram suggests that the density function of the total bit rates appears to be a composite result of several Gaussian densities. And the correlation coefficients indicate that the encoded bit rates have a strong correlation between those of the consecutive frames. Both observations can be explained based on the implemented encoding algorithms in our video codec, the content of the frame sequence, etc.

A motion picture or a section of a motion picture usually consists of various scenes which can be categorized by their motion content. These include scenes with very few motions, with limited, moderate, or violent motions, or zooming and

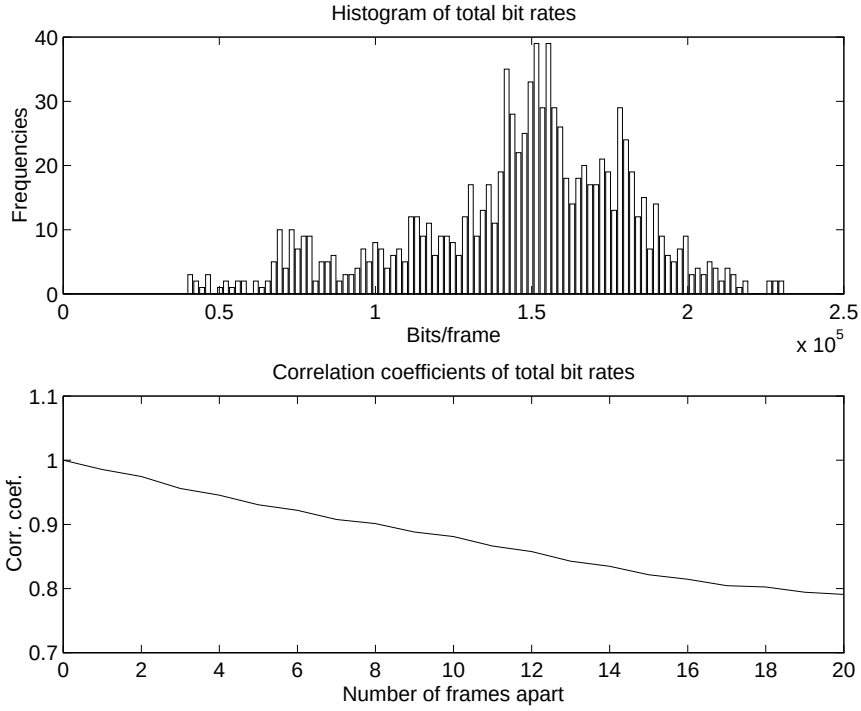


Figure 12: Histogram and Correlation Coefficients of Total Bit Rates

panning, etc. In our study, this categorization maps different scenes to low, medium, and high motion classes for the encoding purpose. Although which encoding schemes to be used depends on the picture content and the thresholds set in the video encoder, we can still expect that three modes of encoding will all be adapted during the entire encoding process of a frame sequence, assuming the number of frames is a reasonable one.

In a very simplified manner, one anticipates that low, medium, and high motion frames will result in a rate with a small, medium, and large magnitude, respectively, and therefore a composite density function of three Gaussians. However, our video codec produces results more complicated than the simplified case. In our codec implementation, a basic encoding unit is a block which varies in size according to the scale of the subimage, starting from 2 by 2 at the top level and increasing to 8 pixel by 8 pixel at the bottom. But each block is still individually categorized as being a low, medium or high block and accordingly encoded. As a result, each frame contains some numbers of blocks from each group and can not be directly labeled as, for example, a low-motion frame. This block-based encoding scheme makes the association of each Gaussian density in Figure 12 to a motion class more difficult. Nevertheless, if a frame contains mostly low-motion blocks, its total bit rate should be from the density function at the lower end of the axis. Similarly, for a frame which is mainly composed of high-motion blocks, its bit rate is expected to come from the density at the higher end. For our encoded data, the distinction between the second and the third higher densities in the histogram curve is not that obvious as shown in Figure 12. This implies that there may be quite a few frames containing certain close numbers of medium-motion and high-motion blocks. The resultant bit rates per frame are then of some close magnitudes and don't necessary indicate which density they are from.

The correlation coefficient curve in Figure 12 suggests that the frame rates are highly correlated between consecutive frames. As previously discussed, within one particular scene, an object translates some distance from one frame to another. Most likely an object moves along in the same direction with a similar rate, from a previous frame to the current one and then to the next. The motions are predictable to some extend and thus present high correlation between consecutive frames.

4.2 Characteristics of the Subimages

One unique feature about the wavelet-based video codec is that an input image is decomposed into several subimages and separately encoded, either intraframely or interframely. To understand the impact and benefit of this decomposition in traffic modeling and rate control, it is important to study the characteristics of these subimages. We ahve already seen that we can take advantage of this hierarchical

structure to develop a rate control scheme at an end node of an integrated network.

4.2.1 Bit Rate Profiles of the Subimages

Figure 13 shows the profiles of average bits per pixel per frame for each subimage, with refreshes removed. Note that here the hierarchical presentation of the statistical curves of all subimages conform to the structure illustrated in both Figure 2 and Figure 3. Most curves have been drawn to the same scale for clarification purpose, except $W10$ whose average magnitude is significantly small compared to those of the other nine wavelets. The unit along the y axis is *bits/pixel,frame* and x axis, frame sequence number.

