Pyramid Vector Quantization for Video Coding

Jean-Marc Valin

Daala Coding Party

Sep 2013

Motivations

- Pyramid vector quantization is a key technique used in Opus (both SILK and CELT parts)
- Investigate PVQ for a video codec (Daala)
- Potential advantages
  - Preserves energy (details) even when details are imperfect (instead of blurring)
  - Implicit activity masking
  - Better representation of coefficients

Gain-Shape Quantization

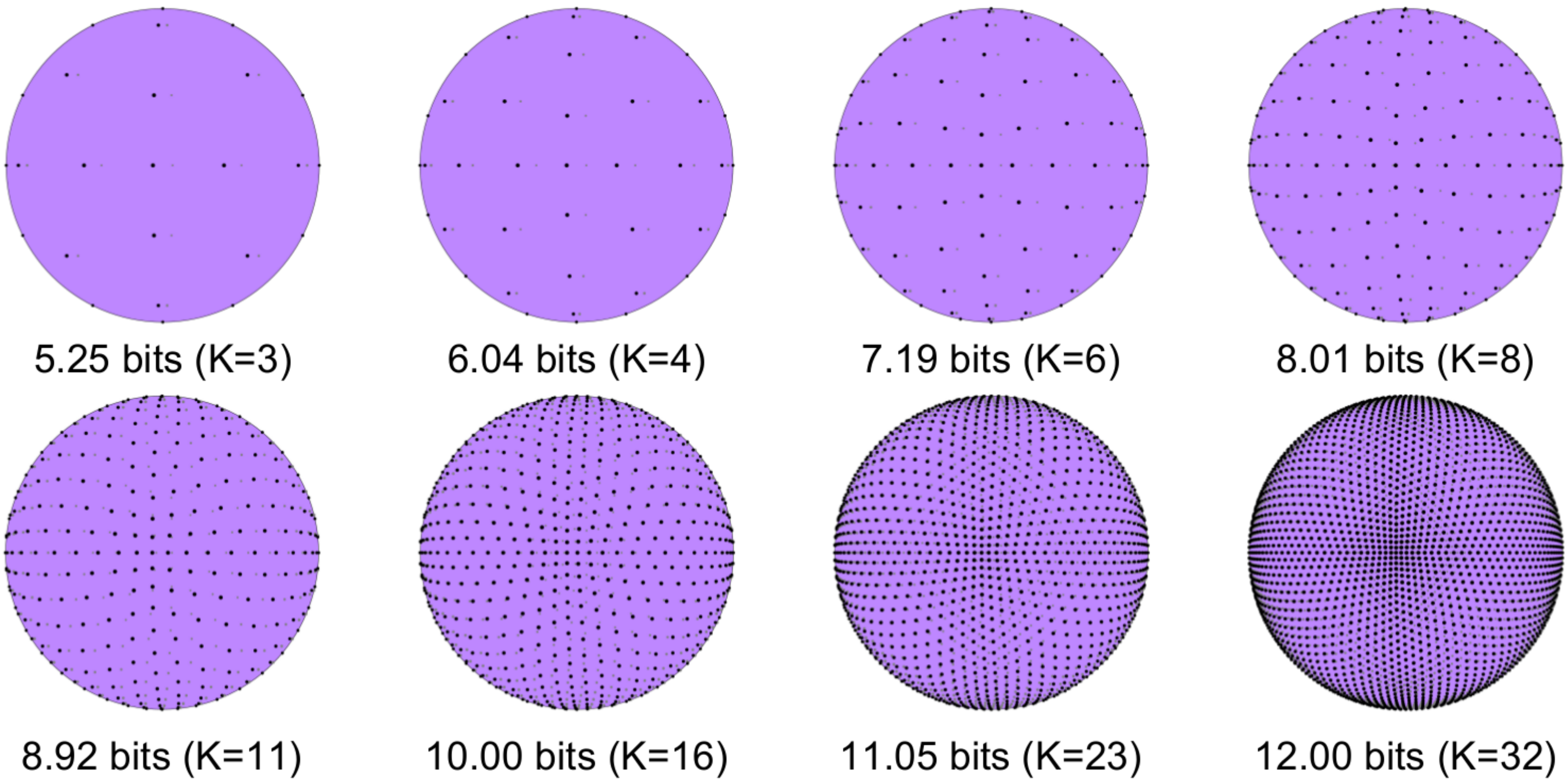
- Represent a vector as magnitude multiplied by unit-norm vector (radius + point on sphere)
  - Amount of texture vs exact details
- Code magnitude separately
  - Adjust resolution of the sphere based on the magnitude

