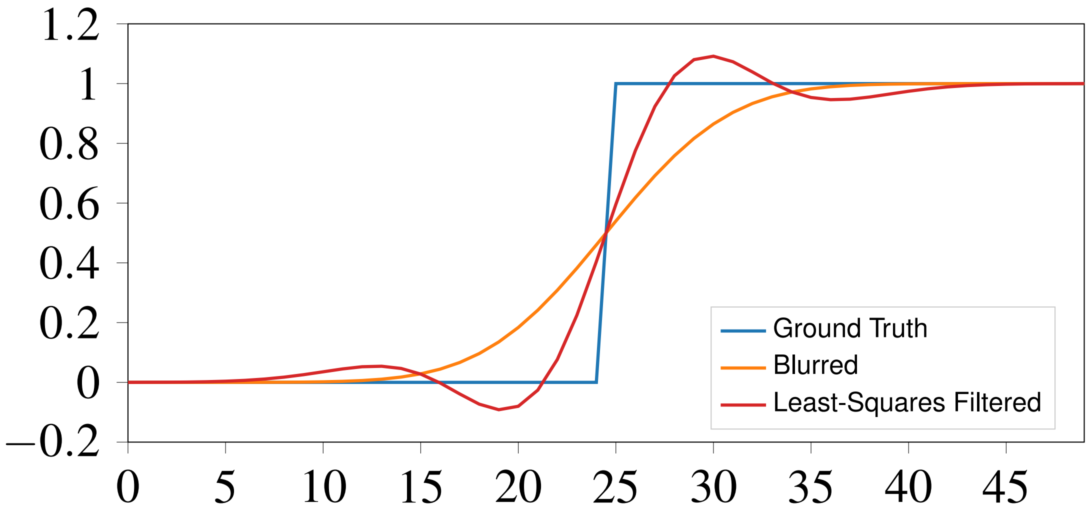
JVET-AE0103

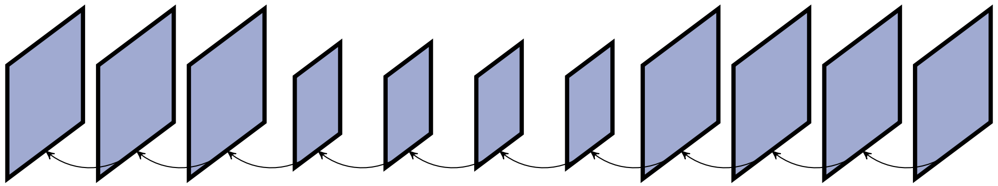
Weighted Filtering for Picture Upscaling and RPR

Tim Claßen, Mathias Wien
Lehrstuhl für Bildverarbeitung
RWTH Aachen University

Motivation

- Upsampled video content in RPR is usually blurred
  - High-frequency components are lost in the downsampling
- Currently implemented linear upsampling filters are pre-defined and not content adaptive
  - Pre-defined filters may be suboptimal
  - Linear filters may exhibit ringing artifacts
- Adaptive filter for edge enhancement without ringing or overshoot and amplification of noise desirable