The average pixel rates in $W1$, $W2$, ... $W8$ basically exhibit a similar variation pattern. This resemblance between certain subimages can also be observed, when the data is presented in its actual rates, *bits/frame*, as well as in its composite profile

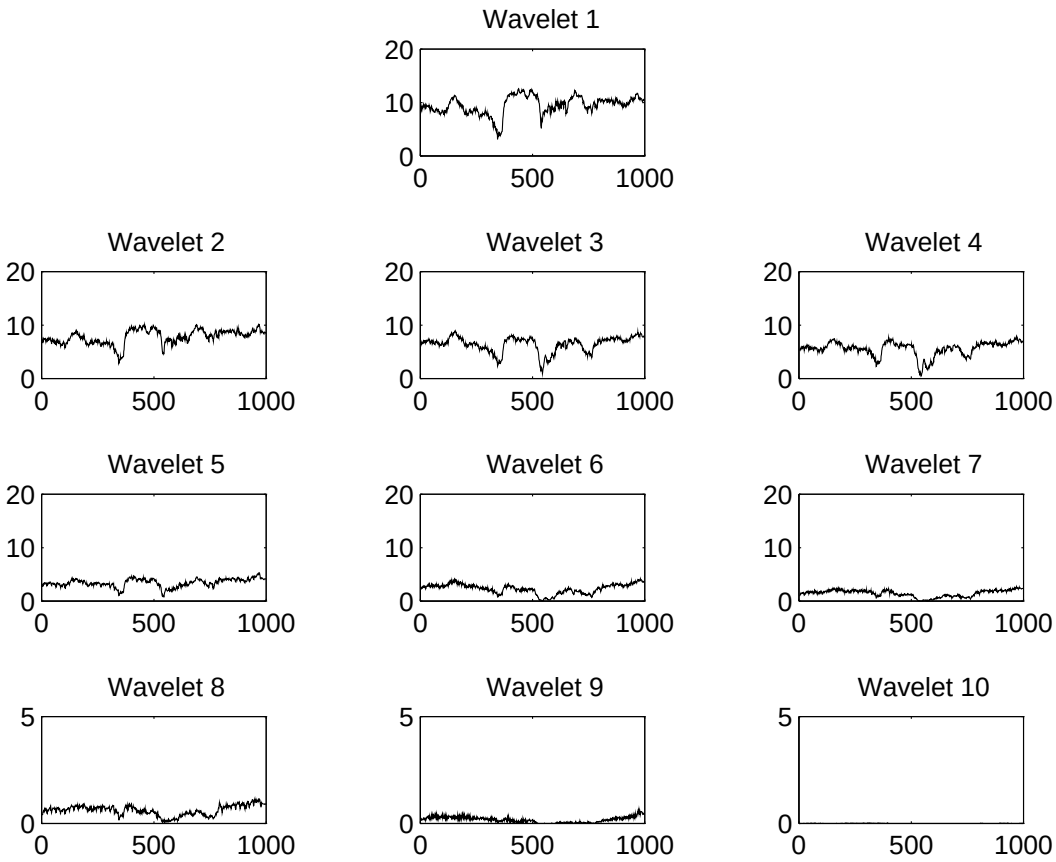


Figure 13: Average Bit Rate Profiles of Subimages

which is shown in Figure 10. The top four profiles, either the actual or average bit rate, exhibit drops at around frame 350 and frame 550 and center around 7 or 8 bits/pixel,frame or 17 kbits/frame. This strong similarity among this group (or level) of the hierarchy is due to the fact that they all are a filtered result of one subimage, W_0^2 (Figure 2). We can further break down the group and apply the same argument to the pair of $W1$ and $W2$, or $W3$ and $W4$. Since $W1$ and $W2$ are the results processed from the same low-passed W_0^2, we expect them to form a pair whose statistical properties are more alike than to those of the $W2$ and $W3$ pair, which is verified by our sample data here. Based on the same reasoning, we will also show in the next section that the correlation within each pair is stronger than that between subimages from different pairs.

4.2.2 Histograms and Correlations of Subimages

Probability densities of the average pixel rates for each subimage is illustrated in Figure 14. The bottom-level densities are not drawn to the same scale as the others. It's noted that, for this football sequence, the means of subimages follow a descending order. By observing Figure 2, we know that most of the signal energy is preserved at the output of the third decomposition process. That is, $W1$, $W2$, $W3$, and $W4$ contain most of the energy and thus have a higher bit rates than the lower level ones. Preceded by $W1$, $W2$ is expected to contain more energy than $W3$ and $W4$, since the former is a detail of a low-passed subimage, while the latters are the details resulting from the highpass filter. Whether the mean of $W3$ is greater than that of $W4$, or visa versa, largely depend on the content of a video sequence. In the football sequence, for example, players run in a horizontal direction and balls fly in the air in a diagonal direction. There appears to be less horizontal details. Thus the mean of pixel rates in $W3$, which contains the diagonal details, are larger than that in $W4$, which contains the horizontal details.

