

# Motion Blur Compensation in Scalable HEVC Hybrid Video Coding

Thorsten Laude, Holger Meuel, Yiqun Liu, Jörn Ostermann  
Institut für Informationsverarbeitung, Gottfried Wilhelm Leibniz Universität Hannover  
Hannover, Germany  
Email: {laude, meuel, liuyiqun, office}@tnt.uni-hannover.de

**Abstract**—One main element of modern hybrid video coders consists of motion compensated prediction. It employs spatial or temporal neighborhood to predict the current sample or block of samples, respectively. The quality of motion compensated prediction largely depends on the similarity of the reference picture block used for prediction and the current picture block. In case of varying blur in the scene, *e.g.* caused by accelerated motion between the camera and objects in the focal plane, the picture prediction is degraded. Since motion blur is a common characteristic in several application scenarios like action and sport movies we suggest the in-loop compensation of motion blur in hybrid video coding. Former approaches applied motion blur compensation in single layer coding with the drawback of needing additional signaling. In contrast to that we employ a scalable video coding framework. Thus, we can derive strength as well as the direction of motion of any block for the high quality enhancement layer by base-layer information. Hence, there is no additional signaling necessary neither for predefined filters nor for current filter coefficients. We implemented our approach in a scalable extension of the High Efficiency Video Coding (HEVC) reference software HM 8.1 and are able to provide up to 1 % BD-Rate gain in the enhancement layer compared to the reference at the same PSNR-quality for JCT-VC test sequences and up to 2.5 % for self-recorded sequences containing lots of varying motion blur.

**Keywords:** Motion blur compensation, SHVC, HEVC, scalable, video coding

## I. INTRODUCTION

Hybrid video coding systems consist of motion compensated prediction as well as transform, quantization and entropy coding. This also applies for the High Efficiency Video Coding (HEVC) [1] standard as well as for predecessor standards like AVC [2] or MPEG-2 [3]. Motion compensated prediction is based on the (block-wise) comparison between already coded reference picture blocks and current picture blocks. Finally, only a displacement vector called motion vector and the prediction error are used for further processing instead of the original picture blocks. As long as the current picture block is similar to the reference picture block the data rate can be reduced significantly. For unblurred videos this holds true as well as for continuous blur effects.

The performance of the (motion compensated) prediction may be degraded by motion blur changing over pictures. Motion blur in general occurs when the relative motion between the objects recorded in a video sequence and the camera is that fast that the object cannot be assumed as static during the

exposure time. This results in reduced compression efficiency when accelerated motion appears. If the extent of motion blur is different in successive pictures, the data rate for the residual of inter predicted blocks increases due to the lack of consideration of varying motion blur by the motion estimation and compensation.

This systematic error can be compensated with reference picture filtering to reduce the prediction error finally resulting in lower coding data rates. For the AVC standard there have been several approaches to compensate motion blur, *e.g.* [4], [5], [6]. Those approaches have in common that they are used in the context of single layer coding and thus need additional signaling in the bitstream for the filtering, *i.e.* the filter coefficients for adaptive filters or an index for predefined filters.

We propose reference picture filtering in the context of scalable video coding. Thereby, we assume a system with two layers, one layer is the high quality enhancement layer (EL), the other layer is called base layer (BL) and is a ( $2 \times$  vertically and horizontally each) downsampled version of the EL. Given that our adaptive filters can be derived implicitly based on BL information no signaling of coefficients or indices is needed. Thus, *e.g.* in comparison to [5], we save three binary flags per CU for the index of predefined filters.

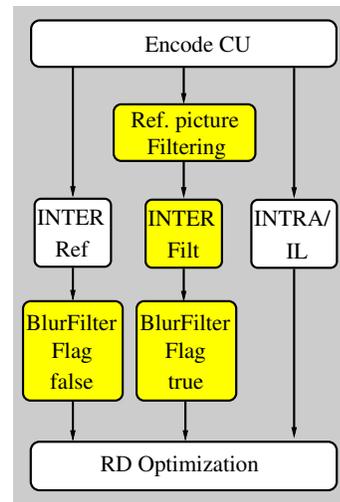


Figure 1. Motion blur compensation flow chart

## II. REFERENCE PICTURE FILTERING

Since motion blur changing over time largely degrades the quality of the picture prediction process we suggest to filter the EL reference picture in order to increase the similarity between these reference pictures and the current picture. Therefore an adjustment of the motion blur extend is applied. The BL does not need a motion blur compensation itself, as motion blur primary appears in high frequency components not present in BL due to downsampling and higher quantization parameters (QPs). However, we employ BL information to derive appropriate filter coefficients assuming the differences are caused by changing motion blur.

