An Overview of Coding Tools in AV1

Debargha Mukherjee
Google

Outline

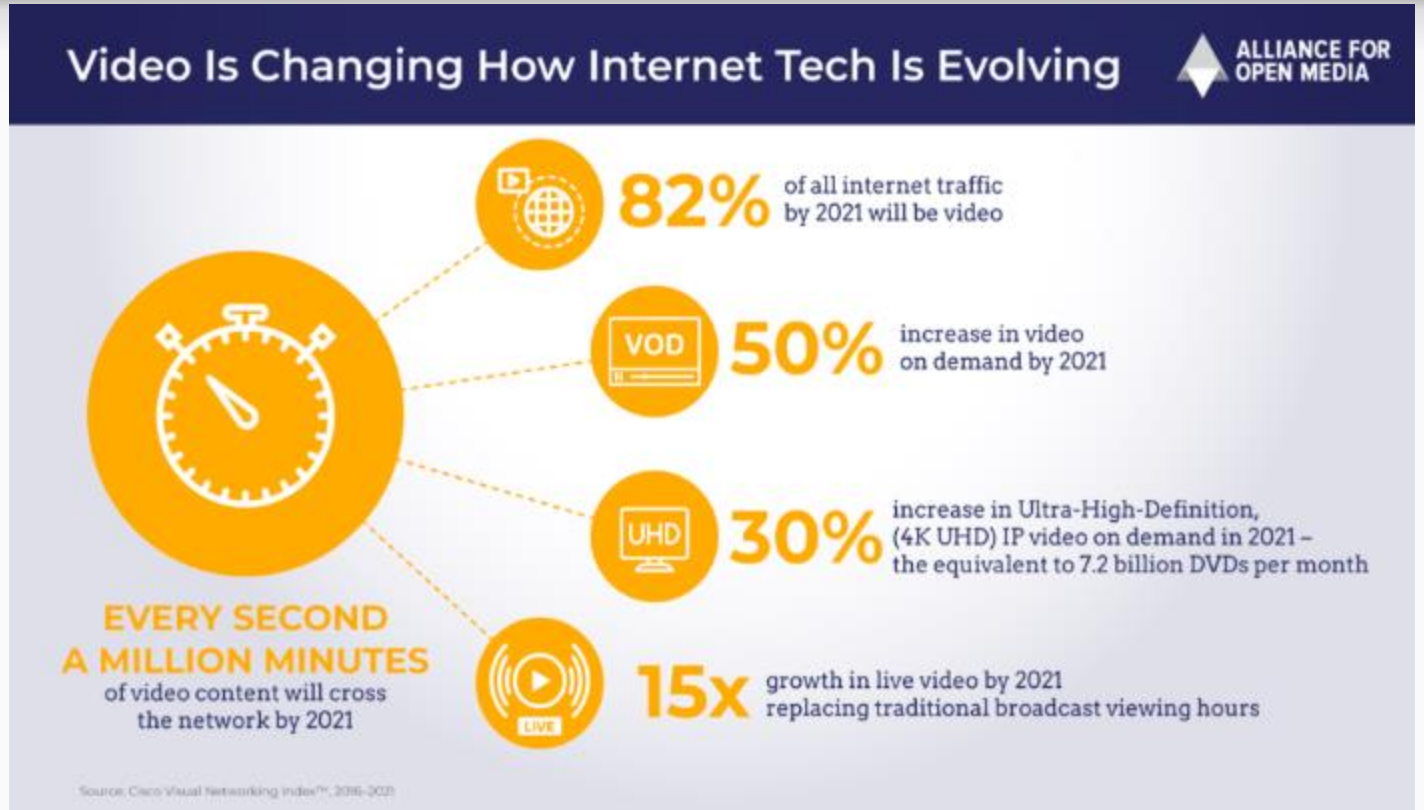
- Alliance for Open Media and AV1
- Coding Tools
- Latest Coding Results
- AV1 Deployment
- Conclusion

Outline

- Alliance for Open Media and AV1
- Coding Tools
- Latest Coding Results
- AV1 Deployment
- Conclusion

Alliance for Open Media and AV1

Projections

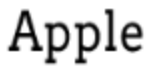


Fast advances in video compression technology is essential to cope with the ongoing explosion in Internet video

Alliance for Open Media and AV1
Founding Members

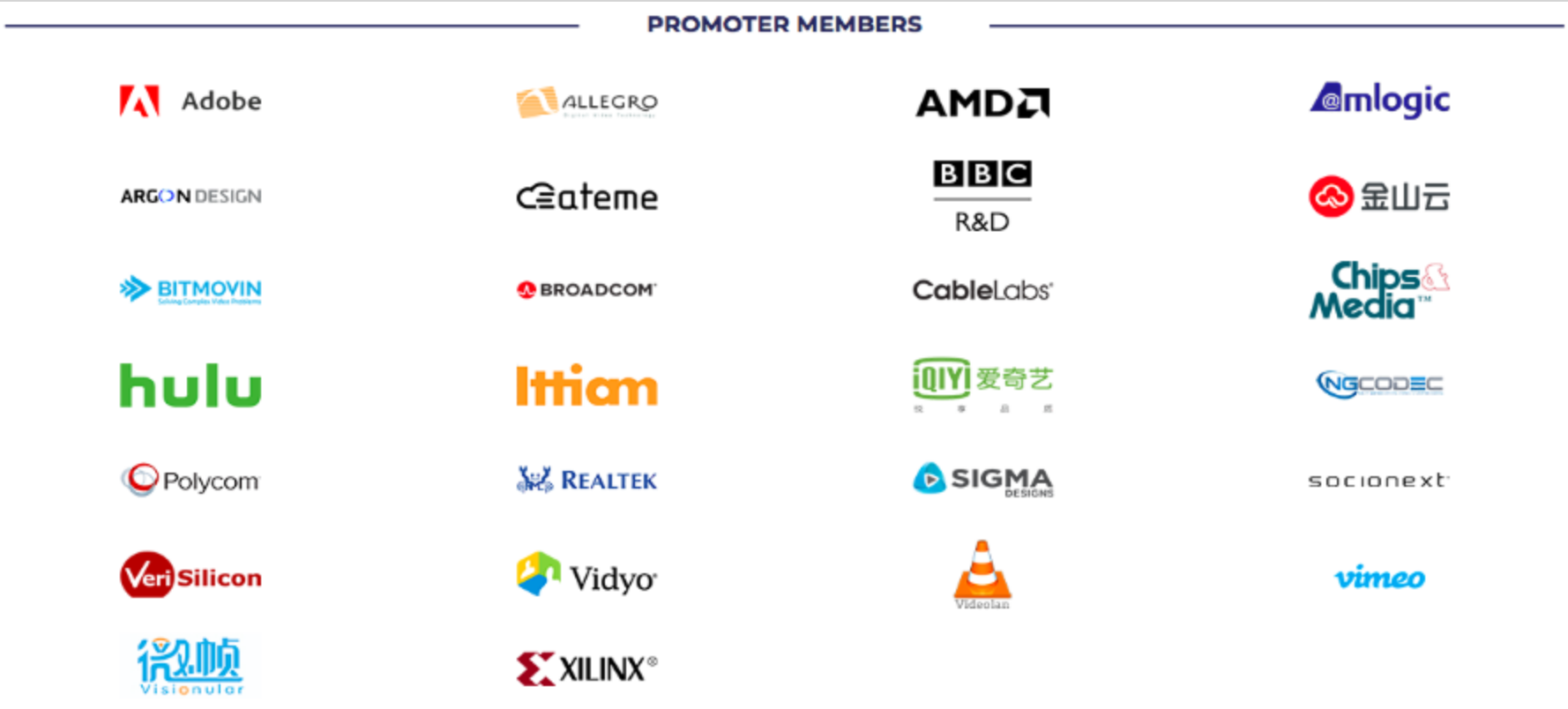
- Many companies began to see the value of a royalty-free codec ecosystem after Google's success with VP9
- AOM: Industry consortium to build royalty-free codecs formed in 2015

FOUNDING MEMBERS



Alliance for Open Media and AV1

Promoter Members

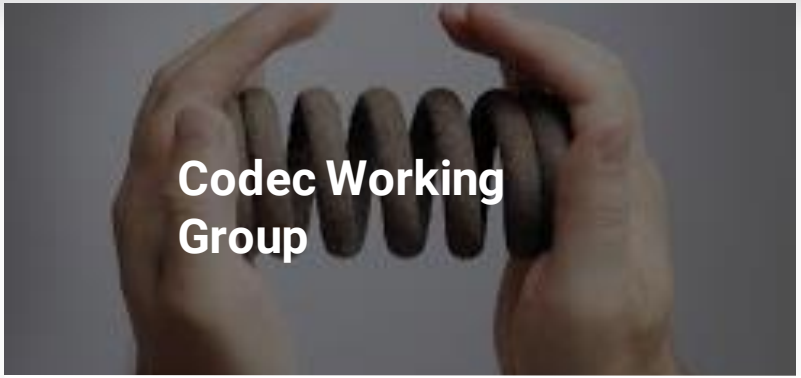


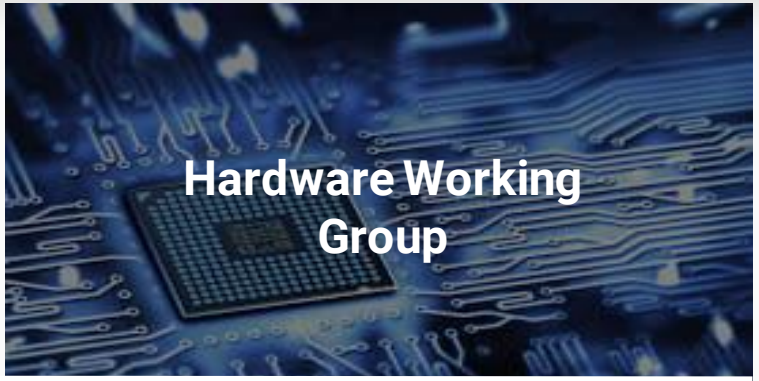
Alliance for Open Media and AV1
AV1

- AV1 - the first video codec from Alliance for Open Media
  - Goal - to achieve about 30% bitrate reduction over VP9 with royalty-free technologies
- Starting point was VP9+
- Tools proposed from:
  - VP10 (Google)
  - Daala (Mozilla)
  - Thor (Cisco)
  - New tools ...

