# NON-CAUSAL ENCODING OF PREDICTIVELY CODED SAMPLES

*Onur G. Guleryuz, Amir Said, and Sehoon Yea*

LG Electronics, Mobile Research Lab, San Jose, CA, USA

**ABSTRACT**

We study the INTRA-prediction operation employed in hybrid video coders where sample values to be compressed are predicted using previously-decoded context values and the prediction errors are transform coded. Given the context values this procedure can be argued to be rate-distortion optimal for Gaussian signals. Yet natural images and video contain many structures that do not readily fit into Gaussian signal assumptions. Targeting such structures we propose a technique that predicts each sample using the context values and *samples that are jointly transform coded* with the predicted sample. We show that this joint, non-causal encoding can be represented with a nonorthogonal transform whose form and parameters we derive. We augment the HEVC standard with our work and show significant compression improvements on images/video that contain directional structures.

***Index Terms***— Nonorthogonal Transforms, DPCM, INTRA Prediction, HEVC, AVC, VP9

## 1. INTRODUCTION

Recent hybrid video coders ([1, 2, 3]) have successfully utilized prediction in the encoding of anchor frames by spatially predicting samples to be compressed using previously-decoded samples (context values) and transform coding the prediction errors. This encoding allows the continued utilization of computationally attractive block transforms even on signals over which block transforms are significantly suboptimal. Signals with significant inter-block correlations, signals with edges and other directional singularities can be cited as examples [4, 5, 6]. The spatial prediction operation can thus be thought of as coarsely adapting to sophisticated random processes by generating prediction residuals that are more amenable to compression with simple transforms. Regardless, because the prediction operation is carried out using context values alone, the efficacy is still strongly tied to the underlying processes exhibiting Gaussian-like behaviour.

In order to make the discussion concrete consider the one-dimensional example where one is tasked with compressing the sequence $x_i,\ i = 1, \ldots, N$, using the context sample $x_0$. $x$ can contain, for example, a horizontal or oriented sequence of pixels from a block of image pixels which will undergo directional prediction using the context sample $x_0$ obtained from the boundary of a previously decoded block [1, 2, 3]. Assume that the identical context sample is available to the encoder-decoder pair. Denote the linear prediction of $x_i$ using $x_0$ with $\mathcal{P}_i(x_0)$ and define the residuals

$$r_i = x_i - \mathcal{P}_i(x_0). \tag{1}$$

One hence has the encoding chain,

$$r_i = x_i - \mathcal{P}_i(x_0) \Rightarrow \texttt{transform\_code}(r), \tag{2}$$

and the decoding chain,

$$\texttt{transform\_decode} \Rightarrow \hat{r} \Rightarrow \hat{x}_i = \mathcal{P}_i(x_0) + \hat{r}_i, \tag{3}$$

where $\hat{x}$ denotes the decoded sequence. It is well-known that if one picks the optimal linear predictor and utilizes the Karhunen-Loeve transform tuned to the second order statistics of $r$ then the above process is asymptotically optimal for the compression of Gaussian sequences [7]. Yet for many image/video structures such Gaussian modeling is also well-known not to be adequate. As we will see in this paper one can significantly improve on the above prediction recipe by utilizing better predictors that make use of *all* decoded information during the decoding process. While for accurately-modeled Gaussian signals this extra information is in effect useless, our results will clearly show its benefit on images/video that contain edges and directional structures.

The outline of the paper is as follows. Section 1.1 discusses the basic ideas of this work by means of the one-dimensional example. Committing to linear predictors, Section 1.2 discusses connections of the presented ideas with DPCM and derives an equivalent nonorthogonal transform. After briefly discussing codec construction in Section 1.3 we consider compression with nonorthogonal transforms and derive rate-distortion optimal quantization parameters in Section 2. Simulation results and implementation details are discussed in Section 3 followed by concluding remarks.