Not considering $W1$ for a moment, the density functions in the middle and the right columns in Figure 14 show more resemblance graphically with each other, than to those in the left column. This similarity results from that their source subimage, from which they are generated, is identical, as anticipated from the previous section. Qqplots between pixel rates of several subimages are included in Figure 15. The qqplots provide information as to whether two datasets have the same distribution or not. An approximately straight line indicates that the distribution from both sets is the same. From Figure 15 we can verify that the pixel rates of two vertically neighboring subimages within one column of the pyramid structure have the same distribution. It is also true for the pixel rates of the adjacent subimages from the diagonal and horizontal columns. For instance, for the first case, we show $W1$ *vs* $W2$ and $W2$ *vs* $W5$, while for the second case, we have $W3$ *vs* $W4$ and $W6$ *vs* $W7$. It is also noted that the plots contain some S shape curves between $W1$, $W3$ and

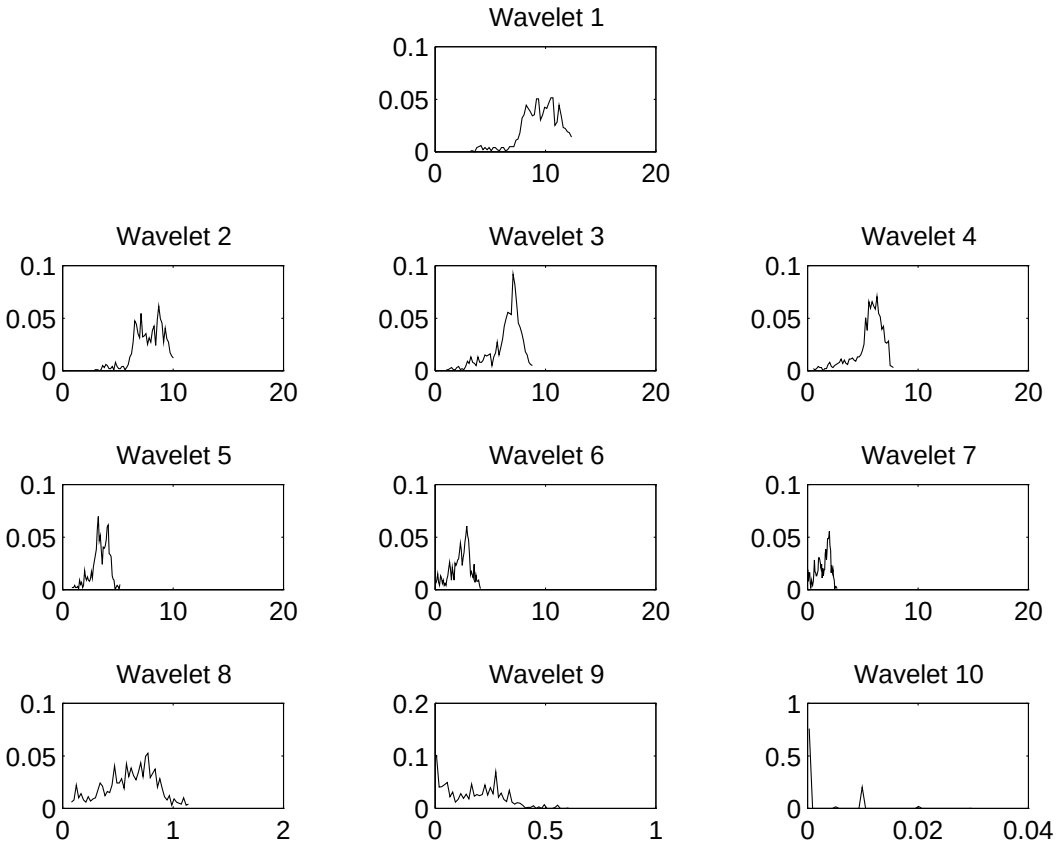


Figure 14: Probability Density Function of Subimages

$W4$, which implies that one distribution has longer tails than the other. When three densities are plotted together, we can easily visualize that the density of $W1$ has tails that are indeed longer than the other two.

We are also interested in learning cross correlation between bit rates of subimages located in immediately adjacent levels, in the same column but not necessarily in the adjacent levels, in the same level but adjacent columns, etc. A few selected combinations of subimages are incorporated in Figure 16. The numbers along the x axis represent the number of frames apart when the cross correlation coefficients are computed. The y axis represents the coefficient itself. These graphs once again verify some of the observations that we found in discussing rate profiles and density functions. $W1$ and $W2$ have strong correlation with a coefficient greater than 0.95, when there is no lagging. And the correlation continues to be strong even after the 10th frame. Although $W5$ is located in the same column as $W2$ in the pyramid data structure, the correlation between $W1$ and $W5$ is significantly reduced due to the skipping of one level. Skipping a level can be interpreted as going though one