Based upon the motion vector (MV) of the co-located BL coding unit (CU) a two-dimensional directional filter is derived and applied to the EL reference pictures. For this derivation we use two different low-pass filters as base, one for the luminance and one for the chrominance signal with the filter coefficients set according to Table I. The corresponding frequency responses are shown in Fig. 2.

Table I  
FILTER COEFFICIENTS

Base filter for	Coefficients
Luma with direction	1 1 1 2 1 1 1
Luma without direction	$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$
Chroma	1 2 1

Since these filters have only one coefficient unequal to “1” each, a computational efficient implementation with additions and only one multiplication is possible. The normalization of the filter result is realized using bit shifts.

If the MV of the co-located BL CU is available and at least one MV component is unequal to zero, the angle of the MV towards the positive horizontal axis is calculated and quantized to the nearest element of  $\{0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ\}$ . This way the possible directions of the two-dimensional filter are restricted to eight: the horizontal and vertical directions and the directions on the diagonals. Using this approach we are able to design 2D filters with very few coefficients by placing the coefficients of a 1D base filter on these directions while the rest of the 2D filter’s coefficients remain zero as illustrated below in (1).

$$\begin{bmatrix} 1 & 2 & 1 \end{bmatrix} \quad \text{and} \quad \swarrow \Rightarrow \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

In cases where the co-located BL CU has no MV, *i.e.* the CU is intra or inter-layer (IL) predicted, or both MV components are equal to zero we tested a 2D non-directional boxfilter.

The motion blur compensation is designed as locally adaptive approach. Consequently, the filter design and the decision process whether to use this method or not within the rate-distortion (RD) process are applied on (sub-)CU level.

Figure 1 shows the process of encoding a CU. We added one additional path to the encoder. In addition to the already ex-

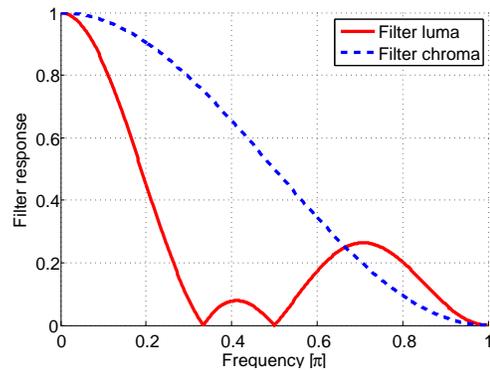


Figure 2. Basefilter frequency responses

isting coding modes (Inter/Intra/IL prediction) inter prediction with a filtered reference picture is tested. Within the encoding process we added one binary “BlurFilter Flag” per CU to signal to the decoder if the reference picture block has to be filtered for proper reconstruction of the current picture block or not. However, this flag is only coded for those CUs for which the motion blur compensation is relevant (*i.e.* inter predicted CUs). This flag is also included in the RD optimization.

## III. EXPERIMENTS

For our simulations we implemented our algorithm in the JCT-VC K0345 [7] software which is a scalable extension of the HEVC reference software HM 8.1.

The evaluation of the proposed approach is based on the JCT-VC *common test conditions* [8] using a  $2 \times$  scalability, the Random Access (RA) and Low Delay (LD-P) configuration and an extended set of test sequences. The latter contains JCT-VC test sequences (Basketball Drive, China Speed, Kimono, Tennis, Traffic) as well as self-recorded test sequences with lots of (varying) motion blur. Our self-recorded sequences ( $1280 \times 720$ , 30 fps, 300 pictures each) were filmed with a camera attached to a bike driver’s helmet. This camera has a fixed focal length. Hence, no blur is introduced by altering the focus. In the following we refer to these sequences as *Bike* sequences. Due to camera shake and the driver looking around to capture the traffic situation these sequences show lots of accelerated motions. The sequence characteristics are demonstrated in Fig. 3.

The usage of the motion blur compensation is illustrated with a usage map next to the reconstructed picture in Fig. 4. For the colored CUs, which are the inter predicted CUs for whom the BlurFilter is potentially applied, the color indicates whether our method has been chosen (yellow/light) by the RD-process or not (magenta/dark). Besides, it is noteworthy that the players’ complex motion can be better described with IL texture prediction than with motion compensated prediction both with or without motion blur compensation.

