Improved High-Definition Video by Encoding at an Intermediate Resolution

Andrew Segall [a], Michael Elad [b*], Peyman Milanfar [c*],

Richard Webb [a] and Chad Fogg [a],

[a] Pixonics Inc., Palo Alto, CA 94306.
[b] The Computer-Science Department - Technion, Haifa 32000 Israel
[b] The Elect. Engineering Department – University of California - Santa Cruz, Santa Cruz, CA 95064.

ABSTRACT

In this paper, we consider the compression of high-definition video sequences for bandwidth sensitive applications. We show that down-sampling the image sequence prior to encoding and then up-sampling the decoded frames increases compression efficiency. This is particularly true at lower bit-rates, as direct encoding of the high-definition sequence requires a large number of blocks to be signaled. We survey previous work that combines a resolution change and compression mechanism. We then illustrate the success of our proposed approach through simulations. Both MPEG-2 and H.264 scenarios are considered. Given the benefits of the approach, we also interpret the results within the context of traditional spatial scalability.

Keywords: High-Definition Video, Video Compression, MPEG-2, H.264, Intermediate-resolution, Spatial scalability.

1. INTRODUCTION

High definition video is becoming increasingly available in the marketplace. With a spatial resolution of up to 1920x1080 pixels per frame, high-resolution sequences contain six times the pixel count of current standard definition video. High-resolution sequences also support frame rates of (up to) 60 frames per second, interlaced and progressive encoding and a 16:9 wide-screen aspect ratio. The resulting image sequence is visually superior to legacy standard definition systems, and it is well suited for evolving plasma, LCD and DLP display technologies. It also approaches the quality of film distributed for general movie viewing.

While high definition video provides an enhanced viewing experience, it requires a significant amount of bandwidth for transmission and storage. Uncompressed, it contains approximately one Giga-bit of data per second of content. Compressing the frames is therefore a requirement for delivery. For example, high definition broadcasts in the United States employ the ATSC standard operating at 19.4 Mbits/second. Pre-recorded high-definition content is also available with the D-VHS tape format that supports a constant video bit-rate of 25Mbits/second. The resulting compression ratios are 52:1 and 40:1, respectively.

The bit-rates of current high-definition systems ensure fidelity in representing the original sequence. However, they preclude widespread availability of high-definition programming. Specifically, satellite and Internet based distribution systems are poorly suited to deliver a number of high-rate channels. Also, video-on-demand applications must absorb a significant increase in storage costs. Finally, pre-recorded DVD-9 stores less than an hour of high-definition video.

With a number of applications enabled by low-rate coding, it is natural to investigate the improved compression of high-definition sequences. In this paper, we consider the impact of down-sampling the image frame prior to compression. We introduce compression through filtering, and exploit two important characteristics of the high-definition scenario. First, the majority of high-resolution image frames do not contain information throughout the high frequency band. Second, the block signaling overhead dominates at lower bit-rates. Here, we concentrate on proof-of-

---

*
Both M. Elad and P. Milanfar are also consultants to Pixonics, Inc.

concept experiments and show that encoding the video sequence using an intermediate resolution improves rate-distortion performance. We also identify and summarize related areas of research. Please note that the theoretical justification of the approach is reserved for future work, perhaps in line with the approach described in [1].

The rest of this paper is organized as follows. In the next section, we define the considered system. In section three, we identify previous work that combines down-sampling and encoding. This work comes from the separate fields of low-rate, still-image coding and estimation problems such as super-resolution for compressed video. In the fourth section, we consider the benefits of coding a high-definition sequence at an intermediate resolution. The discussion is within the context of several experiments, which assess coding efficiency and image quality. Finally, we summarize the paper and extrapolate the results to the problem of spatially scalable coding in section 5.

2. SYSTEM MODEL

The system considered in this paper is defined as follows. Let $f(x,y,t)$ denote the high-definition video sequence with discrete spatial coordinates x,y and temporal coordinate t. For notational convenience, we re-state this in matrix-vector form as $\mathbf{f}$, where $\mathbf{f}$ is a $MNP{\times}1$ vector that contains the lexicographically ordered high-definition frame. The spatial dimensions of the frames are $M{\times}N$, and the sequence contains P frames. We require the high-resolution frame to be filtered and compressed for transmission. This is expressed as

$$\mathbf{d} = Q[\mathbf{Af}],$$

where $\mathbf{d}$ is a $KLP{\times}1$ vector that contains the decoded bit-stream, $Q[\mathbf{x}]$ represents the lossy compression of the vector $\mathbf{x}$, and $\mathbf{A}$ is the $KLP{\times}MNP$ matrix that defines the filtering and sub-sampling procedure. Here, we assume down-sampling prior to compression so that $K{<}M$ and $L{<}N$. However, the temporal resolution is unchanged. At the decoder, a high-resolution sequence is generated by