Pyramid Vector Quantizer (PVQ)

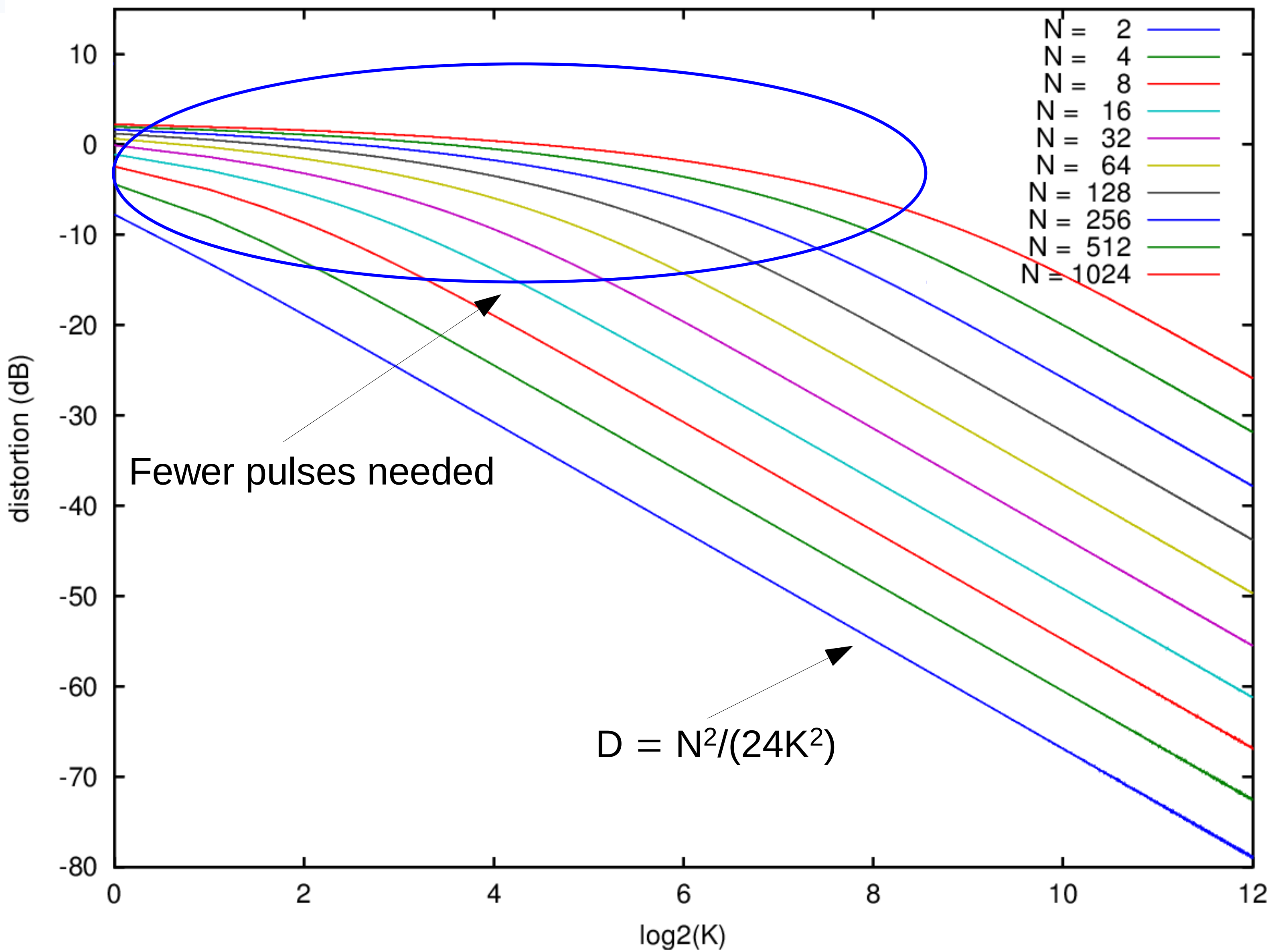
- Place *K* unit pulses in *N* dimensions
  - Up to $N = 1024$ dimensions
- Normalize to unit norm (L_2)

$$S(N,K) = \left\{ \frac{\mathbf{y}}{\|\mathbf{y}\|},\ \mathbf{y} \in \left\{ \mathbb{Z}^{\mathbb{N}} : \sum_{i=0}^{N-1} |y_i| = K \right\} \right\}$$