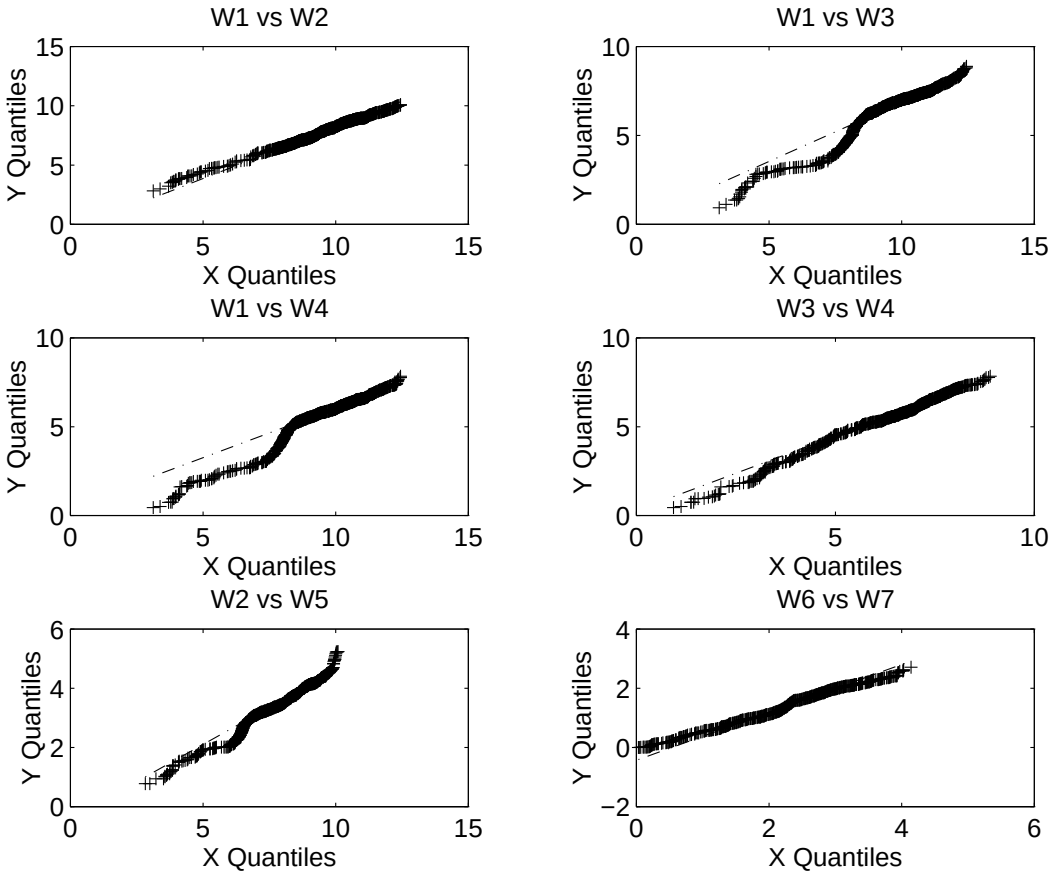


Figure 15: Qqplots of Pixel Rates between Selected Subimages

more transformation process, as demonstrated in Figure 2. The comparing data of two subimages is further decomposed and therefore exhibits less correlation. We can find this decrease in correlation phenomena in subplots $W1$ *vs* $W5$, $W2$ *vs* $W8$, etc. When no levels have been skipped, two subimages demonstrate relatively strong correlation as long as they are located in the same column in the pyramid structure. For example, $W2$ *vs* $W5$, $W3$ *vs* $W6$, $W4$ *vs* $W7$, etc. are pairs as such. Earlier the qqplots, in Figure 15, suggests that the pairs $W3$ and $W4$ as well as $W6$ and $W7$ have the same distributions. Here we also find them having very strong correlation within each pair. It's even more interesting to note that the correlation between $W3$ and $W4$ of the same level appears stronger than those between $W3$ *vs* $W6$ or $W4$ *vs* $W7$ of the same column but adjacent levels. This phenomena is again due to the increasing decorrelation as a subimage traveling down the transformation path in Figure 2. Between $W3$ and $W4$, there is one decomposition process taking place in the vertical direction and they are generated from an identical low-passed subimage. On the other hand, there are more than one decomposition between $W3$ and $W6$.

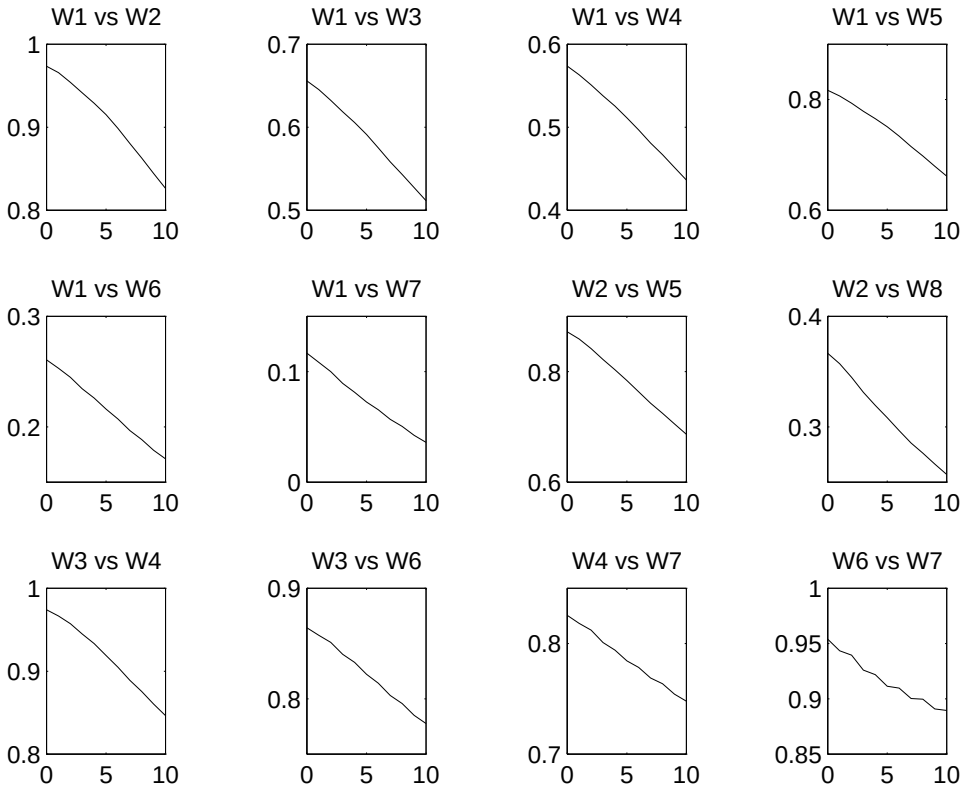


Figure 16: Selected Cross Correlation Coefficients of Bit Rates

5 Modeling of Multiscale Encoded Video

Based on the observation on the data characteristics, a composite model is proposed to capture the behavior of such video traffic. It was noted that the top subimage located in the pyramid structure contains most of the remaining signal energy. This top subimage thus should be modeled as accurately as possible, and can be utilized to estimate the behavior of the remaining subimages. Our modeling approach can, therefore, be grouped into several steps: 1) model the refreshes, 2) model the top subimage, and 3) estimate the remainders as well as obtain the composite model for total bit rates.

