Emerging AV1 Video Codec - Novel Prediction, Transform, and Image Restoration Tools

Zoe Liu (zoeliu@google.com)
Google Inc.

Google

The story continues with AV1

Alliance for Open Media

Alliance for Open Media (AOM) is formed in 2015 Q4

AV1

AOMedia Video 1

THE OPEN AND ROYALTY-FREE CODEC FOR NEXT-GENERATION ULTRA HIGH DEFINITION MEDIA

Alliance of Open Media - Members - 2017

Alliance for Open Media

- Adobe
- Allegro
- amazon
- AMD
- Amlogic
- ARGON DESIGN
- ateme
- ARM
- BBC Research & Development
- BITMOVIN
- BROADCOM
- Chips&Media
- CISCO
- Google
- hulu
- IBM
- intel
- Ittiam
- Microsoft
- mozilla
- NETFLIX
- NGCODEC
- NVIDIA
- Polycom
- REALTEK
- SIGMA DESIGNS
- socionext
- VeriSilicon
- Vidyo
- VideoLAN
- XILINX ALL PROGRAMMABLE

Accessing AV1 Source Code

Alliance for Open Media

- AOMedia git repository:
  git clone https://aomedia.googlesource.com/aom

New Coding Tools in AV1

Alliance for Open Media

- Prediction tools
- Transform tools
- In-loop filtering tools
- Miscellaneous tools

AV1 Technical Overview: Prediction Tools:

Partition Expansion

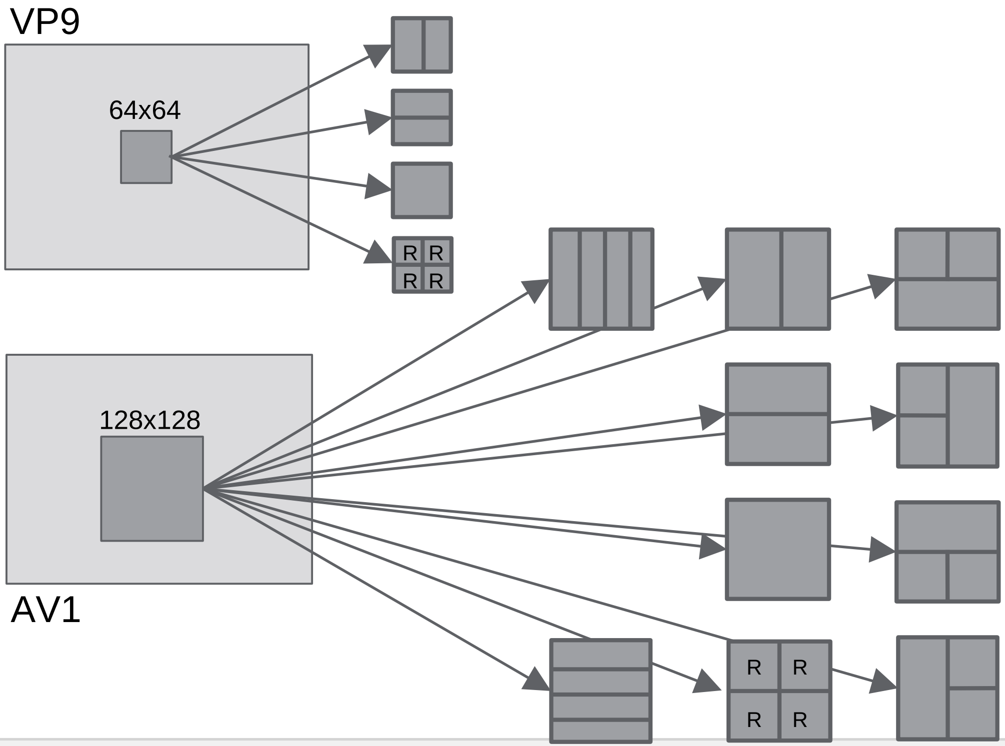
Partition structure

VP9:

- 4-way partition tree
- 64x64 down to 4x4
- Sub 8x8 constraints

AV1

- Expands to a 10-way partition tree
- *Larger superblock size (128x128)*
- More flexibility for 4x4 blocks



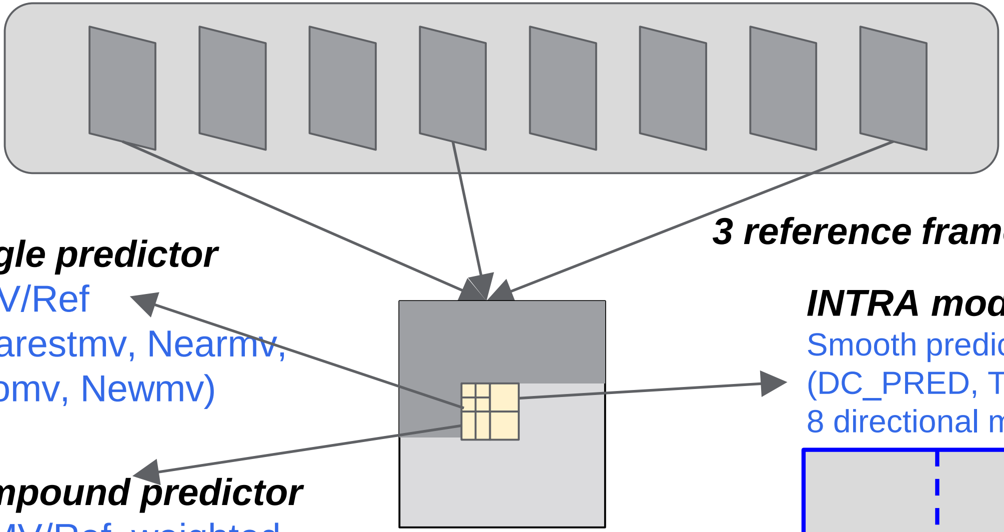
AV1 Technical Overview: Prediction Tools:
Partition Expansion

- AV1 adds more flexibility to sub-8x8 blocks from VP9
- VP9 restrictions for 4x4:
  - References and interpolation filters combined at 8x8 level
  - INTER / INTRA mixing disallowed at sub-8x8 level
- AV1 flexibilities:
  - Allow 2x2 chroma inter prediction
    - Smallest transform is still 4x4
  - INTER / INTRA mixing possible for Luma, not for Chroma
    - For chroma use mode for the bottom-right Luma
  - Compound modes disabled for sub-8x8 blocks
    - Keeps the same hardware throughput as VP9

AV1 Technical Overview: Prediction Tools

VP9 Refresher

***Reference frame store*:**



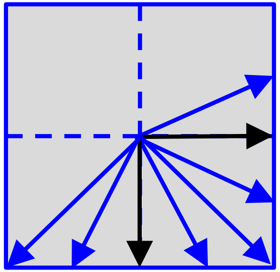
3 reference frames

***INTER modes*:**
Motion compensated prediction from previously encoded frame buffers
- 3 sub-pel filters
- Up to ⅛ pel precision

Single predictor
1 MV/Ref
(Nearestmv, Nearmv, Zeromv, Newmv)

Compound predictor
2+ MV/Ref, weighted combination (½ - ½)
(Nearestmv, Nearmv, Zeromv, Newmv)

***INTRA modes*:**
Smooth prediction modes (DC_PRED, TM_PRED), 8 directional modes



AV1 Technical Overview: Prediction Tools:
INTRA Prediction - Directional Modes

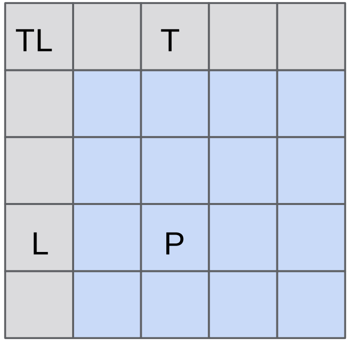
- Directional Modes
  - Keep 8 major directions as in VP9
  - Finer-directions added as delta from the major ones
    - Up to 56 directions overall
    - Depends on block size

AV1 Technical Overview: Prediction Tools:
INTRA Prediction - Non-directional Modes

- Paeth Mode
  - Replaces TM mode in VP9

TM Mode: $P_{TM} = L + T - TL$

Paeth Mode: $P_{Paeth} = \text{argmin}\ |x - P_{TM}|$, over $x \in \{L, T, TL\}$



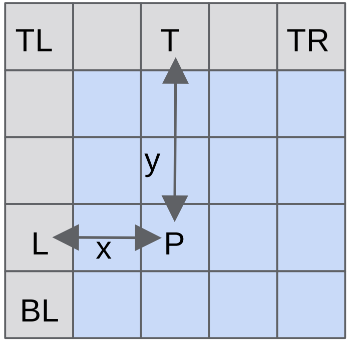
AV1 Technical Overview: Prediction Tools:
INTRA Prediction - Non-directional Modes

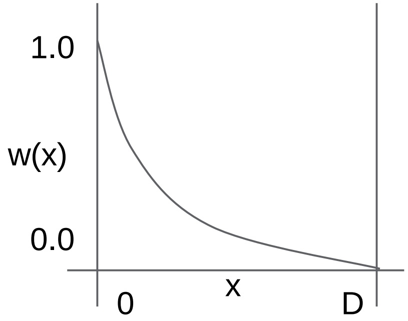
- Smooth Modes
  - Three modes
    - Horizontal, Vertical and Bi-directional

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

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

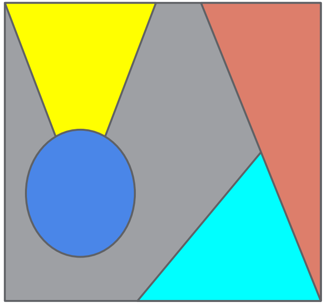
SMOOTH: $P_{SMOOTH\_V} = ½\ (P_{SMOOTH\_H} + P_{SMOOTH\_V})$