For the sequence *Bike 12* the bit rate change per picture is analyzed according to Fig. 5. Thereby a negative change indicates a better compression since the PSNR difference between the coding with and without BlurFilter is at the maximum

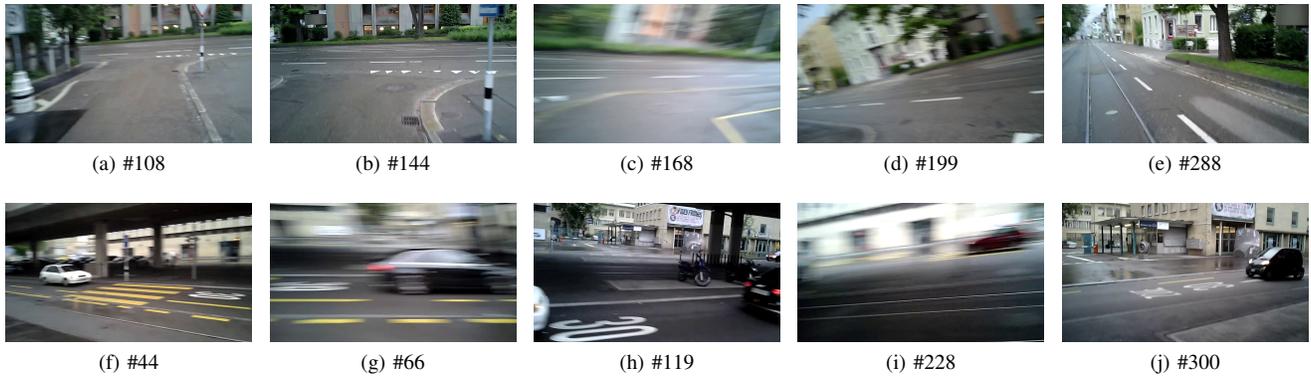


Figure 3. Bike sequences 4 (a–e) and 10 (f–j). [#picture number]

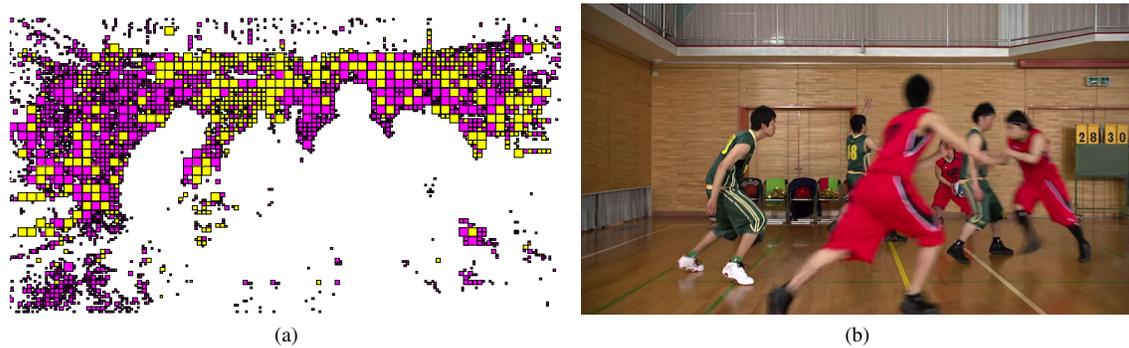


Figure 4. BlurFilter usage map (a) and reconstructed picture (b) for *Basketball Drive*, picture 80 (yellow/light: with BlurFilter, magenta/dark: without BlurFilter)

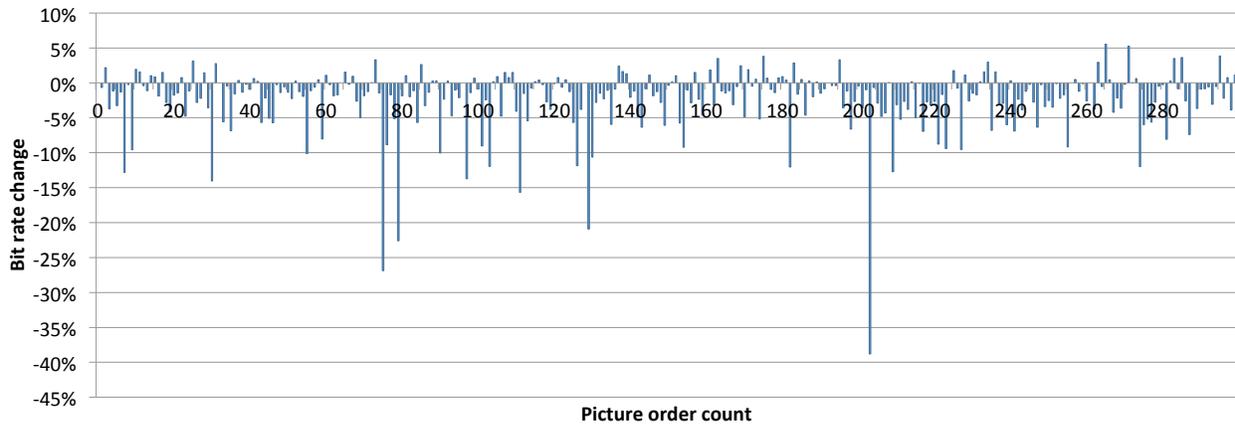


Figure 5. Bit rate change per picture at the same PSNR level, *Bike 12*, RA. Reference: Coding without motion blur compensation

