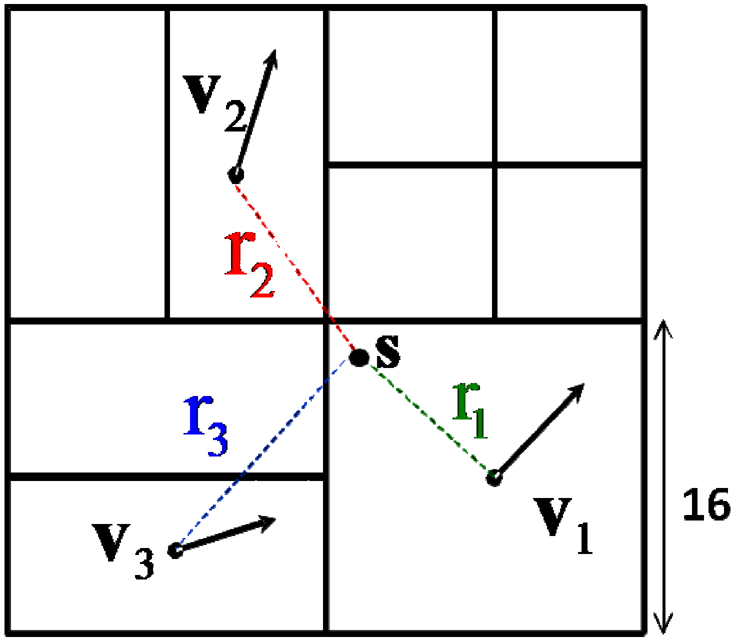
JCTVC-D175
Bi-Prediction Combining Template and Block Motion Compensations

National Chiao Tung University (NCTU)

Wen-Hsiao Peng, Chung-Lin Lee, Chun-Chi Chen,
Hsueh-Ming Hang, et al.

Parametric OBMC (A123)

- A distance-based weighting criterion to combine MVs for OBMC

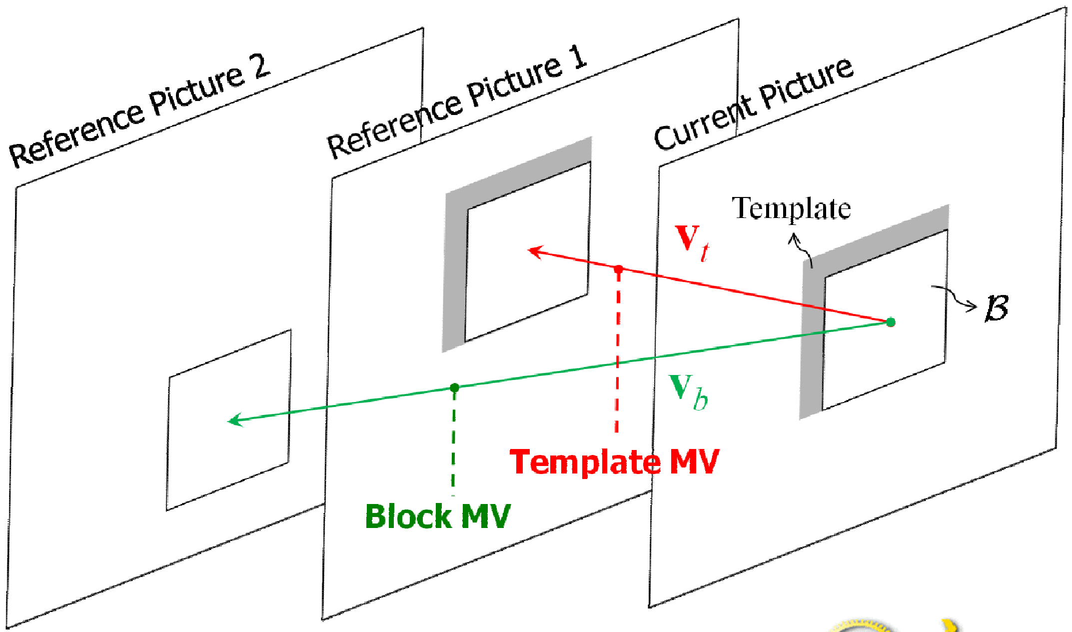


$$w_1^* = \frac{\frac{1}{r_1^2}}{\frac{1}{r_1^2} + \frac{1}{r_2^2} + \frac{1}{r_3^2}}$$

TM-based Bi-Prediction (B072)

