VECTOR RESTORATION FOR VIDEO CODING

J. S. McVeigh[*], *S.-W. Wu*[**], *M. W. Siegel*[*] *and A. G. Jordan*[*]

[*]Carnegie Mellon University, Pittsburgh, PA 15213
[**]AT&T Bell Laboratories, Murray Hill, NJ 07974

ABSTRACT

We present a novel concept, vector restoration, for motion compensated predictive coding of video. Conventional predictive coders generate predictions of the input signal and encode the residual signal. Here, the predicted images are operated upon directly with the goal of restoring the original images at the decoder. The predicted images are processed in a piecewise manner using restoration functions chosen from a codebook. Compression is achieved by transmitting only an index of the chosen restoration function for each image region. We describe the manner in which the restoration functions are generated and chosen. Vector restoration is shown to outperform residual vector quantization and transform-based coders. This superiority is due to vector restoration's ability to adaptively exploit the non-trivial correlation between the predicted and residual images.

1. INTRODUCTION

Motion, or displacement, compensated predictive coders have become the *de facto* standard for the compression of video signals [1, 2]. These coders attempt to exploit the temporal redundancy of video signals by predicting a desired frame from reference frames offset in time. A residual image can then be formed by subtracting the predicted image from the original image. The residual signal often is coded by common intraframe coding techniques, such as vector quantization [3] or transform coding [4], to exploit spatial redundancy.

While displacement compensated predictive coding has been demonstrated to be an effective technique for many applications, conventional methods of residual encoding typically fail to recognize the non-trivial correlation between the residual signal and the predicted signal for practical images. This shortcoming is particularly troublesome in low bit rate video coding, where the number of bits available for coding the residual image is severely constrained.

In this paper we describe a new concept for motion compensated predictive coding that aims at making the best use of the predicted signal to reproduce the original images. Here, we view the decoding process as the operation of restoring the original image from the prediction, which is a degraded version of the original image. For each segment of the predicted image, the encoder calculates the best *restoration function* and transmits this restoration function, or an index for this restoration function to the decoder. We shall call this new class of predictive coding techniques *restoration based coding.*

Prior work most closely related to this class of techniques is "vector classified adaptive filtering" (VCAF), developed by Richardson [5]. VCAF was applied to the problem of image restoration, where only a degraded signal and a model of the degradation are available. The selection of the filter was based on the degraded vector and not the original vector, as in vector restoration based coding.

2. VECTOR RESTORATION

We begin our development of vector restoration by viewing predictive coding in terms of an operation performed on the predicted signal. We wish to find the optimal operation to be performed on the predicted signal to *restore* it to the original signal. To reduce the computational complexity, this operation will be applied to the predicted signal in a piecewise manner.

The main features of vector restoration (VR) based coding are depicted in Fig. 1. A predicted image is generated using any available motion compensated predictive coding technique [1, 2]. The input image and predicted image are sectioned into small subimages, and the pixel values of the subimages are listed in vector form. In general, the input and predicted vectors can be constructed from the images in any manner. For simplicity, in this paper we only consider input vectors constructed from fixed size square blocks and predicted vectors constructed from pixels in the predicted image with the same area and location as the input vector.

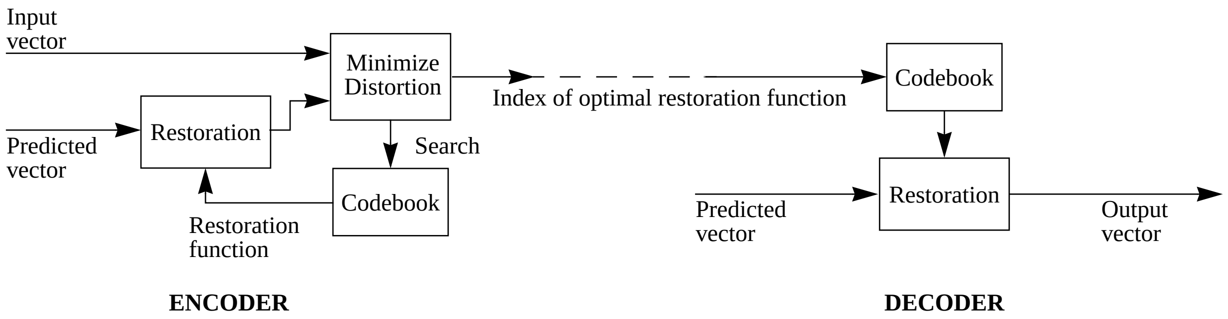


Figure 1: Vector restoration based coding.