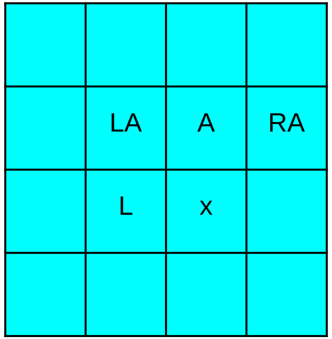




AV1 Technical Overview: Prediction Tools:
INTRA Prediction - Palette mode

- Particularly useful for screen content
- Palette representation:
  - Encode k distinct *base* colors: $k \in \{2, 3, \ldots, 8\}$,
  - Encode a k-ary color index map
- Base colors:
  - Delta encoded w/ colors from causal neighbor blocks
- Color index map:
  - Use causal neighbor pixels within block as context
- Regarded as a new INTRA prediction mode

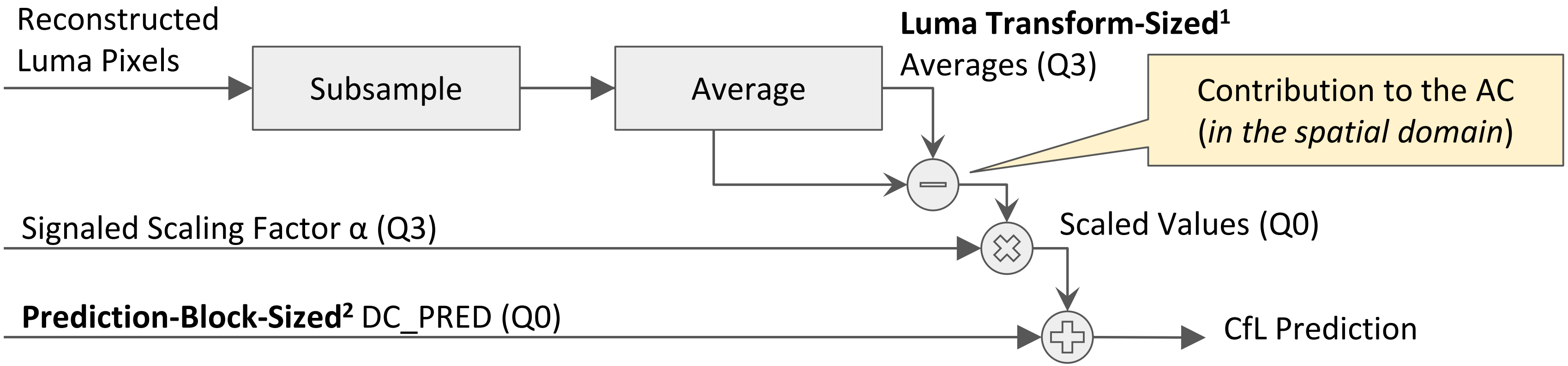




AV1 Technical Overview: Prediction Tools:

INTRA Prediction - Chroma from Luma [Mozilla]

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



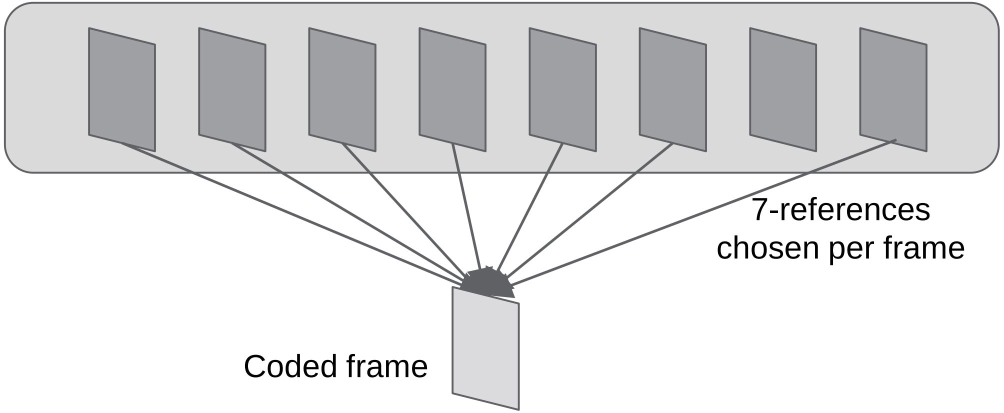
α_{Cb}, α_{Cr} signaled in bit-stream

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

AV1 Technical Overview: Prediction Tools:

INTER Prediction: Reference Expansion

- Number of available references for each frame:
  - Expanded to 7 (from 3 in VP9) out of 8 in frame store
- Provides more flexibility to establish a hierarchical structure
  - Combination of *invisible* reference frames and traditional B-like frames



AV1 Technical Overview: Prediction Tools:

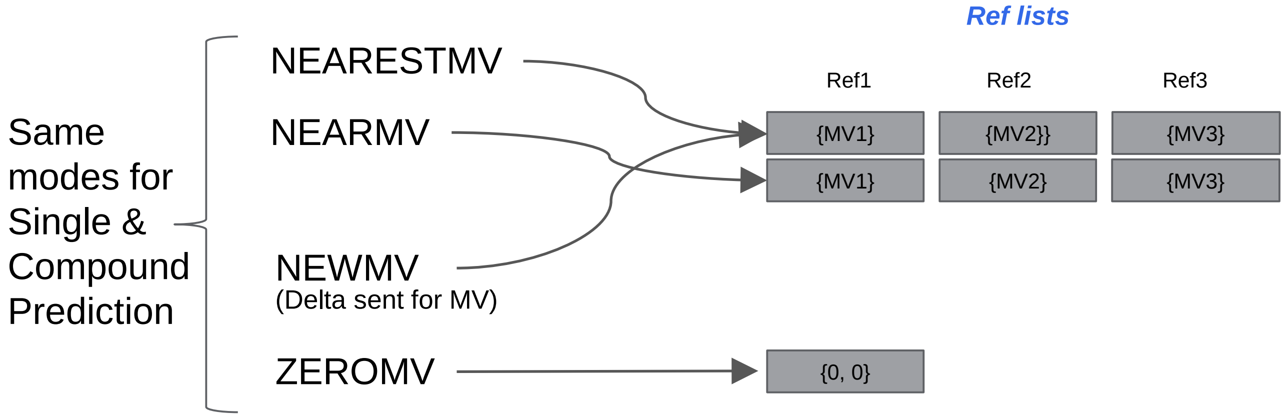
INTER Prediction: MV Referencing

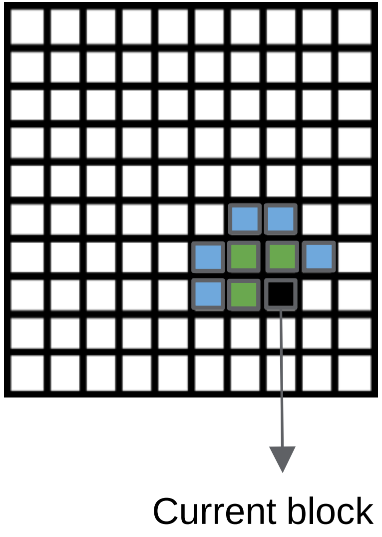
- In modern codecs, motion vectors comprise a fairly large portion of the bitrate
  - Better MV prediction essential for good compression efficiency

AV1 Technical Overview: Prediction Tools:

INTER Prediction: MV Referencing in VP9

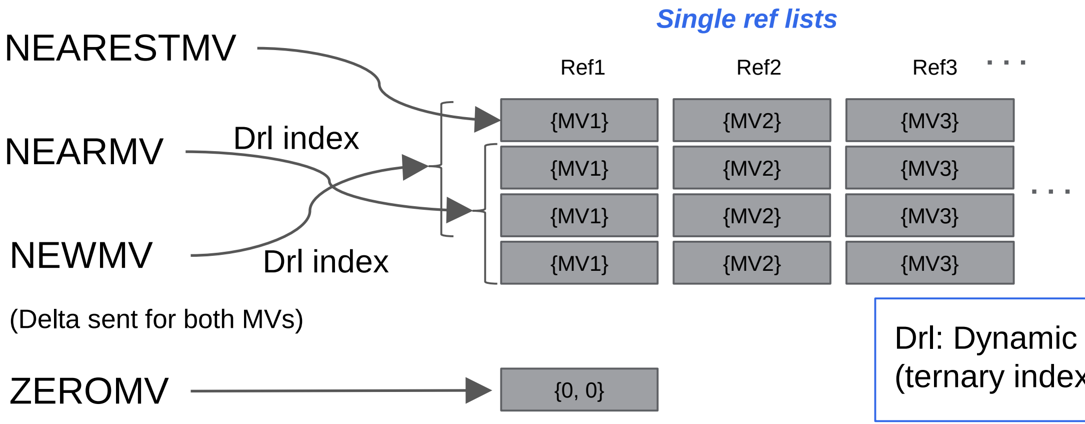
- VP9 scheme:
  - Survey MVs in causal neighborhood of current frame & temporally collocated MV in previous frame
  - Make ranked list of 2 MVs for each reference
  - Send one of 4 modes:





AV1 Technical Overview: Prediction Tools:
INTER Prediction: Dynamic MV Referencing in AV1

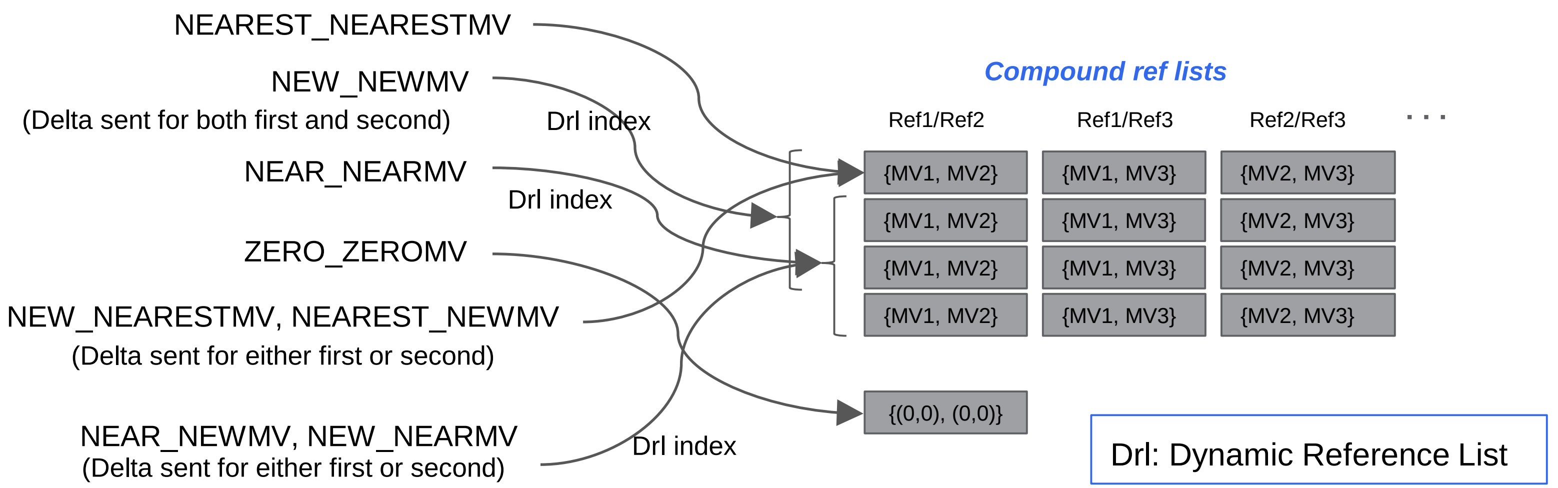
- AV1 scheme:
  - Search range expanded
  - Make ranked list of 4 candidate MVs for Single Reference
  - Same 4 modes used for single reference