- Bi-prediction with reduced overhead by combining template and block MVs

  → Template MV is free from signaling



Optimized Block Motion Search

- Block matching criterion was modified to consider the contribution of the template MV

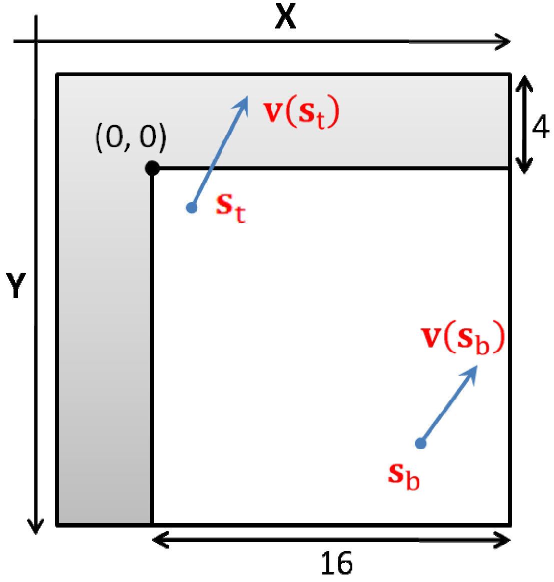
$$\arg\min_{\mathbf{v}_b} \sum_{\mathbf{s}\in B} \Big(I_k(\mathbf{s}) - \overbrace{\big(1 - w(\mathbf{s})\big) \underbrace{I_{k-1}(\mathbf{s}+\mathbf{v}_t)}_{\text{TMP Predictor}} - w(\mathbf{s}) \underbrace{I_{k-1}(\mathbf{s}+\mathbf{v}_b)}_{\text{BMC Predictor}}}^{\text{OBMC Predictor}} \Big)^2$$