A finite set of restoration functions are designed off-line and stored at both the encoder and decoder. We call the set of available restoration functions the *codebook* in analogy to the concept of codebooks in vector quantization. For each input vector, the restoration function that minimizes the distortion between the original and reconstructed vectors is selected from the codebook. The index of this chosen restoration function is transmitted to the decoder. At the decoder, the received index specifies the desired restoration function from the codebook. The output vector is then reconstructed by applying the selected restoration function to the predicted vector.

Let X and $\tilde{X}$ denote the n-dimensional input and corresponding predicted vectors, respectively. In general, the restoration functions can be arbitrary operations on the vector $\tilde{X}$ and any distortion measure can be used for the codebook search. However, for mathematical tractability, we use linear estimation with mean-squared error distortion measure in this paper. Hence, we can write the reconstruction, $\hat{X}$, of the original vector as,

$$\hat{X} = H\tilde{X} + D \qquad (1)$$

where H is a linear transform which can be represented by an $n \times n$ matrix, and D is an n-dimensional offset vector. For each input vector a restoration function, parameterized by (H_i, D_i), is retrieved from the codebook $\{(H_1, D_1), \ldots, (H_M, D_M)\}$ of M candidate restoration functions. The nearest-neighbor encoding rule can be applied to generate the index i such that,

$$\left\|X - (H_i\tilde{X} + D_i)\right\|^2 \le \left\|X - (H_j\tilde{X} + D_j)\right\|^2 \qquad (2)$$

for all $j \ne i$, $1 \le j \le M$.

An iterative algorithm can be applied to design a codebook of restoration functions from a set of representative training vectors, similar to the well-known Generalized Lloyd Algorithm for vector quantization (VQ) codebook design [3]. In each iteration, the training vectors are first partitioned by encoding with the current codebook. Each restoration function in the codebook then is updated by computing the pair, (H_i, D_i), that minimizes the partial mean-squared error,

$$\varepsilon = E\{\left\|X - \hat{X}\right\|^2\} \qquad (3)$$

for the i^{th} partition. The solution of this minimization lies within Wiener filter theory [6], which yields,

$$H_i = C_{X\tilde{X}} C^{-1}_{\tilde{X}\tilde{X}} \quad \text{and} \quad D_i = E\{X\} - H_i E\{\tilde{X}\} \qquad (4)$$

where $C_{X\tilde{X}}$, $C_{\tilde{X}\tilde{X}}$, and $E\{X\}$ respectively denote the conditional cross- and auto-covariance matrices, and the conditional expectation, with respect to the i^{th} partition. These values can be numerically estimated by computing the ensemble averages over the partition. Since the distortion is monotonically decreasing with each iteration, a locally optimal codebook is guaranteed by this design algorithm.

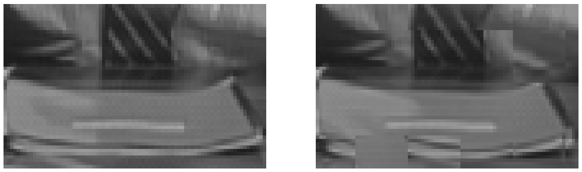
From (1) it is clear that residual VQ is a special, and generally sub-optimal case, of VR, where H equals the identity matrix. In residual VQ, encoding is performed on the residual vector given by $R = X - \tilde{X}$. Therefore, at any given bit rate, vector restoration based coding outperforms, or in the worst case is equivalent to residual vector quantization. It can be easily shown that for each partition the improvement in mean-squared error of VR over VQ is,

$$\varepsilon_{VQ} - \varepsilon_{VR} = \frac{trace\{C_{R\tilde{X}} C^{-1}_{\tilde{X}\tilde{X}} C^{T}_{R\tilde{X}}\}}{n} \qquad (5)$$

