Energy Preservation in PVQ-Based Video Coding

Jean-Marc Valin

July 6, 2015

Abstract

This paper starts with an introduction and ends with a conclusion

1 Introduction

This mini-paper describes a proposal for adapting the CELT energy conservation principle to video coding based on a pyramid vector quantizer (PVQ). One potential advantage of conserving energy of the AC coefficients in video coding is preserving textures rather than low-passing them. Also, by introducing a fixed-resolution PVQ-type quantizer, we automatically gain a simple activity masking model.

The main challenge of adapting this scheme to video is that we have a good prediction (the reference frame), so we are essentially starting from a point that is already on the PVQ hyper-sphere, rather than at the origin like in CELT. Other challenges are the introduction of a quantization matrix and the fact that we want the reference (motion predicted) data to perfectly correspond to one of the entries in our codebook.

2 Pyramid Vector Quantization and Spherical Quantization

A pyramid vector quantization (PVQ) codebook [Fisher] of dimension N is constructed as the sum of K signed unit pulses:

$$\mathbf{y} \in \mathbb{Z}^N : \sum_{i=0}^{N-1} |y_i| = K \ .$$

In the Opus codec [RFC6716], the PVQ codebook is used in the context of spherical quantization, with the gain encoded separately from a unit-norm vector derived from a PVQ codeword $\mathbf{y}/\left\|\mathbf{y}\right\|$.

3 Application to still image coding

In order to apply PVQ-based gain-shape quantization to DCT coefficients, we need to first divide the coefficient into *bands*. For 4x4 blocks, a single band is used for all AC coefficients, while 8x8 and 16x16 blocks are split into 4 and 7 bands, respectively (Fig. XX).

4 Encoder

Let vector $\mathbf{x}_d$ denote the (pre-normalization) DCT band to be coded in the current block and vector $\mathbf{r}_d$ denote the corresponding reference after motion compensation, the encoder computes and encodes the "band gain"

$$g = \sqrt{\mathbf{x}_d^T \mathbf{x}_d} \,. \tag{1}$$

The normalized band is computed as

$$\mathbf{x} = \frac{\mathbf{x}_d}{g} \,, \tag{2}$$

with the normalized reference $\mathbf{r}$ similarly computed based on $\mathbf{r}_d$. The encoder then finds the position and sign of the maximum value in $\mathbf{r}$

$$m = \operatorname*{argmax}_i |r_i| \tag{3}$$

$$s = \operatorname{sgn}(r_m) \tag{4}$$

and computes the Householder reflection that reflects $\mathbf{r}$ to $-s\mathbf{e}_m$. The reflection vector is given by

$$\mathbf{v} = \mathbf{r} + s\mathbf{e}_m \,. \tag{5}$$

The encoder reflects the normalized band to find

$$\mathbf{z} = \mathbf{x} - \frac{2}{\mathbf{v}^T\mathbf{v}}\mathbf{v}\left(\mathbf{v}^T\mathbf{x}\right) \,. \tag{6}$$

The similarity between the current band and the reference band is represented by the angle (assuming no quantization)

$$\theta = \arccos \frac{-s z_m}{\|\mathbf{z}\|} \,. \tag{7}$$

Let N be the number of dimensions in $\mathbf{x}$ and K be the number of pulses in our codebooks, we search for the codebook entry

$$q = \operatorname*{argmax}_i \frac{\mathbf{p}_i^T\left(\mathbf{z} - z_m\mathbf{e}_m\right)}{\sqrt{\mathbf{p}_i^T\mathbf{p}_i}} \,, \tag{8}$$

where $\mathbf{p}_i$ is the i^{th} combination of magnitudes and signs that satisfies $\|\mathbf{p}_i\|_{L1} = K$.

5 Decoder

The decoder starts by decoding the codebook entry $\mathbf{p}_q$ and uses it to reconstruct the unit-norm reflected band as

$$\hat{\mathbf{z}} = -s \cos\hat{\theta}\mathbf{e}_m + \sin\hat{\theta}\frac{\mathbf{p}_q}{\sqrt{\mathbf{p}_q^T\mathbf{p}_q}} . \tag{9}$$

Because the decoder has access to exactly the same reference as the encoder, it is able to apply (3)-(5) to obtain the same $\mathbf{v}$ as used in the encoder. The decoded normalized band is

$$\hat{\mathbf{x}} = \hat{\mathbf{z}} - \frac{2}{\mathbf{v}^T\mathbf{v}}\mathbf{v}\left(\mathbf{v}^T\hat{\mathbf{x}}\right) . \tag{10}$$