$$\mathbf{g} = \mathbf{Bd},$$

where $\mathbf{g}$ is the $MNP{\times}1$ estimate of the high-resolution video, and $\mathbf{B}$ is the $MNP{\times}KLP$ up-sampling matrix. Combining the encoding and decoding procedures, the entire process is stated as

$$\mathbf{g} = \mathbf{Bd} = \mathbf{B}Q[\mathbf{Af}],$$

where $\mathbf{A}$ and $\mathbf{B}$ exploit the temporal and spatial relationships within $\mathbf{f}$ for compression. Note that the resulting $\mathbf{g}$ is not equal to $\mathbf{f}$, as both the operator $Q[]$ is lossy and there is unavoidable loss due to the design of $\mathbf{A}$ and $\mathbf{B}$. Finally, we mention that ignoring temporal relationships and processing each frame independently can simplify filtering. In this case, the system model becomes

$$\mathbf{g}_k = \mathbf{B}_k\mathbf{d}_k = \mathbf{B}_k Q[\mathbf{A}_k\mathbf{f}_k],$$

where $\mathbf{g}_k$ and $\mathbf{f}_k$ are $MN{\times}1$ vectors that respectively contain the estimated and original high-resolution frames at time k, $\mathbf{B}_k$ is the $MN{\times}KL$ up-sampling matrix for time k, and $\mathbf{B}_k$ is the $KL{\times}MN$ down-sampling matrix for time k.

3. SYSTEM DESIGN

Selection of the up-sampling and down-sampling matrices is an important component of the intermediate resolution approach. In the next section, we consider these matrices within the context of high-definition sequences. This is done experimentally, and it allows us to assess the usefulness of an intermediate resolution for high-definition transmission. In this section though, we discuss work related to construction of the $\mathbf{A}$ and $\mathbf{B}$ matrices. For a non-proprietary solution, the matrix $\mathbf{B}$ is computed at the decoder and not explicitly provided in the bit-stream. This is an open-loop problem, and it is similar in goal to previous estimation work. As a second area of work, construction of the up-sampling and down-

sampling procedures is accomplished at the encoder. The up-sampling matrix is then transmitted explicitly in the bit-stream, which results in a closed loop system.

3.1 Open Loop Design

Several estimation problems naturally lead to a definition for the up-sampling procedure. For these problems, the original signal is unknown during the construction of **B**. For example, the problem of super-resolution for compressed video assumes that a number of image frames are filtered and down-sampled prior to compression. The super-resolution algorithm then tries to estimate the original high-resolution frames from the decoded observations. This effectively defines the up-sampling procedure.

The primary differentiator between super-resolution methods is the underlying model for the compression system [4]. In one class of approaches, the quantized transform coefficients describe the compression process. The quantization values are available in the bit-stream, and they constrain the solution space **Bd**. In words, the resulting **B** approximates the inverse of **A**, subject to the constraint that $Q[\mathbf{ABd}] = Q[\mathbf{Af}]$. (The result $Q[\mathbf{Af}]$ defines the quantized transform coefficients in the bit-stream.) An explicit form for **B** is rarely found in practice. Instead, an iterative solution for **g** is employed that incorporates a projection onto convex sets (POCS) algorithm such as

$$
P_i[\mathbf{g}] = \begin{cases} \mathbf{A}^{-1}\mathbf{T}^{-1}\left(\mathbf{TAg} + \mathbf{Td} - \dfrac{\mathbf{q}}{2}\right)_i, & (\mathbf{TAg} - \mathbf{Td})_i \leq -\dfrac{\mathbf{q}_i}{2} \\[2ex] \mathbf{A}^{-1}\mathbf{T}^{-1}\left(\mathbf{TAg} - \mathbf{Td} + \dfrac{\mathbf{q}}{2}\right)_i, & (\mathbf{TAg} - \mathbf{Td})_i \geq \dfrac{\mathbf{q}_i}{2} \\[2ex] (\mathbf{g})_i, & Otherwise \end{cases}
$$

where **T** is the matrix defining the transform operator utilized for compression, **q** is the vector describing the width of the quantization interval, and i is the scalar referencing the ith entry in the vector. More sophisticated algorithms are necessary to addresses the general case of a non-invertible **A**.

A second class of super-resolution methods relies on a Gaussian noise model to describe the encoder. This is motivated by the use of linear de-correlating transforms and scalar quantizers within the compression system. The inverse transform of the noise is then a linear sum of independent noise processes, which tends towards a Gaussian distribution irrespective of the noise distribution in the transform domain. Leveraging this noise model, the design of **B** must invert (undo) **A**, subject to the constraint that **ABd** is "close" to **d**. This is more precisely stated by

$$
\underset{\mathbf{B}}{\text{ArgMax}} \ \exp\left\{-\frac{1}{2}[\mathbf{d} - \mathbf{ABd}]^T \mathbf{K}_Q^{-1}[\mathbf{d} - \mathbf{ABd}]\right\},
$$

where $\mathbf{K}_Q$ is the covariance matrix estimated from the compressed bit-stream.