→ Window function has to be designed properly

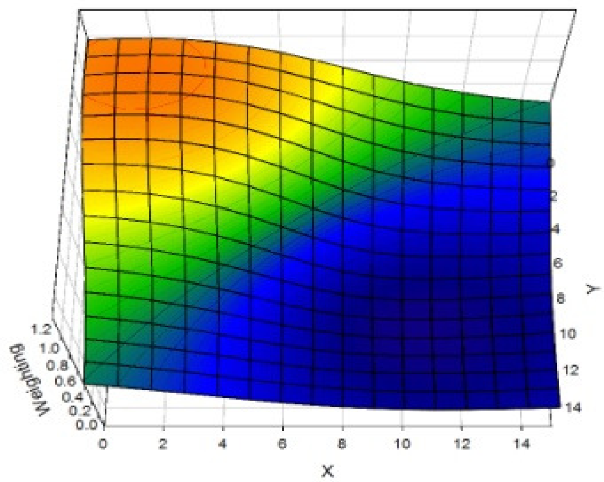
Window Functions

- A particular type of geometry partitioning

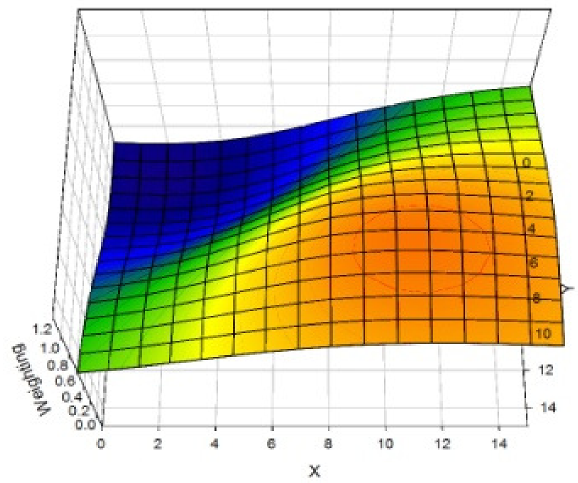
InvL



Template MV



Block MV

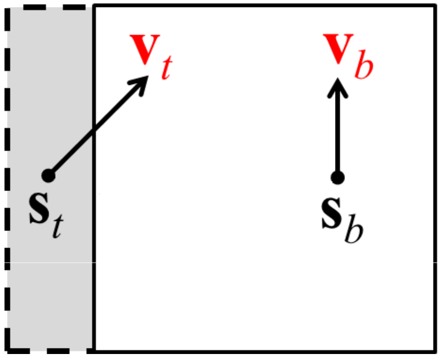


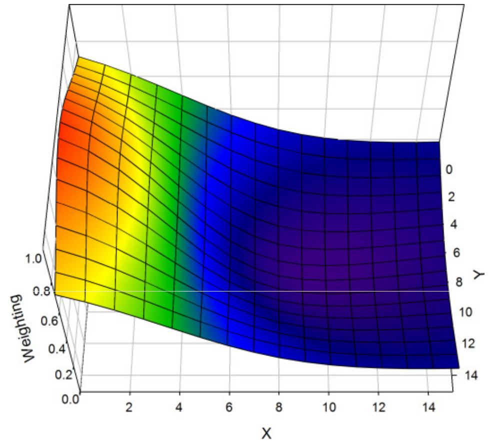
More Examples

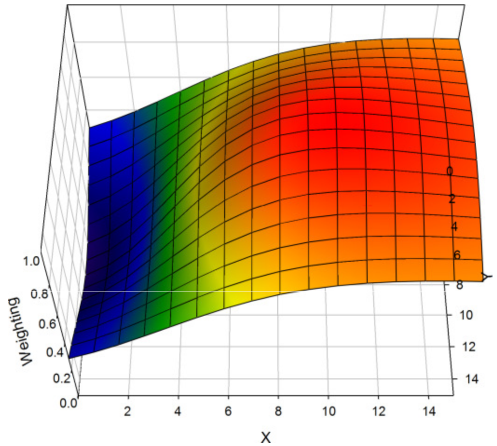
Template MV

Block MV

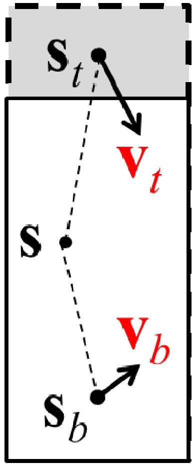
Rect-L

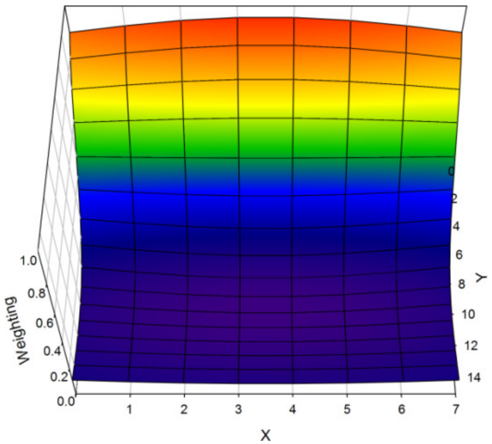


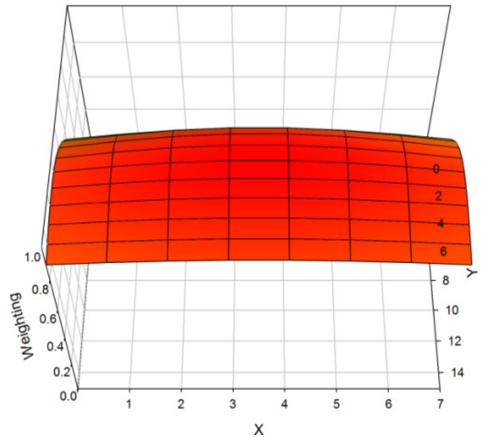




Rect-T

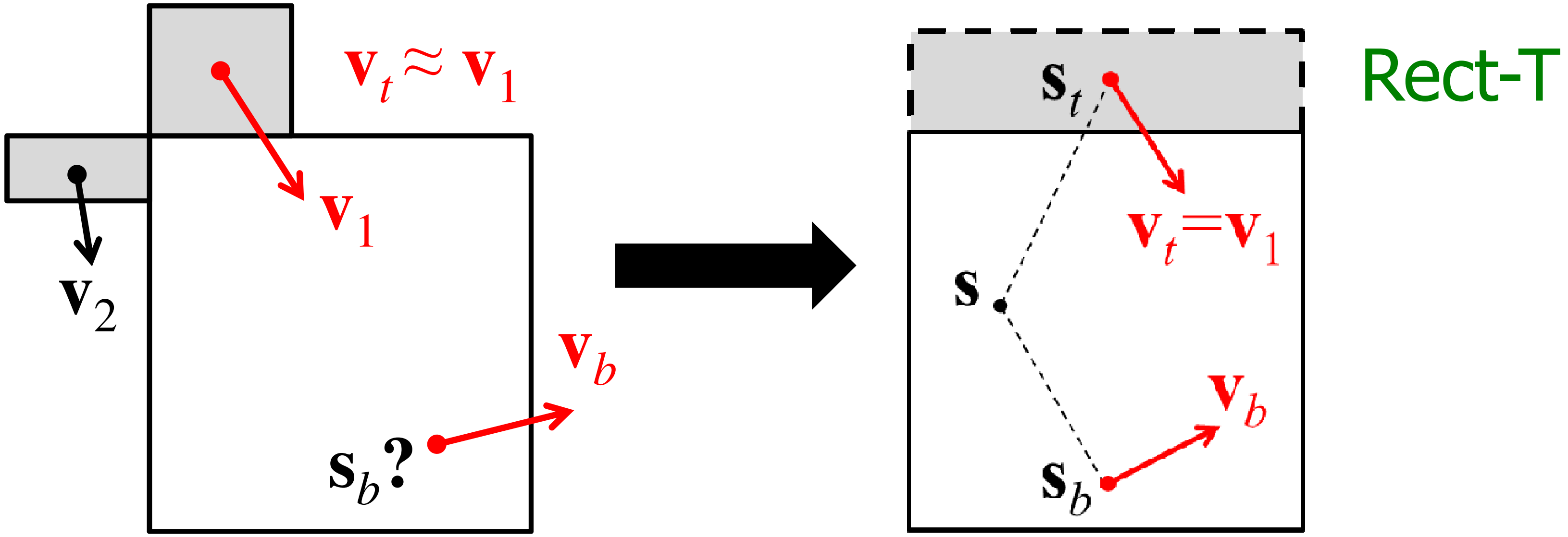






TM-free Implementation

- Replace template MV with a decoded MV specified by motion merging
- Select window according to merge direction



Experiments

- 3 algorithms integrated into TMuC0.9-hm
  - CU-adaptive on/off
  - PU-adaptive window selection (only for #2 & #3)

Algo.	Supported PU	Window	Motion Inference
#1	2Nx2N	Fixed (InvL only)	TMP (MVP ± 8)
#2	2Nx2N	Adaptive	Motion Merging
#3	2Nx2N, NxN, 2NxN, Nx2N	Adaptive	Motion Merging

Performance

- Anchor: TMuC0.9-hm
- HE-RA/LD, LC-RA/LD
- No loss is observed

AVG	#1	#2	#3
HE-RA	-1.0%	-1.4%	**-2.1%**
HE-LD	-1.5%	-1.9%	**-2.8%**
LC-RA	-0.8%	-1.3%	**-2.1%**
LC-LD	-0.9%	-1.1%	-1.8%

MAX	#1	#2	#3
HE-RA	-1.8%	-1.8%	-2.8%
