RATE-DISTORTION BASED VIDEO CODING
WITH ADAPTIVE MEAN-REMOVED VECTOR QUANTIZATION

Raouf Hamzaoui[1], Dietmar Saupe[2], Marcel Wagner[1]

[1]Universität Freiburg, Institut für Informatik, Am Flughafen 17, 79110 Freiburg, Germany
[2]Universität Leipzig, Institut für Informatik, Augustusplatz 10–11, 04109 Leipzig, Germany
[1]hamzaoui,wagner@informatik.uni-freiburg.de
[2]saupe@informatik.uni-leipzig.de

ABSTRACT

In this paper we improve the rate-distortion performance of a previously proposed video coder based on frame replenishment and adaptive mean-removed vector quantization. This is realized by determining for each block of a given frame the optimal encoding mode in the rate-distortion sense. The algorithm is a new contribution to very low bit rate video coding with adaptive vector quantization suitable for videophone applications. Experimental results comparing the two coders for several test sequences at different bit rates are provided.

1. INTRODUCTION

To ensure acceptable image quality at very low bit rates, most of the current video coding schemes use motion compensation. In [1] we have shown that an alternative is possible by combining frame replenishment with adaptive mean-removed vector quantization. Moreover, our coder offered the following features:

- It can be implemented in software on computers of PC class.
- The bit rate is scalable within a range from 10 kb/s to 100 kb/s.
- The bit rate per frame is constant.

In this paper we propose to keep these nice features and enhance the rate-distortion performance of the coder by optimally assigning the quantizers to the source vectors of a given frame. This yields a new approach to rate-distortion based adaptive VQ (AVQ). Most of the existing AVQ algorithms rely on either a distortion or a rate criterion [2, 3]. Previous attempts that consider both rate and distortion include the schemes of Lightstone *et al.* [4, 5] and Fowler *et al.* [6]. However, unlike [4, 5], our algorithm does not include a training procedure for the adaptive codebook. Instead, the codebook is updated by replacing ineffective codebook vectors with source vectors. Also our technique incorporates frame replenishment (not all blocks are updated) which is not considered in both [4, 5] and [6].

The rest of the paper is organized as follows. Section 2 describes the algorithm of [1]. In Section 3 we propose our new encoding strategy, where the encoding mode of a source vector is chosen such as to ensure optimal rate-distortion performance at the frame level. In Section 4 we provide experimental results.

2. THE ORIGINAL ALGORITHM

We begin by briefly reviewing the video coder proposed in [1]. The target bit rate and the frame refresh rate together define the bit budget and the time available for the coding of a single frame. A new frame is read and compared block-wise with the frame buffer, which contains the previously encoded frame. Blocks are of fixed size (e.g., 4×4). The blocks are sorted with respect to decreasing mean square difference. From this priority list, blocks are encoded until the allocated bit budget or the available time is used up. The remaining blocks are unchanged with reference to those of the same position in the frame buffer. The block encoding is done with mean-removed VQ (MRVQ) [7]. The block means are uniformly quantized, and the shape codebook $\{\mathbf{s}_1, \ldots, \mathbf{s}_{n_s}\}$ is adaptive. For each coded frame a target quality t is computed. If the mean square error given by the MRVQ encoding of the block is larger than t, then the quantized mean and the residual (shape) block are transmitted. The shape block is encoded using DPCM of quantized prediction errors. The new quantized shape block is included in the codebook while another one is removed. The shape codebook is maintained in a form of a priority queue where the codebook blocks are sorted according to a biased frequency count, which is incremented by one each time the corresponding vector is referenced by the encoder. When a new codebook vector is inserted into an already full codebook, the vector with the smallest count is deleted. The frequency count of this new codebook vector is initialized to

$f\left(\frac{n_s}{2}\right) + 1$, where $f\left(\frac{n_s}{2}\right)$ is the frequency of the codevector whose index is $\frac{n_s}{2}$.[1] The quantized mean values, the shape codevector indices and the quantized prediction errors for the components of the blocks that are inserted in the codebook have variable length codewords defined by Huffman tables. The bitmap indicating the locations of the updated blocks is run length encoded with Huffman coding of the runs. For each updated block a bit is sent to specify the VQ/codebook update alternative.

The above encoding technique is restricted to the Y component of the image. For the U and V color components the same block size is used but only the means of the blocks are encoded.

