Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 11, 2015

JM. Valin
Mozilla

June 9, 2015

Pyramid Vector Quantization for Video Coding
draft-valin-netvc-pvq-00

Abstract

This proposes applying pyramid vector quantization (PVQ) to video coding.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This draft describes a proposal for adapting the Opus RFC 6716 [RFC6716] energy conservation principle to video coding based on a pyramid vector quantizer (PVQ) [Pyramid-VQ]. 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. This proposal is described in greater details in [Perceptual-VQ], as well as in demo [PVQ-demo].

2. Gain-Shape Coding and Activity Masking

The main idea behind the proposed video coding scheme is to code groups of DCT coefficient as a scalar gain and a unit-norm "shape" vector. A block’s AC coefficients may all be part of the same group, or may be divided by frequency (e.g. by octave) and/or by directionality (horizontal vs vertical).

It is desirable for a single quality parameter to control the resolution of both the gain and the shape. Ideally, that quality parameter should also take into account activity masking, that is, the fact that the eye is less sensitive to regions of an image that have more details. According to Jason Garrett-Glaser, the perceptual analysis in the x264 encoder uses a resolution proportional to the variance of the AC coefficients raised to the power a, with a=0.173. For gain-shape quantization, this is equivalent to using a resolution of g^(2a), where g is the gain. We can derive a scalar quantizer that follows this resolution:

$$g = Q_g \, \gamma^{b} ,$$

where gamma is the gain quantization index, b=1/(1-2*a) and Q_g is the gain resolution and main quality parameter.

An important aspect of the current proposal is the use of prediction. In the case of the gain, there is usually a significant correlation with the gain of neighboring blocks. One way to predict the gain of a block is to compute the gain of the coefficients obtained through intra or inter prediction. Another way is to use the encoded gain of the neighboring blocks to explicitly predict the gain of the current block.

3. Householder Reflection

Let vector x_d denote the (pre-normalization) DCT band to be coded in the current block and vector r_d denote the corresponding reference (based on intra prediction or motion compensation), the encoder computes and encodes the "band gain" g = sqrt(x_d^T x_d). The normalized band is computed as

$$x = \frac{x_d}{\| x_d \|} ,$$

with the normalized reference vector r similarly computed based on r_d. The encoder then finds the position and sign of the largest component in vector r:

$$m = \operatorname{argmax}_i | r_i |$$
$$s = \operatorname{sign}(r_m)$$

and computes the Householder reflection that reflects r to -s e_m, where e_m is a unit vector that points in the direction of dimension m. The reflection vector is given by

$$v = r + s \, e_m .$$

The encoder reflects the normalized band to find the unit-norm vector

$$z = x - 2 \frac{v^T x}{v^T v} v .$$

The closer the current band is from the reference band, the closer z is from -s e_m. This can be represented either as an angle, or as a coordinate on a projected pyramid.

4. Angle-Based Encoding

Assuming no quantization, the similarity can be represented by the angle

$$\theta = \arccos(-s\, z_m) .$$

If theta is quantized and transmitted to the decoder, then z can be reconstructed as

$$z = -s \cos(\theta)\, e_m + \sin(\theta)\, z_r ,$$

where z_r is a unit vector based on z that excludes dimension m.

The vector z_r can be quantized using PVQ. Let y be a vector of integers that satisfies

$$\sum_i(|y[i]|) = K ,$$

with K determined in advance, then the PVQ search finds the vector y that maximizes y^T z_r / (y^T y) . The quantized version of z_r is

$$z_{rq} = \frac{y}{\| y \|} .$$

If we assume that MSE is a good criterion for optimizing the resolution, then the angle quantization resolution should be (roughly)

$$Q_\theta = \frac{dg}{d(\gamma)} * \frac{1}{g} = \frac{b}{\gamma} .$$

To derive the optimal K we need to consider the normalized distortion for a Laplace-distributed variable found experimentally to be approximately