some hundredths of a decibel. The data rate decreases for the majority of pictures, at some points significantly. Nevertheless there is a minor increase of the data rate for some pictures which is caused by the necessity to signal non-usage of the motion blur compensation. By adding additional high-level syntax (HLS) to the slice header or the *Picture Parameter Set* (PPS) the motion blur compensation could be deactivated in such cases.

For the same sequence with the LD-P configuration and a more comprehensive set of QPs the RD curve is shown in Fig. 6.

Thereby for all QPs the curve of the coding with motion blur compensation is above the reference curve, meaning that the coding efficiency was increased. As expected the additional coding gain of the motion blur compensation depends on the amount of motion blur and its relative variation in the sequences. In case of no motion blur, e.g. in the JCT-VC sequence *Traffic*, the gain for the improved residual is just sufficient to compensate the additional signaling flag. Nevertheless there is no loss. Given that lots of changing motion blur appears, e.g. in *Basketball Drive* or the self-recorded *Bike* sequences,

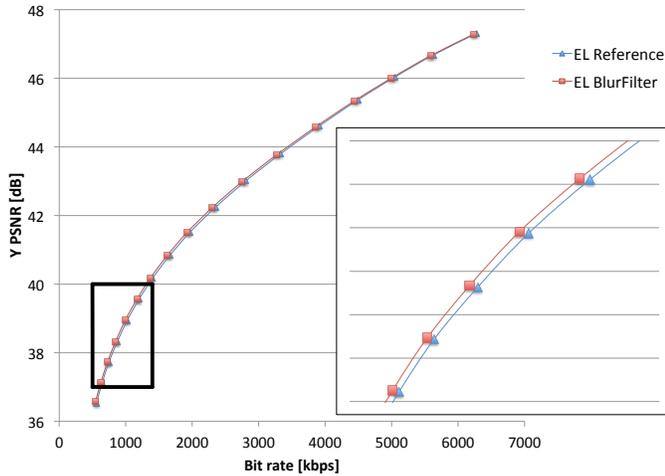


Figure 6. Rate-distortion (RD) plot for *Bike 12* sequence, LD-P, QP 18–34

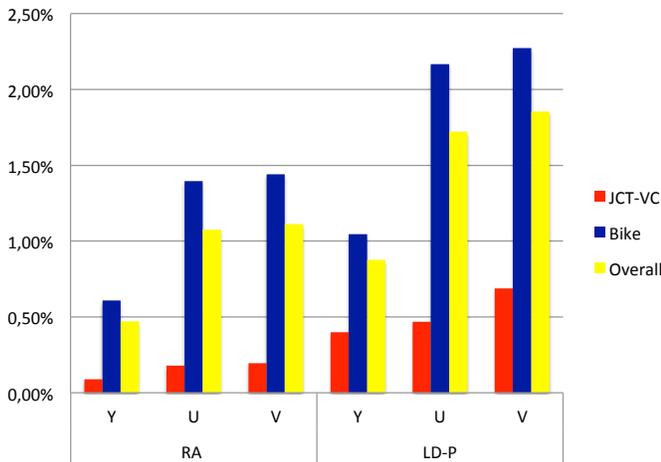


Figure 7. Average BD-Rate gains

Table II  
BD-RATES FOR LOW DELAY P CONFIGURATION

Sequence	EL			EL+BL
	Y	U	V	Y
Basketball Drive	-0.97%	-1.17%	-1.42%	-0.73%
China Speed	-0.04%	-0.16%	-0.41%	0.01%
Kimono	-0.21%	-0.31%	-0.49%	-0.10%
Tennis	-0.69%	-0.77%	-0.82%	-0.50%
Traffic	-0.08%	0.07%	-0.29%	-0.03%
Bike 1	-1.13%	-2.63%	-2.53%	-0.75%
Bike 2	-0.90%	-2.15%	-2.18%	-0.58%
Bike 3	-1.46%	-2.57%	-3.10%	-1.01%
Bike 4	-0.95%	-1.96%	-2.17%	-0.56%
Bike 5	-0.38%	-1.03%	-1.06%	-0.20%
Bike 6	-0.66%	-1.83%	-2.09%	-0.37%
Bike 7	-1.17%	-2.72%	-2.90%	-0.72%
Bike 8	-0.93%	-1.73%	-1.79%	-0.55%
Bike 9	-0.87%	-1.93%	-2.40%	-0.53%
Bike 10	-0.39%	-0.78%	-0.71%	-0.20%
Bike 11	-1.13%	-2.08%	-2.20%	-0.72%
Bike 12	-2.41%	-3.98%	-4.07%	-1.71%
Bike 13	-1.46%	-3.32%	-3.04%	-0.95%
Bike 14	-0.83%	-1.63%	-1.57%	-0.49%

high gains are achieved. Due to the shorter temporal distance between the current picture and the reference pictures in the LD-P configuration the gains are higher than for the RA configuration.

Figure 7 shows the average Bjøntegaard delta (BD)-Rate gains [9]. When averaged over all test sequences the BD-Rate gains (EL) for {Y, U, V} are {0.47%, 1.08%, 1.11%} for RA and {0.88%, 1.72%, 1.86%} for LD-P, respectively. For test sequences containing lots of varying motion blur BD-Rate gains of up to {2.41%, 3.98%, 4.07%} are achieved. Detailed results for single sequences of the LD-P simulations can be found in Table II.