### 1.1. Basic Ideas

It is clear from (3) that after transform decoding the decoder has access to *all* of the residual samples. However, it only uses $x_0$ and $r_i$ when decoding the $i^{th}$ sample, $\hat{x}_i$. In particular, note that when decoding $\hat{x}_{i+1}$ the decoder has already reconstructed $\hat{x}_i$, which is typically a far better predictor of $\hat{x}_{i+1}$ compared to $x_0$. In this paper we design the decoding chain,

$$\texttt{transform\_decode} \Rightarrow \hat{r} \Rightarrow \hat{x}_i = \mathcal{P}'_i(x_0, \hat{r}_1, \ldots, \hat{r}_N) + \hat{r}_i. \tag{4}$$

Since the decoder has all of the transform-decoded residuals available it is clear that this chain and the augmented predictor $\mathcal{P}'$ are feasible. The corresponding encoding chain can be described as the selection of optimal coded transform coefficients which, when fed into the transform decoder in (4), result in $\hat{x}$ that has the minimum distortion at a given target bit-rate.

While our work can be generalized to nonlinear prediction functions we will keep the computationally simple, linear predictors in [1, 2] but accomplish prediction using the closest available samples rather than using $x_0$ everywhere. For the one dimensional example we hence construct,

$$\begin{aligned}\hat{x}_1 &= \mathcal{P}_1(x_0) + \hat{r}_1 = x_0 + \hat{r}_1,\\ \hat{x}_2 &= \mathcal{P}_2(\hat{x}_1) + \hat{r}_2 = x_0 + \hat{r}_1 + \hat{r}_2,\\ &\vdots\\ \hat{x}_N &= \mathcal{P}_N(\hat{x}_{N-1}) + \hat{r}_N = x_0 + \hat{r}_1 + \ldots + \hat{r}_N, \quad (5)\end{aligned}$$

where we have assumed that the prediction is linear with a prediction weight of unity as in [1, 2]. Observe that in this setting the prediction $\mathcal{P}_i(x_0)$ in (3) is simply replaced with $\mathcal{P}_i(\hat{x}_{i-1})$. Other weights and types of linear predictors are straightforward generalizations.

### 1.2. Relationship to DPCM and Equivalent Nonorthogonal Transforms

It is interesting to note that (5) resembles a first-order DPCM decoder that is operating with a prediction weight of unity. Note that while a DPCM system will encode the residuals *causally and independently* [8], the decoder of (5) corresponds to decoding of residuals that have been encoded *non-causally and jointly*. This is due to $\hat{r}$ being the output of the transform decoder shown in (4). *It can be said that the proposed system gains the prediction accuracy of a DPCM system while exploiting residual dependencies (and other DPCM R-D inefficiencies [8]) via transform coding.*

Observe also that (5) leads to the matrix equation

$$\hat{x} = \mathbf{F}\hat{r} + \mathbf{B}x_0, \quad (6)$$

where $\mathbf{F}$ is a $(N \times N)$ lower triangular prediction matrix with

$$\mathbf{F}_{i,j} = \begin{cases} 1, & i \geq j \\ 0, & otherwise, \end{cases} \quad (7)$$

and for this simple example $\mathbf{B}$ is a $(N \times 1)$ matrix with unit entries. Augmenting (6) to accommodate transform coding we arrive at

$$\hat{x} = \mathbf{FT}\hat{c} + \mathbf{B}x_0, \quad (8)$$

where $\mathbf{T}$ $(N \times N)$ is the transform used in compression (e.g., the block DCT/DST in HEVC [2]) and $\hat{c}$ are the de-quantized transform coefficients. Letting $\mathbf{G} = \mathbf{FT}$ it is hence clear that (8) corresponds to the transform coding of $\hat{x} - \mathbf{B}x_0$ with the nonorthogonal transform $\mathbf{G}$ via,

$$\hat{x} - \mathbf{B}x_0 = \mathbf{G}\hat{c}, \quad (9)$$

In a nutshell then, in this simple linear form, the proposed work is the transform compression of $x - \mathbf{B}x_0$ using the nonorthogonal transform $\mathbf{G}$.

### 1.3. Codec Construction

Using mode-based linear predictors it is clear that the proposed decoding chain can be incorporated within a baseline hybrid codec like HEVC by designing $\mathbf{F}$ and $\mathbf{B}$ matrices and deriving the equivalent nonorthogonal transform $\mathbf{G}$ for each prediction mode (Section 3). Observe that such a decoding chain will have only marginal complexity increase compared to the baseline since all it will do is predict using the closest samples rather than the boundary samples. The encoding chain is more complex however as it must pick optimal coefficients to transmit for the decoding chain. In the next section we discuss an iterative quantization algorithm which the encoder must carry out and derive rate-distortion optimal quantization parameters.