Alliance for Open Media and AV1

Workgroups

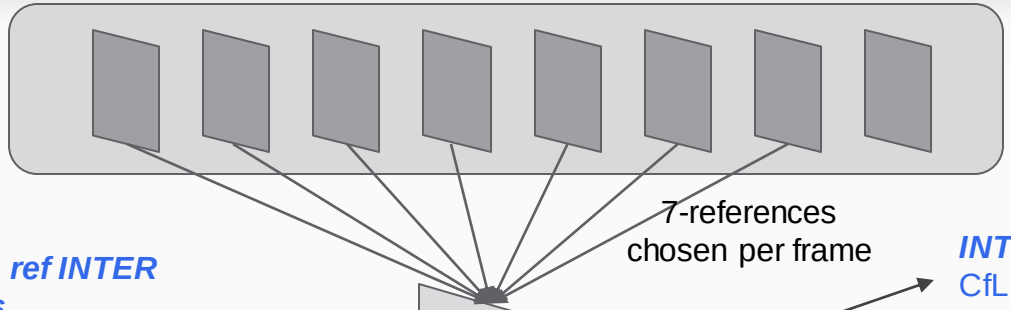




Outline

- Alliance for Open Media and AV1
- Coding Tools
- Latest Coding Results
- AV1 Deployment
- Conclusion

Coding tools
Prediction Framework



Single ref INTER modes
Translation
Global / Local Warp

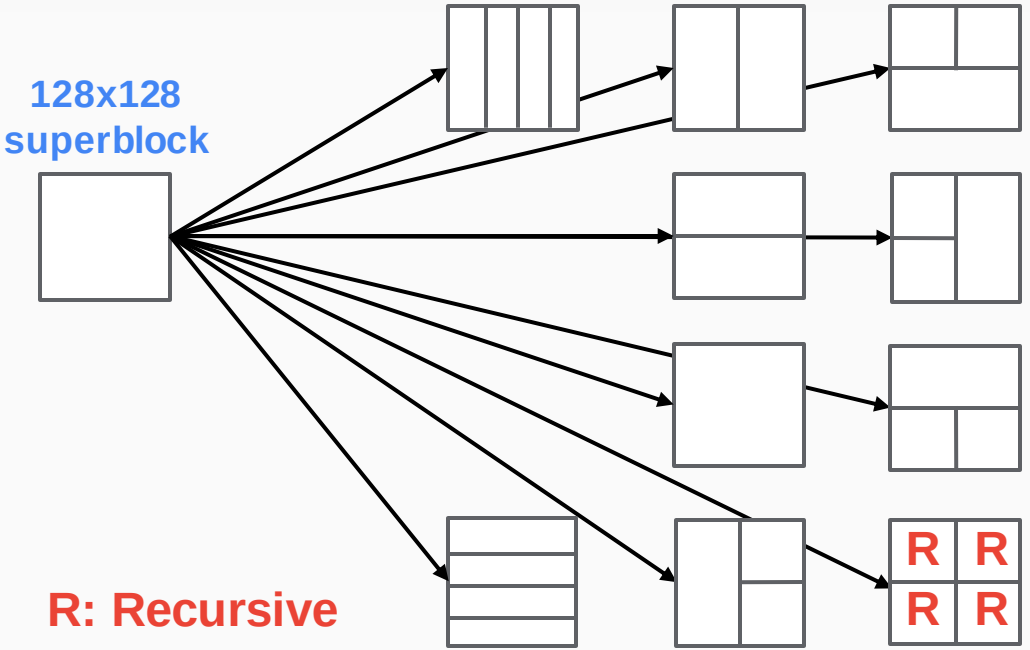
Compound ref INTER modes
Average, Distance-wtd, Difference-wtd, Wedge
OBMC

***INTRA modes*:**
CfL, DC, PAETH, SMOOTH, SMOOTH_H, SMOOTH_V, 8 directional modes + finer directions, Filter-intra modes, palette, intra-block-copy

***INTER-INTRA modes*:**
Gradual Inter-intra
Wedge Inter-intra

Coding tools
Partition Structure

- 10-way recursive partition

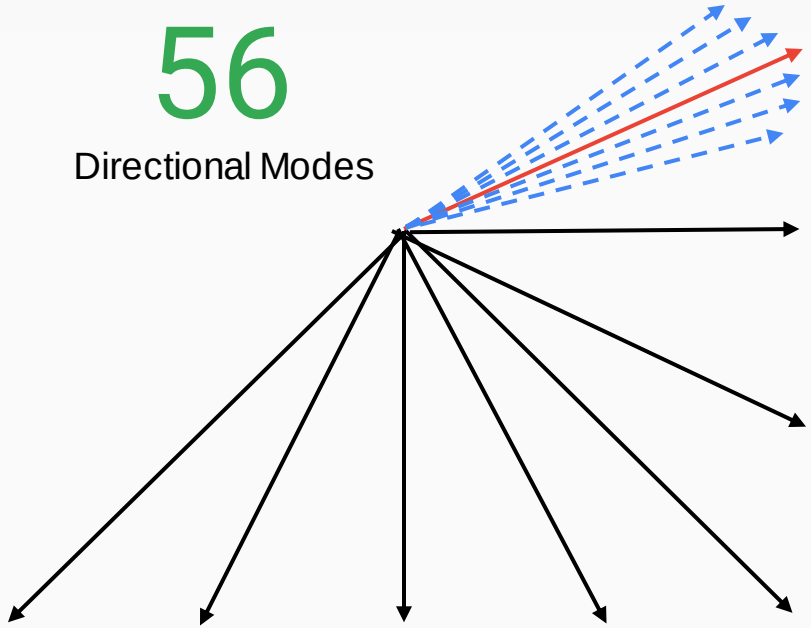


Coding Tools
Intra Prediction

- DC Mode
- Directional Modes
- Paeth Mode
- Smooth/Smooth_h/Smooth_v Modes
- Chroma from Luma
- Recursive Intra Filter
- Intra block copy Mode
- Palette Mode

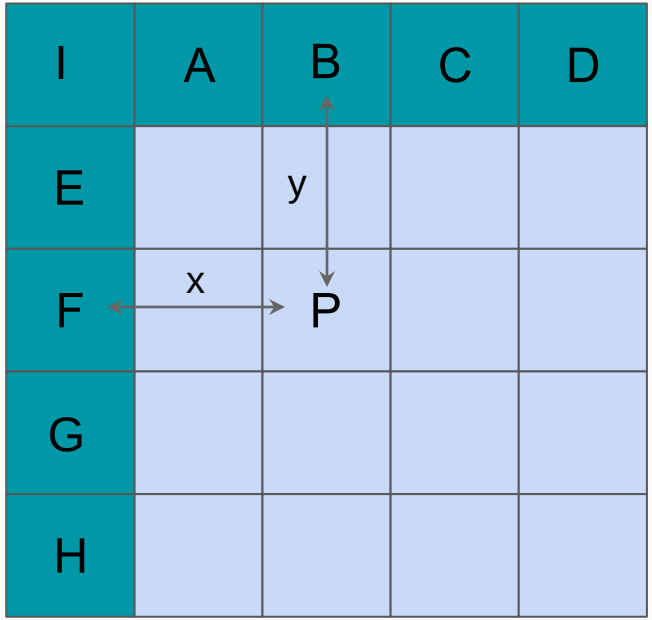
Coding Tools
Intra Prediction - Directional Modes

- 8 main extrapolation directions
- [-3, 3] x 3° angle delta is enabled
- Extended modes are realized by bi-linear interpolation of spatial references



Coding Tools

Intra Prediction - Paeth/Smooth modes



Quadratic interpolation:

SMOOTH_H: $P_{SMOOTH\_H} = w(x)\ F + (1-w(x))\ D$

SMOOTH_V: $P_{SMOOTH\_V} = w(y)\ B + (1-w(y))\ H$

SMOOTH: $P_{SMOOTH} = ½\ (P_{SMOOTH\_H} + P_{SMOOTH\_V})$

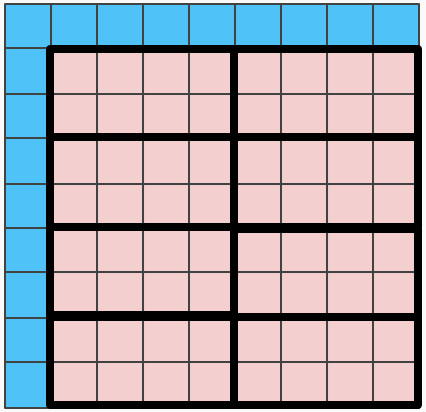
Paeth Mode:

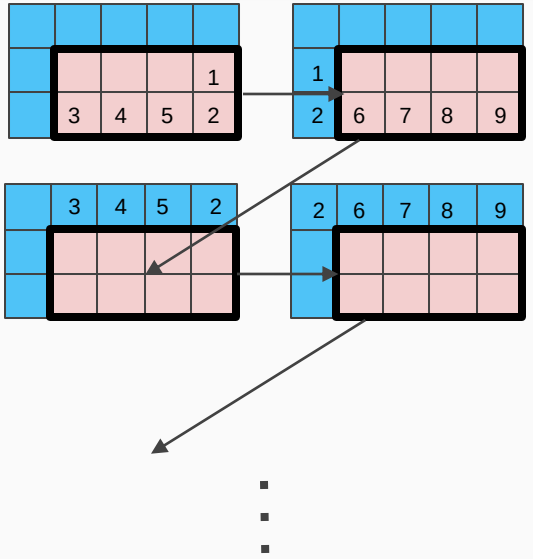
$P_{Paeth} = \text{argmin}\ |x - B+F-I|, \text{over}\ x \in \{B, F, I\}$

Coding Tools

Intra Prediction - Recursive Intra Prediction

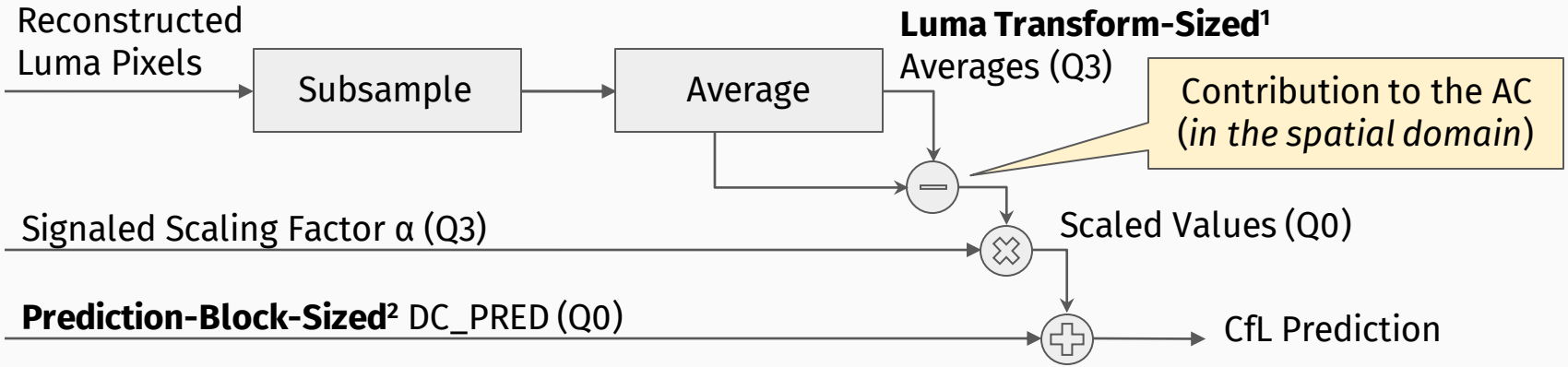
- Recursive Intra Prediction
  - Predict in batches of 4x2 pixels blocks
  - Apply eight 7-tap filters to get prediction for each of the 8 pixels
    - 5 sets of filters for 5 modes
  - Apply recursively using the predicted pixels as neighbors for adjacent blocks.