Other estimation problems are also relevant to the design of **B**. For example, the field of post-processing considers the construction of the matrix **A** [6]. The matrix **A** is constrained to be the identity matrix though, so that there is no change in resolution. Disregarding the fact that **B** is *MN×MN* (and not *MN×KL*), post-processing methods still model the compression system and address its degradations. For example, the block-based structure of an encoder often leads to blocking errors. These structured errors are bothersome and addressed by a **B** that filters across the block boundaries. Other errors such as ringing, mosquito and corona artifacts are also addressed with a suitable choice of **B**.

De-blurring algorithms for compressed video suggest additional designs for **B**. As in post-processing, the matrix is constrained to be *MN×MN*. However, the matrix **A** is no longer the identity – it now defines a filtering procedure. The goal of de-bluring is then to estimate the original image from a blurred and compressed observation. This is similar to the super-resolution problem. Its application to traditional block based coding algorithms is considered in [5], where

both spatial and transform model for the noise are considered. Work that de-blurs a frame after wavelet coding is presented in [7].

Finally, we mention work that considers down-sampling along the temporal dimension. In this case, the matrix **A** is *MNQ×MNP* with Q<P. The up-sampling matrix **B** then maps the lower frame-rate sequence to the higher rate *P* [3].

3.2 Closed Loop Design

The previous design approach for **B** assumed a non-proprietary bit-stream, where the up-sampling matrix is not signaled. When a proprietary solution is acceptable, the up-sampling matrix can be designed at the encoder. This provides a closed-loop that incorporates the original image frame into the procedure; it also allows for optimizing the scale factors *M*/*K* and *N*/*L*. As an example of a closed loop system, the up-sampling procedure can be defined as

$$\underset{\mathbf{B}}{\operatorname{ArgMax}} \left\| \mathbf{B}Q[\mathbf{Af}] - \mathbf{f} \right\|_2^2,$$

where **A**, *Q*[] and **f** are all known. In fact, one can optimize with respect to both **A** and **B** jointly [9]. While complicated in general, this method is tractable under simplifying assumptions such as structured matrices **A** and **B** representing linear space invariant filters merged with rate-conversion. Then the unknowns defining these matrices are the filters coefficients, and those could be found by the VARPRO method [2].

4. SIMULATIONS

Assessing the benefits of an intermediate resolution for high-definition coding is an important contribution of this paper. In considering the value of the methodology, we process several high-definition sequences at several rates and resolutions. Results are reported here for the "Rolling Tomatoes" and "Man in Car" sequences that are part of the JVT test suite. Frames in both sequences contain 1920x1080 pixels, which are stored in progressive format. The frame rate is defined as 24 frames per second, with the "Tomatoes" and "Car" sequences containing 222 and 334 frames, respectively.

We are interested in the performance of both MPEG-2 and H.264 based coding systems. For the simulations, we use the TMPEG MPEG-2 [8] and VSS H.264 encoders [10] (Demonstration versions were available from both vendors website at the time of this writing). Both encoders are representative of the underlying coding technology. Specifically, the MPEG-2 encoder is quite mature. It supports two-pass variable bit-rate modes and most profiles and levels. On the other hand, the H.264 encoder is relatively new. It currently supports the baseline profile; we selected a constant quality approach for rate control.

Simulations utilize three frame sizes for the intermediate resolution. Specifically, we down-sample the image sequences to 720x360, 960x540, and 1440x720 pixels, which are denoted as 360p, 540p and 720p, respectively. Each of the low-resolution sequences is then compressed. For the MPEG-2 experiments, the encoder operates at main-profile/high-level (MP@HL), and the target of the rate-control varies between 0.25-12Mbps. For the H.264 experiments, the encoder operates in baseline mode and the quantizer value varies between 15-41. The decoded frames are then up-sampled to 1920x1080 pixels using a linear filter, designed with a 5-by-5 windowing function.

Results from the "Tomatoes" experiment appear in Figure 1. In the figure, the peak signal-to-noise ratio (PSNR) for the intermediate resolutions is plotted as a function of bit-rate. The direct encoding of the high-definition source also appears (denoted as 1080p). Comparing the plots, we see significant compressions gains at the lower rates. For example, the H.264 encoder processes the intermediate 720p frame at 1.7Mbps and produces a PSNR of 38.8dB. Directly encoding the 1080p frame requires 2.5Mbps to achieve the same level of quality. Thus, the intermediate resolution provides a 30% reduction in bit-rate. (Visual examples from the sequence are shown in Figures 2 and 3.) This reduction actually increases as the bit-rate decreases – the bit-savings is 60% for an image quality of 37.6dB.