## 2. COMPRESSION WITH NONORTHOGONAL TRANSFORMS

Consider the random vector $x$ $(N \times 1)$. For notational convenience assume that the context prediction is absorbed within $x$. The vector $x$ is represented using the linear transform $\mathbf{G}$ $(N \times N)$, whose columns $g_i,\ i = 1, \ldots, N$ form the transform basis. Assume $\mathbf{G}$ is full rank but is otherwise general, i.e., $\mathbf{G}$ is not necessarily orthogonal and $g_i$ are not necessarily unit norm. We have

$$x = \mathbf{G}c, \quad (10)$$

where $c$ $(N \times 1)$ are the transform coefficients. The coefficients are scalar quantized to yield $\hat{c} = \mathcal{Q}(c)$ which are then entropy coded and transmitted to a decoder.

### 2.1. Quantization

The scalar quantization problem with respect to the nonorthogonal basis $\mathbf{G}$ where one aims to minimize the quantization distortion can be written as,

$$||x - \mathbf{G}\hat{c}||^2. \quad (11)$$

While our work can accommodate a variety of quantizers for compatibility with video coders [1, 2] we will assume

$$\hat{c} = \mathbf{\Lambda}\iota \quad (12)$$

where $\iota$ $(N \times 1)$ is a vector of integers and $\mathbf{\Lambda}$ is a diagonal matrix of quantizer step-sizes, i.e., $\mathbf{\Lambda}_{i,j} = \lambda_i \delta_{i,j}$ with $\lambda_i$ the $i^{th}$ step-size and $\delta_{i,j}$ is the Kronecker delta function. Equation (11) hence becomes

$$||x - \mathbf{G}\mathbf{\Lambda}\iota||^2, \quad (13)$$

which can be recognized as a lattice quantizer whose optimal solution in terms of $\iota$ requires solving an integer problem [7]. Many suboptimal techniques have been proposed for the solution of (11) (e.g., [9, 4]). In order to accommodate fast solutions we incorporate a method similar to [9, 4] where one iteratively solves scalar quantization problems concentrating on each coefficient in turn. Assume all coefficients except for the $i^{th}$ coefficient have been quantized. The error vector can be defined as

$$\epsilon_i = x - \sum_{\{k|1\leq k\leq N, k\neq i\}} g_k \hat{c}_k. \quad (14)$$

Observe that without the integer constraint the distortion is minimized by choosing the $i^{th}$ coefficient to be

$$c_i^* = \arg\min_d ||\epsilon_i - g_i d||^2 = g_i^T \epsilon_i/(g_i^T g_i). \quad (15)$$

For the uniform de-quantization process in (12), it is easy to see that the optimal quantized coefficient can be obtained as

$$\hat{c}_i = \lambda_i round(c_i^*/\lambda_i) = \mathcal{U}(\epsilon_i, g_i, \lambda_i), \quad (16)$$

assuming a nearest-neighbor encoding rule[1]. This leads to the quantization algorithm,

**Algorithm 1 (Quantization)**

for $n = 1, 2, \ldots$ //until convergence

*1.* $e^n = x - \mathbf{G}\hat{c}^{n-1}$ $(\hat{c}^0 = \hat{c}^{init})$ //current error

*2.* for $k = 1, \ldots, N$

*(a)* $\epsilon_k = e^n + g_k \hat{c}_k^{n-1}$ //error when $\hat{c}_k = 0$

*(b)* $\hat{c}_k{}^n = \mathcal{U}(\epsilon_k, g_k, \lambda_k)$ //optimal $\hat{c}_k$

*(c)* $e^n \rightarrow e^n + g_k(\hat{c}_k^{n-1} - \hat{c}_k{}^n)$. //updated error

Observe that any change induced by step 2c is guaranteed to reduce distortion. The algorithm is hence guaranteed to converge with $\hat{c}_k^{n-1} - \hat{c}_k{}^n \rightarrow 0$.