Coding Tools
Intra Prediction - Chroma from Luma

- Mechanism for INTRA chroma prediction
  - Use reconstructed luma to predict chroma components



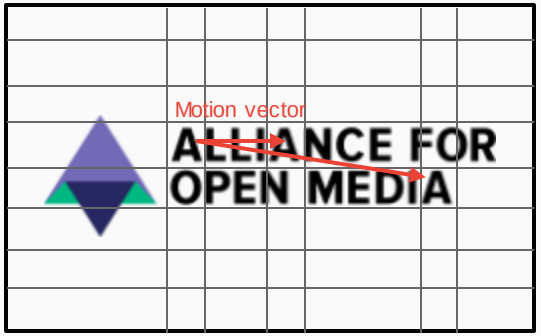
α_{Cb}, α_{Cr} signaled in bit-stream

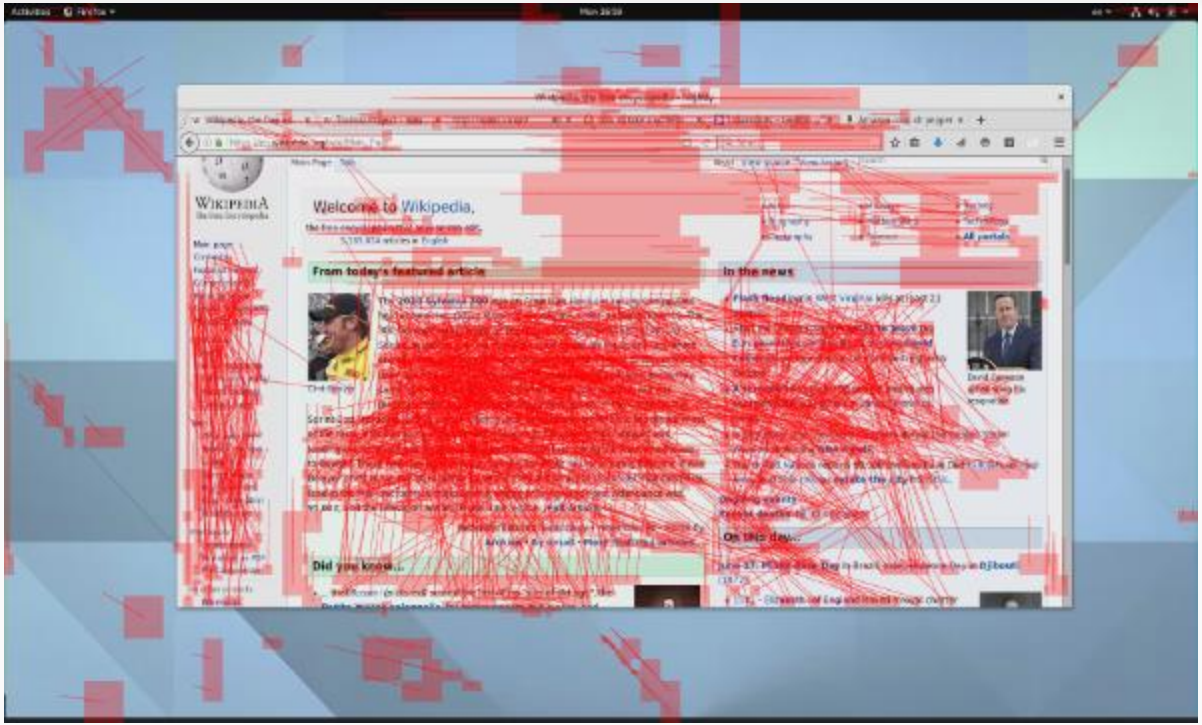
1. Luma average computed over the luma transform block
2. Chroma DC_PRED computed over prediction block

Coding Tools

Intra Prediction - Intra Block Copy

- Screen content tool
- Referencing reconstructed pixels in same frame
- Wavefront reference area

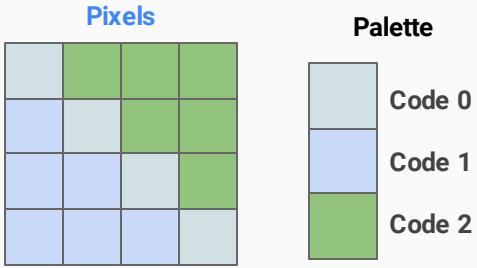




Coding Tools

Intra Prediction - Palette mode

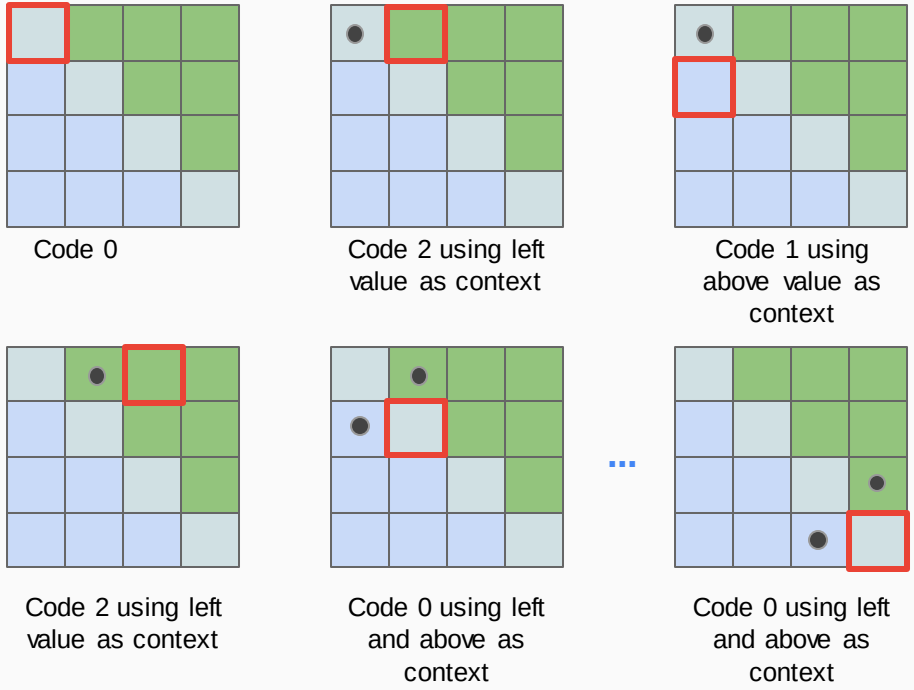
- Screen content tool



Wavefront Order

0	1	3	6
2	4	7	10
5	8	11	13
9	12	14	15

Encoding process proceeds in wavefront order

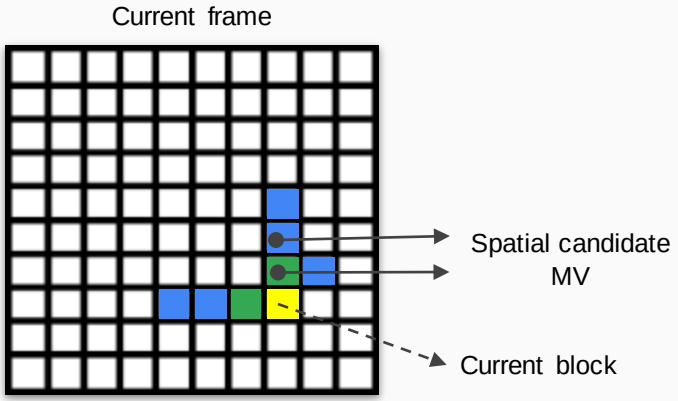


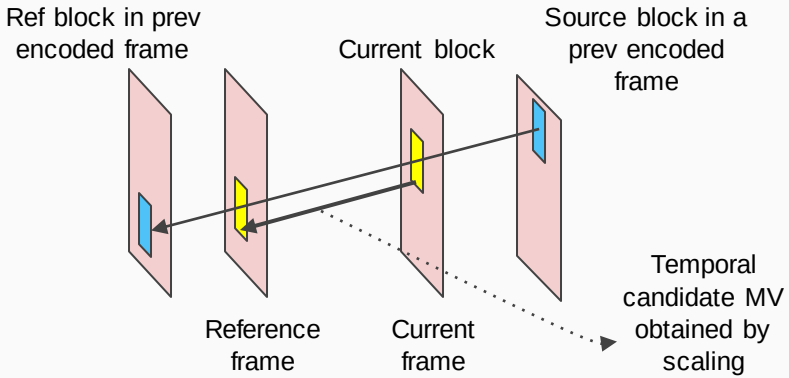
Coding Tools
Inter Prediction

- Expand reference frames (up to 7 / frame) used for prediction
- Dynamic Motion Vector Referencing
  - Spatial neighborhood and Temporal neighborhood candidates
- Subpel filtering
  - Dual Switchable Subpel filters
  - High intermediate precision for filtering and compounding
- Scaled Inter Prediction
- Compound Inter-Inter prediction
- Compound Inter-Intra prediction
- Overlapped Block Motion Compensation
- Affine warp
  - Global warp and Local warp

Coding Tools
Inter Prediction - Dynamic MV Referencing

- MV coding
- Create an ordered list of reference motion vectors from:
  - Spatial neighbors
  - Temporal neighbors
- Separate lists for single reference and compound references

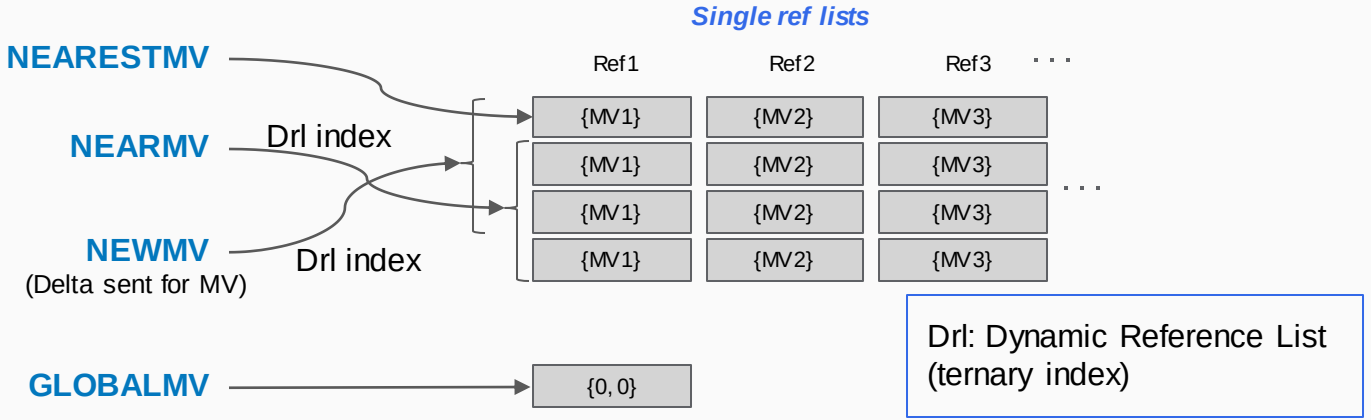




Coding Tools
Inter Prediction - Coding modes (Single-ref)