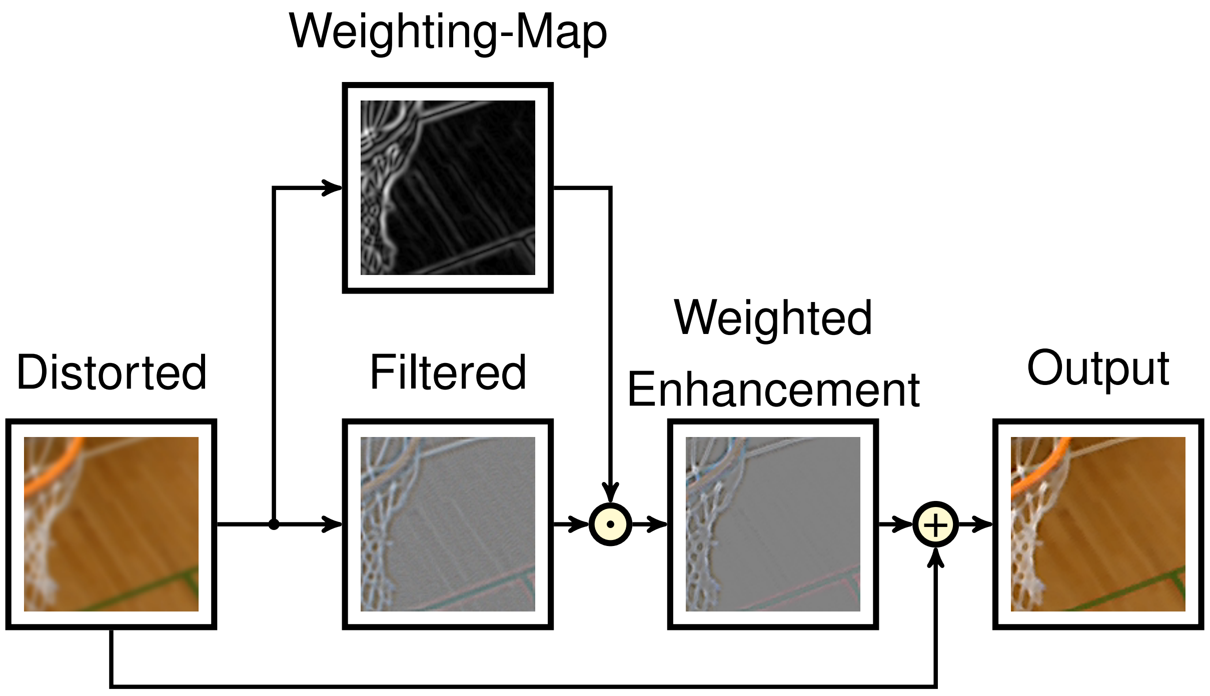
RPR example

Weighted Adaptive Filtering

Filter Setup

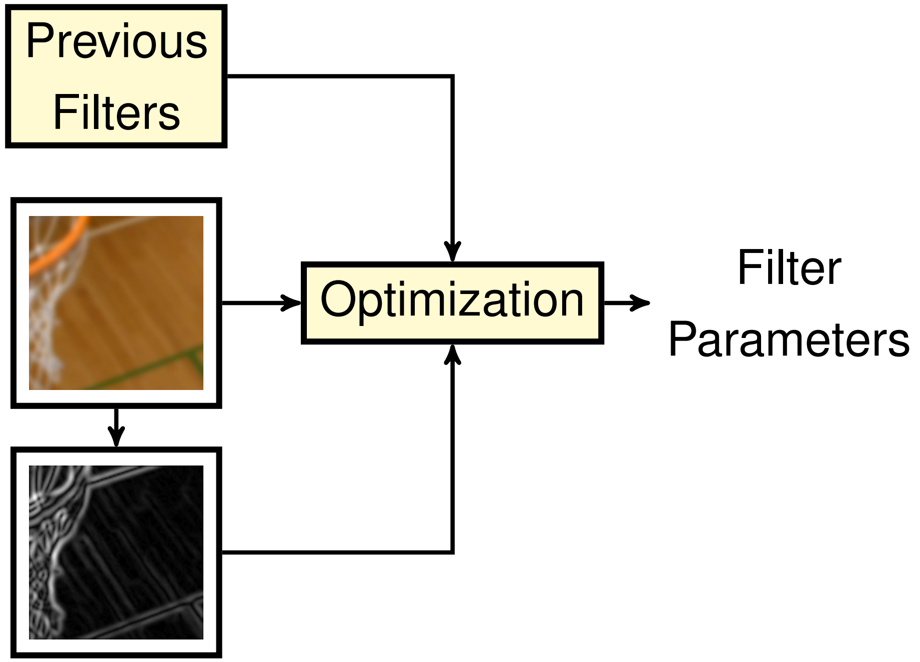
1. Weighting map is computed based on the input picture
2. Filtered picture is computed based on input picture
3. Multiplication of weighting map and filtered picture
4. Result of multiplication is added to the input picture

- Filter is applied to luma and chroma components separately
- Weighting map: Magnitude of the local gradient (e.g. Sobel-operator)
- Diamond shaped filters similar to ALF



Optimization

1. Calculate weighting map
2. Find least-squares filter
3. Find encoding costs of filter / RD-optimize filter
   – Find optimal encoding of filter coefficients
   – Decide for luma/chroma on-/off-flags
   – Compare to RD-Performance of re-using previous filter
4. Transmit result of optimization (new filter, re-using old filter) in APS if RD-improvement is achieved