Inspecting the MPEG-2 simulations for the "Tomatoes" sequences follow a similar trend. The intermediate 720p frame provides a 54% reduction in rate for a quality of 37.1dB. For the lower quality frame of 36.0dB, the 360p intermediate resolution provides a bit-savings of approximately 87%.

Results from the 'Car' sequence appear in Figure 4. This sequence differs from the 'Tomatoes' sequence in that it contains more motion. The motion further differentiates the intermediate resolution and direct coding methods. (This derives from the smaller motion vectors present in the lower resolution frames.) Inspecting the H.264 simulations, we see that the intermediate 720p frame provides 39.2dB of quality at 1.2Mbps. Compressing the original high-definition frame yields a PSNR of 39.0dB and a rate of 1.8Mbps. Thus, the intermediate resolution provides over 33% savings in the bit-rate. (A visual example appears in Figure 5.) Further efficiencies appear at the lower rates, where the rate is reduced by approximately 45% for the 360p data points.

Compressing the 'Car' sequence with the MPEG-2 encoder also shows the advantage of an intermediate resolution. For example, the 720p frame size leads to a PSNR of 37.7dB at 3.8Mbps. This same level of quality requires 8.2Mbps when the sequence is encoded at native resolution. The decrease in rate is over 54%. If a lower quality frame is acceptable, the 360p intermediate resolution provides a PSNR of 36.1dB at 1.25Mbps. Equivalent quality with a direct encoding of the sequence requires 8.2Mbps. The resulting bit-rate savings is approximately 85%.

In the above experiments, we designed the up-sampling filter without explicit knowledge of the original image sequence. This is an instance of an open-loop design approach. For comparison, we re-processed the sequences with an up-sampling operator designed by the encoder and described in Section 3.2. A comparison of the open-loop and closed-loop solutions appears in Figure 6. In the figure, we plot the difference between the two experiments as a function of image quality. Here, image quality is equal to the PSNR of the open-loop up-sampled frame. Thus, the scatter plots show the improvement of the re-designed up-sampling operator as a function of compressed image quality. Interpreting the MPEG-2 and H.264 results, we see that the benefit of a closed-loop approach varies as a function of decoded image fidelity. This is true in both MPEG-2 and H.264 experiments. Interestingly, the two simulations differ in relating the closed-loop design and the intermediate frame size. For H.264 systems, we observe that smaller intermediate resolutions benefit more from the closed-loop approach. For MPEG-2 type systems, the opposite is true; larger frame sizes show more benefit.

5. CONCLUSION AND FURTHER WORK

Utilizing an intermediate resolution for high-definition video coding is both practical and beneficial. In this paper, we have shown bit-savings for both MPEG-2 and H.264 type systems. These efficiencies are most pronounced at the lower rates and increase as the rate decreases. For the highest rate reduction, we observe image qualities that may not be suitable for distribution. However, bit-savings of around 30% are observed for the H.264 system with acceptable image quality. Gains of over 50% occurred with the MPEG-2 system.

The coding gains of the intermediate resolution motivate comments on the related field of spatially scalable video coding. In a spatially scalable system, video is encoded at a lower resolution. This low-resolution data is then up-sampled and refined with a transmitted residual. The motivations for such as system are varied; however, spatially scalable methods are often assumed inferior to coding at the display resolution. (Signaling overheads usually motivate the statement.) For high rate scenarios, this may be true. However, results in this paper show that it is not true at the lower rates. Decreasing the resolution of the base-layer is inherently more efficient at these rates. Thus, a spatially scalable system that does not spend bits for residual would outperform the low-rate, single resolution approach. We expect that judiciously transmitting the residual would lead to further improvements.

REFERENCES

[1]   A. Bruckstein, M. Elad and R. Kimmel, "Down Scaling for Better Transform Compression", *IEEE Trans. on Image Processing*, Vol. 12, No. 9, pp. 1132-44, Sept. 2003.

[2]   G.H. Golub and V. Pereyra, The Differentiation of Pseudo-Inverses and Non-linear Least Squares Problems Whose Variables Separate, *SIAM Journal on Numerical Analysis*, Vol. 10, No. 2 (1973), pp. 413-432.

[3]   Mark A. Robertson and Robert L. Stevenson, "Temporal Resolution Enhancement in Compressed Video Sequences," *EURASIP Journal on Applied Signal Processing: Special Issue on Nonlinear Signal Processing*, pp.230-238, Dec. 2001.