- Create ranked candidate list of 4 MVs for each reference
- 4 Single-ref modes signaled

Single Reference Modes (4)

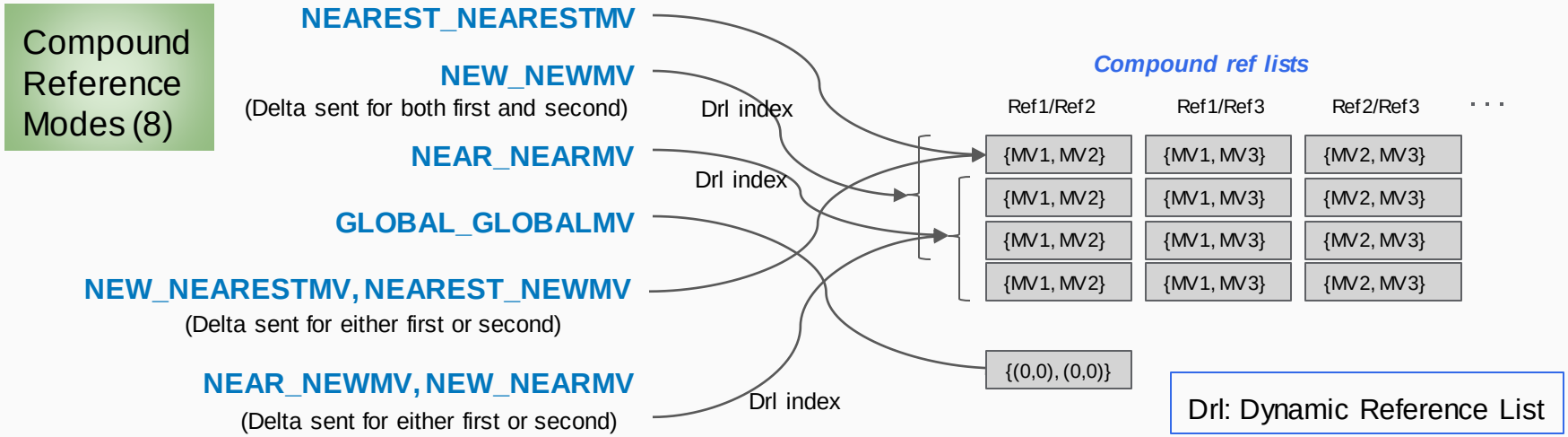


Drl: Dynamic Reference List (ternary index)

Coding Tools

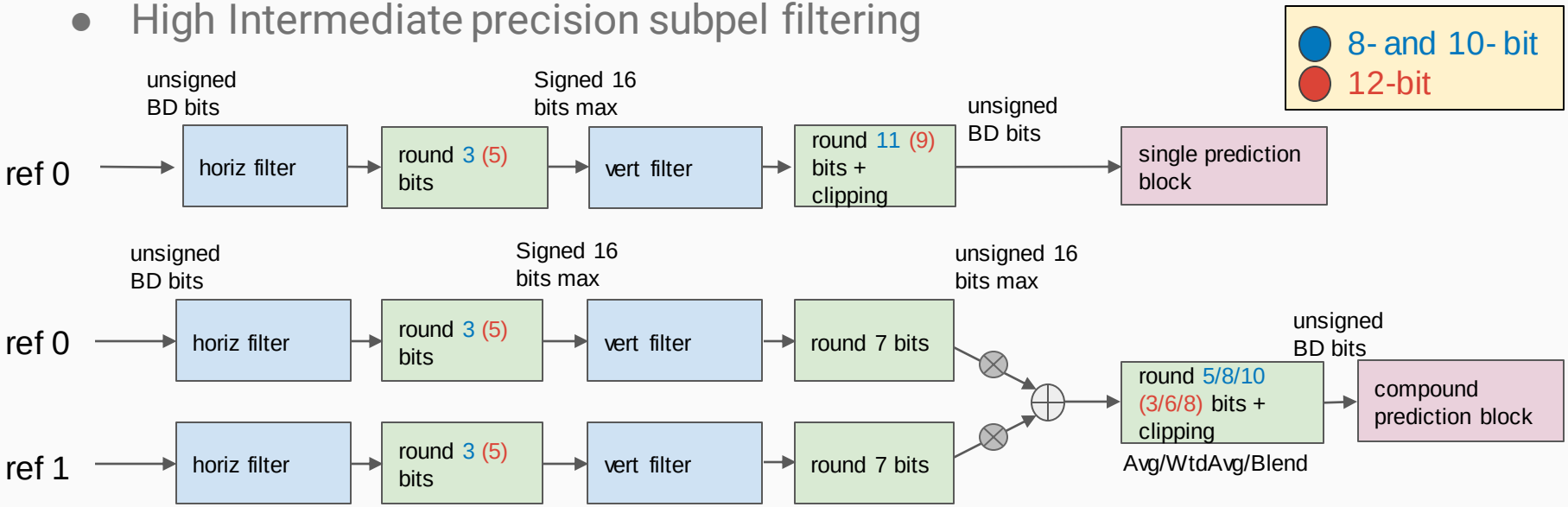
Inter Prediction - Coding modes (Compound-ref)

- Create ranked candidate list of 4 MVs for each reference-pair
- 8 Compound-ref modes signaled



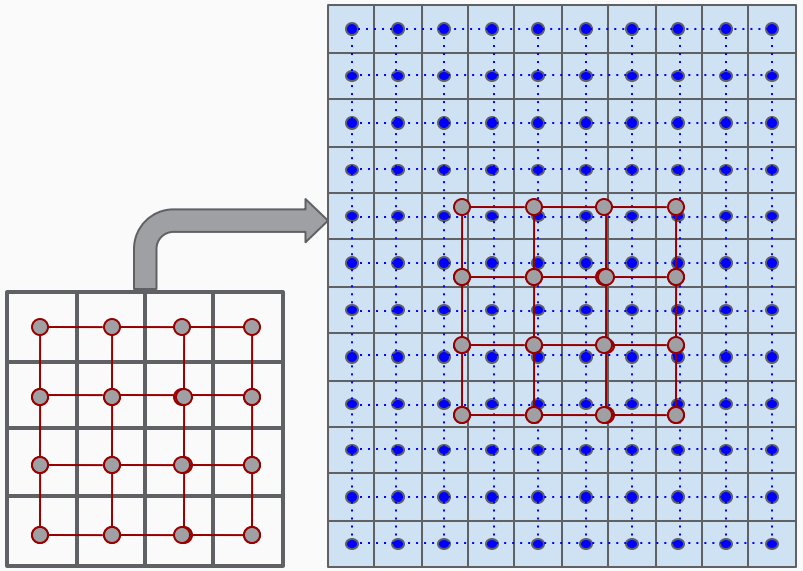
Coding Tools
Inter Prediction - Subpel filtering

- Dual Switchable Subpel filters
  - {SMOOTH, REGULAR, SHARP} x {SMOOTH, REGULAR, SHARP} in hor/vert directions
- High Intermediate precision subpel filtering



Coding Tools
Scaled INTER Prediction

- AV1 can predict from references at different resolution (limit: ½ to 2)
- Relative pixel positions hor (vert) are the same in each row (col)
  - Implies scaling can be implemented as separable filtering where step-sizes between pixels in hor (vert) directions are same
  - Need higher precision for starting offset and steps, but the same 1/16th pel filters as in unscaled prediction can be used.



Coding Tools
Compound Inter Prediction

- Compound = combine two INTER predictors
  - Average weighted
  - Distance weighted
    - Uniform weight based on distance of ref frame from current frame
  - Difference weighted
    - Weight depends on difference of two prediction values per pixel
    - Average when pixel values are close, prefer one when they are different.
  - Wedge weighted
    - Wedge codebook index provides the weights

Coding Tools

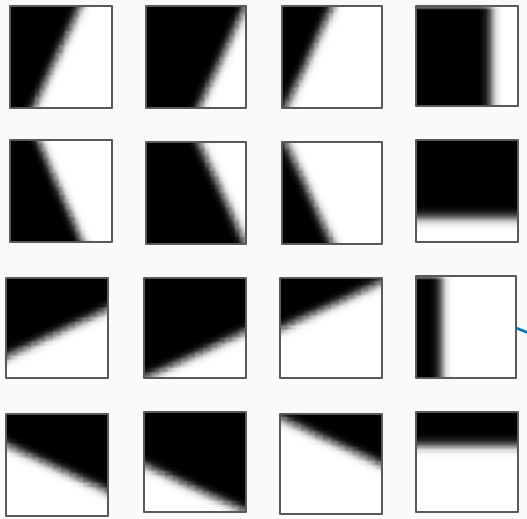
Inter Prediction - Wedge Compound Prediction

Wedge codebook:

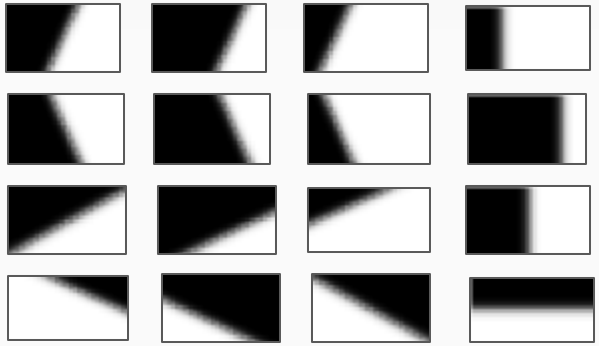
Inter-Inter
4-bit shape
1-bit sign

5 bits total

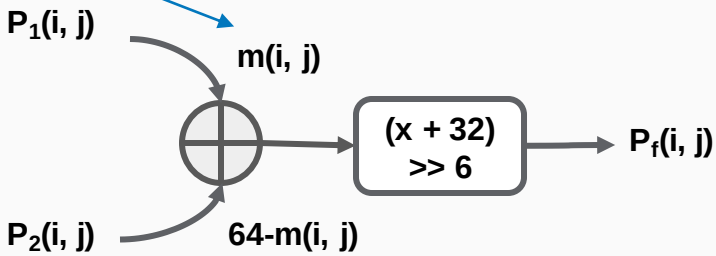
Used for 8x8 up to 32x32 sizes



Square codebook



Rectangular codebook



Coding Tools

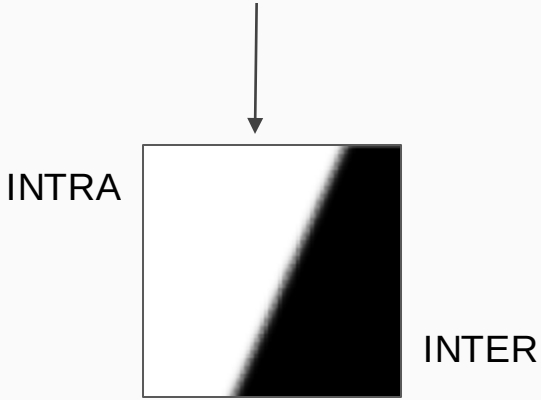
Inter Prediction - Wedge Weighted Compound