Drl: Dynamic Reference List (ternary index)

AV1 Technical Overview: Prediction Tools:
INTER Prediction: Dynamic MV Referencing in AV1

- AV1 Scheme (cont'd):
  - Compound modes use separate pair-wise ranked lists
  - Compound modes expanded to 8

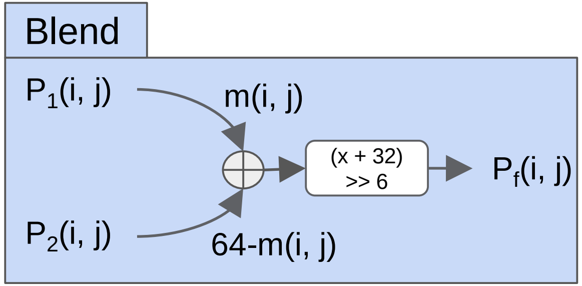


AV1 Technical Overview: Prediction Tools:
INTER Prediction: Extended Compound Modes

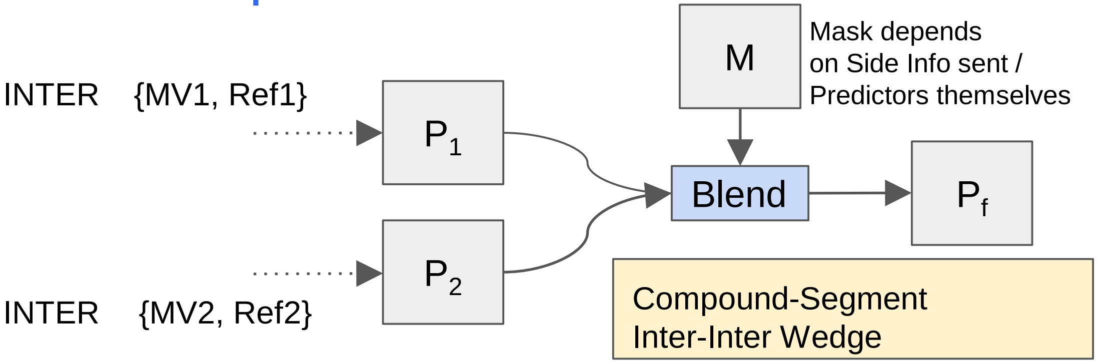
- VP9 Compound modes
  - Only allows ½, ½ combination of two predictors
  - Same-sided compound modes not allowed
- AV1 Compound Modes
  - Allow compound from two predictors on the same side
  - Allow flexible blending of two predictors:
    - INTER-INTER
      - Compound Segment
      - Wedge Inter-Inter
    - INTER-INTRA
      - Gradual
      - Wedge Inter-Intra

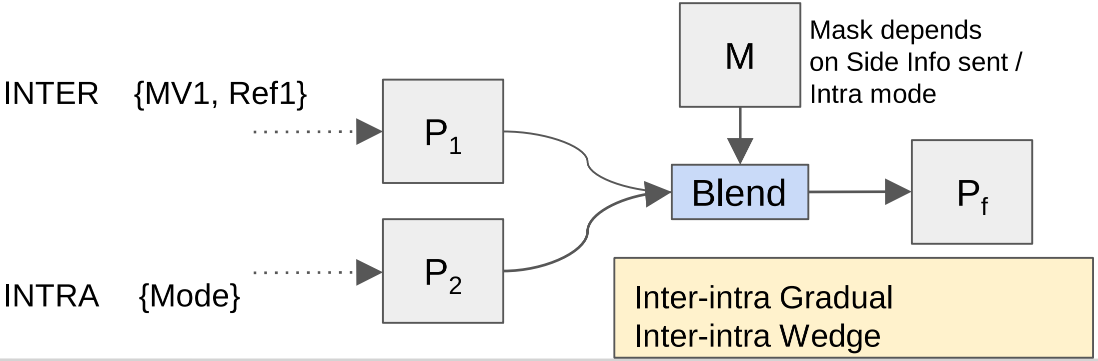
AV1 Technical Overview: Prediction Tools:
INTER Prediction: Extended Compound Modes

Per-pixel blending



Mask $m(i, j) \in [0, 64]$
(integerized)



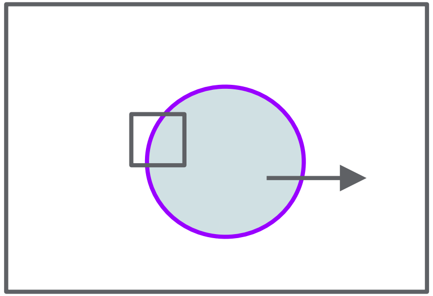


AV1 Technical Overview: Prediction Tools:

INTER Prediction: Interinter Compound Segment Mode

- Difference modulated mask generation
  - Pixels with a larger difference between two Inter predictor candidates get a larger blending mask differential between one predictor vs. the other
    - $m(i, j) = 38 + |P_1(i, j) - P_2(i, j)| >> 4$
  - Send an additional bit to indicate whether to switch P_1 and P_2

AV1 Technical Overview: Prediction Tools:
INTER Prediction: Interinter Wedge Mode

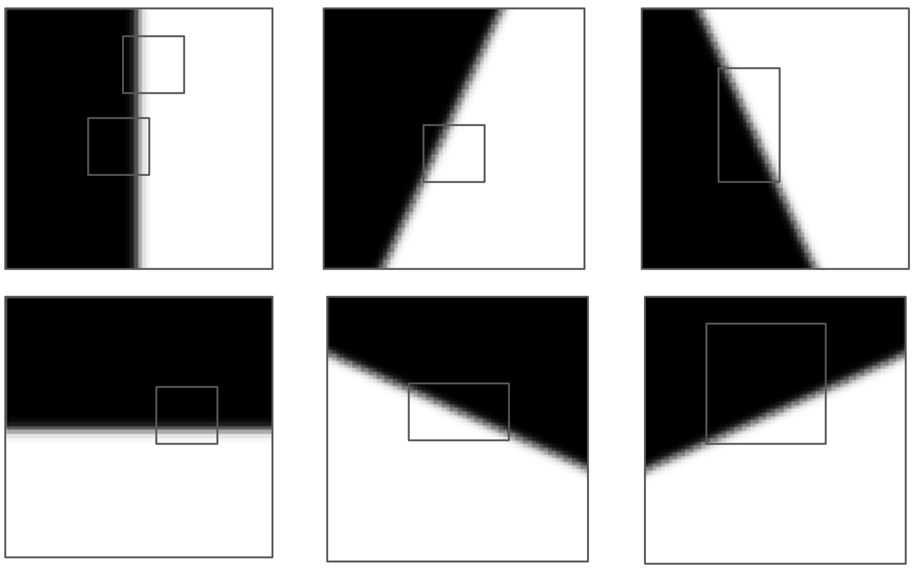


- Blocks can be on boundary of moving objects.
    - Boundary not necessarily horizontal or vertical
    - Going down to too small blocks not efficient.
- One Approach:
    - Allow blocks to have arbitrary partitions.
    - Predict each side with a different MV/Ref combination.
    - Problem:
        - Inexact partitioning introduces spurious high frequencies
- Solution:
    - Soft-partitioning with a cliff-like mask for blending - wedge
    - Only a few masks are enough and can be compactly conveyed

AV1 Technical Overview: Prediction Tools:

INTER Prediction: Interinter Wedge Mode

- Wedge mask codebook
  - Send index from codebook
    - Fully determines the mask
- Codebook generation
  - Conveniently generated for each block-size from three length-64 arrays row-shifting
    - On-the-fly generation
- *Overlapped wedge motion compensation*



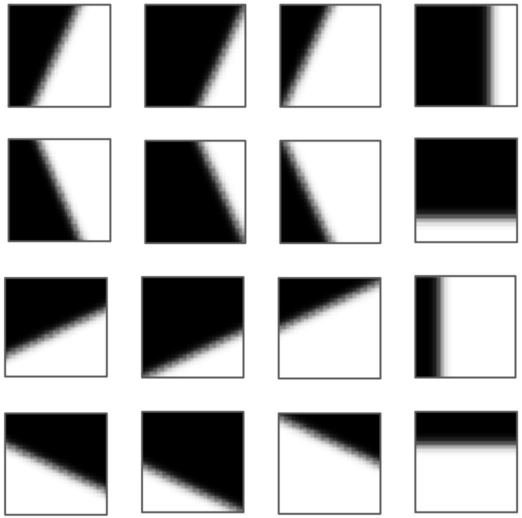
AV1 Technical Overview: Prediction Tools:

INTER Prediction: Interinter Wedge Mode

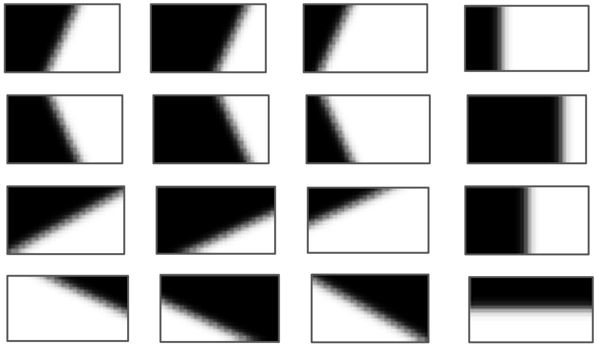
Wedge codebook:

Inter-Inter
4-bit shape
1-bit sign

5 bits total



Square codebook

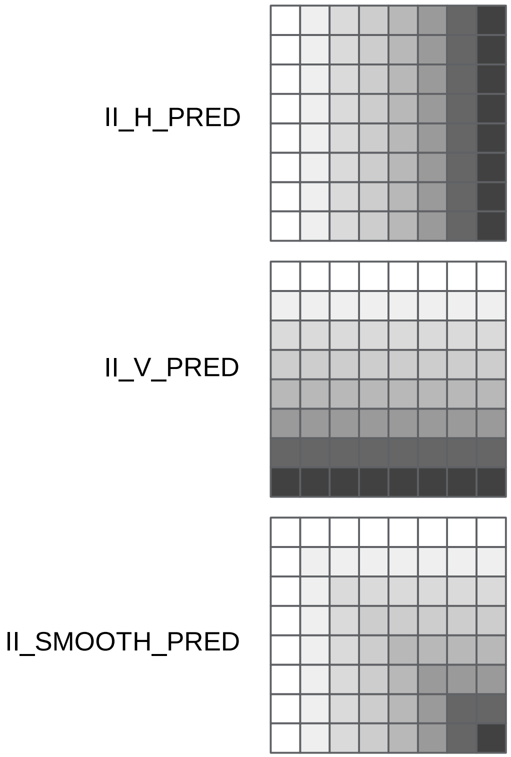


Rectangular codebook

AV1 Technical Overview: Prediction Tools:

INTER Prediction: InterIntra Gradual Modes

- Interintra - compound between Inter and Intra
- Interintra Gradual Modes
  - Decay weight gradually for Intra from prediction boundary - increase weight for Inter correspondingly
  - Four modes supported:
    - II_H_PRED
      - Use Hor Intra pred - decay weight horizontally
    - II_V_PRED
      - Use Ver Intra pred - decay weight vertically
    - II_SMOOTH_PRED
      - Use SMOOTH Intra pred - decay weight based on distance from closest left or top boundary
    - II_DC_PRED
      - Use DC Intra pred - simple average with Inter



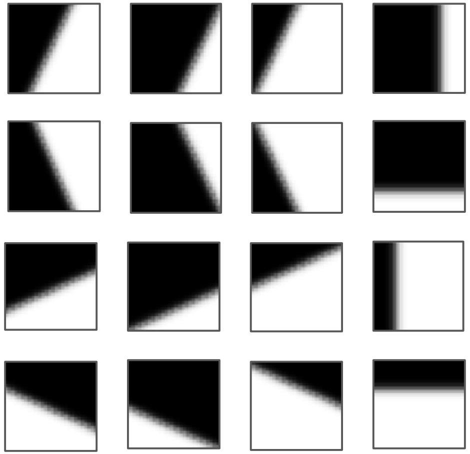
AV1 Technical Overview: Prediction Tools:

INTER Prediction: InterIntra Wedge Modes

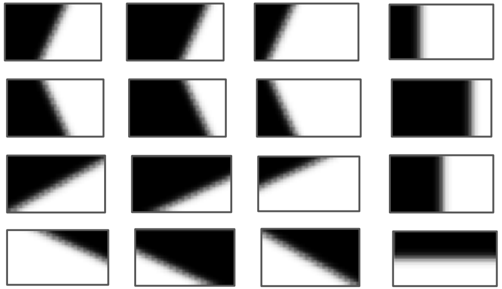
- Interintra Wedge Modes
  - One side of Wedge closer to top and left boundaries uses Intra
    The other side uses Inter - same wedge codebooks as Interinter

Wedge codebook:

Inter-Intra
4-bit shape
(For inter-intra wedge sign is fixed)



Square codebook

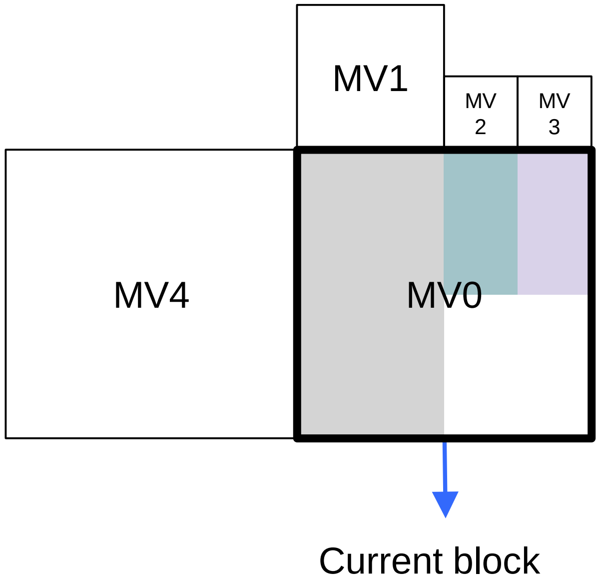


Rectangular codebook

AV1 Technical Overview: Prediction Tools:

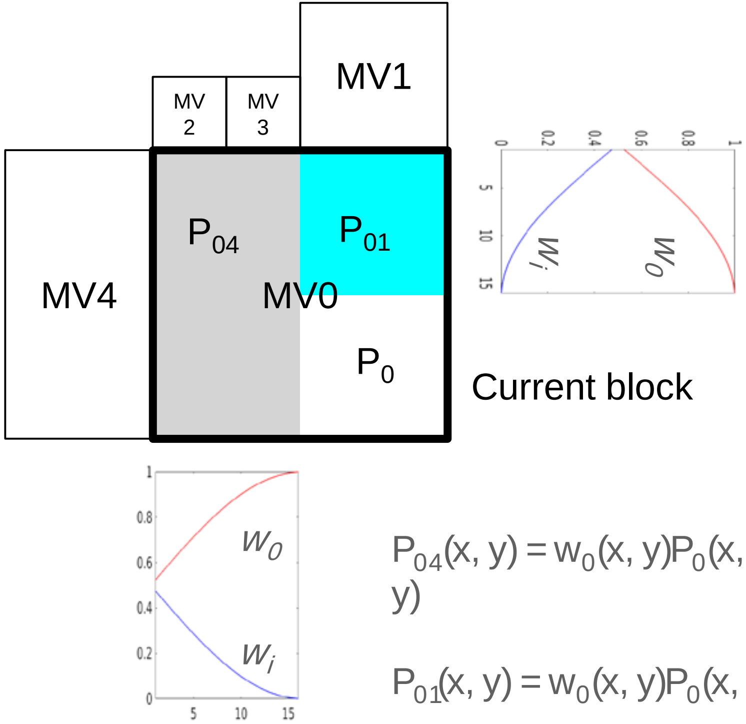
INTER Prediction: Overlapped Block Motion Compensation

- OBMC
  - Blend multiple predictors from neighboring blocks
  - Not a new concept
    - But H.263 in the 90s was the last codec to use it when block partitioning was much more straightforward
- AV1 reintroduces OBMC
  - Much harder to incorporate in a modern codec and still get gains



AV1 Technical Overview: Prediction Tools:
INTER Prediction: Overlapped Block Motion Compensation

- Causal OBMC mode
  - Blend multiple predictors using motion information from neighboring causal blocks on top or left.
  - 1-D Raised Cosine function for blending extending half-way to the width / height of current block B



$P_{04}(x, y) = w_0(x, y)P_0(x, y) + w_1(x, y)P_4(x, y)$

$P_{01}(x, y) = w_0(x, y)P_0(x, y) + w_1(x, y)P_1(x, y)$

AV1 Technical Overview: Prediction Tools:
INTER Prediction: Affine Motion

- True motion in a scene is never translational.
- Modern video codecs however use only translational models.
  - Flexibility in block size partitioning in modern codecs obviate the need for more complicated motion
    - But there is a cost for going down to too small blocks
  - Non-translational motion parameters are expensive to send
  - Warping is expensive in hardware and software
- To better the state-of-the-art we need to break this barrier
- AV1 does this by introducing two new coding modes
  - Global Affine
  - Local Affine

AV1 Technical Overview: Prediction Tools:
INTER Prediction: Affine Warp Implementation

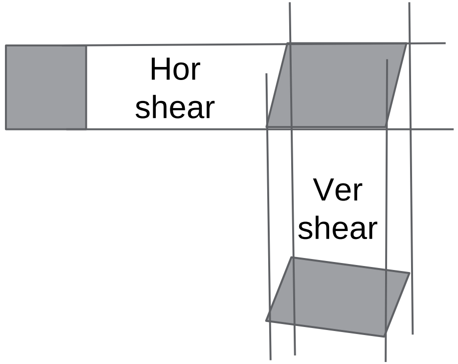
- AV1 - warped model
  - Limit to Affine
  - Only small motions allowed
- Small Affine Advantage:
  - Can be efficiently implemented in hardware / software with two shears just as in separable sub-pel interpolation.
  - 1/64th precision for shears

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

Affine model

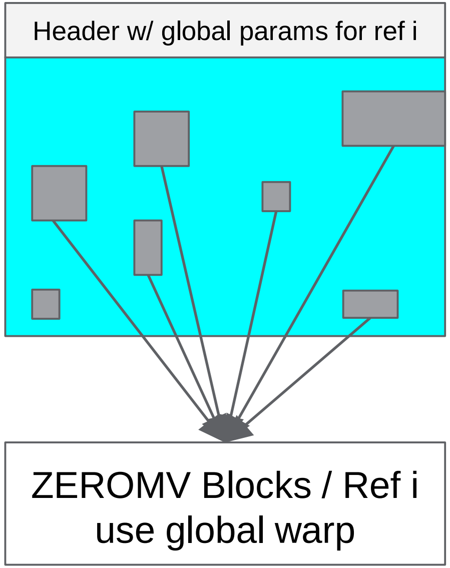
$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \gamma & 1+\Delta \end{bmatrix} \begin{bmatrix} 1+\alpha & \beta \\ 0 & 1 \end{bmatrix}$$