[4]   C. Andrew Segall, Rafael Molina and Aggelos K. Katsaggelos, "High-Resolution Images from Low-Resolution Compressed Video," *IEEE Signal Processing Magazine*, pp.37-48, May 2003.

[5]   C. Andrew Segall and Aggelos K. Katsaggelos, "Approaches for the Restoration of Compressed Video," *Proceedings of the IEEE International Conf. on Image Processing*, Barcelona, Spain, Sept. 14-17, 2003.

[6]   M.-Y. Shen and C.C. Jay Kuo, "Review of Postprocessing Techniques for Compression Artifact Removal," *Journal of Visual Communication and Image Representation*, pp. 2-14, March 1998.

[7]   C. Parisot, M. Antonini, M. Barlaud, S. Tramini, C. Latry and C. Lamber-Nebout, "Optimization of the Joint Coding/Decoding Structure," *Proceedings of the IEEE International Conference on Image Processing*, Thessaloniki, Greece, Oct. 7-10, 2001.

[8]   Pegasys Inc., TMPGEnc, Version 2.521, 2003. http://www.tmpgenc.net/

[9]   Y. Tsaig, M. Elad, G.H. Golub and P. Milanfar, "Optimal Framework for Low Bit-rate Block Coders," *Proceedings of the IEEE International Conf. on Image Processing*, Barcelona, Spain, Sept. 14-17, 2003.