- Capture shapes of differently moving objects better
- Soft transition allows a limited number of wedges to be sufficient

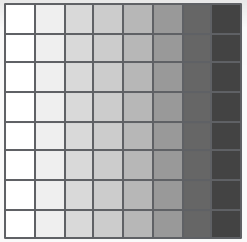
Coding Tools
Inter-Intra Prediction tools

- Combination of an INTER and INTRA predictor
  - Only 4 INTRA modes allowed
- Types
  - Gradual INTER-INTRA modes
  - Wedge INTER-INTRA mode

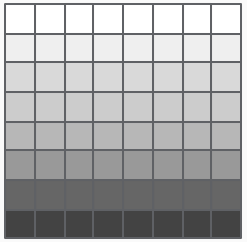
4 modes



II_H_PRED

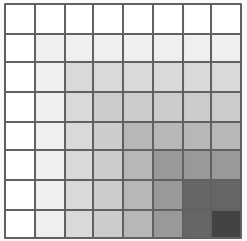


II_V_PRED

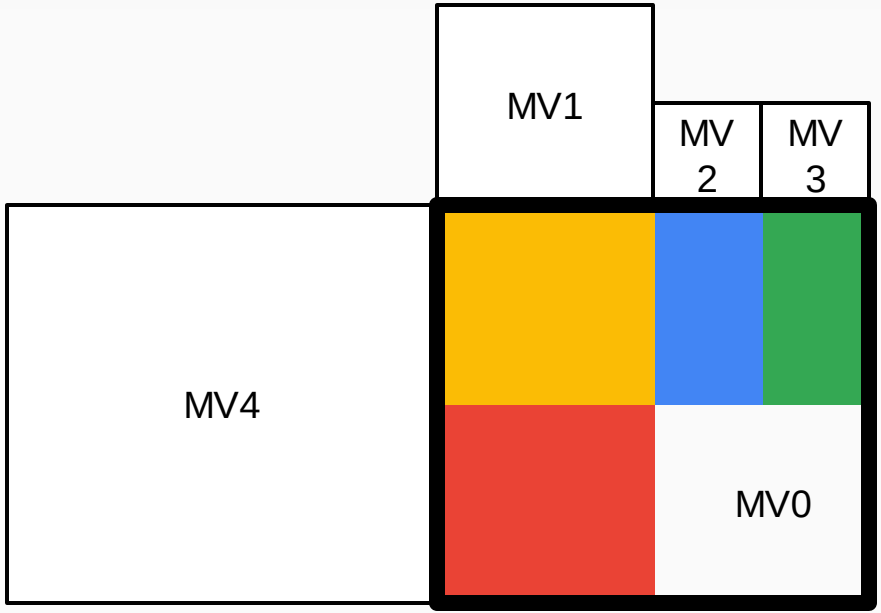


II_SMOOTH_PRED

II_DC_PRED



Coding Tools
Variable block-size OBMC



- Create predictions from neighbors' MVs
- Mitigate the effect of discontinued motion field
- 2-sided causal overlapped predictor (top/left halves)
- Blend with 1-D smooth filters

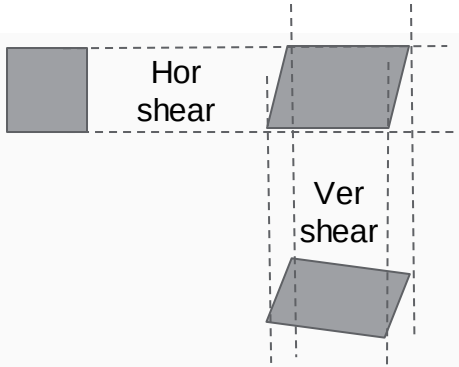
Coding Tools
Affine warps

- True motion is never translational
- More complex motion models have not traditionally been used in video codecs
  - Parameter cost
  - Computation complexity
- AV1
  - Introduces an efficient affine motion compensation technique

Affine model $\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} u \\ v \end{bmatrix}$

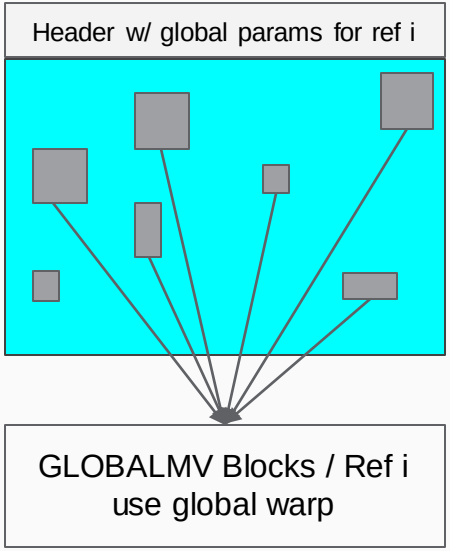
$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \gamma & 1+\Delta \end{bmatrix} \begin{bmatrix} 1+\alpha & \beta \\ 0 & 1 \end{bmatrix}$$

Original warping matrix | Vertical Shear | Horizontal Shear

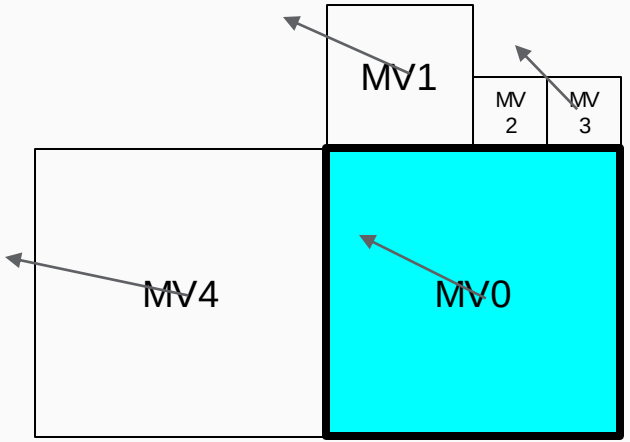


Coding Tools
Affine Warps - Global and Local

- Global warp mode
  - Estimate and send affine parameters for every frame per reference
  - Invoke Global warp with GLOBALMV mode



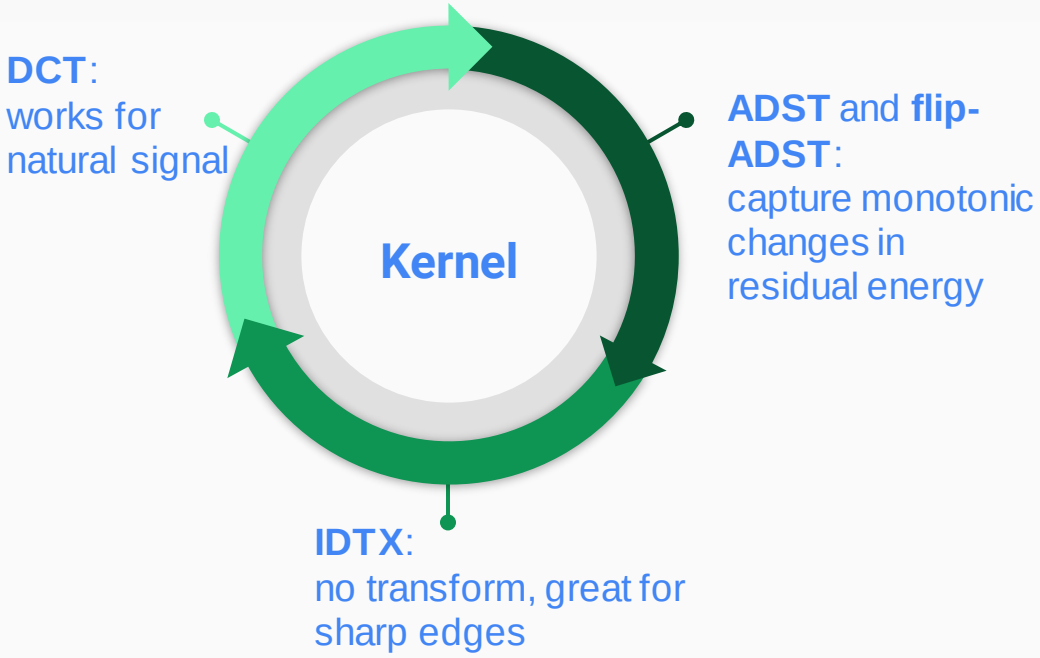
- Local warp mode
  - Estimate affine parameters from neighborhood of current block



Blocks 1, 3, 4 have same reference as current block

Coding Tools
Switchable Transforms

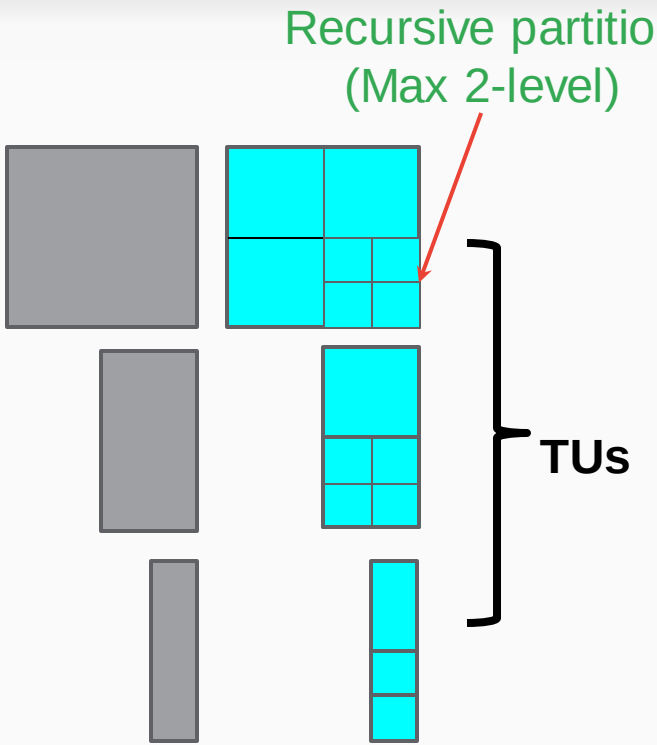
- Expanded transforms
  - 16 possible transform kernels
  - Separable kernels where in each direction one of {DCT, ADST, FlipADST, IDTX} can be selected
  - Larger transforms reduce the number of possible transforms