The renormalized band is computed by taking into account the quantization resolution:

$$\hat{\mathbf{x}}_d = \hat{g}\hat{\mathbf{x}} . \tag{11}$$

6 Coding Resolution

It is desirable for a single quality parameter to control K and the resolution of gain and angle. That quality parameter should also take into account activity masking to some extent. According to Jason Garrett-Glaser, x264's activity masking uses a resolution proportional to $g^{2\alpha}$, with $\alpha = 0.173$. We can derive a scalar quantizer based on quantization index γ that follows this resolution:

$$\begin{aligned}
\frac{d\hat{g}}{d\hat{\gamma}} &= Q\hat{g}^{2\alpha} \\
\hat{g}^{-2\alpha}d\hat{g} &= Qd\hat{\gamma} \\
\frac{\hat{g}^{1-2\alpha}}{1-2\alpha} &= Q\hat{\gamma} \\
\hat{g} &= \left((1-2\alpha)\,Q\hat{\gamma}\right)^{1/(1-2\alpha)} \\
\hat{g} &= Q_g\hat{\gamma}^{1/(1-2\alpha)}
\end{aligned} \tag{12}$$

where $Q_g = \left((1-2\alpha)\,Q\right)^{1/(1-2\alpha)}$ is the companded gain resolution and "master" quality parameter. Let $\beta = 1/\left(1-2\alpha\right)$, the MSE-optimal angle quantization resolution is the one that matches the gain resolution. Considering the distance to be approximately equal to the arc distance, we have

$$Q_\theta = \frac{d\hat{g}/d\hat{\gamma}}{\hat{g}} = \frac{Q_g\beta\hat{\gamma}^{\beta-1}}{Q_g\hat{\gamma}^{\beta}} = \frac{\beta}{\hat{\gamma}} . \tag{13}$$

Alternatively, the number of quantization *steps* for θ is

$$S_\theta = \frac{\pi\hat{\gamma}}{2\beta} .$$

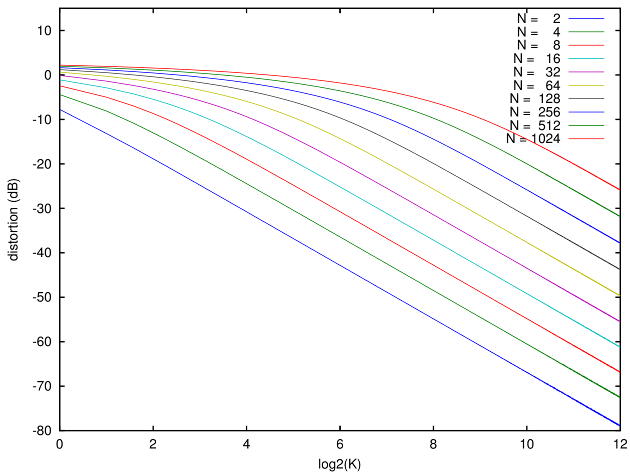

Figure 1: PVQ distortion as a function of N and K for a Laplace-distributed source

6.1 Setting K

Using an *i.i.d.* Laplace source normalized to unit norm, we simulated quantization with different values of N and K. The resulting distortion is shown in Fig. 1. The asymptotic distortion for large values of K is approximately

$$D_{pvq} = \frac{(N-1)^2 + C_K\,(N-1)}{24K^2}\ ,$$

with $C_K = 4.2$.

The distortion due to scalar quantization of the gain is (asymptotically)

$$\begin{aligned} D_g &= \frac{1}{12}\left(d\hat{g}/d\hat{\gamma}\right)^2 \\ &= \frac{\beta^2 Q_g^2 \hat{\gamma}^{2\beta-2}}{12}\ . \end{aligned}$$

To achieve uniform distortion along all dimensions, the distortion due to the $N-2$ PVQ degrees of freedom must be $N-2$ times greater than that due to quantizing the gain, so

$$
\begin{aligned}
(N-2)\, D_g &= \left(\hat{g}\sin\hat{\theta}\right)^2 D_{pvq} \\
\frac{(N-2)\,\beta^2 Q_g^2 \hat{\gamma}^{2\beta-2}}{12} &= \left(Q_g\hat{\gamma}^{\beta}\sin\hat{\theta}\right)^2 \frac{(N-2)^2 + C_K\,(N-2)}{24K^2} \\
(N-2)\,\beta^2 &= \frac{\hat{\gamma}^2\sin^2\hat{\theta}\left[(N-2)^2 + C_K\,(N-2)\right]}{2K^2} \\
K &= \frac{\hat{\gamma}\sin\hat{\theta}}{\beta}\sqrt{\frac{N+C_K-2}{2}}
\end{aligned}
$$

In this way, we can avoid having to signal K because it is determined only by $\hat{\gamma}$ and $\hat{\theta}$, both of which are available to the decoder.

7 Bi-Prediction

We can use this scheme for bi-prediction by introducing a second θ parameter. For the case of two (normalized) reference frames $\mathbf{r}_1$ and $\mathbf{r}_2$, we introduce $\mathbf{s}_1 = (\mathbf{r}_1 + \mathbf{r}_2)/2$ and $\mathbf{s}_2 = (\mathbf{r}_1 - \mathbf{r}_2)/2$. We start by using $\mathbf{s}_1$ as a reference, apply the Householder reflection to both $\mathbf{x}$ and $\mathbf{s}_2$, and evaluate θ_1. From there, we derive a second Householder reflection from the reflected version of $\mathbf{s}_2$ and apply it to $\mathbf{x}_r$. The result is that the θ_2 parameter controls how the current image compares to the two reference images. It should even be possible to use this in the case where the two references are before the frame being encoded, i.e. P frames based on two parents. This might help for fades.

8 Conclusion

While it seems like a good idea, we're still experimenting with the details.