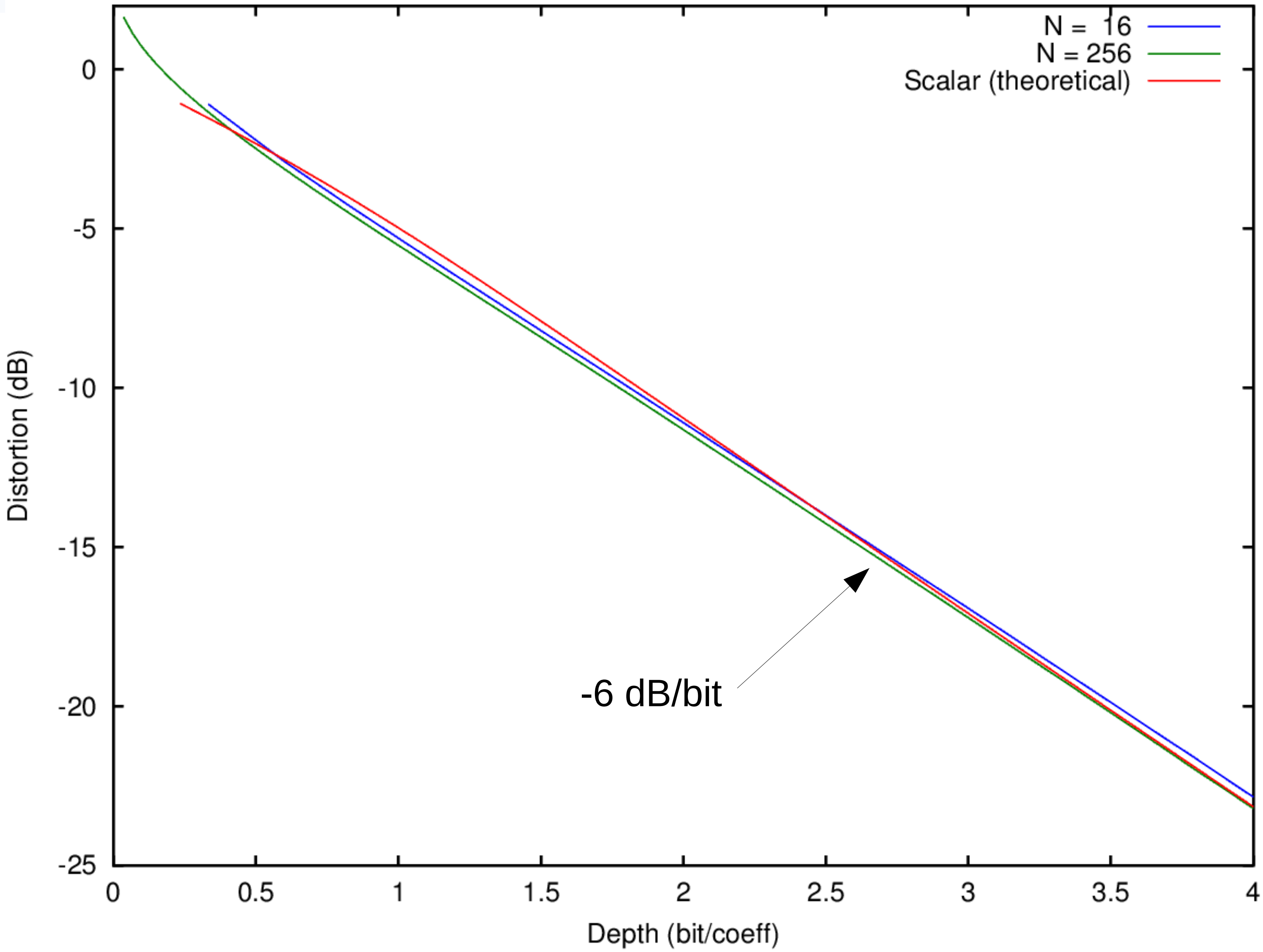
Codebook for *N*=3 and different *K*



Distortion, *N* and *K*



PVQ vs Scalar Quantization

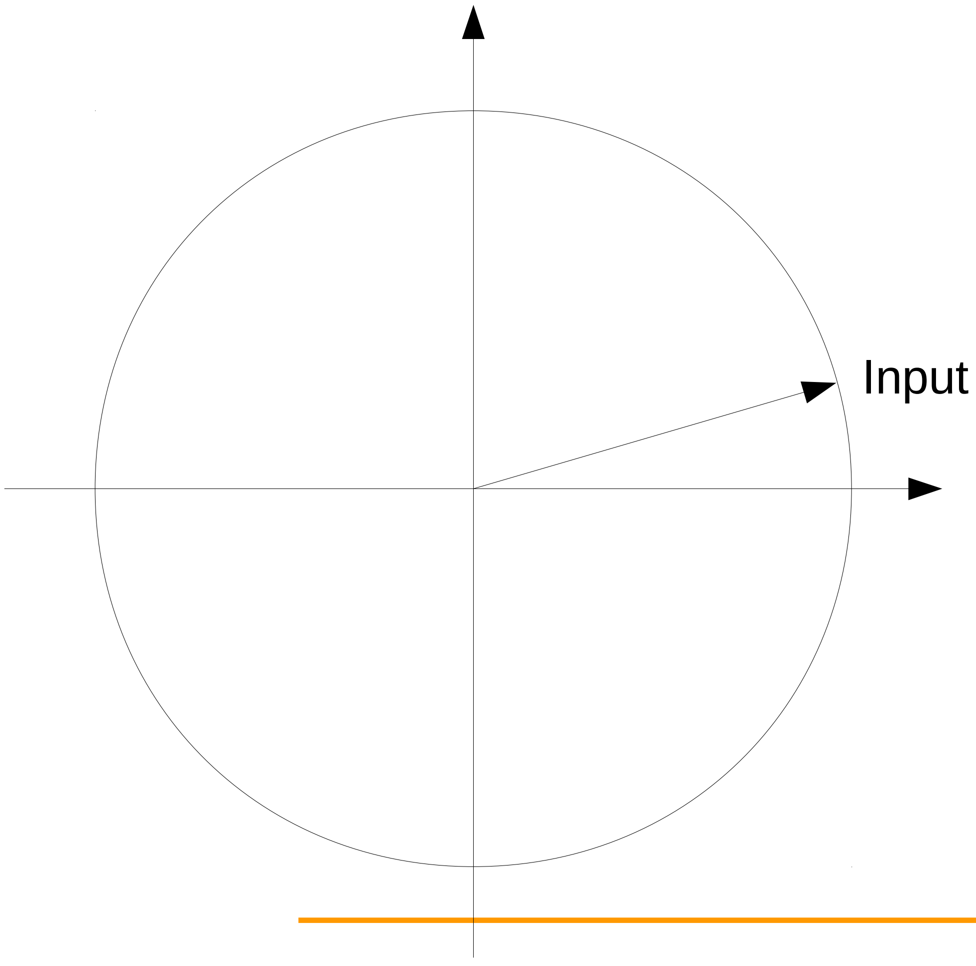


Prediction

- Unlike CELT, we want to predict the vectors
- PVQ on the residual loses energy preservation
- Apply prediction in the normalized vector
  - Use Householder reflection to align prediction with one axis
  - Encode magnitude of the residual as an angle