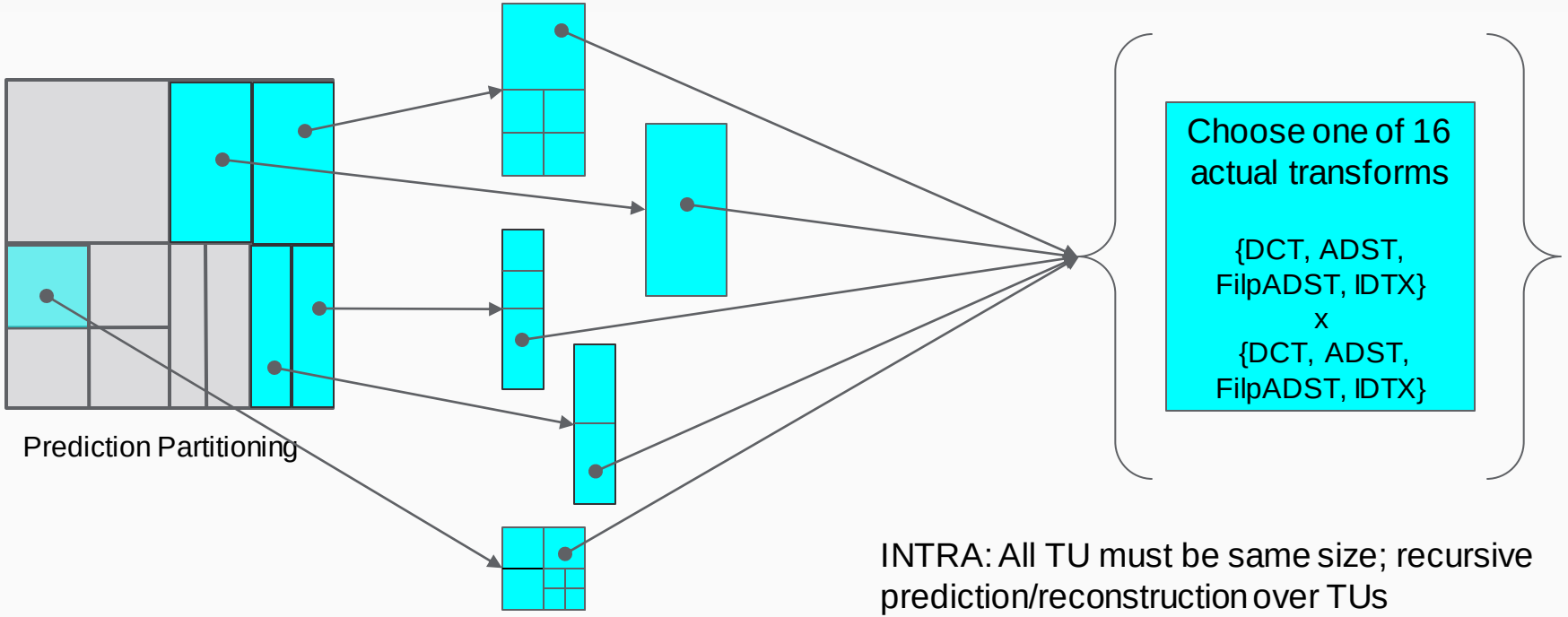
Coding Tools
Transform Partitions

AV1

64x64, 64x32, 32x64, 64x16, 16x64
32x32, 32x16, 16x32, 32x8, 8x32
16x16, 16x8, 8x16, 16x4, 4x16
8x8, 8x4, 4x8
4x4



Coding Tools
Transform Summary



INTRA: All TU must be same size; recursive prediction/reconstruction over TUs

Coding Tools
Coeff Coding

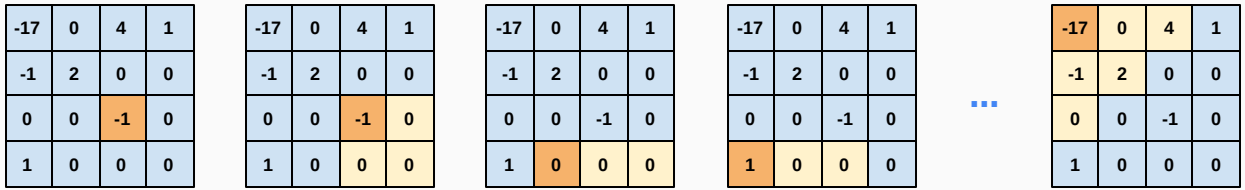
zig-zag scan

0	1	5	6
2	4	7	12
3	8	11	13
9	10	14	15

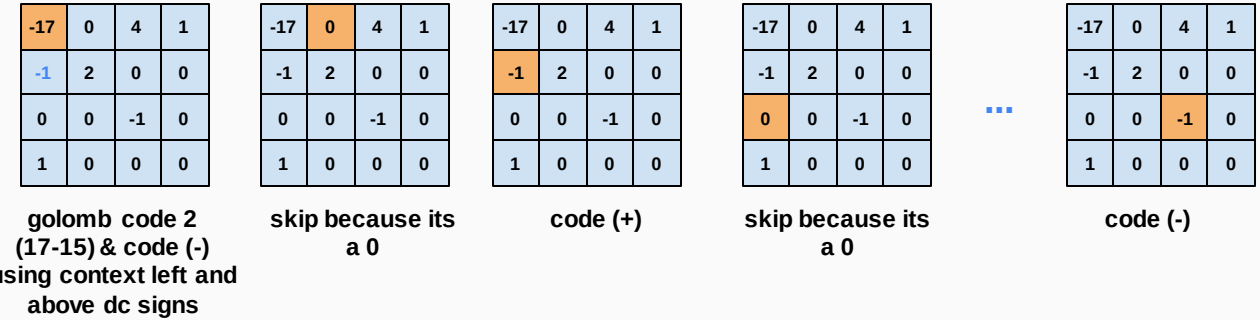
TX coeffs

-17	0	4	1
-1	2	0	0
0	0	-1	0
1	0	0	0

Encoding process

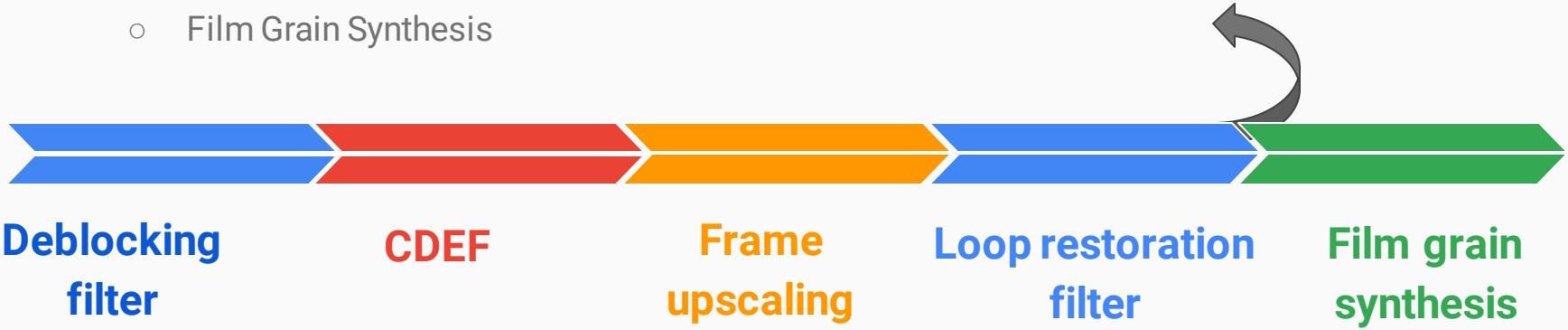


Reverse zigzag scan with moving spatial context



Coding Tools
In-loop filtering and postfilter

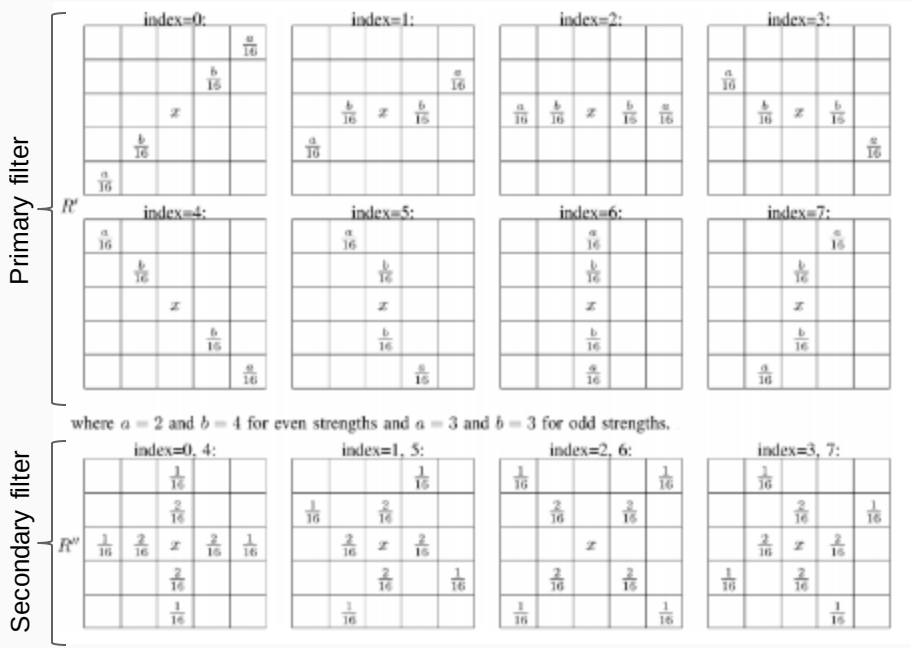
- AV1 pushes the envelope in in-loop filtering
  - Deblocking
  - CDEF (Constrained Directional Enhancement Filter)
  - Upscaling
  - Loop-restoration
- Also adds an out-of-loop filter
  - Film Grain Synthesis



Coding Tools - CDEF

In-loop filtering / postfilter - CDEF

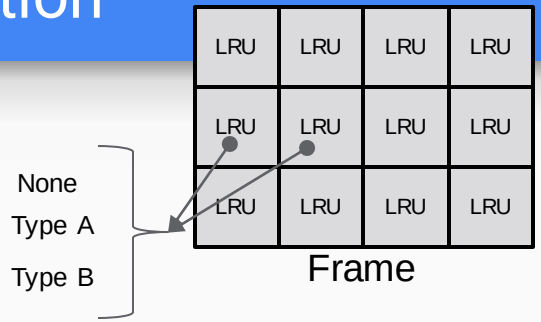
- CDEF:
  - Combination of Daala Deringing filter and Cisco's Constrained Low Pass Filter
  - Adapts filtering to direction of edges and patterns in an image
- Direction estimation:
  - Conducted at 8x8 level by minimizing variances along predefined lines
- Use a nonlinear directional filter
  - 5x5 support region
  - Combined Primary + Secondary filter



Coding Tools

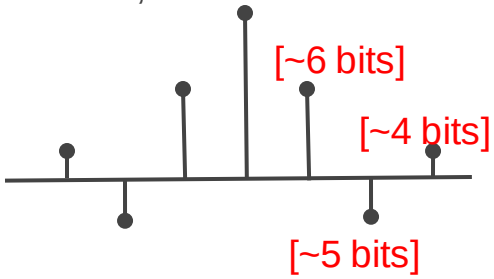
In-loop filtering / postfilter - Loop Restoration

- Switchable restoration filter in blocks referred to as Loop-restoration unit (LRU)
  - Filter type and corresponding parameters