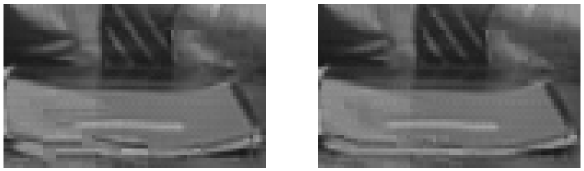
Since $C_{\tilde{X}\tilde{X}}$ is a positive semi-definite matrix, we are guaranteed that $\varepsilon_{VQ} \ge \varepsilon_{VR}$. Equality holds if, and only if, R and $\tilde{X}$ are uncorrelated (i.e., $C_{R\tilde{X}} \equiv \mathbf{0}$).

To reduce the storage requirement of the VR codebook and to reduce the computational complexity of the nearest

Original *Salesman* Frame



Original Predicted



Residual VQ VR

Figure 2: Original and coding results for subimage of *Salesman* frame. The lower edge of the notebook exhibits much less blockiness in the VR encoded frame than the residual VQ result.

neighbor search, we may constrain the two-dimensional transform H to be separable. In this case, the size of the VR codebook is on the order of the size of a VQ codebook. Separable transforms also reduce the number of multiplications and additions that must be performed. The mathematics of separable transforms can be derived similarly to the general case that we have shown.

3. EXPERIMENTAL RESULTS

Codebooks were generated by modifying VQ codebook splitting techniques [7]. The vectors used to train the codebooks and the test vectors used to examine the performance of the techniques were independent. The predicted images were generated using block-based techniques with 16×16 blocks and half-pixel displacement estimation accuracy.

As an initial experiment to evaluate the potential performance achievable with the VR coding technique, we used three image frames from the *Salesman* sequence as the training set. An original frame of this sequence is shown in Fig. 2. Four-bit codebooks were generated for both VR and residual VQ using 4×4 regions from the training frames. The results for an 80×50 subimage of a test frame are shown in Fig. 2. For this frame VR produced a gain of 1.6 dB over VQ.

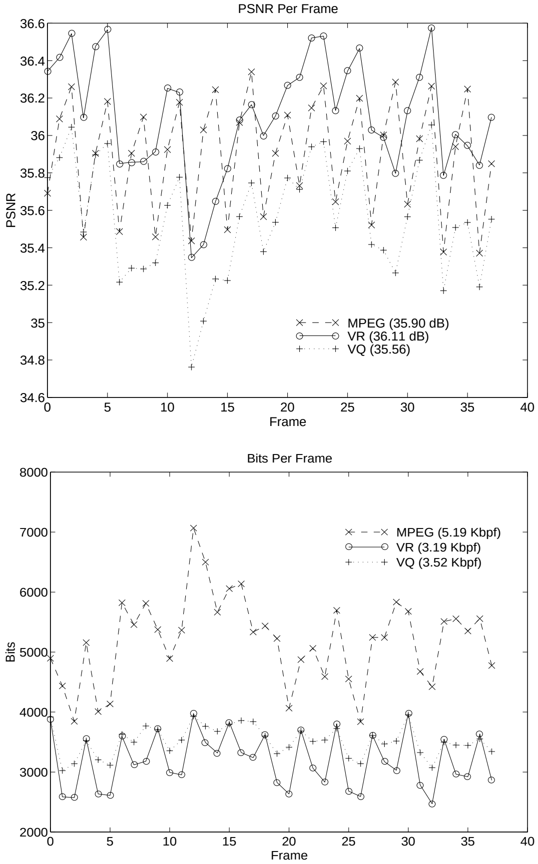


Figure 3: P-frame performance comparison for *Grandmom* sequence. Overall bit rate for MPEG encoded sequence was 0.6 Mbps, assuming 30 frames per second.