### 2.2. Optimal Quantizer Step-Sizes

Rate-Distortion optimal design of quantizer step-sizes is in general a difficult problem since tractable expressions for rate and distortion are codec dependent and hard to obtain. In this section we use high rate approximations in order to optimize the vector of step-sizes, $\lambda$, used in (12).

*Rate Constraint:* The transform coding recipe followed by successful image and video coders utilize scalar entropy coders. Thus the rate required to convey the quantized coefficients in $\hat{c}$ can be approximated as

$$\mathcal{R}(\hat{c}) = \sum_k H(\hat{c}_k), \quad (17)$$

where $H(.)$ denotes entropy. Since coefficient $\hat{c}_i$ is scalar quantized using the step-size $\lambda_i$, at high bit-rates one can invoke the approximation,

$$H(\hat{c}_i) \approx h(c_i) - \log(\lambda_i), \quad (18)$$

where $h(c_i)$ is the differential entropy of the continuously-valued coefficient [10]. Hence in order to meet a rate constraint one needs,

$$\sum_k \log(\lambda_k) \sim constant. \quad (19)$$

[1]Related non-nearest-neighbor encoding rules (e.g., ”offset”-based rules of AVC/HEVC reference encoders) can also be straightforwardly derived.

*Distortion Expression:* Had $\mathbf{G}$ been orthonormal, a straightforward approximation for average distortion in terms of $\lambda$ would have been $\mathcal{D}_{orth}(\lambda) = \sum_k \lambda_k^2/12$, which is obtained by assuming uniformly distributed quantization error [7]. With a nonorthogonal $\mathbf{G}$, signal domain and coefficient domain distortions are not the same and one cannot use this approximation. Assume all quantities are zero mean. The signal domain average distortion can be written as

$$\mathcal{D}(\lambda) = E[e^T e] = Tr(E[ee^T]), \quad (20)$$

where $E[.]$ denotes expectation and $Tr(.)$ is the trace of a matrix and $e = \mathbf{G}(c - \hat{c})$. We have,

$$\mathcal{D}(\lambda) = Tr(\mathbf{G}E[(c-\hat{c})(c-\hat{c})^T]\mathbf{G}^T) \quad (21)$$
$$= Tr(\mathbf{G}E[pp^T]\mathbf{G}^T) \quad (22)$$

where we have set $p = c - \hat{c}$ to denote the coefficient domain error. Assuming that coefficient domain error is decorrelated we have that $E[pp^T]$ $(N \times N)$ is diagonal with,

$$E[pp^T]_{i,j} = \begin{cases} \pi_i, & i = j \\ 0, & \text{otherwise.} \end{cases} \quad (23)$$

Then straightforward algebra yields,

$$\mathcal{D}(\lambda) = \sum_{k=1}^N \sum_{l=1}^N G_{k,l}^2 \pi_l. \quad (24)$$

It is important to note that since the quantization is carried out through the quantization algorithm, approximations of the form $\pi_l = \lambda_l^2/12$ are not valid. In order to relate $\pi$ to $\lambda$ let us concentrate on the rounding error induced by the quantization algorithm. At the point of convergence, using steps 2a and 2b in conjunction with Equation (12) we obtain,

$$\iota_k = \hat{c}_k/\lambda_k = round(\frac{g_k^T(e + g_k\hat{c}_k)}{(g_k^T g_k)\lambda_k})$$
$$= round(\frac{g_k^T e}{(g_k^T g_k)\lambda_k} + \iota_k), \quad (25)$$

which leads to the rounding error satisfying $|\frac{g_k^T e}{(g_k^T g_k)\lambda_k}| < 0.5$. Set $\omega_k = \frac{g_k^T e}{(g_k^T g_k)}$ and observe that if we assume that the rounding error is uniform we obtain,

$$E[(g_k^T e/(g_k^T g_k))^2] = E[\omega_k^2] \simeq \lambda_k^2/12. \quad (26)$$

Let $\tilde{\mathbf{G}}$ be the matrix with the $i^{th}$ column $\frac{g_i}{(g_i^T g_i)}$. We have

$$\omega = \tilde{\mathbf{G}}^T e = \tilde{\mathbf{G}}^T \mathbf{G}(c - \hat{c}). \quad (27)$$