Original warping matrix — Vertical Shear — Horizontal Shear



AV1 Technical Overview: Prediction Tools:
INTER Prediction: Affine Motion: Global Motion Mode

- For each frame in its header for each reference, optionally send:
  - 6-parameter affine parameters
  - 4-parameter similarity parameters (rotation-zoom)
  - 2-parameter translational parameters
- Within the frame if ZEROMV is signaled for a prediction block with a reference that has global parameters, warped prediction is used
  - For sub-8x8 blocks translational motion is used where the block's MVs are computed using global parameters at its center



AV1 Technical Overview: Prediction Tools:

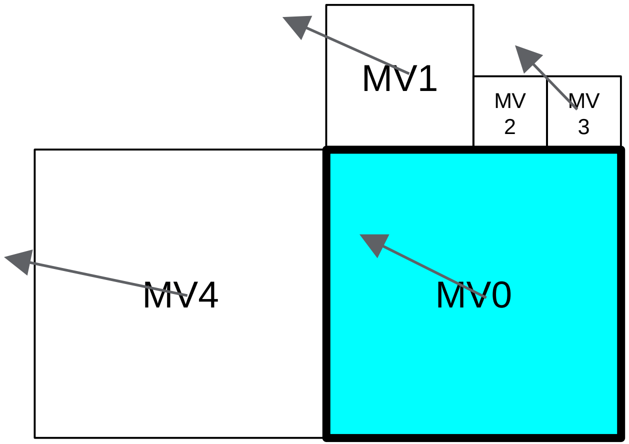
INTER Prediction: Affine Motion: Local Warped Motion Mode

- Global motion mode works great when there is a strong consistent camera motion in a frame
  - Motion in a scene may not be regular
- Ideally need to send multiple parameters within each frame
  - Too expensive
- Local Warped motion in AV1:
  - Estimate affine parameters from translational MVs in causal neighborhood of a block, as well as the translational MV transmitted for the current block.
  - Low-complexity least squares problem solved on decoder side.

AV1 Technical Overview: Prediction Tools:

INTER Prediction: Affine Motion: Local Warped Motion Mode

- Goal: Estimate affine parameters using neighboring blocks' MV with same reference.
- Affine least squares:
  - Can be always decomposed into two 3-dim least squares problems
  - MV transmitted is constrained to be consistent with model at the block center
  - This constraint converts each 3-dim problem into a simpler 2-dim problem that is readily integerized.



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

AV1 Technical Overview: Prediction Tools:

INTER Prediction: Sub-pel interpolation filter expansion

- VP9 interpolation filters:
    - 8-tap 3 choices: {REGULAR, SHARP, SMOOTH}
    - Signaled in the bit-stream
- AV1 interpolation filters:
    - Allow different filters horizontally and vertically
    - Expanded to {REGULAR, SHARP, SMOOTH} in each direction to make 9 filters overall
    - Signaled in the bitstream

AV1 Technical Overview: Prediction Tools:

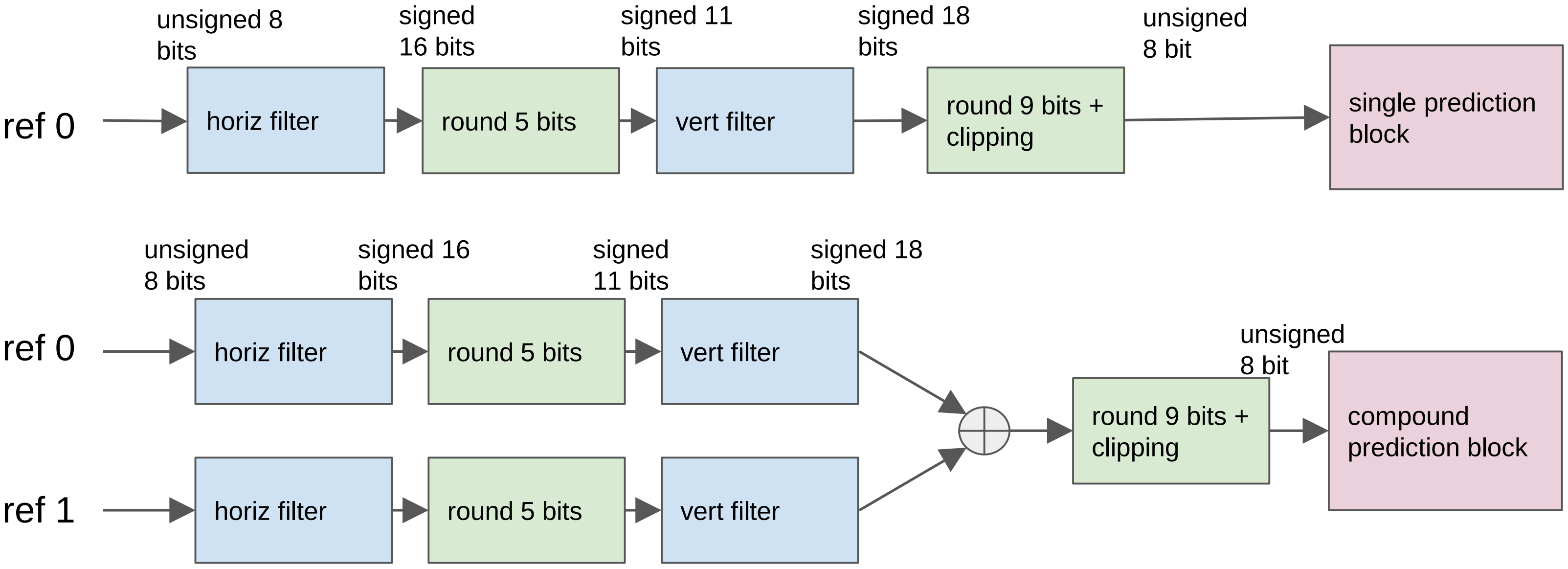
INTER Prediction: Sub-pel interpolation intermediate precision

- VP9 filtering precision:
  - Brings pixel precision after horizontal filtering down to pixel precision before vertical pass
  - Compound prediction:
    - Each predictor is maintained at pixel precision before averaging
- AV1 intermediate precision:
  - Maintains higher intermediate precision between horizontal and vertical filtering passes
  - Compound prediction:
    - Additionally maintain higher precision before averaging two predictors

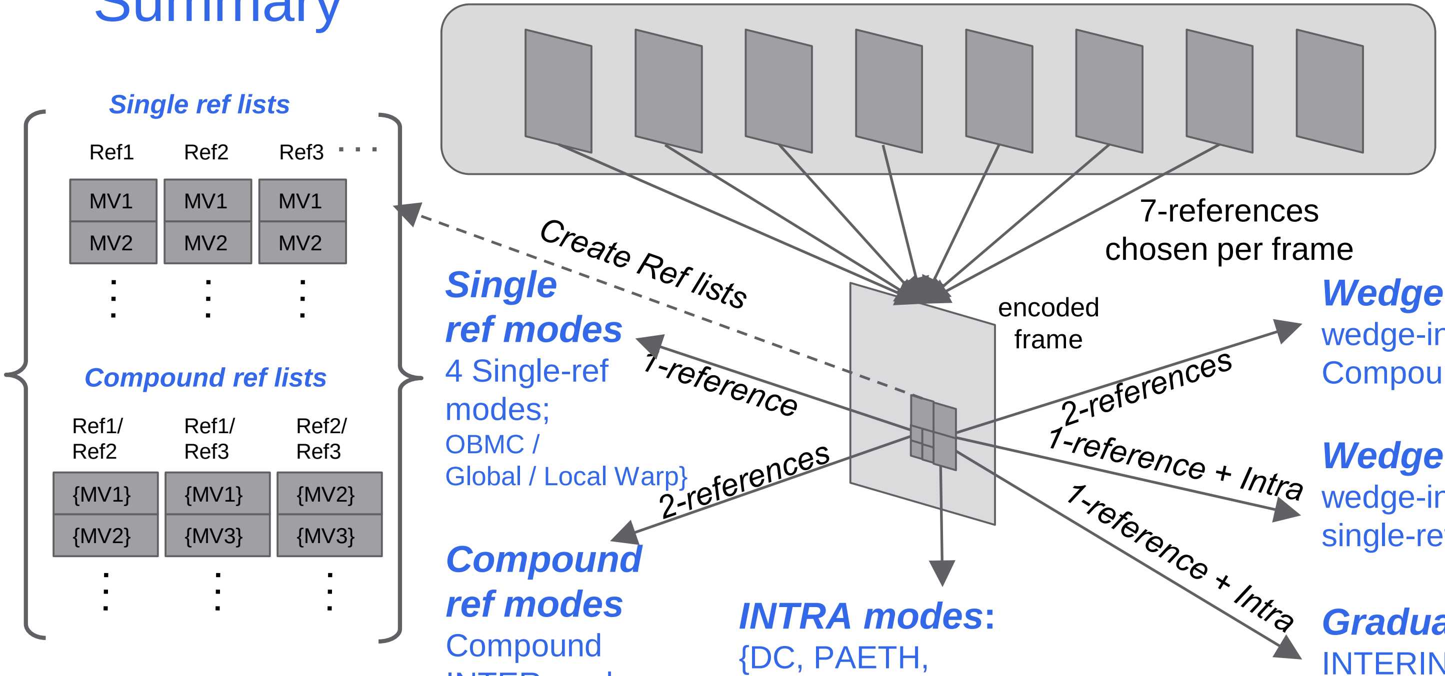
AV1 Technical Overview: Prediction Tools:

INTER Prediction: Sub-pel interpolation intermediate precision

Keep 3 extra bits between horizontal and vertical filters from source bit-depth
[NOTE: filters are 7 bits]