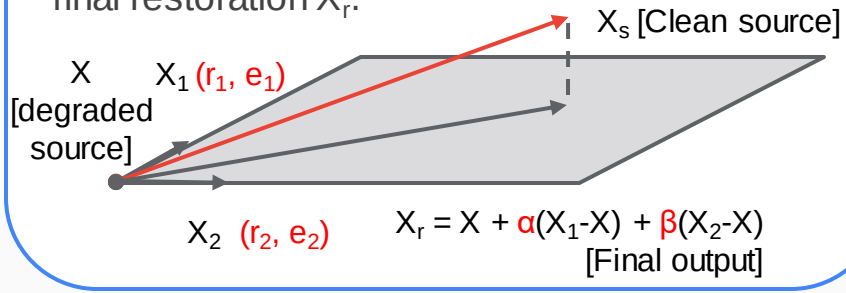
Type A: Wiener filter

Separable (horz + vert filter) 7-tap, symmetric, normalized



Type B: Dual Self-guided filters

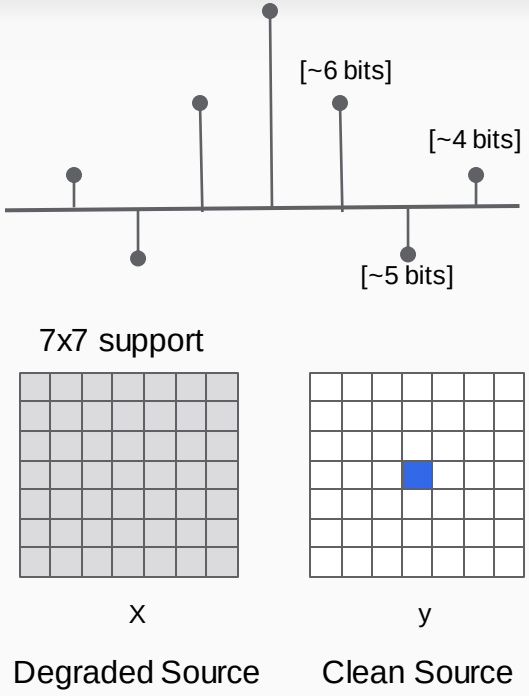
X_1 and X_2 are cheap restored versions. Subspace projection can yield a much better final restoration X_r.



Coding Tools

In-loop filtering / postfilter - Loop Restoration - Wiener

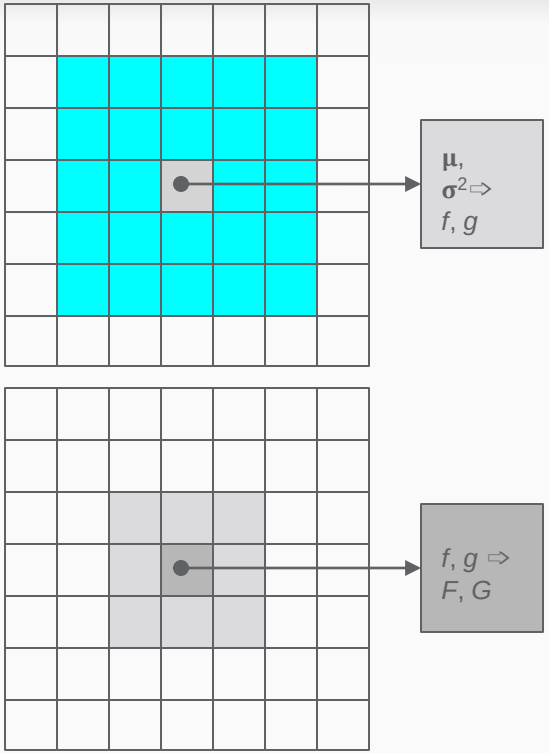
- Separable Symmetric Normalized Wiener filter:
  - 7-tap horizontal and 7-tap vertical separable
  - Symmetric, Normalized
  - Decoder complexity is kept small
- Encoder:
  - Design filters s.t. when applied to a deblocked frame produces an output closer to source
  - Design:
    - Classical non-separable LMMSE design: $H = R_{xx}^{-1} R_{xy}$ not possible
    - Impose separability, symmetry, normalization
    - Iterative design for hor/vert filters



Coding Tools

In-loop filtering / postfilter - Loop Restoration - Guided

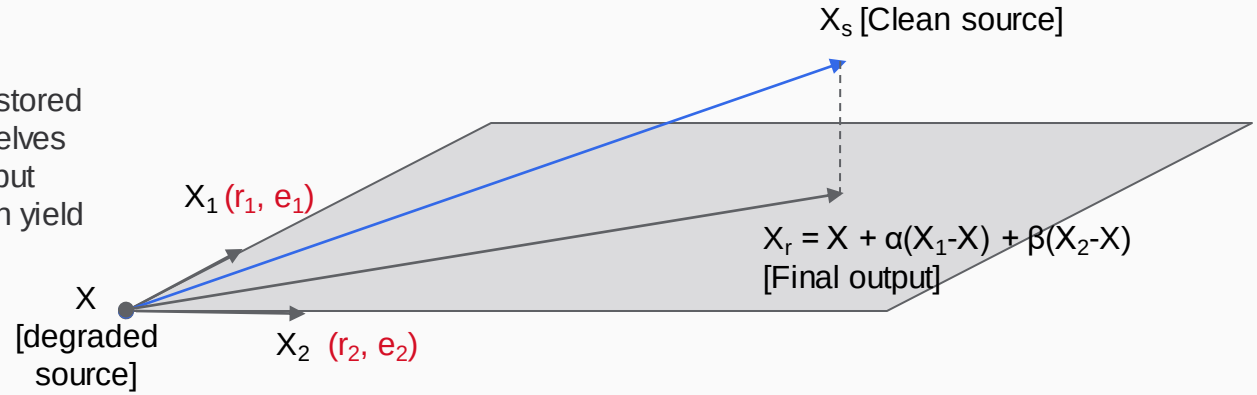
- Guided image filtering [2010]
  - Fit a local linear model considering a guide image
    - Self-guided: Guide image = Input Image
  - Can be used as an edge-preserving smoother
- Steps with parameters (r, e) in AV1:
  - Obtain mean μ and variance σ^2 of pixels in a $(2r + 1) \times (2r + 1)$ window around every pixel.
  - Compute: $f = \sigma^2/(\sigma^2 + e)$; $g = (1 - f)\mu$
  - Compute F, G as average of f, g in a 3x3 area around a pixel.
  - Filter pixel value using: $y = x.F + G$
  - Efficient implementation: readily integerized / vectorized



Coding Tools
In-loop filtering / postfilter - Loop Restoration - Guided

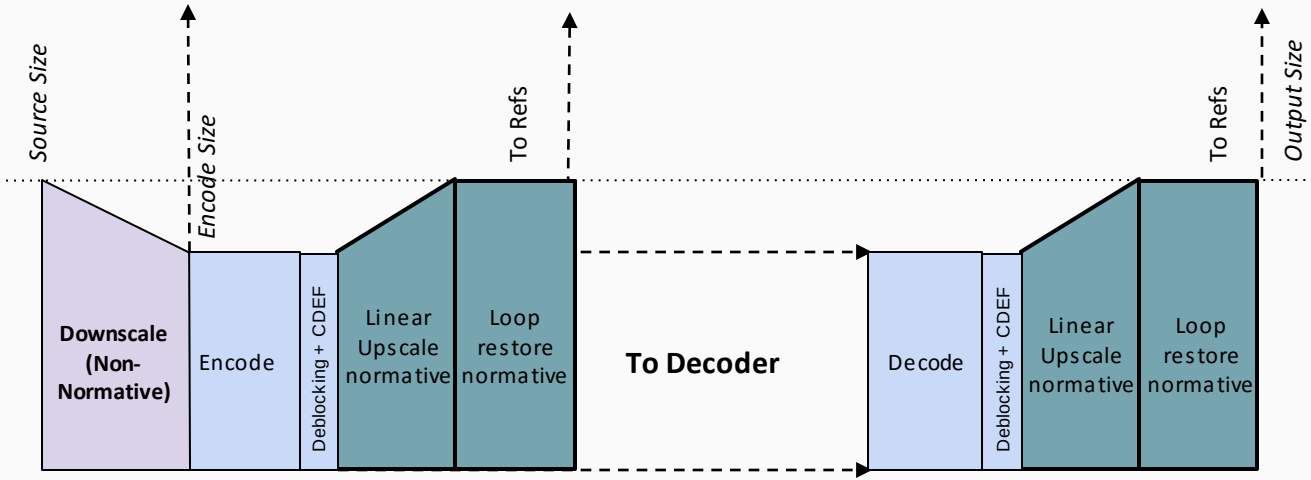
- Dual Self-guided filter with subspace projection in AV1:
  - Filter twice: Once with (r_1, e_1) yielding X_1, and once again with (r_2, e_2) yielding X_2
  - Use weighted combination w/ weights $\{\alpha, \beta\}$ of X_1, X_2, to get final output
  - (r_1, e_1, r_2, e_2) taken from a small 4-bit codebook ($r_1 = 1$; $r_2 = 2$);
  - $\{\alpha, \beta\}$ uses 7-bits each
  - ~18 total bits per RU

X_1 and X_2 are cheap restored versions, and by themselves may not be very good; but subspace projection can yield a much better final restoration X_r.



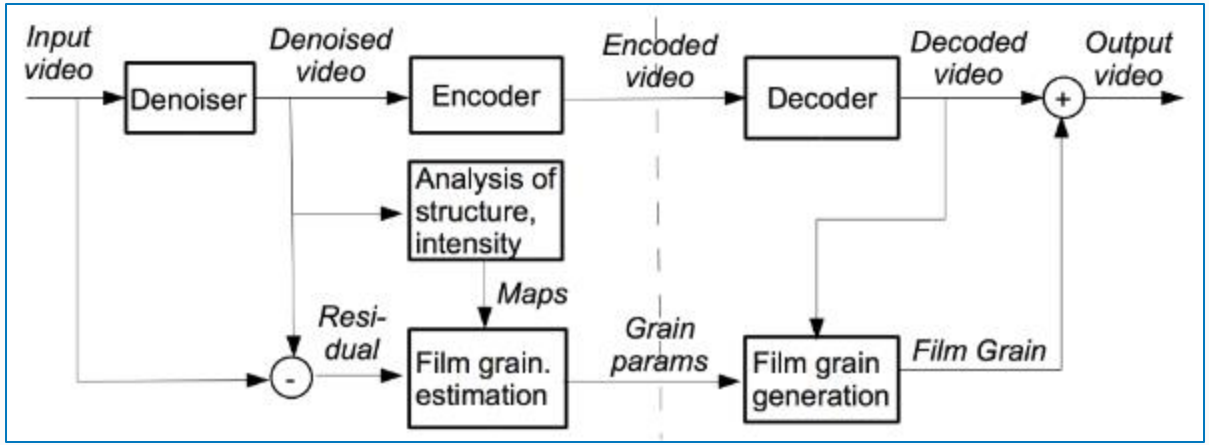
Coding Tools
In-loop filtering / postfilter - Frame superresolution