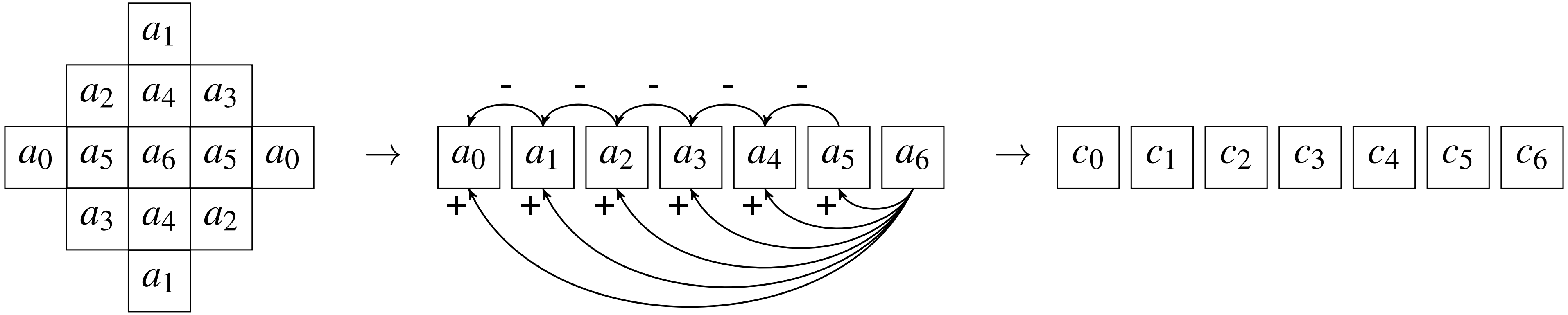
Coefficient Coding

Steps

1. Quantization of filter coefficients
   - Quantization step size of 2^{-10}
2. Prediction of filter coefficients and parameters
   - Intra filter prediction
   - Inter filter prediction
   - Filter copy
   - Fallback
3. Encoding of filter coefficients and parameters

Parameter Prediction

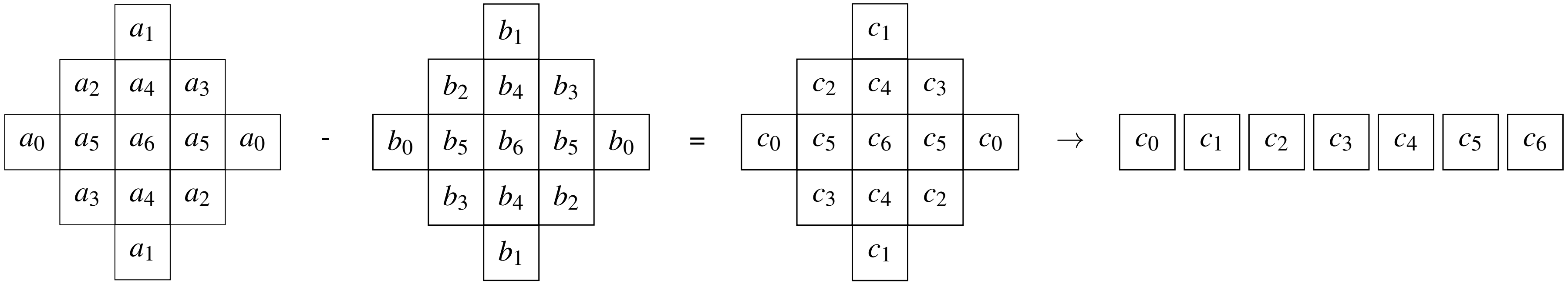
Coding Mode: Intra Filter Prediction



- The optimized filter is defined as $\mathbf{a}$
- Prediction from previously decoded coefficients such that $c_i = a_i - a_{i-1},\ n-1 > i > 0$ and $c_0 = a_0$ are encoded
- Last filter coefficient $n-1$ is the sum of all coefficients
  - Filter does not induce a strong offset in most cases $\rightarrow$ sum should be close to zero
- Reduces coding costs if there are similarities inside the filter