[10]   Vsofts, VSS H.264 Codec, Beta 3 Preview, 2003. http://www.vsofts.com/
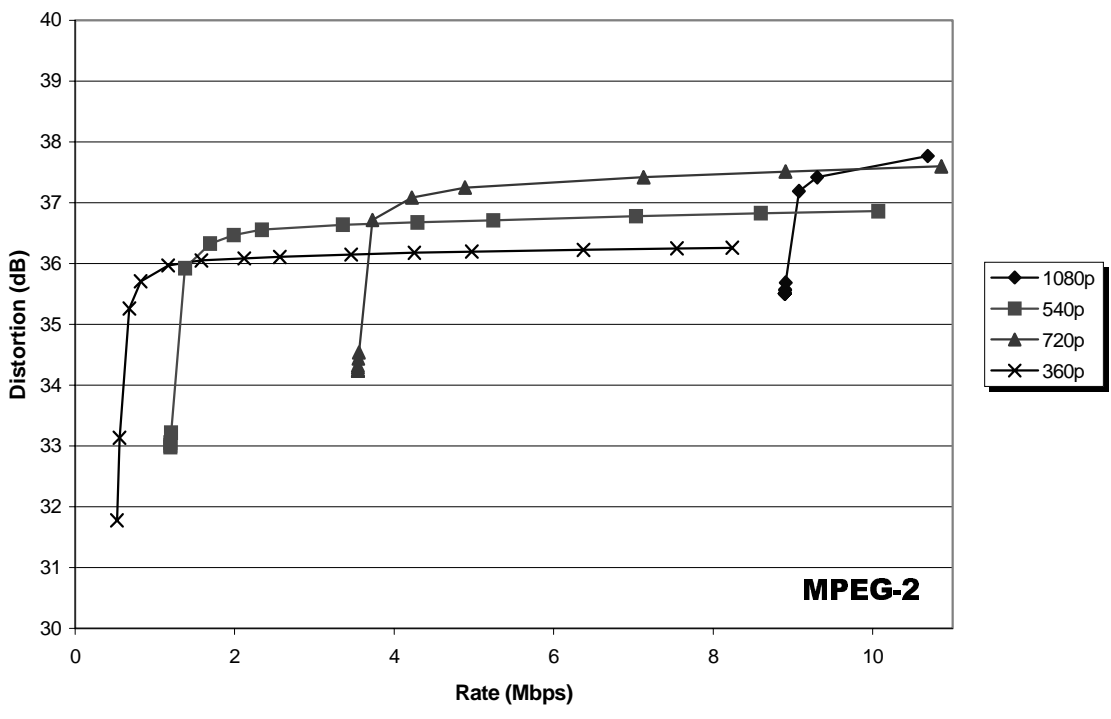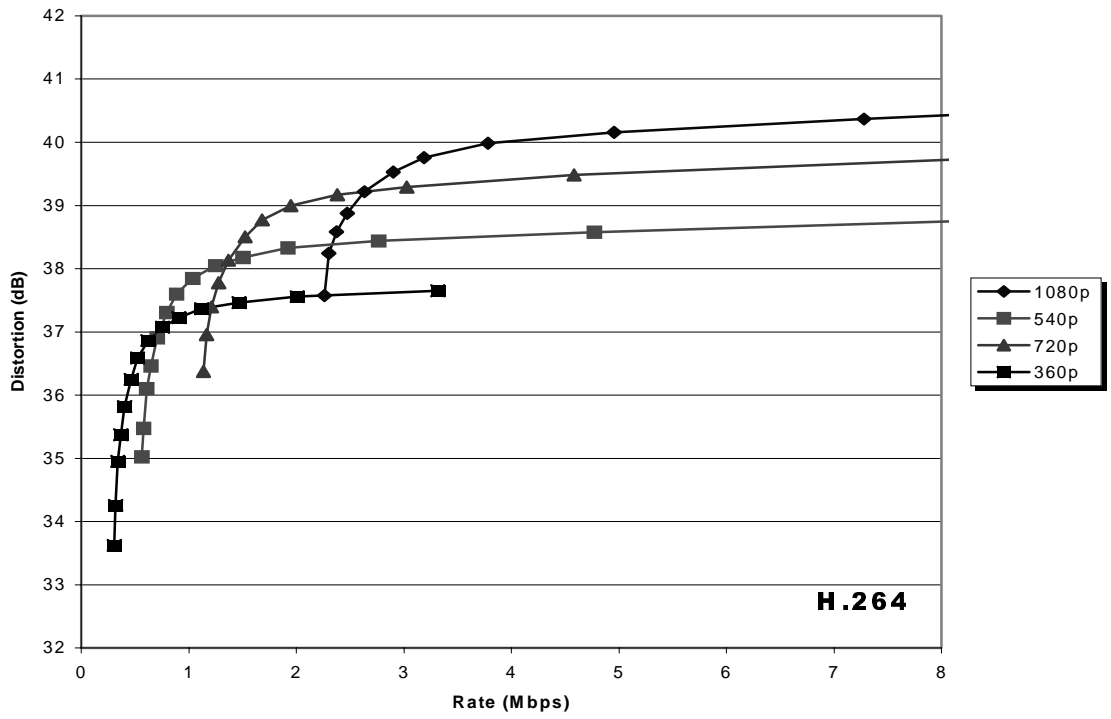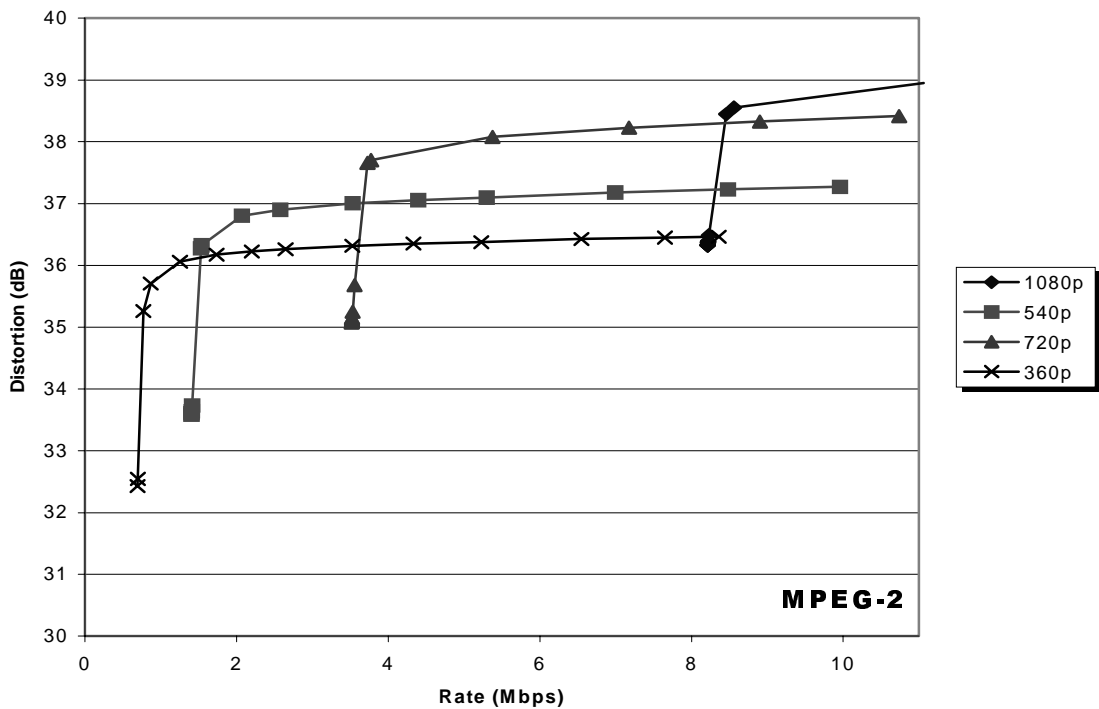
Figure 1. Rate distortion curves for the "Rolling Tomatoes" sequence . The frames are encoded at four resolutions with an H.264 and MPEG-2 encoder, respectively. The decoded frames are then up-sampled with a linear filter. The intermediate resolution approach leads to significant bit savings at the lower rates.