AV1 Technical Overview: Prediction Tools:
Summary



Single ref modes
4 Single-ref modes;
OBMC / Global / Local Warp}

Compound ref modes
Compound INTER modes

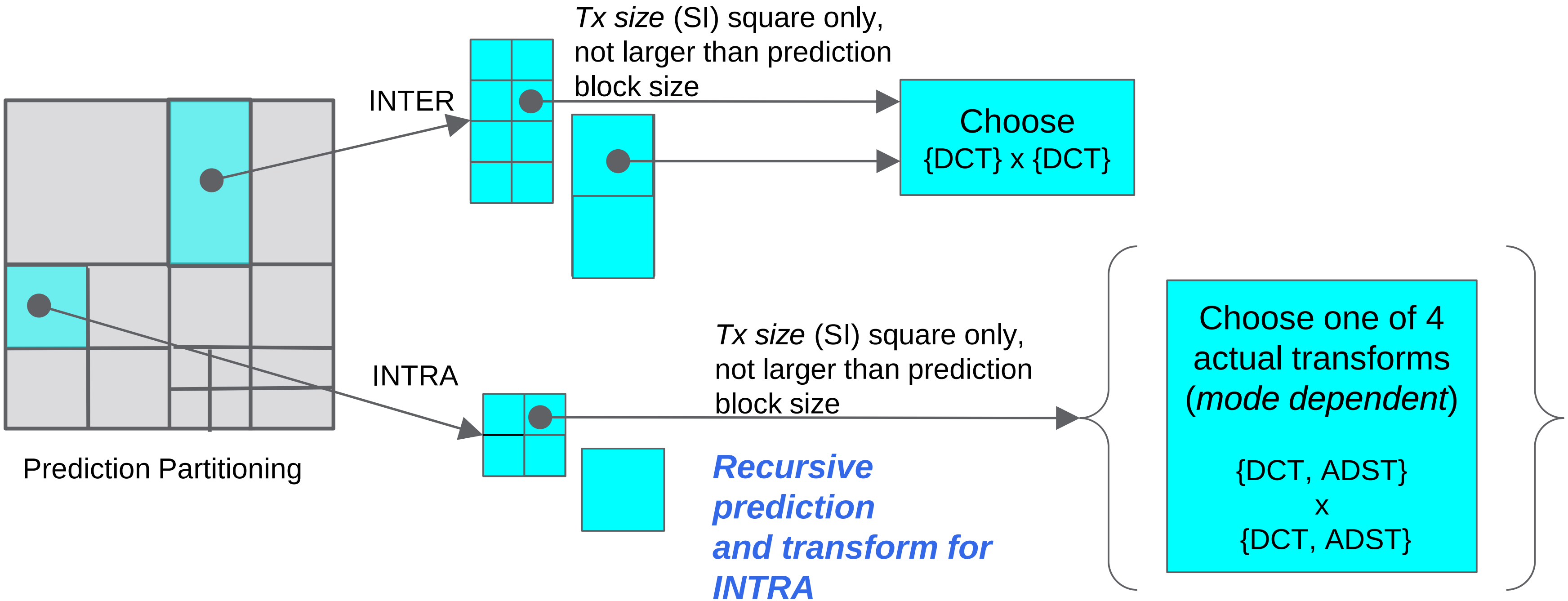
***Wedge INTER-INTER modes*:**
wedge-index + Compound INTER mode

***Wedge INTER-INTRA modes*:**
wedge-index + INTRA mode + single-ref INTER mode

***Gradual INTER-INTRA modes*:**
INTERINTRA mode + Single-ref INTER mode

***INTRA modes*:**
{DC, PAETH, SMOOTH, SMOOTH_H, SMOOTH_V
8 directional modes + finer directions

AV1 Technical Overview: Transform Tools:
VP9 Transforms Refresher

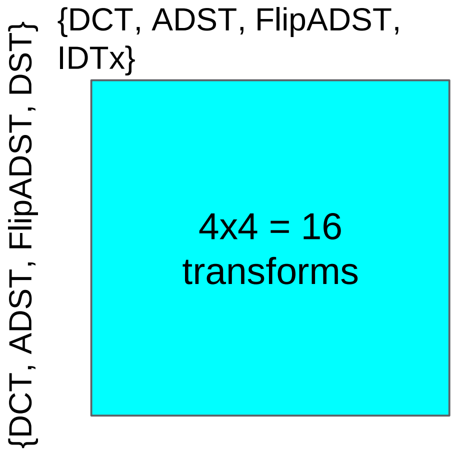


Recursive prediction and transform for INTRA

AV1 Technical Overview: Transform Tools:

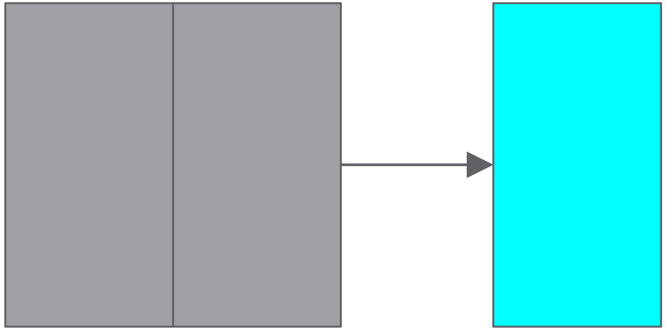
AV1 Expanded Transform Set

- AV1 Expanded transforms
  - Expands transforms to cover all combinations of:
    {DCT, ADST, FLIPADST, IDTX}
    horizontal and vertical.
    - 16 transforms for 4x4/8x8 sizes
    - Fewer at larger sizes
    - Overheads: Indicate transform used
  - Transform Skip = IDTX x IDTX
    - Very good for screen content

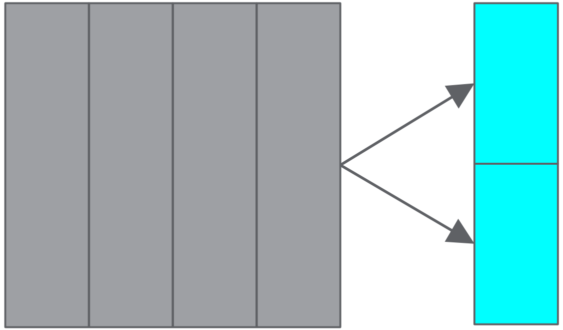


AV1 Technical Overview: Transform Tools:
Rectangular Transform Units

- AV1 adds support for 2:1 rectangular transform units
  - Can only be used for rectangular partitions
  - For 2:1 rect partitions, rect transform can only be at the same size
  - For 4:1 rect partitions, rect transform can only be used at half size



Rect transform for 2:1 partition

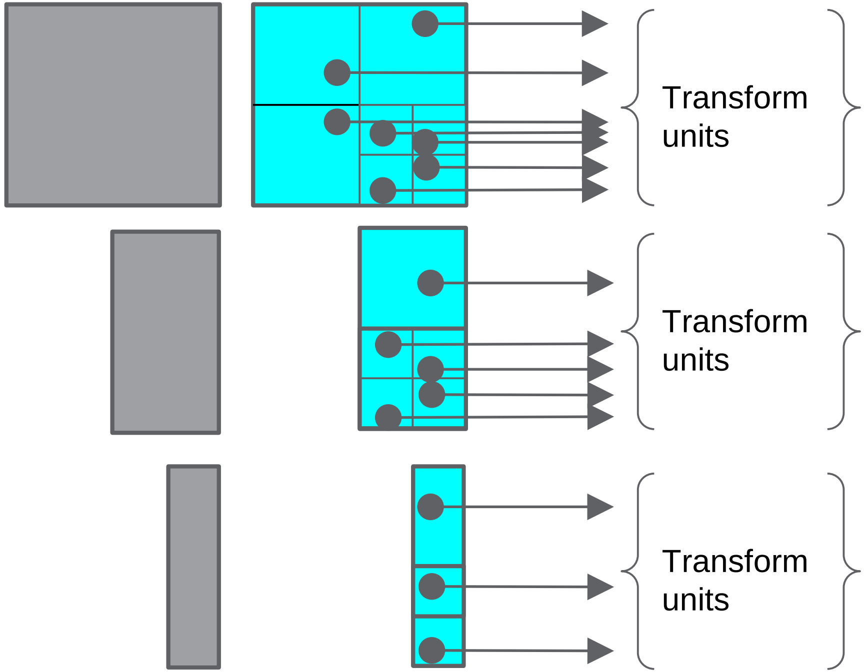


Rect transform for 4:1 partition

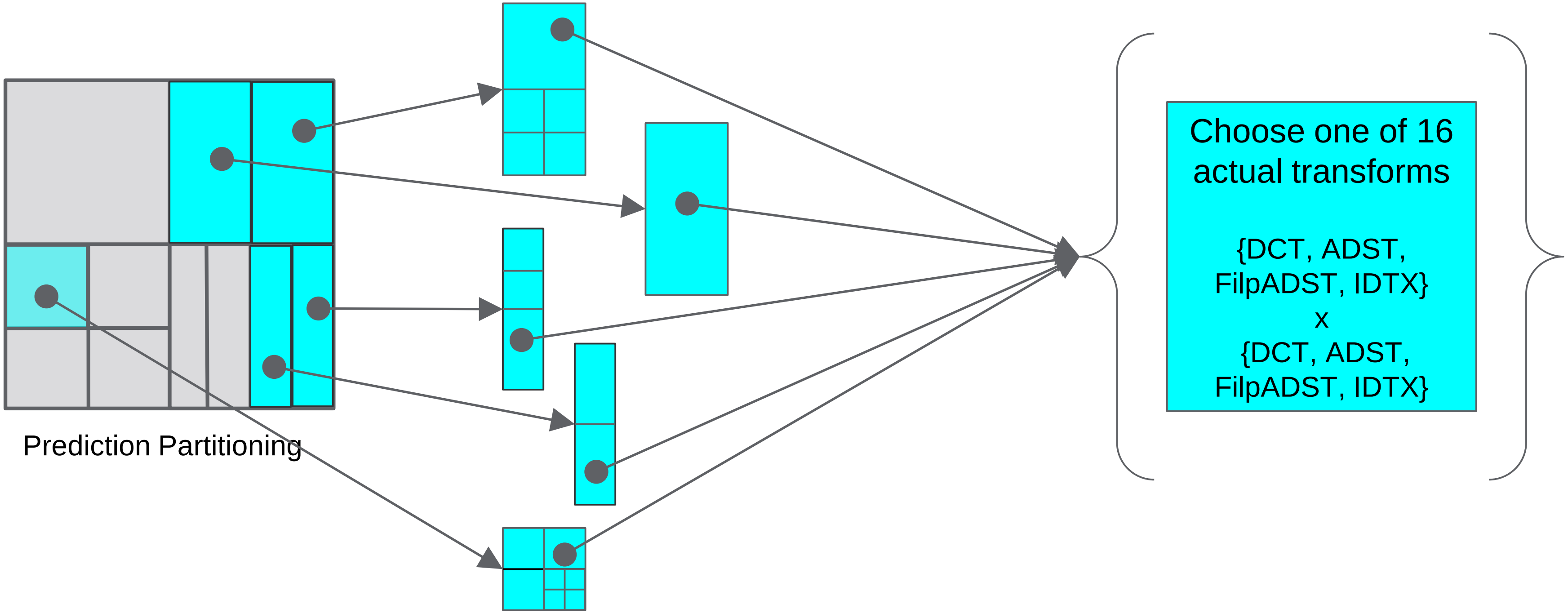
AV1 Technical Overview: Transform Tools:

Recursive partitioning of Transform units

- AV1:
  - Allows transforms within a single prediction unit to have quadtree partitions up to two levels
    - More flexibility - can isolate part of a larger block with residue concentration
  - Max transform size still smaller than Partition Unit size
  - Each transform unit has its own transform type



AV1 Technical Overview: Transform Tools:
Transform Tools Summary

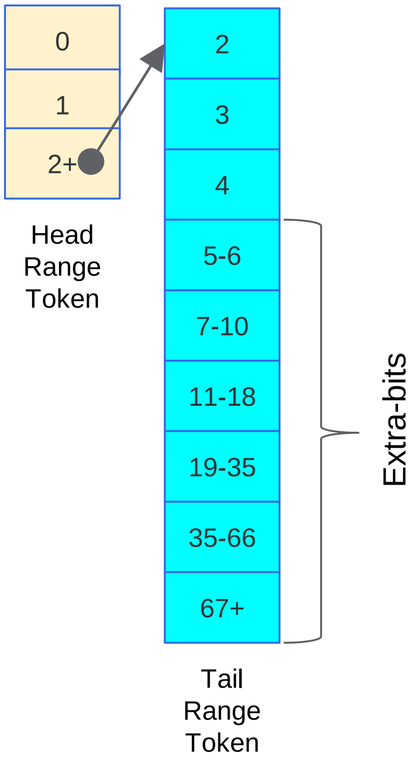


AV1 Technical Overview: Transform Tools:
Coefficient Coding

- VP9:
  - Binary non-adaptive arithmetic coding used throughout
  - Coefficient coding: Uses a binary tree with alphabet size of 11 or 12
- AV1:
  - Moving to a multi-symbol adaptive arithmetic coding model
    - Non-binary symbols are coded directly leading to increase in throughput
  - Coefficient coding modifications:
    - Intrinsically multi-symbol and symbol-adaptive

AV1 Technical Overview: Transform Tools:
Coefficient Coding

- AV1 Coefficient coding:
  - A coefficient consists of:
    - A Sign
    - Magnitude Range combined with EOB and Zero Block. Has Head and a Tail Token:
      - A Head token from:
        - {BLOCK_ZERO, ZERO, ONE_EOB, ONE_NEOB, TWOPLUS_EOB, TOWPLUS_NEOB}
      - A Tail token: codes other ranges as a 9-valued symbol
    - Extra bits indicating the value within a range
      - Coded 4 at a time in 16-valued symbols



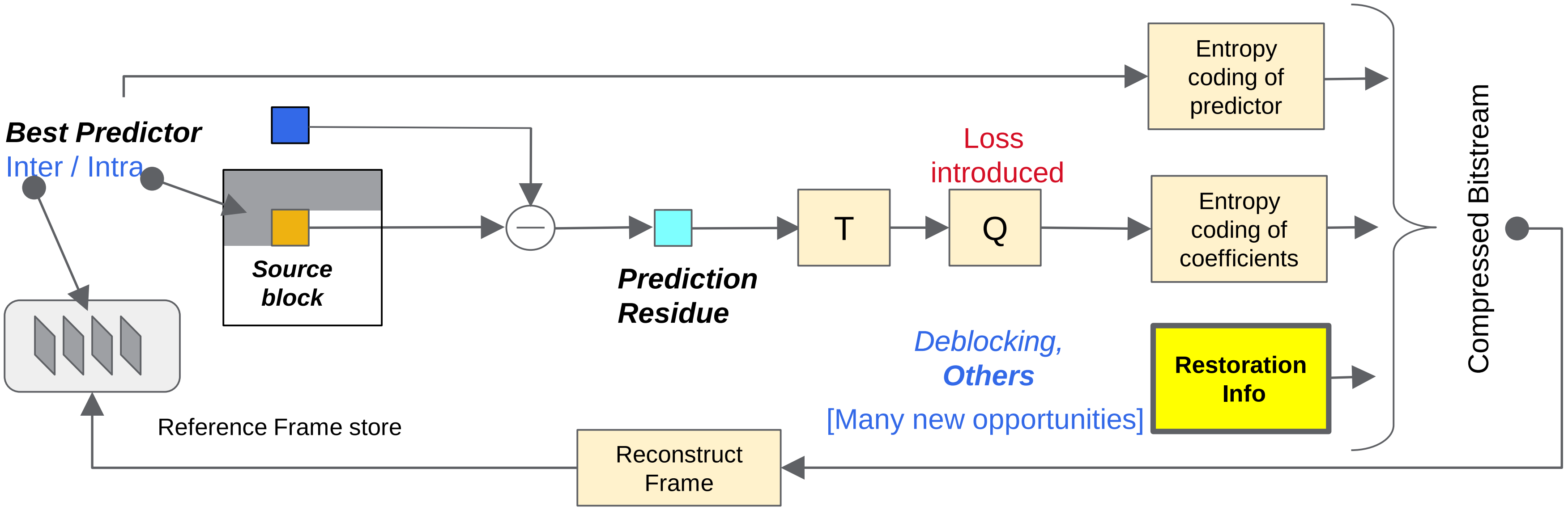
AV1 Technical Overview: Transform Tools:
Adaptive Scan of Coefficients

- Optimize scan order for different transform sizes and types
  - Gather stats for non-zero transform coefficient locations in a frame for each transform size / type
  - At the end of a frame:
    - Optimize for the best scan order for subsequent frames
      - Constraint: Maintain causality of coefficients visited
      - Topological sort
  - Enabled for 9 transform types:
    - {DCT, ADST, FlipADST} x {DCT, ADST, FlipADST}

AV1 Technical Overview: In-loop filtering tools

- Many sophisticated image restoration techniques known
  - Deblocking, Deringing, Deblurring, Denoising, Superresolution
- Seldom used for compression
  - Except Deblocking - which is standard
  - Computation is a major concern for video
- Restorability can be equated with compressibility
  - Information that can be restored, can be saved !

AV1 Technical Overview: In-loop filtering tools



- Several tools in AV1 for in-loop restoration other than Deblocking
  - CDEF (CLPF, Deringing)
  - Loop Restoration (Wiener, Dual Self-guided filter)

AV1 Technical Overview: In-loop filtering tools

Deblocking filter Advances:

- VP9:
  - One filter level sent per frame
- AV1:
  - Four filter levels per frame:
    - Luma plane: horizontal and vertical filter levels
    - Chroma plane: U, V each has its own filter level.
  - Adds ability to change filter level from superblock to superblock
  - Parallel deblocking [Intel/Microsoft]
    - Apply deblocking for all vertical edges first, then horizontal
    - Adds ability for parallelism

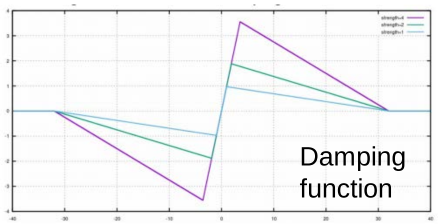
AV1 Technical Overview: In-loop filtering tools

CDEF (Constrained Directional Enhancement Filter) [Mozilla/Cisco]

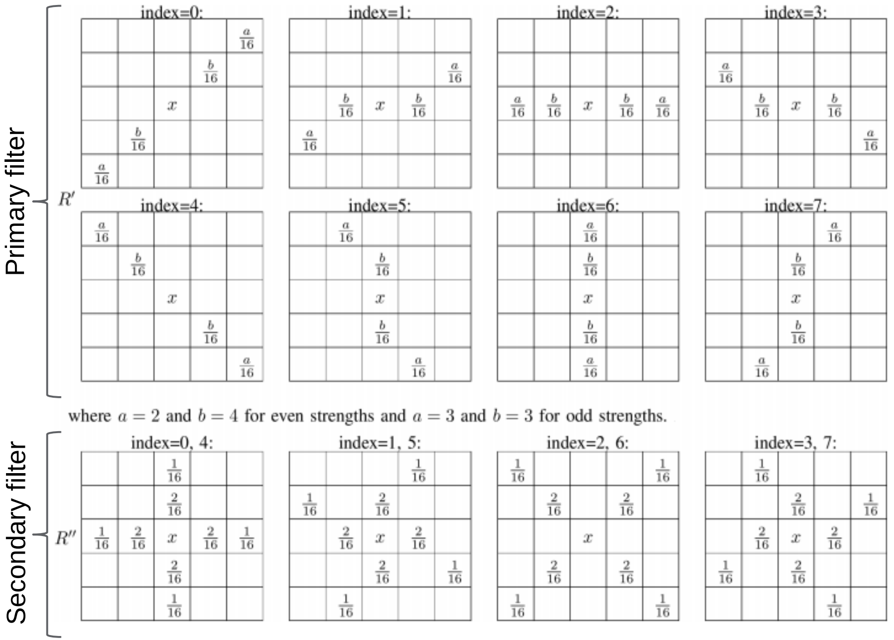
- CDEF:
  - Combination of Daala Deringing filter and Cisco's Constrained Low Pass Filter
  - Adapts filtering to direction of edges and patterns in an image
- Steps:
  - Direction estimation:
    - Conducted at 8x8 level by minimizing variances along predefined lines
  - Based on direction use a nonlinear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

AV1 Technical Overview: In-loop filtering tools

CDEF [Mozilla/Cisco]



- Direction-adaptive nonlinear filtering in CDEF:
  - Filter expressed as difference to center pixel of 5x5 support region
  - Differences go through nonlinear damping function f() before being filtered through a *primary* and a *secondary* 2D filter
  - Sum the two filter outputs, clip to the max difference within the 5x5 block, add to the center pixel.