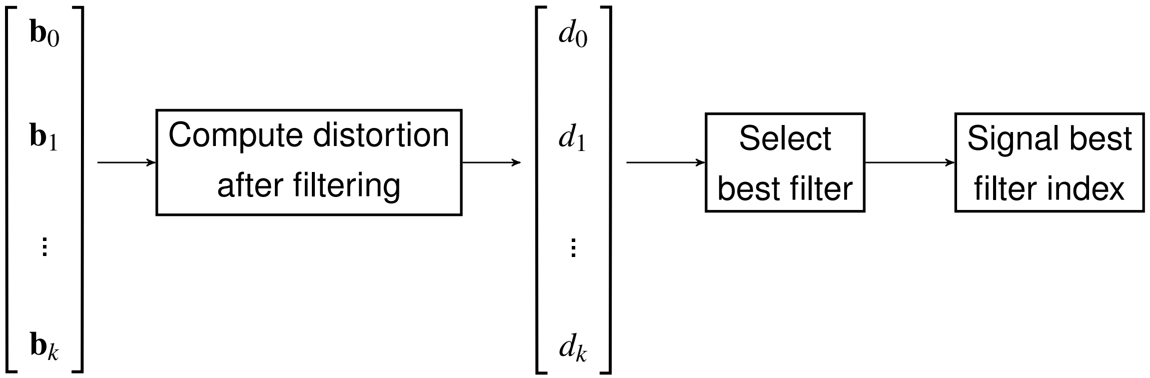
Parameter Prediction

Coding Mode: Inter Filter Prediction



- The optimized filter is defined as $\mathbf{a}$
- Filter $\mathbf{a}$ is predicted by $\mathbf{b}$ such that the coefficients $c_i = a_i - b_i$ are encoded
- Reduces coding costs if $\mathbf{a}$ and $\mathbf{b}$ are similar
- Filter $\mathbf{b}$ is selected from list of available reference filters

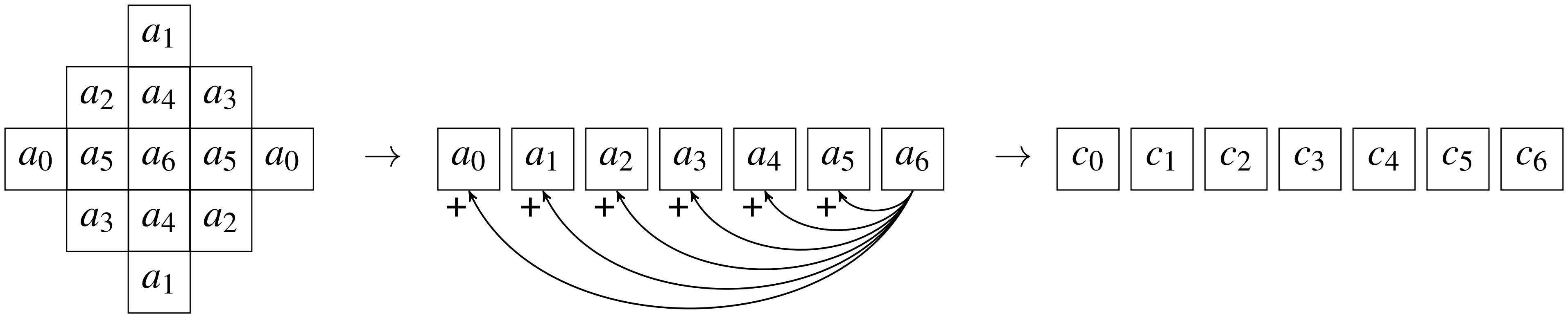
Coding Mode: Filter Copy



- Selection by minimizing distortion d_i
- Copies the entire filter without altering it
- Very low coding costs
- Reduced quality gain
  - Characteristics of the video change in temporal direction
  - Statistics of the residual change in temporal direction