5.1 Modeling of the Refreshes

Refreshes occur periodically with a significantly large magnitude. The time instances at which refreshes take place are predictable. Their large magnitudes separate the corresponding probability distribution from those of the relatively lower rates in the density plot such as those shown in Figure 12. We therefore propose to individually model the refreshes so that both the refreshes and the slow-varying video data can be better explored and modeled.

The average pixel rates at refresh frames are first identified in our video sequence and filtered. Its probability density is then plotted, which is shown in Figure 17. We start by considering a Gaussian distribution to approximate that of the refreshes. It is not unreasonable to assume a Gaussian distribution for this case. The density plot itself approximates a normal one. Moreover, the encoded rates are actually content dependent. Given a video sequence, such rates can have magnitudes that would fall most likely anywhere on the bit rate axis. When there are sufficient number of frames, they are then expected to result in a Gaussian distribution. Various statistics of this data set, including mean, standard deviation, the third and the fourth moments, are measured and presented in Table 1.

Two approaches were implemented to model the refreshes. The correlation coefficient between two consecutive refresh frames has a value of 0.93. Taking correlation into consideration, our first method thus models the data stream as a sequence from

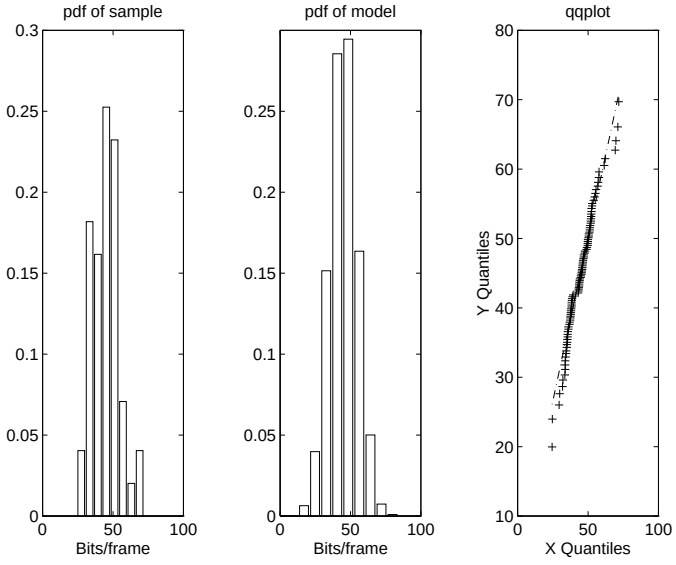


Figure 17: Probability Density Functions of Sample and Modeled Data

x	Sample Data	Modeled Data	% of Error
Mean	44.95	44.90	0.11%
Std	9.692	9.704	0.12%
3rd Moment	$1.038 * 10^5$	$1.032 * 10^5$	0.55%
4th Moment	$5.318 * 10^6$	$5.234 * 10^6$	1.57%

Table 1: Statistics of Refreshes for Sample and Modeled Data

an autoregressive process, eqn. 2, with the expected correlation and a probability distribution similar to the first one in Figure 17.

$$r(n) = a * r(n-1) + G \tag{2}$$

where n is frame sequence number, r is refresh bit rate, a is correlation coefficient, and G is Gaussian noise. We started by considering a first order autoregressive model, because the density plot shows only one distribution. In spite that the simulated results yield a good correlation coefficient, the remaining statistics do not produce a good match as the second approach does. Our second method treats the data as if it is coming from a Gaussian distribution with the mean, std, etc. as stated in Table 1: $mean = 44.95$ and $std = 9.692$. Data points are then randomly generated from such a Gaussian distribution: $G(\mu, \sigma) = G(44.95, 9.692)$.

The associated statistics of our modeling data are presented in Table 1. The percentage of error suggests that it is possible to utilize such a Gaussian model to represent the refreshes with a fair statistical match. The modeling probability distribution and its qqplot against the encoded samples are illustrated in Figure 17. The linear qqplot further confirms that a reasonably good match between the sample and the simulated refreshes can be achieved by using the proposed Gaussian model. Although our video sequence contains a reasonable number of frames, after removing the non-refresh frames, the resulting number of frames (i.e. refresh frames) are dramatically reduced. This reduction in sample numbers could explain why the statistical properties of the model are not as accurate as we would like them to be, since they are simulated based on those measurements of the original samples. With an increase in the amount of input data, the statistics measurements of the samples, especially of refreshes, are expected to be more accurate and can possibly lead to better simulation results: the means and std's closer matched and the "peakness" as reflected in the discrepancy of the 4th moment improved.