Letting $\mathbf{H} = \tilde{\mathbf{G}}^T \mathbf{G}$ we obtain

$$\mathbf{H}E[(c-\hat{c})(c-\hat{c})^T]\mathbf{H}^T = \mathbf{H}E[pp^T]\mathbf{H}^T$$
$$= E[\omega\omega^T]. \quad (28)$$

Considering the diagonal elements of (28) leads to

$$\sum_{l=1}^N H_{k,l}^2 \pi_l = E[\omega_k^2] \simeq \lambda_k^2/12, \quad (29)$$

which is the expression we need that relates $\pi$ to $\lambda$. Let $\bar{\mathbf{G}}$ and $\bar{\mathbf{H}}$ denote the matrices that have the matrix elements squared of $\mathbf{G}$ and $\mathbf{H}$ respectively. Equations 24 and 29 become

$$\mathcal{D}(\lambda) = u^T \bar{\mathbf{G}} \pi \quad (30)$$

$$\bar{\mathbf{H}} \pi = \bar{\lambda}/12 \quad (31)$$

where $u$ is the vector of all-ones and $\bar{\lambda}_i = \lambda_i^2$.

*Rate-Distortion Optimization:* The optimization can be put in the form of the minimization of average distortion (30) subject to the rate constraint (19) to obtain,

$$\min_{\pi} \quad \left\{ u^T \bar{\mathbf{G}} \pi + \gamma \sum_k log(\bar{\lambda}_k) \right\}, \quad (32)$$
$$\text{such that } \pi > 0,\ \bar{\mathbf{H}}\pi = \bar{\lambda}/12.$$

In (32), $\gamma$ is a Lagrange multiplier enforcing the rate constraint and $\pi > 0$ is enforced since $E[pp^T]$ is a positive definite covariance matrix (Equation 23). Observe that (32) poses a convex optimization problem which can be solved to yield $\pi$ and $\lambda$ using standard techniques[2] [11].

## 3. SIMULATION RESULTS AND CONCLUSION

We implemented our work within HEVC reference software (HM-12.0) as a set of extra INTRA prediction modes. The codec was augmented to transmit/receive overhead information that signaled the usage of the new modes. Our results include this overhead. The reference and the augmented codecs both used the same default INTRA main profile configuration parameters. For each INTRA prediction mode we designed a specific prediction matrix $\mathbf{F}$ (Equation 6). This matrix is so that INTRA prediction is repeated using the closest samples rather than the boundary samples within each coded block. We then constructed the nonorthogonal transform $\mathbf{G} = \mathbf{FT}$ (Equation 8) and obtained $\lambda$ through (32). For each coded block quantized coefficients were obtained using Algorithm 1. Mode decisions and encoding of quantized coefficients and overhead were handled using the reference software which generated bit-streams similar to HEVC except for the addition of overhead bits.

We report results for standard test images (Figure 1) in Table 1 (a) and on standard video sequences (30 frames compressed) in Table 1 (b). As seen in Table 1 (a), the proposed work obtains substantial improvements especially on images with significant directional structure. *The reader should note that these improvements are over a state-of-the-art codec that is already employing very high-performance INTRA prediction and that we have not changed prediction weights, etc.* Video sequences benefit from the proposed work based on the extent of directional singularities depicted. Gains are pronounced on computer graphics and "screen content" (Table 1 (a), top) but reduced on classical sequences (Table 1 (b), bottom) where HEVC INTRA prediction is already working well. For each prediction mode the proposed work obtains improvements, with the bulk of the gains coming from directional modes (modes 2-35 in HEVC). This is consistent with earlier discussion regarding Gaussian processes.

[2]The unconstrained solution can be shown to be $\lambda_i = \sqrt{\gamma/a_i}$, where $a^T = u^T \bar{\mathbf{G}} \bar{\mathbf{H}}^{-1}/12$. When this solution was not admissible we have observed quick convergence using descent techniques.