Parameter Prediction

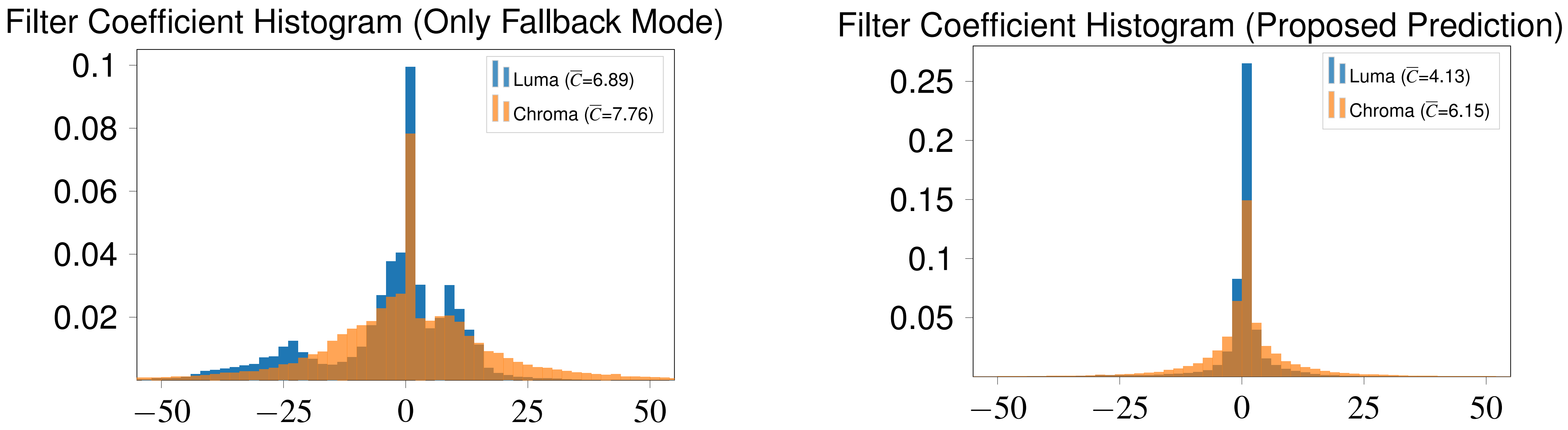
Coding Mode: Fallback



- The optimized filter is defined as $\mathbf{a} = [a_0, ..., a_{n-1}]$
- Filter coefficients $c_i = a_i, i \in \{0, ..., n-2\}$ are directly encoded
- The last filter coefficient is encoded as the sum of all other values $c_{n-1} = \sum_j^{n-1} a_j$
- Fallback mode is applied if other modes do not provide a better encoding
- Prediction of last coefficient is considered as robust

Entropy Coding

- Parameters and flags: fixed-length codes
  - Luma- and chroma-flag / filter shape / coding mode
- Filter coefficients: exponential Golomb code



$\overline{C}$: average coding costs per filter coefficient

Results (ECM 8 with 2.0x scaling)

	All Intra Main 10				
	Over ECM-8.0				
	Y	U	V	EncT	DecT
Class A1	#WERT!	#WERT!	#WERT!	#DIV/0!	#DIV/0!
Class A2	#WERT!	#WERT!	#WERT!	#DIV/0!	#DIV/0!
Class B	-4,89%	-0,80%	-1,11%	125%	169%
Class C	-3,71%	-0,90%	-1,05%	136%	160%
Class E	-8,35%	-0,25%	-0,02%	153%	197%
Overall	#WERT!	#WERT!	#WERT!	#ZAHL!	#ZAHL!
Class D	-3,12%	-0,50%	-1,09%	148%	178%
Class F	-7,36%	-4,26%	-4,67%	148%	196%
Class TGM	#WERT!	#WERT!	#WERT!	#DIV/0!	#DIV/0!