5.2 Modeling of the $W1$ Subimage

Recall that $W1$ is the last horizontally and vertically low-passed subimage in our decomposition process and contains most of the signal energy. Also, other low-

passed subimages at a different pyramid level exhibit strong correlation with this top one, either directly or indirectly, as discussed with Figure 16 on cross correlations. We therefore propose to first model this subimage and use its model as a base to predict the behavior of other subimages. A distribution and correlation coefficients of the average pixel rates of the $W1$ subimage, without refreshes, is depicted in Figure 18. Similar to our previous discussion on the total bit rates, due to the encoding schemes implemented in our wavelet-based codec, it is not surprising to observe three partially overlapped densities for $W1$. In addition, we can also expect that the correlation between two consecutive frames are high, since this subimage preserves most of the characteristics of the composite image.

A Markov-modulated renewal process is proposed to model this particular set of encoded video data. Each density function in Figure 18 is associated with a Markov state and the transition from one state to another is modulated by a Markov process. The statistical characteristics of the data are state dependent. That is, data at each state possesses different statistical properties from those of another state. Our model starts at one particular state, simulates video traffic with correspond-

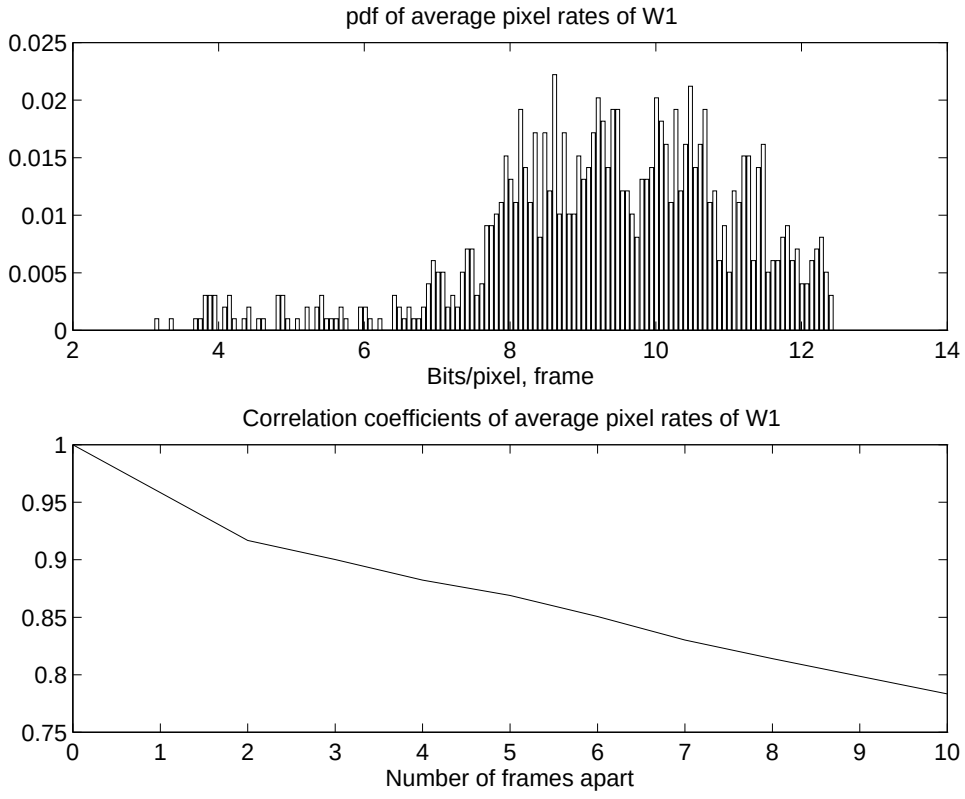


Figure 18: Probability Density Function and Correlation Coefficients of $W1$

	Class 1	Class 2	Class 3	Total
mean	5.022	8.589	10.86	9.364
Std	0.985	0.745	0.703	1.728
a	0.881	0.842	0.884	0.958

Table 2: Statistics of $W1$ with 3 Classes

ing characteristics, moves on to another state based on the Markov transitional probability.

The reasons such a model is considered are as follow. From our previous discussion, we noted that there are three encoding algorithms incorporated and the outputs can be subsequently categorized into three major groups: low, medium, and high motion classes. Since scene changes rarely occur, we usually do not anticipate large changes in bit rate magnitudes. Therefore, the transition from one class (i.e. state) to another is mostly predictable, i.e. mostly transferring from one state to itself or its adjacent ones. The transition thus relies on the current state, rather than any of the previous ones. Therefore, the transition can be described as memoryless and possibly modeled by a first order Markov process. Depending on the content of the video sequence, the encoded bit rates can remain in one state for any period of duration. The length of stay does not appear related to a previous visit at the same state. The *time* of which each visit to a state lasts then appears as a random variable with a memoryless quality. As a result, we propose to model the duration of a stay by a geometric process. In our early sections, we noted that the bit rates between consecutive frames are highly correlated in each class. Their distribution approximates some composite Gaussian densities. Therefore, a first-order autoregressive model is considered in an attempt to model data with such attributes.