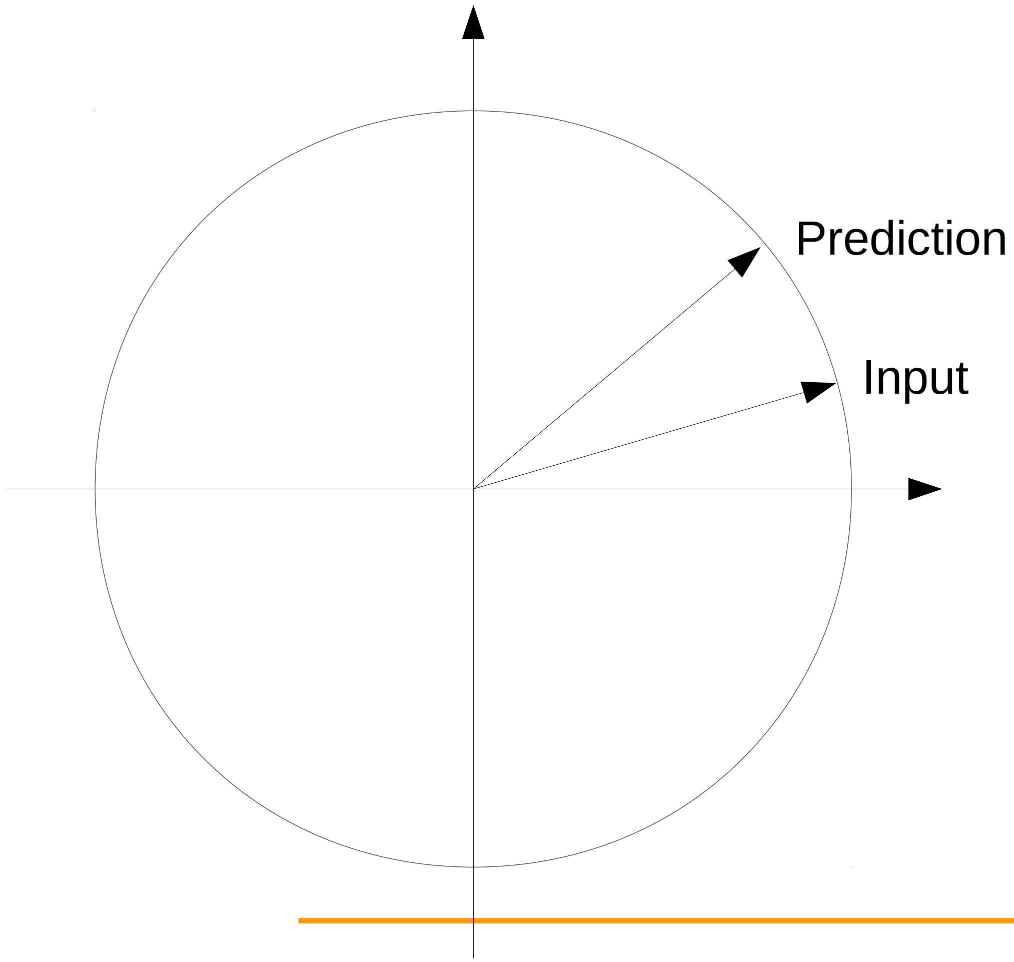
2-D Projection

- Input



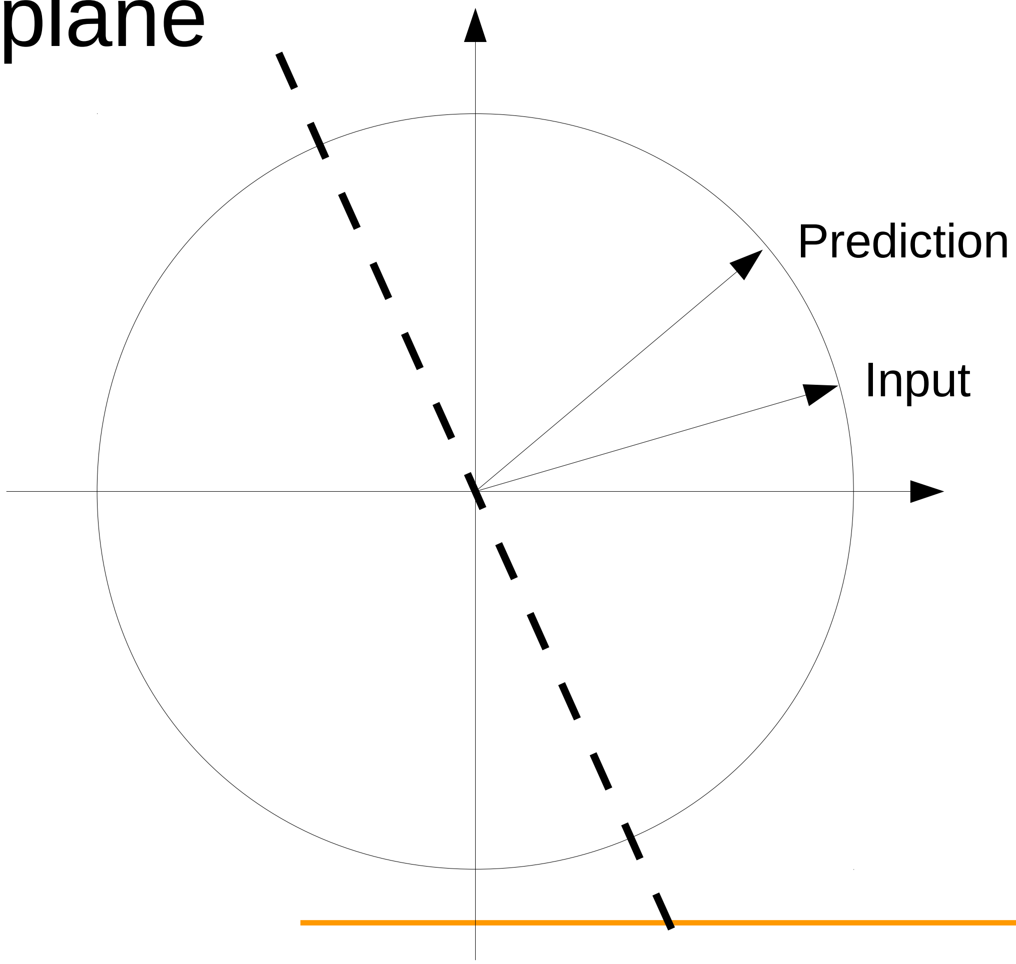
2-D Projection

- Input+prediction



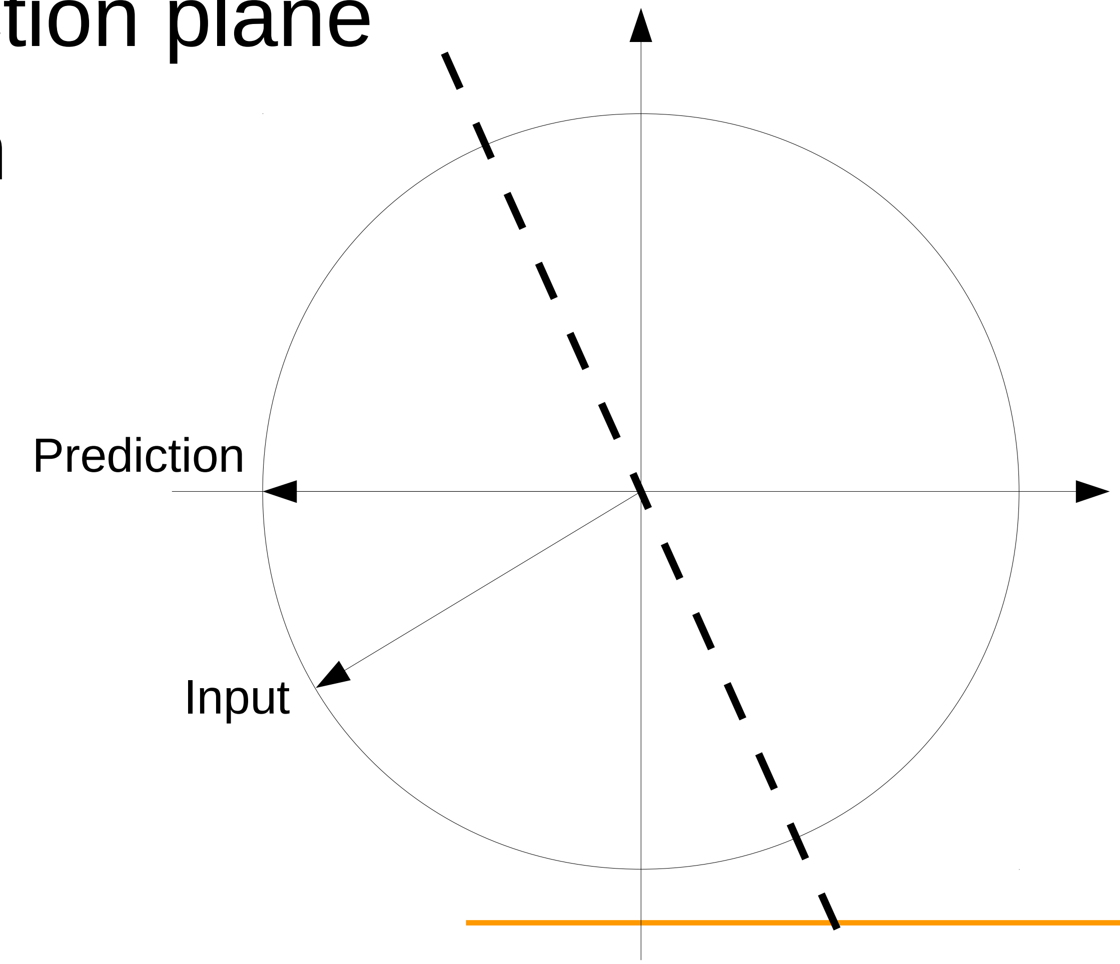
2-D Projection

- Input+prediction
- Compute reflection plane



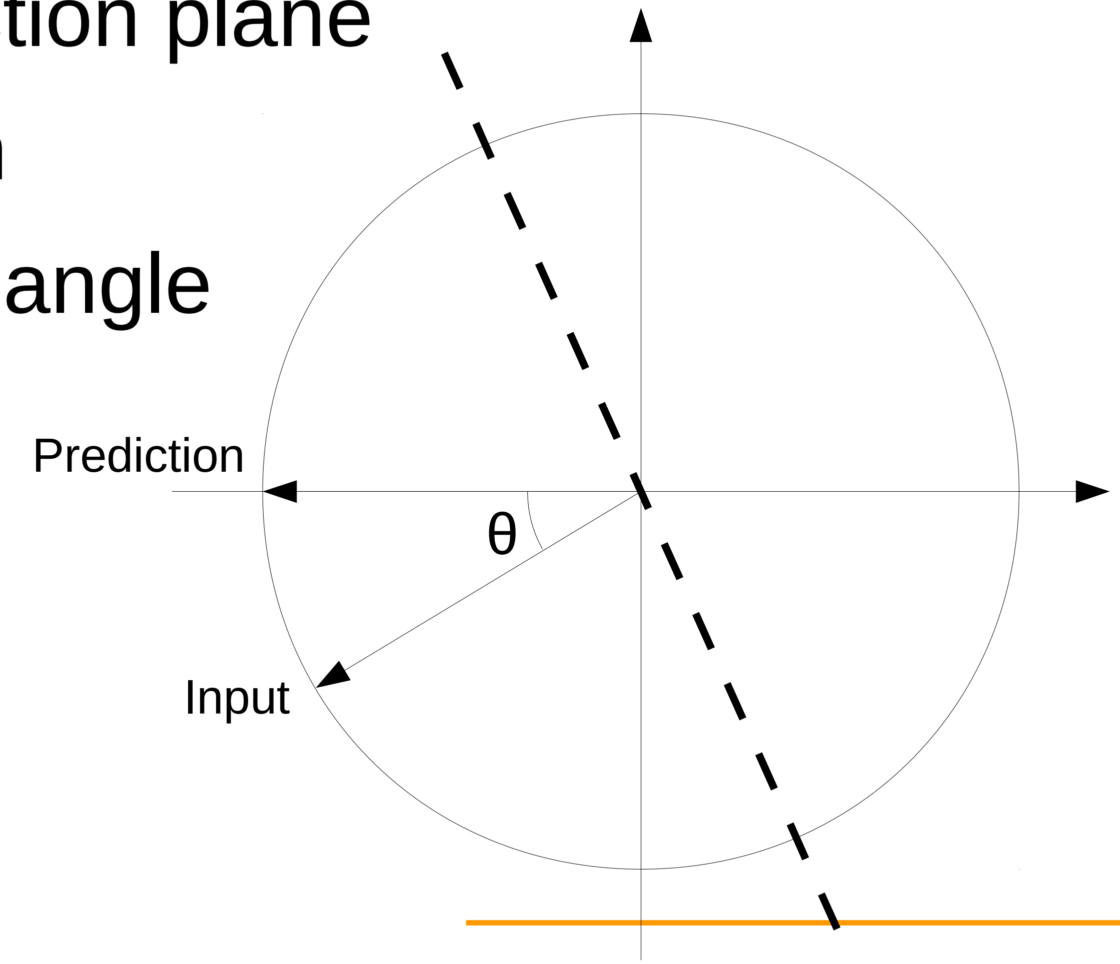