AV1 Technical Overview: In-loop filtering tools

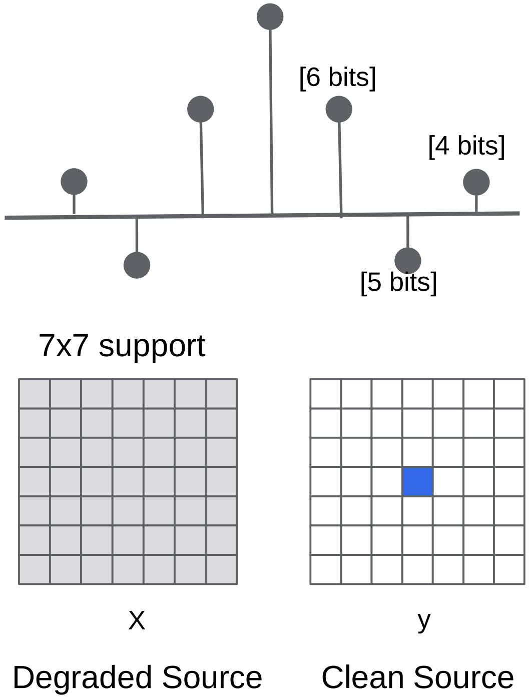
Loop-Restoration: Switchable framework

- Loop-Restoration:
  - Switchable restoration framework
    - Rationale:
      - Same restoration scheme may not be suitable to restore all parts of frame
      - A highly adaptive scheme may be too complex on decoder side
    - How it works
      - Divide a frame into multiple restoration units (RU)
      - Apply a different scheme to restore each RU
      - Send parameters needed for each RU
    - The number of tools can grow over time
      - Manageable hardware and software complexity since only one is done for any RU.

AV1 Technical Overview: In-loop filtering tools

Loop-Restoration: Wiener Filter

- Separable Symmetric Normalized Wiener filter:
  - 7-tap horizontal and 7-tap vertical separable
  - Symmetric, Normalized
  - ~30 bits per RU, Easy for decoder
- 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 and normalization constraints
    - Iterative design for hor/vert filters

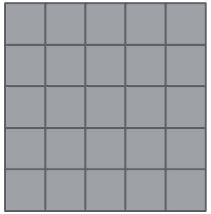


AV1 Technical Overview: In-loop filtering tools

Loop-Restoration: Dual Self-guided filter

- 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) x (2r + 1) window around every pixel.
  - Compute: $f = \sigma^2/(\sigma^2 + e)$; $g = (1 - f)\mu$
  - Compute F, G by averaging f, g in a 3x3 area around each pixel.
  - Filter pixel value using: $y = x. F + G$
  - Efficient implementation possible: readily integerized / vectorized

Parameters: (*r*, *e*)
r can be 1 or 2



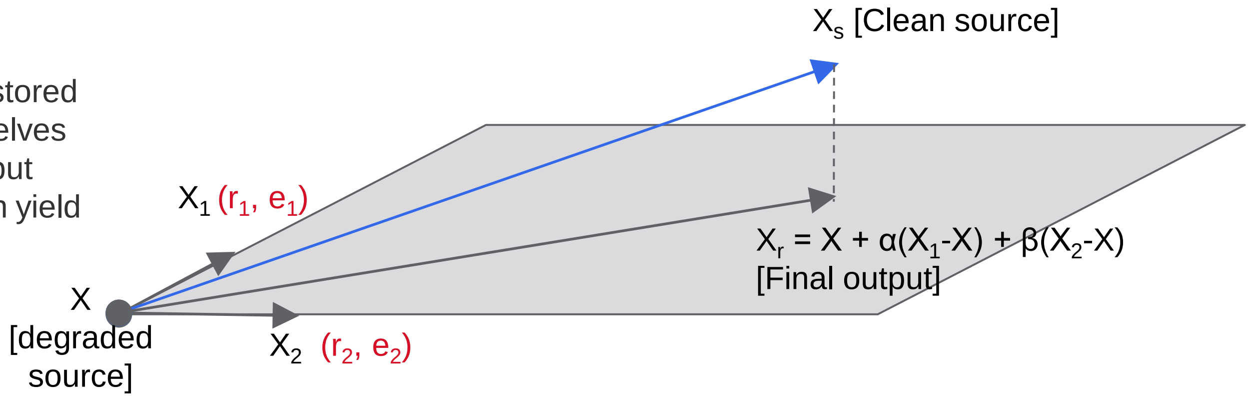
(2*r* + 1) x (2*r* + 1)

AV1 Technical Overview: In-loop filtering tools

Loop-Restoration: Dual Self-guided filter

- 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 {α, β} of X_1, X_2, to get final output
  - (r_1, e_1, r_2, e_2) taken from a small 4-bit codebook ($r_1 = 1$; $r_2 = 2$);
  - {α, β} 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.



AV1 Technical Overview: In-loop filtering tools

Deblocking/CDEF/LR Integration

- Cascade:
  - Deblocking, CDEF and LR needs to work in cascade mode
- Line-buffer problem
  - A straightforward application of one after the other will increase line-buffers significantly for a hardware implementation
- Ongoing work
  - Shift grid of deblocking and CDEF/LR
  - LR modifications to use pre-CDEF input as borders of a processing unit
  - Will eventually reduce line buffers to the same level as VP9

AV1 Technical Overview: Miscellaneous tools

Underlying Entropy Coder

- VP9:
  - Bitstream is binary non-adaptive arithmetic encoded
  - Multi-symbol alphabet decomposed into a binary tree
  - Forward and Backward updates at frame level
- AV1:
  - Moving to a multi-symbol adaptive arithmetic coding model [Mozilla]
    - Taken from Daala
    - Non-binary symbols are coded directly leading to increase in throughput
  - Multi-symbol adaptation
  - Forward updates not needed
  - Backward update is much simpler

AV1 Technical Overview: Miscellaneous tools

Quantization Advances [CISCO]

- Quantization matrices (QMs)
  - AV1 supports quantizing coefficients using 15 sets of QMs based on Contrast-Sensitivity Functions
  - QMs are applied to a frame based on selectable scaling of its base quantizer level – higher levels of quantization imply flatter matrices
  - Allows finer adjustments than just inter versus intra
- Superblock delta-quantization
  - Allow per-superblock changes in quantization parameter to support sub-frame (and low-latency) rate control
  - Additive changes on top of segmentation-level parameters to support both ROI and RC simultaneously

AV1 Technical Overview: Miscellaneous tools

Tiling Features

- Motivation:
  - Parallel encoding/decoding for speed
  - Random Access: Allow decoder to extract only an interesting section in a frame
  - Error-resilience
- AV1 tiling features:
  - Independent tiles so that tiles in a frame can be encoded / decoded in parallel.
  - Tile width ranges from 64 pixels to 4096 pixels.
  - Tile area limited to a maximum of 4096 x 2304 pixels.
  - Support flexible tile, including uniform and non-uniform tile spacing.
  - Tiles can be grouped into *tile_groups* and each *tile_group* is independently decode-able to achieve error resilience.
  - Loop-filtering can be enabled or disabled across tiles.

AV1 Technical Overview: Miscellaneous tools

Hardware Simplification Contributions

- Several contributions from various partners to simplify hardware
  - MV referencing simplifications
  - 7-bit filter taps
  - Deblocking + CDEF + LR pipeline

Preliminary Results
AWCY

- www.arewecompressedyet.com
  - An automatic service for compressing videos and comparing coding efficiency among AV1, VP9, Daala, Thor, and other codecs.
    - Maintained by: xiph.org foundation [Mozilla]
  - Chosen as official testing infrastructure for AOM
  - Uses AWS Cloud resources
  - Supports several test sets, and test conditions

Preliminary Results
AWCY: High Latency

- Objective-1 Fast - High Latency
  - Constant Quality mode, 30 clips w/ 60 frames
- Candidates
  - Baseline: VP9 tip of tree; Test: AV1 (08/26/2017), X.265 ver 1.9

BDRATE	PSNR-Y	PSNR-Cb	PSNR-Cr	PSNR-HVS	CIEDE 2000	SSIM	MS-SSIM
AV1*	-25.58%*	-21.78%	-23.69%	-24.14%	-24.43%	-23.63%	-22.58%
X.265+	-4.46%	N/A	N/A	-2.25%	+13.33%	+8.24%	+7.88%

+X.265: --preset placebo --tune psnr --no-wpp

*AV1 target: -30% over VP9

Preliminary Results

AWCY: Low Latency

- Objective-1 Fast - Low Latency
  - Constant Quality mode, 30 clips w/ 60 frames
- Candidates
  - Baseline: VP9 tip of tree; Test: AV1 (09/15/2017)

BDRATE	PSNR-Y	PSNR-Cb	PSNR-Cr	PSNR-HVS	CIEDE 2000	SSIM	MS-SSIM
AV1*	-29.31%*	-28.43%	N/A	-27.11%	-28.10%	-30.48%	-28.44%

*AV1 target: -30% over VP9

Conclusion

- AV1 Coming Soon!
  - AOM has made decent progress towards a next generation royalty-free codec **AV1**
    - 25+% BDRATE reduction over VP9 and HEVC at high delay
    - 29+% BDRATE reduction over VP9 at low delay
    - With high-bit-depth internal there is an additional 3-4% gain
    - Our target: 30%
  - Google VP10 effort evolved into AV1
  - Expected AV1 freeze: December 2017
  - Encoder / decoder optimization will continue

Thank you!

LiveVideoStackCon

聚 音 视　研修不止于形

关注LiveVideoStackCon公众号

回复 **Zoe Liu** 为讲师评分