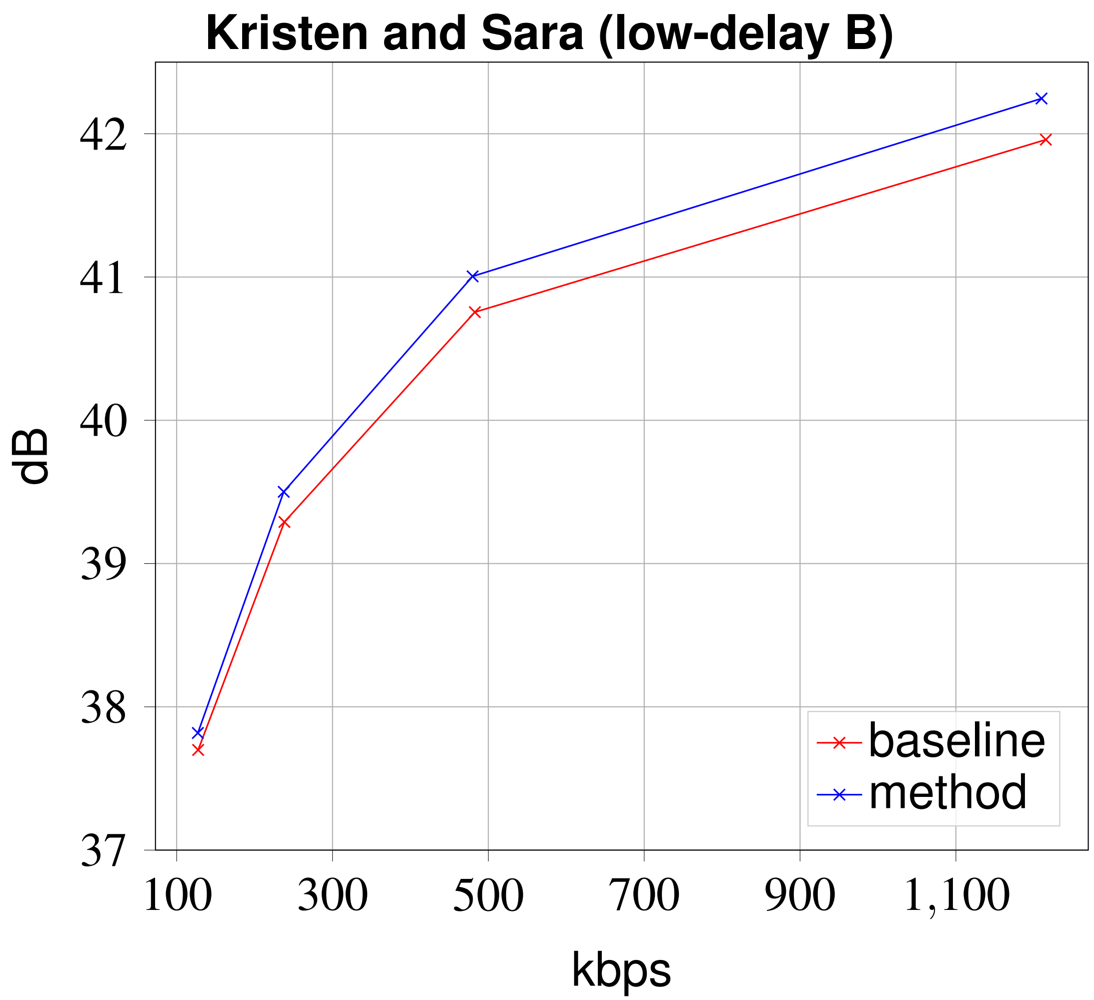
	Low delay B Main 10				
	Over ECM-8.0				
	Y	U	V	EncT	DecT
Class A1					
Class A2					
Class B	-6,87%	-0,79%	-1,04%	102%	115%
Class C	-3,55%	-1,24%	-0,63%	83%	92%
Class E	-12,90%	0,67%	1,29%	70%	90%
Overall	-7,27%	-0,57%	-0,32%	87%	100%
Class D	-2,62%	1,03%	-1,06%	68%	77%
Class F	-7,06%	-3,63%	-4,81%	80%	101%
Class TGM	#WERT!	#WERT!	#WERT!	#DIV/0!	#DIV/0!