2-D Projection

- Input+prediction
- Compute reflection plane
- Apply reflection



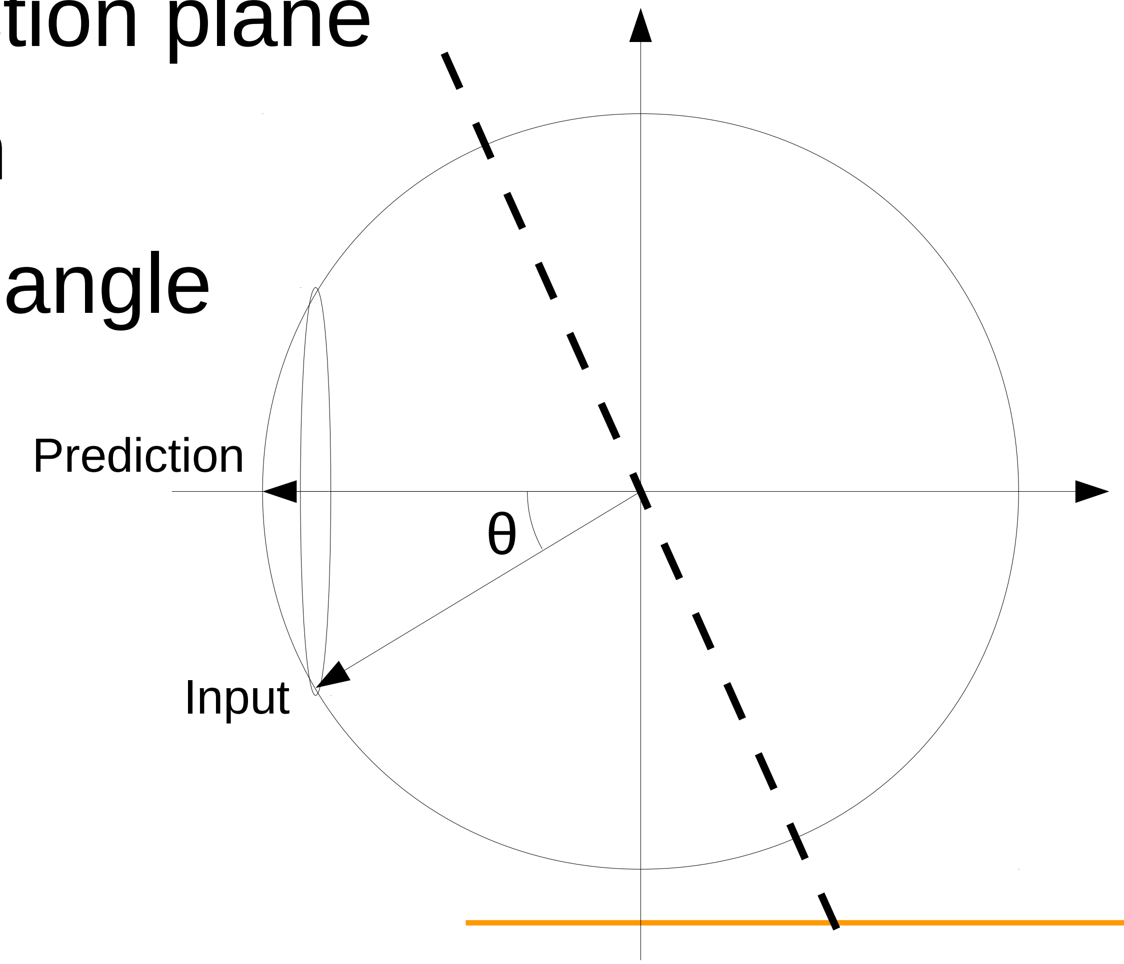
2-D Projection

- Input+prediction
- Compute reflection plane
- Apply reflection
- Compute/code angle



2-D Projection

- Input+prediction
- Compute reflection plane
- Apply reflection
- Compute/code angle
- Code other dimensions



Activity Masking

- Artefacts are easier to detect on flat areas they on textured areas
  - Code unit-norm vector with a resolution that depends on the gain (texture)
- Code companded gain $g_c = g^\gamma$
  - Implicit activity masking built into the bitstream

Open Questions

- How to split into bands
- Avoid wasting bits on still video
- Quantization matrix
- Take advantage of correlation/prediction in gain and angle
- Rate-Distortion Optimization
  - Fast RDO PVQ search?