3. RATE-DISTORTION BASED ALGORITHM

One limitation of the scheme described in Section 2 is that only distortion criteria are employed in order to decide how a source vector is to be coded. Thus, it may happen for example that a mean-removed block is appended to the shape codebook even though the cost of the side information is not compensated by the reduction in distortion. However, if we assume that the codebook vectors together with their frequency counts are kept unchanged during the encoding of a given frame, then it is possible for this frame to guarantee an optimal encoding in the rate-distortion sense. This encoding can be found by the bit allocation algorithm of Westerink *et al.* [8], which can be applied as follows. As in Section 2 for a given frame with a shape codebook $\{\mathbf{s}_1, \ldots, \mathbf{s}_{n_s}\}$ a source block $\mathbf{x}_t$ can be coded in three different modes:

- **Mode 0: No update**: The block remains unchanged. This gives a rate cost $r_t^0 = 0$ and a distortion d_t^0.
- **Mode 1: MRVQ**: The block is encoded with MRVQ. Note that since the shape codevectors have variable length codewords, each shape codevector $\mathbf{s}_k$, $k = 1, \ldots, n_s$ provides a distinct rate cost r_t^k and a distinct distortion d_t^k.
- **Mode 2: Codebook update**: Both the quantized mean and the residual block (the block with its quantized mean removed) are transmitted to the decoder, which gives a rate cost $r_t^{n_s+1}$ and a distortion $d_t^{n_s+1}$.

The encoding of the frame proceeds as follows:
Initialization: Start by assuming that each block is not updated. Thus each block is encoded with the smallest rate $r_t^0 = 0$. Set $i = 0$.
Maximization procedure: Calculate for each block $\mathbf{x}_t$ the parameter

$$\lambda_t^k = \frac{r_t^k - r_t^i}{d_t^k - d_t^i}$$

for all k such that $d_t^k < d_t^i$. Note that $\lambda_t^k < 0$. Then, find $k = k_t$ for which λ_t^k is maximal. Next find the block $\mathbf{x}_t$ for

which $\lambda_t^{k_t}$ is maximal.
Stopping criterion: Assuming that this block is encoded with the corresponding distortion-rate couple $(d_t^{k_t}, r_t^{k_t})$, compute the total rate of the frame. If this total rate is larger than the bit budget or if the time budget is spent, then stop the encoding of the frame. Otherwise, set $i = k_t$ *only for the previously selected block* and repeat the maximization procedure. Thus λ_t^k need be recomputed only for the selected block.

After the modes of all blocks of the current frame have been determined, the frequency counts of the shape codebook vectors referenced by the encoder are updated. Then, the sorting of the shape codevectors is changed accordingly. Next, all shape vectors of blocks encoded with mode 2 are appended to the codebook and their frequency is set according to the rule in Section 2. Finally, the shape codevectors with the smallest frequency counts are removed.

The side information consists of the bits for the frame bitmap specifying if a block is coded with mode 0 or not. The bit discriminating between mode 1 and mode 2 is included in the rate cost r_t^k, $k \geq 1$.

4. RESULTS AND COMPARISON

We provide in this section encoding results at various bit rates for the codec in [1] and the new one based on the algorithm of Westerink *et al.* [8]. The encoding parameters are:

- Block size: 4×4.
- Shape codebook size: 512.
- Number of quantization levels for mean values: 64.
- Number of quantization levels for DPCM prediction error for block components: 64.

The image sequences are in 4:2:0 QCIF color format (176×144 pixels). For both algorithms no time constraint (time budget) is imposed for the coding of a frame. We point out, however, that the coder in [1] is able to encode at a frame rate of 8.3 Hz. This is not shared by the rate-distortion coder due to the complexity of the bit allocation algorithm. In Figures 1 and 4 the peak-signal-to-noise ratio (PSNR) for the Y component is shown as a function of the frame number for the Salesman sequence. Figures 2 and 5 show similar results for the Miss America sequence.

The results show that the rate-distortion based coder (denoted "rate-distortion" in the figures) can significantly improve the performance of the distortion based coder at all bit rates. Note in particular the dramatic edge in the first frames.

Figures 3 and 6 illustrate the mode in which the blocks are coded in a given frame for the rate-distortion based coder. Blocks that are not updated (mode 0) are not represented here. It is remarkable that mode 2 is rarely used.