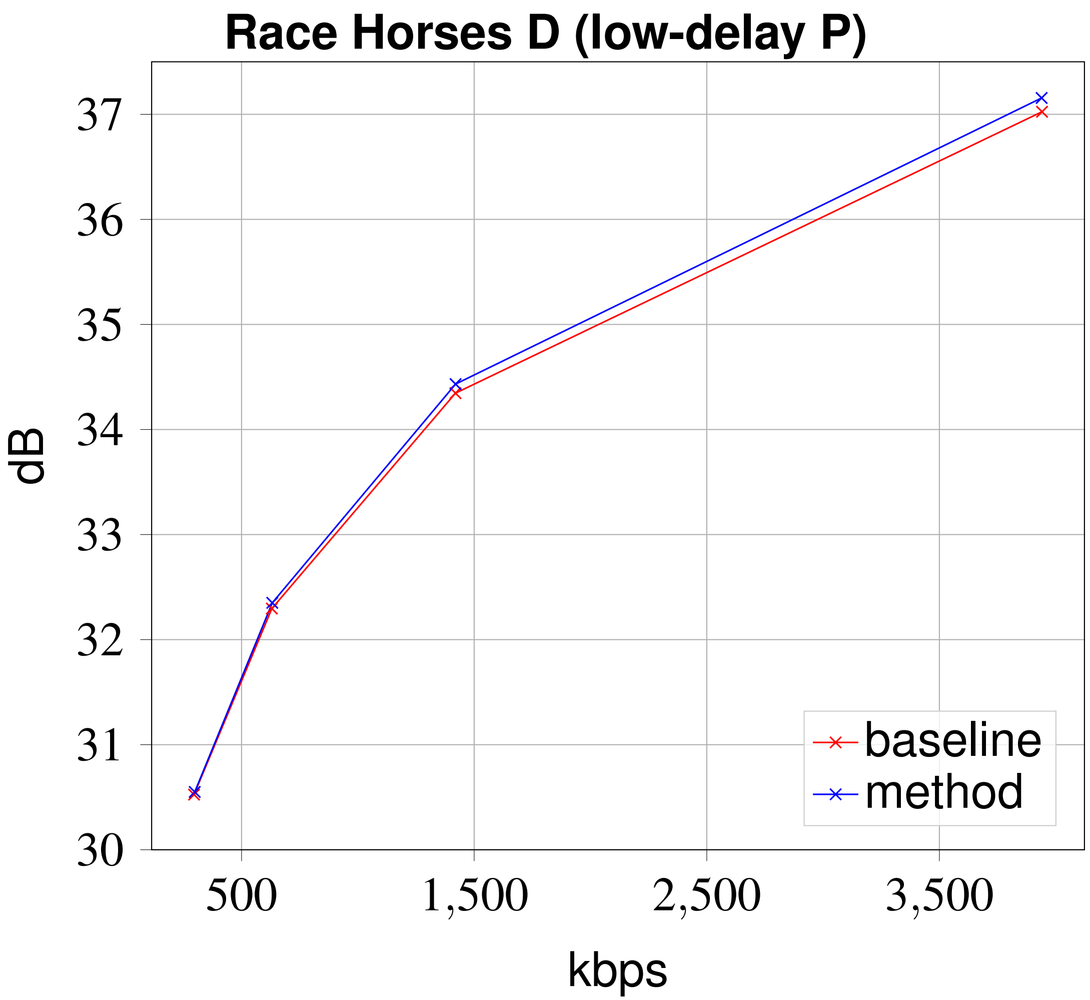
	Random Access Main 10				
	Over ECM-8.0				
	Y	U	V	EncT	DecT
Class A1	#WERT!	#WERT!	#WERT!	#DIV/0!	#DIV/0!
Class A2	#WERT!	#WERT!	#WERT!	#DIV/0!	#DIV/0!
Class B	-6,02%	-0,65%	-0,96%	125%	132%
Class C	-4,17%	-1,22%	-1,36%	89%	100%
Class E					
Overall	#WERT!	#WERT!	#WERT!	#DIV/0!	#DIV/0!
Class D	-3,05%	-0,30%	-1,99%	66%	84%
Class F	-7,87%	-5,79%	-6,55%	105%	145%
Class TGM	#WERT!	#WERT!	#WERT!	#DIV/0!	#DIV/0!

	Low delay P Main 10				
	Over ECM-8.0				
	Y	U	V	EncT	DecT
Class A1					
Class A2					
Class B	-6,96%	-1,44%	-1,20%	118%	128%
Class C	-3,93%	-1,39%	-1,92%	106%	114%
Class E	-13,93%	-1,50%	-0,73%	112%	170%
Overall	-7,69%	-1,44%	-1,33%	112%	132%
Class D	-2,82%	-0,74%	-1,44%	119%	115%
Class F	-7,45%	-4,55%	-5,68%	109%	146%
Class TGM	#WERT!	#WERT!	#WERT!	#DIV/0!	#DIV/0!

Computation times are unreliable

Results (ECM 8)





Conclusion

- Filtering method for upscaled video content proposed
- Applications:
  - Picture upscaling
  - Reference picture resampling
- Method:
  - Weighted linear filter
  - Weighting calculated as "Sobel magnitude map"
  - Linear filter is least squares optimized
- Method show consistent gains for all coding configurations and classes
- Suggestion to investigate this in an Exploration Experiment (EE)