(a)

(b)

(c)

Figure 2. Visual example from the intermediate resolution experiments: (a) original frame from the "Rolling Tomatoes" sequence, (b) frame coded directly at 1080p, and (c) frame coded at 720p and up-sampled. The frames are compressed with an H.264 encoder and cropped for display. Inspection of the results shows similar image quality. The average bit-rate for the 1080p sequence is 2.5Mbps, while the average bit-rate for the proposed method is 1.7Mbps. The bandwidth savings is approximately 30%.

(a)

(b)

Figure 3. Expanded example from the intermediate resolution experiments: (a) frame coded directly at 1080p and (b) frame coded at 720p and up-sampled. The frames are alternative views of the images in Figure 2, and the peak signal-to-noise ratio for both sequences is 38.8dB. Notice that while both frames contain blocking artifacts, the errors are more pronounced in the upper-left portion of (a). The proposed method attenuates these structured errors during coding and up-sampling.
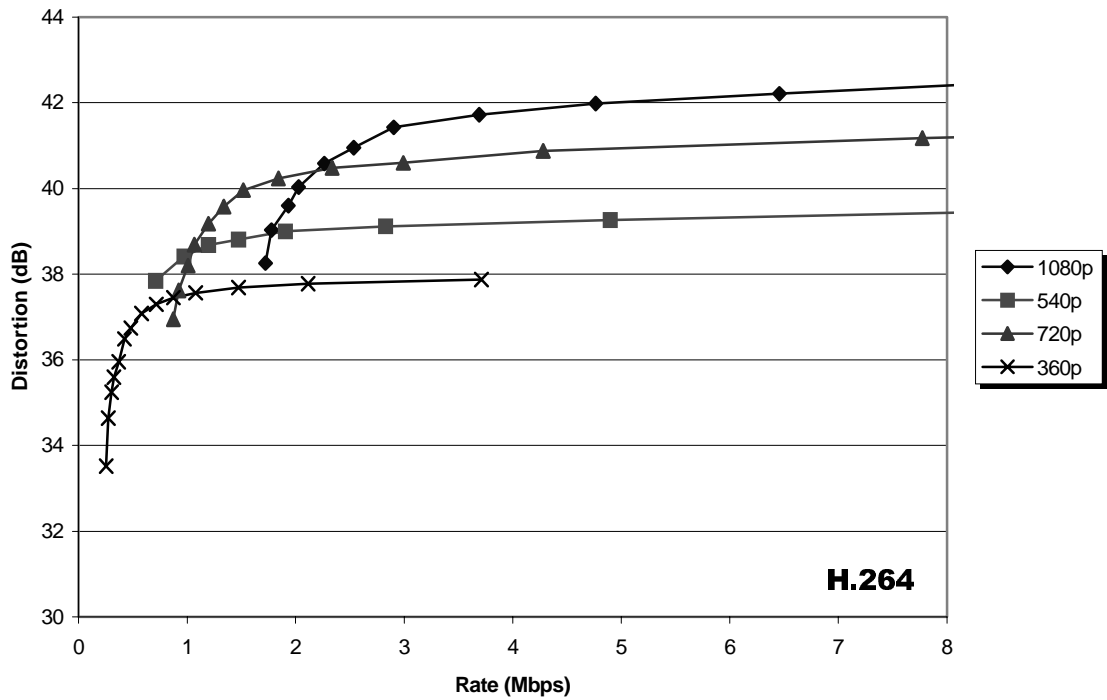
Figure 4. Rate distortion curves for the "Man in Car" sequence. The frames are encoded at four resolutions with an H.264 and MPEG-2 encoder, respectively. The decoded frames are then up-sampled with a linear filter. The intermediate resolutions provide higher quality frames at the lower bit-rates.

(a)

(b)

(c)