---

[1]This initialization rule is more efficient than the one given in the original paper.
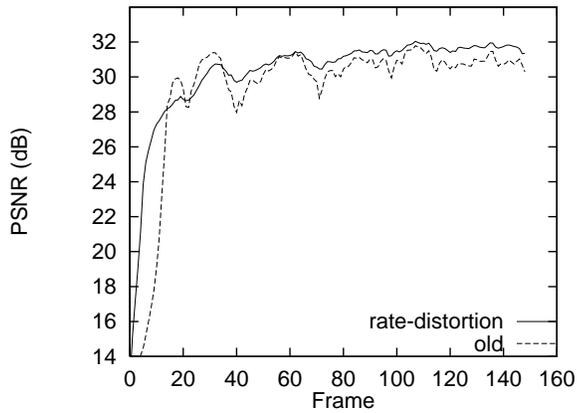
Figure 1: Frame course of PSNR for Y component of the Salesman sequence at 0.137 bpp.
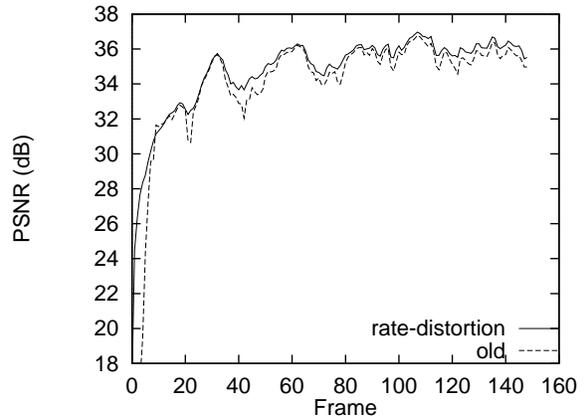
Figure 4: Frame course of PSNR for Y component of the Salesman sequence at 0.312 bpp.
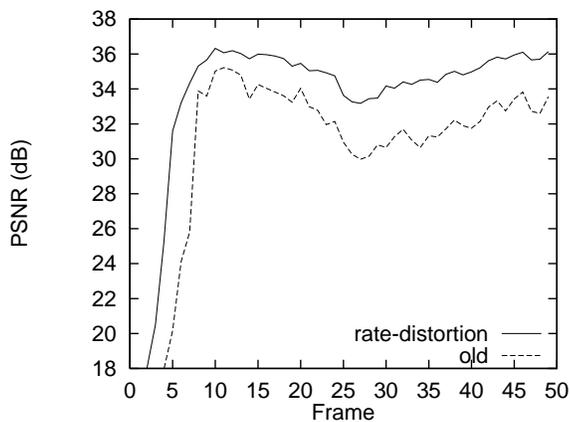
Figure 2: Frame course of PSNR for Y component of the Miss America sequence at 0.137 bpp.
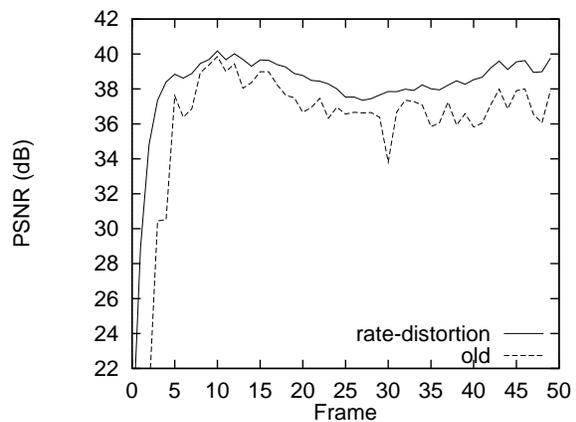
Figure 5: Frame course of PSNR for Y component of the Miss America sequence at 0.312 bpp.
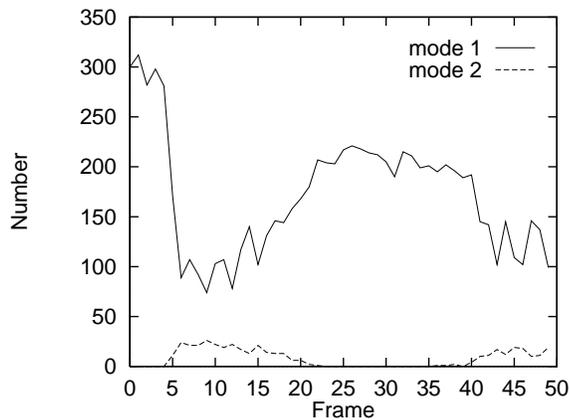
Figure 3: Statistics on the mode in which source vectors are encoded in the Miss America sequence coded at 0.137 bpp.
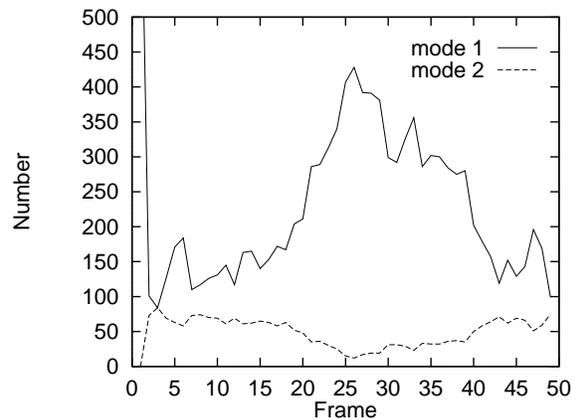
Figure 6: Statistics on the mode in which source vectors are encoded in the Miss America sequence coded at 0.312 bpp.