- AV1 supports predicting blocks from references at different resolutions
- Loop-restoration filter can super-resolve
- Frame-superresolution depends on the above two properties
  - Only horizontal upscaling supported to control line-buffer



Coding Tools
In-loop filtering / postfilter - Film Grain Synthesis

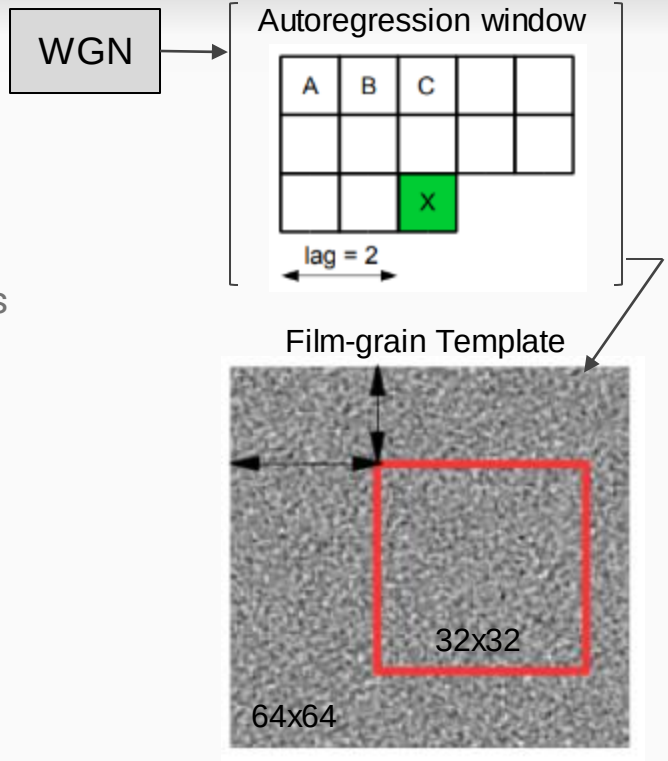
- Film-grain is hard to compress, but present in most commercial content.
  - Need to be preserved as part of creative intent
- AV1 supports film grain synthesis via a normative post-processing outside the coding loop.



Coding Tools

In-loop filtering / postfilter - Film Grain Synthesis

- Generate WGN sequence
  - Generated by taking random samples from a 2048-ary predefined Gaussian noise sequence
- Generate Film-grain template
  - Filter AWGN sequence with autoregressive parameters sent in the bitstream
- For each 32x32 block:
  - Take pseudo random offsets from the Film-grain template
  - Scale based on intensity of reconstruction using piecewise Lut specified in the bitstream
  - Add scaled grain to reconstruction and clip
  - 2-line overlapped grain addition to prevent sudden changes in grain



Outline

- Alliance for Open Media and AV1
- Coding Tools
- Latest Coding Results
- AV1 Deployment
- Conclusion

Latest Coding Results
AWCY

- www.arewecompressedyet.com
  - Official test infrastructure for AV1
  - Test set - Objective-1-fast, constant quality encoding, 60 frames, resolutions 360p-1080p
- Compare VP9 vs. HEVC vs. AV1
  - VP9 - tip-of-tree Oct 6, 2018, hash 4a47ef814b57d16787e6331e4ac1bd9dc093459e
  - AV1 - tip-of-tree Oct 6, 2018, hash 9b21428c86af1c081ae87cc546a360eaefa8ba8a
  - HEVC - X.265 hash e293b13373b72dfd88a91e196fbb595b027da3a3; HM 16.17

Latest Coding Results

Constant Quality BDRATE - AWCY Objective-1-Fast

- VP9 vs. X.265 vs. AV1
  - Testset: Objective-1-Fast on www.arewecompressedyet.com;
    - 60 frames, single keyframe, constant quality
  - Baseline: VP9 (libvpx); Tests: X.265 (--preset placebo --no-wpp --tune psnr), AV1 (libaom)

Codec	PSNR-Y	PSNR-HVS	SSIM	CIEDE2000	PSNR-Cb	PSNR-Cr
X.265	-4.04	-1.35	+7.34	+10.76	+20.51	+19.70
AV1 (cpu-used=0)	-30.16	-29.70	-30.46	-31.85	-33.01	-34.67
AV1 (cpu-used=1)	-28.75	-28.40	-29.21	-30.94	-32.76	-34.16
AV1 (cpu-used=2)	-26.10	-25.68	-26.67	-28.54	-30.61	-31.77

Latest Coding Results

Constant Quality BDRATE - AWCY Objective-1-Fast

- After recent VP9 improvements with pyramid structure
- Baseline VP9* (--auto-alt-ref=5); Tests: X.265, AV1

Codec	PSNR-Y	PSNR-HVS	SSIM	CIEDE2000	PSNR-Cb	PSNR-Cr
X.265	-1.60	+1.92	+7.74	+10.12	+18.20	+17.21
AV1 (cpu-used=0)	-27.74	-26.88	-29.28	-31.68	-33.58	-35.25
AV1 (cpu-used=1)	-26.31	-25.55	-28.03	-30.78	-33.28	-34.76
AV1 (cpu-used=2)	-23.55	-22.69	-25.41	-28.33	-31.11	-32.31

Latest Coding Results

Constant Quality BDRATE - Google testsets

- HM16.17 vs. VP9 vs. AV1
  - Testset Google's internal (Lowres, Midres, Hdres), 120 frames
    - Single keyframe, constant quality mode
  - Baseline: HM16.17 (-ip -1 --ConformanceWindowMode=1); Tests: VP9 (libvpx), AV1 (libaom)

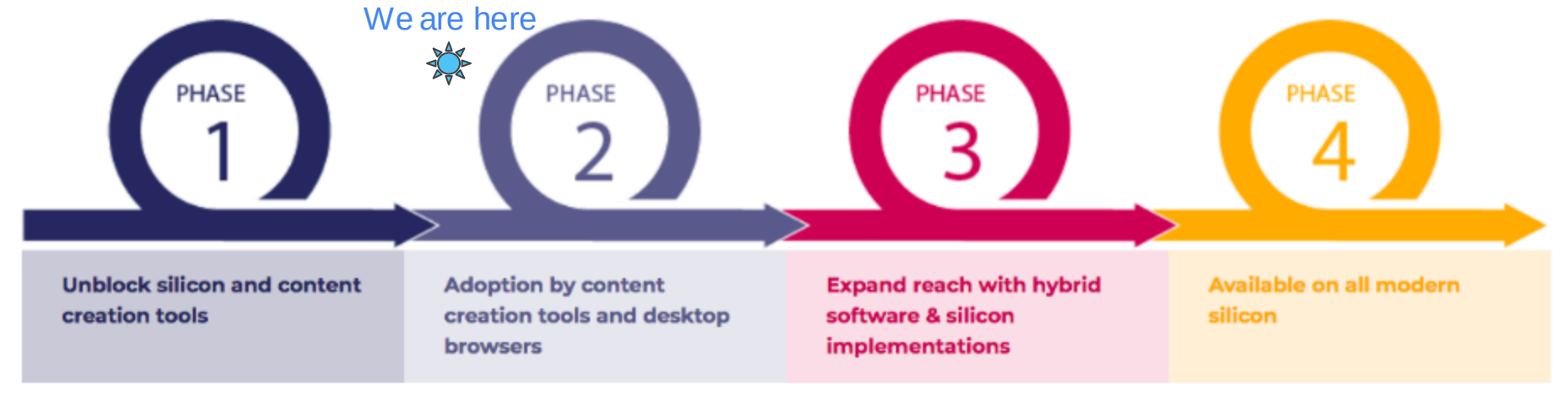
	Av. PSNR	SSIM
Lowres (120 frames), VP9	+10.448%	+5.437%
Lowres (120 frames), AV1	-19.578%	-22.703%
Midres (120 frames), VP9	+12.215%	+7.477%
Midres (120 frames), AV1	-19.840%	-23.015%
Hdres (120 frames), VP9	+7.637%	+5.240%
Hdres (120 frames), AV1	-23.979%	-26.152%

Outline

- Alliance for Open Media and AV1
- Coding Tools
- Latest Coding Results
- AV1 Deployment
- Conclusion

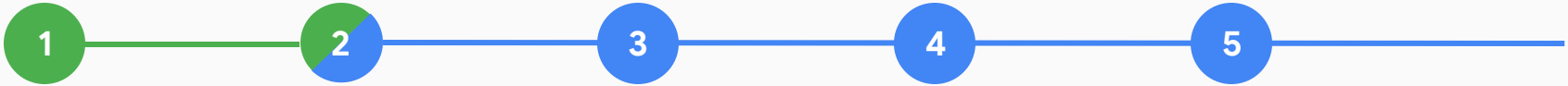
AV1 Deployment
Phases

- Codec Deployment Phases
  - Bit-stream freeze is simply the first phase



AV1 Deployment
Rollout plans in Google

- AV1 Rollout plans in Google



Spec Published

(Completed)

Chrome Support

(Q4 2018)

WebRTC Integration

(H2 2019)

Android Q Support

(Q3 2019)

SoC Hardware

(2020)

AV1 Deployment
Software speed-ups

- Early deployment of AV1 to depend on SW decode and encode
- SW Encoder speed-up
  - Encoder was very slow 6 months ago 1000x of VP9 cpu-used=0
  - Substantially faster today
    - AV1 cpu-used=0 (libaom): 40x of VP9 encode time at cpu-used=0 (libvpx)
    - AV1 cpu-used=1 (libaom): 16x of VP9 encode time at cpu-used=0 (libvpx) +1.5% PSNR
    - AV1 cpu-used=2 (libaom): 10x of VP9 encode time at cpu-used=0 (libvpx) +4.0% PSNR
  - Extensive use of machine learning based mode/partition decisions, early terminations, etc.
- SW Decoder speed-up
  - SIMD optimizations
  - Today 3.4x of VP9 decoder (single-threaded)
- Ongoing work … more speed-ups expected by end of the year.

AV1 Deployment
Quality Improvements

- Improvement in frame management
  - Better hierarchical arrangement and quality control
  - 2+% gain beyond the bit-stream freeze.
- Adaptive quantization
  - Perceptual quality improvement - validated with perceptual tests
- Forward reference keyframe implementation
  - Substantial improvement in short key-frame interval scenarios for trick-play
  - 4+% improvement for 30-frame key frame interval.
- Coming:
  - Intelligent use of superresolution mode
    - Already shows gains at low bitrate for key-frames.

Outline

- Alliance for Open Media and AV1
- Coding Tools
- Latest Coding Results
- AV1 Deployment
- Conclusion

Conclusion

- AV1 is undoubtedly the most advanced video codec in the world today
  - ~30% better than VP9, and no slower than ~3x of VP9 decoding
  - ~20% better than HEVC
- May not remain the best in a couple of years - with VVC coming
  - A 2-horse race?
- Advancement in codec technology will continue
  - AV2

Questions

Questions ?