$$D_p = \frac{(N-1)^2 + C*(N-1)}{24*K^2} ,$$

with C ˜= 4.2. The distortion due to the gain is

$$D_g = \frac{b^2*Q_g^2*\gamma^{(2*b-2)}}{12} .$$

Since PVQ codes N-2 degrees of freedom, its distortion should also be (N-2) times the gain distortion, which eventually leads us to the optimal number of pulses

$$K = \frac{\gamma*\sin(\theta)}{b} \sqrt{\frac{N + C - 2}{2}} .$$

The value of K does not need to be coded because all the variables it depends on are known to the decoder. However, because Q_theta depends on the gain, this can lead to unacceptable loss propagation behavior in the case where inter prediction is used for the gain. This problem can be worked around by making the approximation sin(theta)˜=theta. With this approximation, then K depends only on the theta quantization index, with no dependency on the gain. Alternatively, instead of quantizing theta, we can quantize sin(theta) which also removes the dependency on the gain. In the general case, we quantize f(theta) and then assume that sin(theta)˜=f(theta). A possible choice of f(theta) is a quadratic function of the form:

$$f(\theta) = a1\ \theta - a2\ \theta^2.$$

where a1 and a2 are two constants satisfying the constraint that f(pi/2)=pi/2. The value of f(theta) can also be predicted, but in case where we care about error propagation, it should only be predicted from information coded in the current frame.

5. Bi-prediction

We can use this scheme for bi-prediction by introducing a second theta parameter. For the case of two (normalized) reference frames r1 and r2, we introduce s1=(r1+r2)/2 and s2=(r1-r2)/2. We start by using s1 as a reference, apply the Householder reflection to both x and s2, and evaluate theta1. From there, we derive a second Householder reflection from the reflected version of s2 and apply it to z. The result is that the theta2 parameter controls how the current image compares to the two reference images. It should even be possible to use this in the case of fades, using two references that are before the frame being encoded.

6. Coefficient coding

Encoding coefficients quantized with PVQ differs from encoding scalar-quantized coefficients from the fact that the sum of the coefficients magnitude is known (equal to K). It is possible to take advantage of the known K value either through modeling the distribution of coefficient magnitude or by modeling the zero runs. In the case of magnitude modeling, the expectation of the magnitude of coefficient n is modeled as

$$E(|y_n|) = \alpha \cdot \frac{K_n}{N - n} ,$$

where K_n is the number of pulses left after encoding coeffients from 0 to n-1 and alpha depends on the distribution of the coefficients. For run-length modeling, the expectation of the position of the next non-zero coefficient is given by

$$E(|run|) = \beta \cdot \frac{N - n}{K_n} ,$$

where beta also models the coefficient distribution.

7. Development Repository

The algorithms in this proposal are being developed as part of Xiph.Org’s Daala project. The code is available in the Daala git repository at <https://git.xiph.org/daala.git>. See <https://xiph.org/daala/> for more information.

8. IANA Considerations

This document makes no request of IANA.

9. Security Considerations

This draft has no security considerations.

10. Acknowledgements

Thanks to Jason Garrett-Glaser, Timothy Terriberry, Greg Maxwell, and Nathan Egge for their contribution to this document.

11. Informative References

[Perceptual-VQ]
Valin, JM. and TB. Terriberry, "Perceptual Vector Quantization for Video Coding", Proceedings of SPIE Visual Information Processing and Communication , February 2015, <http://jmvalin.ca/papers/spie_pvq.pdf>.

[PVQ-demo]
Valin, JM., "Daala: Perceptual Vector Quantization (PVQ)", November 2014, <https://people.xiph.org/~jm/daala/pvq_demo/>.

[Pyramid-VQ]
Fischer, T., "A Pyramid Vector Quantizer", IEEE Trans. on Information Theory, Vol. 32 pp. 568-583, July 1986.

[RFC6716] Valin, JM., Vos, K., and T. Terriberry, "Definition of the Opus Audio Codec", RFC 6716, September 2012.

Author's Address

Jean-Marc Valin
Mozilla
331 E. Evelyn Avenue
Mountain View, CA 94041
USA

Email: jmvalin@jmvalin.ca