Figure 5. Visual example from the intermediate resolution experiments: (a) original frame from the "Man in Car" sequence, (b) frame coded directly at 1080p, and (c) frame coded at 720p and up-sampled. The frames are compressed with an H.264 encoder and cropped for display. The image quality of the frames is similar. However, the average bit-rate for the 1080p encoding is 1.8Mbps, while the average bit-rate for the proposed method is 1.2Mbps. The bandwidth savings is approximately 33%.
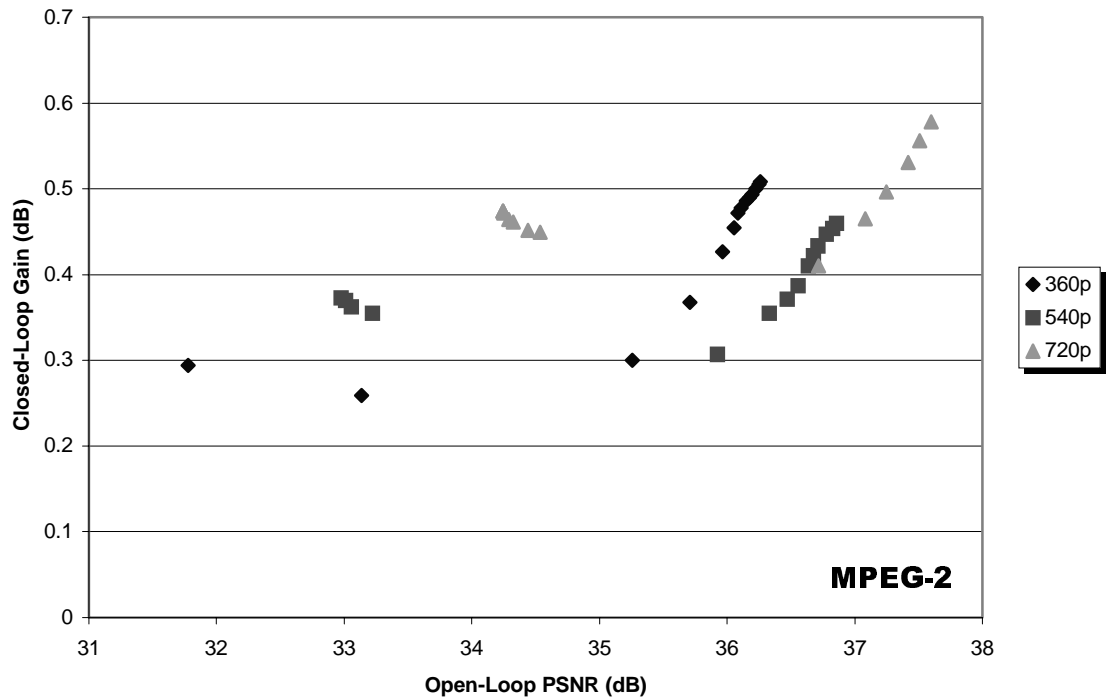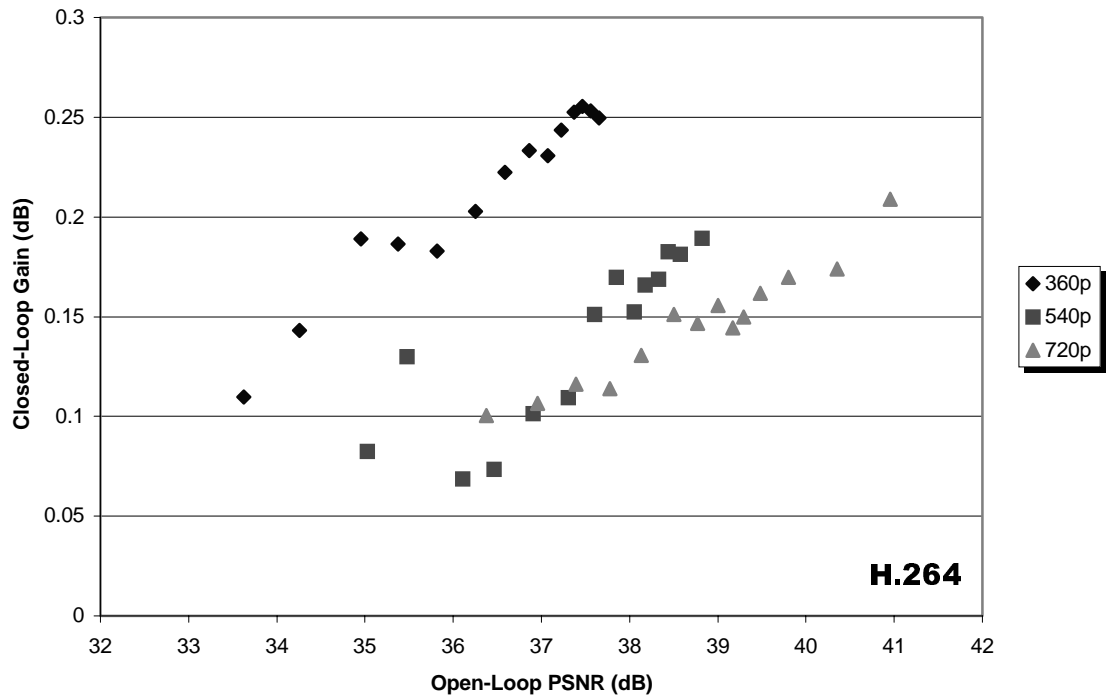
Figure 6. Comparison of an open-loop and closed-loop approach. The 'Rolling Tomatoes' sequen ce is re -processed with an up-sampling operator designed at the encoder. The level of improvement is then plotted as a function of the open-loop image quality. Gains are more significant for high quality frames.