To evaluate the performance of vector restoration in a more practical situation, we compared VR using separable transforms with residual VQ and with an IP-only MPEG-1 [2] coder, which is comparable to the H.261 [1] video telephony standard. Six-bit codebooks were generated using 102598 training vectors from regions of size 8×8 obtained from two video-conferencing sequences (*Mom* and *Mom and Daughter*). A third sequence (*Grandmom*) was used as the test sequence. To ensure a fair comparison of the encoding techniques, we generated nearly identical predicted frames by intraframe coding one of every four frames. A fixed quantizer step size was used for the MPEG coder. A block was coded in the VR and VQ techniques if the prediction error was greater than an empirical threshold. The peak signal-to-noise ratio (PSNR) and per frame bit count for interframe coded blocks using the three techniques are shown in Fig. 3 (average values are given in parentheses). The results for I-frames have been omitted

Original

IP-only MPEG-1 encoded

VR encoded using 40% fewer bits than MPEG

Figure 4: Original and encoded twelfth P-frame of *Grandmom* sequence.

since these frames were identical for all three coders. The bit counts exclude the displacement vector and header information, which was basically equivalent for all three techniques. The original and encoded twelfth P-frame of the *Grandmom* sequence are shown in Fig. 4.

4. CONCLUSION

We have presented a new concept for predictive video coding, called restoration based coding. Vector restoration outperforms residual vector quantization both theoretically and experimentally. The gain achieved by VR is a result of its ability to adaptively exploit the non-trivial correlation between the predicted and residual images.

The *Salesman* sequence results provide an indication of the upper bound in the performance of VR with an unconstrained codebook when the test images conform to the training statistics. The more elaborate comparison using the *Grandmom* sequence provides encouraging results for a practical coding scheme. Vector restoration achieved a consistently higher PSNR than vector quantization along with a slightly lower bit count. On average, VR produced a gain over VQ of approximately 0.55 dB per frame. Vector restoration also produced a small average PSNR gain over the MPEG simulation while using approximately 40% fewer bits.

Our initial study of VR coding focused on constant rate coding with exhaustive codebook search. Future research will include the use of structured VR codebooks, variable bit-rate VR techniques, such as tree-structured coding [8-9], and adaptive restoration functions.

5. REFERENCES

[1] CCITT Study Group XV - Report R95, Recommendation H.261, “Video Codec for Audiovisual Services at p x 64 kbits,” May 1992.

[2] ISO/IEC/JTC1/SG29/WG11, ISO/IEC 11172-2, “Information Technology - Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbits/s - Part 2: Video,” May 1993.

[3] A. Gersho and R. Gray, Vector Quantization and Signal Compression, Kluwer Academic Publishers, 1991.

[4] R. J. Clarke, Transform Coding of Images, Academic Press, 1985.

[5] P. Richardson, “Image Restoration Using Vector Classified Adaptive Filtering,” *Proc. SPIE Inter. Conf. on Visual Communications and Image Processing*, vol. 2094, pp. 1581-1591, November 1993.

[6] N. Wiener, Extrapolation, Interpolation, and Smoothing of Stationary Time Series, MIT Press, 1949.

[7] Y. Linde, A. Buzo and R. M. Gray, “An Algorithm for Vector Quantization Design,” *IEEE Tran. on Communications*, vol. COM-28, no. 1, pp. 84-95, January 1980.

[8] P. A. Chou, T. Lookabaugh and R. M. Gray, “Optimal Pruning with Application to Tree-structured Source Coding and Modeling,” *IEEE Tran. on Information Theory*, vol. 35, no. 2, pp. 299-315, March 1989.

[9] T. Lookabaugh, E. A. Riskin, P. A. Chou and R. M. Gray, “Variable Rate Vector Quantization for Speech, Image, and Video Compression,” *IEEE Trans. on Communications*, vol. 41, no. 1, pp. 186-199, January 1993.