## IV. CONCLUSIONS

The performance of motion compensated prediction is degraded by changing motion blur. To compensate this degradation we present an algorithm based on reference picture filtering. In the context of scalable video coding we derived an adaptive two-dimensional directional filter with a base filter and an already known motion vector from the BL. Thus no additional signaling of filter coefficients is needed. Depending on the amount of varying motion blur our proposed reference picture filtering results in BD-Rate gains of up to {2.41%, 3.98%, 4.07%} for {Y, U, V} compared to the unmodified scalable HM.

## REFERENCES

- [1] B. Bross, W.-J. Han, G. J. Sullivan, J.-R. Ohm, and T. Wiegand, “High Efficiency Video Coding (HEVC) text spec. draft 10 (for FDIS&Consent),” in *JCT-VC Doc. JCTVC-L1003*, 12th Meeting: Geneva, Switzerland, Jan. 2013.
- [2] ISO/IEC and ITU-T, *Recommendation ITU-T H.264 and ISO/IEC 14496-10 (MPEG-4 Part 10): Advanced Video Coding (AVC)-3rd Ed.*, Geneva, Switzerland, Jul. 2004.
- [3] —, *Recommendation ITU-T H.262 and ISO/IEC 13818-2 (MPEG-2 Part 2): Information technology - Generic coding of moving pictures and associated audio information: Video*, Mar. 1995.
- [4] Y. Vatis and J. Ostermann, “Adaptive Interpolation Filter for H.264/AVC,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 2, pp. 179–192, 2009.
- [5] J. Jang, H. Lee, T.-y. Jung, S.-M. Hong, and J. Jeong, “Pre-filtering with locally adaptive filter set for reference frame in video compression,” in *IEEE International Conference on Multimedia and Expo (ICME)*, 2011, pp. 1–6.
- [6] M. Budagavi, “Video compression using blur compensation,” in *IEEE International Conference on Image Processing (ICIP)*, vol. 2, 2005, pp. II–882–5.
- [7] T. Wiegand, J. Park, J. Boyce, H. Schwarz, C. Bartnik, P. Helle, T. Hinz, A. Khairat, H. Kirchhoffer, H. Laksman, D. Marpe, M. Siekmann, J. Stegemann, K. Sühring, K. McCann, J. Kim, C. Kim, J.-H. Min, E. Alshina, A. Alshin, I.-K. Kim, T. Lee, B. Choi, Y. Piao, S. Jeong, S. Lee, Y. Cho, J. Y. Choi, F. C. A. Fernandes, Z. Ma, D. Hong, W. Jang, A. Abbas, and S. Reddy, “JCT-VC K0345: Suggested design of initial software model for scalable HEVC extension proposal by Fraunhofer HHI, Vidyo and Samsung,” 2012.
- [8] X. Li, J. Boyce, P. Onno, and Y. Ye, “L1009: Common Test Conditions and Software Reference Configurations for the Scalable Test Model. Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11. 12th Meeting, Geneva, CH, 14–23 Jan,” 2013.
- [9] G. Bjøntegaard, “Calculation of average PSNR differences between RD curves,” in *ITU-T SG16/Q6 Output Document VCEG-M33*, Austin, Texas, April 2001.