To obtain the Markov transitional probability matrix, we first estimate the thresholds which separate our data into three classes. The statistical qualities of the data and each subset are then measured, as included in Table 2, where a is the correlation coefficient. The transitional probability from state to state is next obtained and presented in the transitional probability matrix, Π.

$$\Pi = \begin{bmatrix} 0.900 & 0.100 & 0 \\ 0.012 & 0.918 & 0 \\ 0 & 0.081 & 0.929 \end{bmatrix} \tag{3}$$

To simulate the data in each state, we begin by using a first-order autoregressive model:

$$y_i(n) = a_i * y_i(n-1) + G(\mu_i, \sigma_i) \tag{4}$$

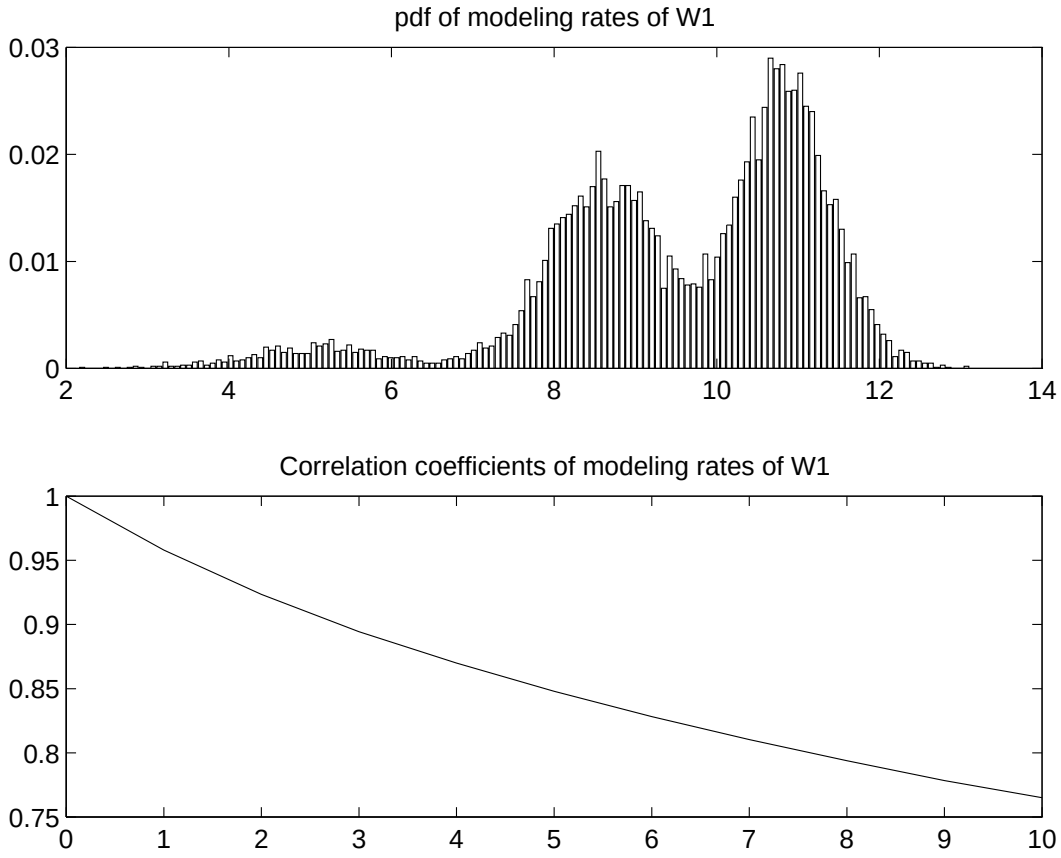


Figure 19: Probability Density Function and Correlation Coefficients of Modeled $W1$

where n is the frame sequence number, i is the state, $y_i(n)$ is the bit rate of state i at frame n, a_i is the measured correlation coefficient between consecutive frames for state i, and G is a Gaussian distribution with a mean of μ_i and std of σ_i. Numerical results of our simulation will be presented in the next section.

By using the statistics measurements from our sample data, $10,000$ data points are generated. The probability distribution of our modeling results is then plotted in Figure 19 We can see that the modeling densities are similar to the sample ones, while the correlation coefficients are closely modeled as the sample coefficients plotted in Figure 18. In addition, we classify the generated data with the same threshold values as those for the sample data. The resulting statistics are shown in Table 3 As the figures show, our Markov-modulated renewal process can generate data with a reasonably good match in statistical attributes. The percentage of errors appears small for most of the moments. The discrepancy reflected in the 4th moment mostly is due to the slightly smaller value of modeling standard deviation then

	mean	std	a	3rd moment	4th moment
sample	9.364	1.728	0.958	9.009	9.139
model	9.584	1.684	0.960	9.564	9.825
% of error	2.3%	2.5%	0.2 %	6.1%	7.5%

Table 3: Statistics of Model $W1$

the sample standard deviation. To improve such differences, one can try another estimation method to obtain a better set of measurements of the attributes, such means, std's for the three classes of our sample data.

5.3 Modeling of the Composite Video

Due to the highly correlated behavior between frames within each subimage and among different subimages, we propose a model to take advantage of this characteristics. The bit rates of the remaining subimages are than derived from those of the one located in an upper level in the pyramid structure.

$$X_j(n) = c_{ij} * X_i(n) + G(\mu_{ij}, \sigma_{ij}) \tag{5}$$

where $X_i(n)$ is the bit rates of Wi at frame n, c_ij is the correlation between subimages i and j, and G is a Gaussian noise with mean μ_ij and std σ_ij, which can be derived from the measurements of our sample.

Our composite model consists of the refresh model r, the Markov-modulated renewal process model for $W1$, Y, and the estimation model for the remaining subimages X. Let Y represent the total average pixel rate at frame n, we then have the following analytical model for the composite bit rate:

$$\mathbf{Y} = \mathbf{r} + \mathbf{Y} + \mathbf{X} \tag{6}$$

where Y, r, Y, and X are all column vectors.