In Figure 7 we show how the performance of the rate-distortion coder can be improved by using adaptive arithmetic coding (the probabilities are initialized with a uniform distribution and updated after the encoding of each frame) instead of Huffman coding for the entropy coding of the shape codevector indices, the quantized mean values, the quantized prediction errors of the components of the transmitted vectors and the symbols discriminating between mode 1 and mode 2. Here the rate costs r_t^k, $k \geq 1$ are estimated using the frequency of the symbols in the previously coded frames. For example, for mode 1 we have for $k \in \{1, \ldots, n_s\}$

$$r_t^k = -\log_2(p_1) - \log_2(p_t) - \log_2(p_k)$$

where p_1 is the probability that a block is encoded with mode 1, p_t is the probability of the quantized mean of the block and p_k is the probability of the index of the codebook block $\mathbf{s}_k$.
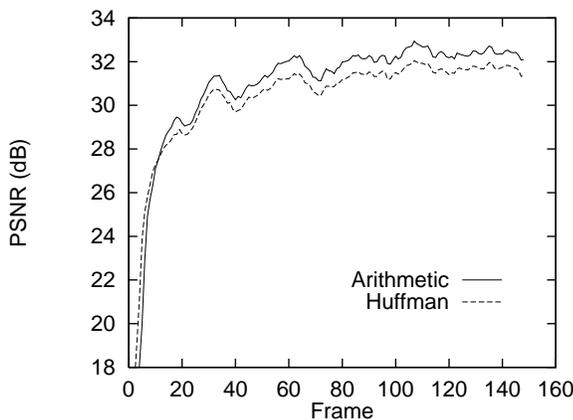
Figure 7: Comparison between Huffman coding and adaptive arithmetic coding for the rate-distortion coder for the Salesman sequence at 0.137 bpp.

However, this improvement is penalized by an increase in time complexity. We have investigated in this respect the effect of reducing the number of shape codevectors and come to the conclusion that using only 256 shape codevectors can approximately halve the encoding time of a frame without any significant loss in quality.

5. CONCLUSION AND FUTURE WORK

We have presented a new adaptive vector quantization video coder for very low bit rate codings with an optimal bit allocation strategy at the frame level. Improvements in the rate-distortion performance are achieved over a similar distortion constrained codec. The price is an increase in encoding complexity, which is not a severe drawback when we take into consideration the enlarged CPU power of future PC generations.

Currently we are investigating the translation of the encoding in the wavelet domain in order to enhance the efficiency of VQ-encoding and to reduce the blocking artifacts which typically occur in block-based coders in the spatial domain.

6. REFERENCES

[1] Saupe, D., Butz, B., *Real-time very low bit rate video coding with adaptive mean-removed vector quantization,* in: *Proc. ICIP-97 IEEE International Conference on Image Processing,* Santa Barbara, California, Oct. 1997.

[2] Gersho, A., Yano, M., *Adaptive vector quantization by progressive codevector replacement,* in: *Proc. ICASSP-85 IEEE International Conference on Acoustics, Speech and Signal Processing,* Vol. 1, pp. 133–136, 1985.

[3] Wang, X., Shende, S., Sayood, K., *Online compression of video sequences using adaptive VQ codebooks*, in: *Proc. DCC'94 Data Compression Conference,* J. A. Storer, M. Cohn (eds.), IEEE Computer Society Press, Snowbird, Utah, March, 1994.

[4] Lightstone, M., Mitra, S., *Adaptive vector quantization for image coding in an entropy-constrained framework*, in: *Proc. ICIP-94 IEEE International Conference on Image Processing,* Austin, Texas, Nov. 1994, Vol. 1, pp. 618–622.

[5] Lightstone, M., Mitra, S. K., *Image-adaptive vector quantization in an entropy-constrained framework,* IEEE Trans. Image Proc. 6,3 (1997) 441–450.

[6] Fowler, J., Ahalt, S. C., *Adaptive vector quantization using generalized threshold replenishment,* in: *Proc. DCC'97 Data Compression Conference,* J. A. Storer, M. Cohn (eds.), IEEE Computer Society Press, Snowbird, Utah, March 1997.

[7] Gersho, A., Gray, R. M., *Vector Quantization and Signal Compression,* Kluwer Academic Publishers, 1992.

[8] Westerink, P. H., Biemond, J., Boekee, D. E., *An optimal bit allocation algorithm for sub-band coding,* in: *Proc. ICASSP-88, IEEE International Conference on Acoustics, Speech and Signal Processing,* pp. 757–760, 1988.