(a) image 1 (b) image 2 (c) image 3

(d) image 4 (e) image 5 (f) image 6

**Fig. 1**. Test images.

| | % Rate Gain |
|---|---|
| image 1 | 12.0% |
| image 2 | 13.0% |
| image 3 | 11.6% |
| image 4 | 8.9% |
| image 5 | 12.4% |
| image 6 | 7.9% |

(a) Images

| | % Rate Gain |
|---|---|
| Slide Editing | 6.9% |
| China Speed | 5.0% |
| Slide Show | 4.1% |
| Foreman | 1.3% |
| Mobile | 1.3% |
| Flower | 1.0% |

(b) Video

**Table 1**. Percentage rate gains (at constant distortion) the proposed work obtains over HEVC (computed with aid of [12] at $QP = 22, 27, 32, 37$).

We proposed a technique that incorporates changes in the decoding chain to allow the decoder to make effective use of all available decoded information, i.e., all decoded prediction residuals, with marginal increases in decoding complexity. Encoder complexity increases through application of Algorithm 1, which is known to converge rapidly in typical cases [9, 4]. We note that for the case of simple linear prediction (as employed within AVC/HEVC) the algorithm further simplifies due to simplifications in Equations 15 and 16. We leave a full complexity analysis to another paper.

Our work primarily impacts pictures with significant directional structures and obtains substantial improvements over a codec that is already targeting such structures with mode-based predictors. Our future work will consider more sophisticated linear and nonlinear predictors for further performance improvements. Another interesting avenue is to augment multi-resolutional coders that use directional prediction [13].

## 4. REFERENCES


[1] D. Marpe, T. Wiegand, and G.J. Sullivan, "The h.264/mpeg4 advanced video coding standard and its applications," *Communications Magazine, IEEE*, vol. 44, no. 8, pp. 134–143, Aug 2006.

[2] G.J. Sullivan, J. Ohm, Woo-Jin Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1649–1668, Dec 2012.

[3] D. Mukherjee, J. Bankoski, A. Grange, Jingning Han, J. Koleszar, P. Wilkins, Yaowu Xu, and R. Bultje, "The latest open-source video codec vp9 - an overview and preliminary results," in *Picture Coding Symposium (PCS), 2013*, Dec 2013, pp. 390–393.

[4] O.G. Guleryuz and M.T. Orchard, "Optimized nonorthogonal transforms for image compression," *Image Processing, IEEE Transactions on*, vol. 6, no. 4, pp. 507–522, Apr 1997.

[5] D.L. Donoho, M. Vetterli, R.A. DeVore, and I. Daubechies, "Data compression and harmonic analysis," *Information Theory, IEEE Transactions on*, vol. 44, no. 6, pp. 2435–2476, Oct 1998.

[6] A. Cohen, I. Daubechies, O.G. Guleryuz, and M.T. Orchard, "On the importance of combining wavelet-based nonlinear approximation with coding strategies," *Information Theory, IEEE Transactions on*, vol. 48, no. 7, pp. 1895–1921, Jul 2002.

[7] Allen Gersho and Robert M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, Norwell, MA, USA, 1991.

[8] O.G. Guleryuz and M.T. Orchard, "On the dpcm compression of gaussian autoregressive sequences," *Information Theory, IEEE Transactions on*, vol. 47, no. 3, pp. 945–956, Mar 2001.

[9] P. Moulin, "A multiscale relaxation algorithm for snr maximization in nonorthogonal subband coding," *Image Processing, IEEE Transactions on*, vol. 4, no. 9, pp. 1269–1281, Sep 1995.

[10] Thomas M. Cover and Joy Thomas, *Elements of Information Theory*, Wiley, 1991.

[11] Stephen Boyd and Lieven Vandenberghe, *Convex Optimization*, Cambridge University Press, New York, NY, USA, 2004.

[12] G. Bjontegaard, "Calculation of average psnr differences between rd-curves," *ITU-T Q.6/SG16 VCEG-M33 Contribution*, vol. 48, Apr 2001.

[13] O.G. Guleryuz and A.L. da Cunha, "Image compression with a geometrical entropy coder," in *Image Processing, 2006 IEEE International Conference on*, Oct 2006, pp. 1161–1164.