HE-LD	-2.5%	-3.3%	**-4.7%**
LC-RA	-1.6%	-2.1%	-3.3%
LC-LD	-1.7%	-2.1%	-3.3%

MIN	#1	#2	#3
HE-RA	-0.7%	-0.9%	-1.3%
HE-LD	-0.5%	-0.5%	-1.3%
LC-RA	-0.3%	-0.8%	-1.2%
LC-LD	-0.1%	-0.0%	-0.2%

Runtime

Dec. Time	#1	#2	#3
HE-RA	204%	95%	99%
HE-LD	194%	97%	100%
LC-RA	231%	114%	123%
LC-LD	213%	111%	121%

Enc. Time	#1	#2	#3
HE-RA	153%	126%	208%
HE-LD	126%	116%	172%
LC-RA	165%	142%	263%
LC-LD	133%	127%	204%

Dec. Time	#1	#2	#3
HE-RA	175%	97%	99%
HE-LD	177%	95%	95%
LC-RA	189%	94%	98%
LC-LD	190%	93%	94%

- Dec. time w/ yuv
- Dec time w/o yuv

Table Sizes

- A separate set of windows are pre-computed for each distinct PU size and shape
- They may as well be computed on the fly at the cost of extra computational complexity

Algo.	PU	Window	Table Size (Kbytes)
#1	2Nx2N	Fixed (InvL)	5.6
#2	2Nx2N	Adaptive	11
#3	2Nx2N, NxN, 2NxN, Nx2N	Adaptive	19.2

Conclusions

- A bi-prediction scheme with a motion overhead similar to that as for unidirectional prediction
- Three implementations featuring different performance-complexity trade-offs—they all show moderate coding gains
- An effect similar to always having one of the geometry/asymmetric partitions coded in merge mode
- We suggest the committee establish a CE to further investigate this proposed technique

Q&A