Figure 20 shows the distributions of our modeling and sample data. We can observe from the linear line in the qqplot that we have obtained two similar composite distributions. However, the modeling data appears to be closer towards the higher end than our sample. This could be contributed by the discrepancies of the modeling results of $W1$ subimage, propagated through estimation of other subimages. After the summation of ten subimages, the supposedly minor differences can be enlarged. Another possible explanation is that, besides $W1$, $W5$ and $W8$ also contain relatively high signal energy, and merely an estimation approach may not be sufficient enough to obtain the best results for these two subimages. In order to improve

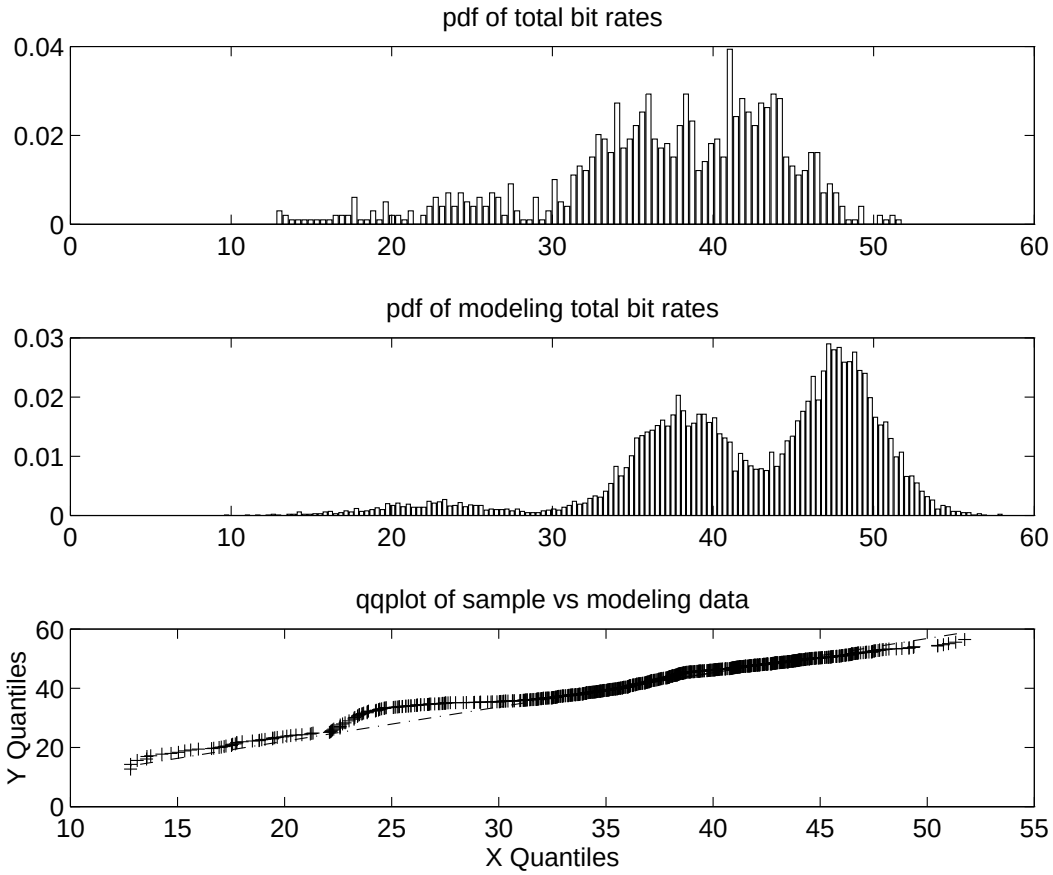


Figure 20: Probability Density Function of Sample and Modeling Data and their Qqplot

the accuracy of the modeling of high-energy subimages, it may become necessary to model the first three subimages, followed by the estimation of the remaining low-energy subimages.

6 Conclusion

In this paper, we examine the statistical characteristics of the encoded data from a wavelet-based video codec. The SNR analysis provides clues as to which subimages have a strong impact on the performance in terms of SNR and prioritizes subimages for possible band dropping. By analyzing the statistical characteristics of the overall image and individual subimages, we propose an analytical composite model to describe the traffic behavior of such encoded data. Our model can become useful

while performing rate control study at an end node, since each subimage has its own entry in the final bit rate matrix. Our results, presented in terms of density functions, various moment statistics, and qqplots, suggest that this analytical tool can estimate such data to some reasonable extend. We note that by applying a better estimation algorithm to the sample data while gathering measurements can most likely improve the modeling results. We also find that by SNR analysis, we can prioritize the signal level of each subimage and determine the modeling criteria for the relatively high-energy subimages such that the resulting modeling can also be improved.

References

[1] S. Brofferio and F. Rocca, "Interframe redundancy reduction of video signals g enerated by translating objects," *IEEE Trans. on Communications*, vol. 25, pp. 448 – 455, April 1977.

[2] D. L. Gall, "MPEG: A Video Compression Standard for Multimedia Applications," *Communications of the ACM*, vol. 34, pp. 46 – 58, April 1991.

[3] S. Zafar, *Motion Estimation and Encoding Algorithms for Hierarchical Representation of Digital Video.* PhD thesis, George Mason University, 1994.

[4] S. Zafar, Y.-Q. Zhang, and B. Jabbari, "Multiscale Video Representation Using Multi-Resolution Motion Compensation and Wavelet Decomposition," *IEEE Transaction on Circuit and Systems for Video Technology*, 1993.