

Adaptive Global Motion Temporal Filtering for High Efficiency Video Coding

Andreas Krutz, *Member, IEEE*, Alexander Glantz, *Student Member, IEEE*, Michael Tok, *Student Member, IEEE*, Marko Esche, *Student Member, IEEE*, and Thomas Sikora, *Senior Member, IEEE*

Abstract—Coding artifacts in video codecs can be reduced using several spatial in-loop filters that are part of the emerging video coding standard High Efficiency Video Coding (HEVC). In this paper, we introduce the concept of global motion temporal filtering. A theoretical framework for a concept combining the temporal overlapping of several noisy versions of the same signal is introduced. This includes a model of the motion estimation error. As an important result, it is shown that an optimum number of frames N for filtering exists. An implementation of the concept based on several versions of the HEVC test model using global motion-compensated temporal filtering shows that significant gains can be achieved.

Index Terms—Higher order motion estimation, temporal filtering, theoretical model, video coding.

I. INTRODUCTION

OUR CHALLENGE is the reduction of coding artifacts in coded videos using temporal filtering approaches. We explore the fact that noise reduction using overlapping of a number of noisy versions of the same signal is very powerful for noise reduction. Because the pixels of consecutive frames of a video sequence are highly correlated, it is possible to use this technique to reduce noise in a video sequence. For that, adjacent frames of the current frame to be filtered are assumed to be multiple versions of the same frame. Usually, motion appears between consecutive frames; therefore, motion estimation has to be performed to identify identical pixels. After motion compensation, the pixels of the current frame can be temporally filtered and the noise can be reduced. The concept of this technique has already been used in the early stage of video processing [1]. Since then, it has been developed in various video processing fields. Temporal filtering is often used in subband and wavelet video coding techniques, where motion compensation is used to align pixels along consecutive frames for filtering [2]–[6]. Another coding technique that uses temporal filtering was developed within the MPEG-4 standard more than a decade ago [7]. Long-term motion-compensated background models were developed to achieve

Manuscript received April 2, 2012; revised July 18, 2012; accepted August 21, 2012. Date of publication October 5, 2012; date of current version January 8, 2013. This paper was recommended by Associate Editor O. C. Au.

The authors are with the Communication Systems Group, Technische Universität Berlin, Berlin 10587, Germany (e-mail: krutz@nue.tu-berlin.de; glantz@nue.tu-berlin.de; tok@nue.tu-berlin.de; esche@nue.tu-berlin.de; sikora@nue.tu-berlin.de).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSVT.2012.2223012

very high coding efficiency for certain video content [8], [9]. The major advantage of this approach is the compact representation of the background information over a long period of the video sequence (e.g., 300 frames). The potential of this technique was outlined in [10] and it was evaluated with H.264/Advanced Video Coding in [11]. In the well-established hybrid video coding approach [12], the concept of motion-compensated temporal filtering is implicitly used in hierarchical B-frames [13] and further discussed, e.g., in [14] and [15], where translational motion vectors are used to filter pixels over a small number of frames.

Higher order motion compensation as used in [8] and [16] gives improved coding efficiency for test sequences having substantial camera parameter changes. The idea of combining these compensation techniques including a superresolution approach within a common hybrid coding scheme was outlined in [17]–[19]. In [19], a promising technique that includes a theoretical model was proposed. Here, higher order motion compensation was applied to improve the prediction efficiency of a hybrid encoder loop. It was shown theoretically that the use of the proposed technique is reasonable. Inspired by this work and further theoretical approaches in image processing, e.g. [20]–[22], we developed a new global motion temporal filtering (GMTF) technique based on a fundamental theoretical background. It has been shown in [23] that the use of higher order motion parameters can lead to a significant improvement depending on the application.

In this paper, the focus lies on improving distorted decoded video data. Block distortions (blocking artifacts) can occur due to the block-based design of such coding systems. Therefore, deblocking filters are used to reduce these distortions, e.g. [24]. This deblocking filter, well established in [12], is a spatial filter only. We propose a temporal deblocking filter design employing parametric motion-compensated temporal filtering. Our approach employs the noise filter as an adaptive decoder postfiltering, where side information is transmitted with the video data to adapt the filter at the receiver. First results were presented in [25]–[27].

Fig. 1 illustrates the concept of GMTF. Adjacent frames of a reference to be filtered are motion compensated using a higher order motion estimation method. In this paper, we use an algorithm based on feature correspondences and a robust Helmholtz estimator along with the well-known 8-parameter perspective motion model [28]. The estimation method is first applied on a frame-by-frame basis. Then, the resulting motion

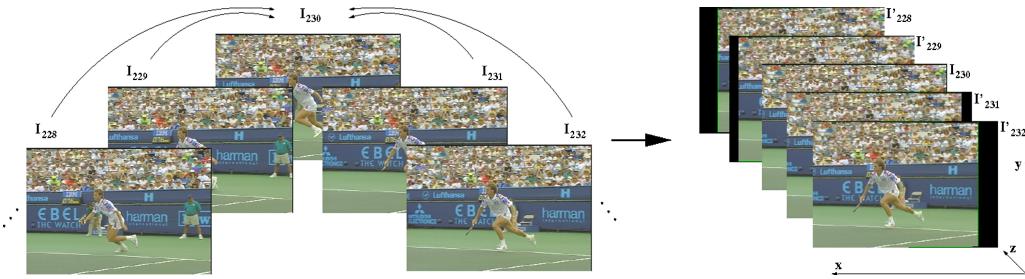


Fig. 1. Concept of GMTF.

parameters are accumulated to build long-term parameters. Every adjacent frame to be used for filtering is compensated with respect to the reference frame. This results in a frame stack, on which the temporal filtering can be performed along every pixel trajectory using averaging.

In this paper, we introduce a theoretical framework with the goal to understand the impact of motion estimation errors on filter performance. One of our fundamental results imply that an optimum filter length N exists for a given motion estimation error variance. This implies that at the encoder for each coded sequence the length N of the filter stack needs to be identified in an optimization procedure. Experimental results confirm the theoretical findings and illustrate the excellent performance of the filtering approach on sequences with global camera motion.

II. A THEORETICAL MODEL FOR GLOBAL MOTION TEMPORAL FILTERING

In this section, we analyze the potential of motion temporal filtering using a motion-compensated approach. As it turns out, the accumulated motion estimation errors will bound the maximum achievable filter gain. This provides useful insight into optimization of practice temporal filtering approaches. It is assumed that a number N of distorted versions Y of an original image X are available in the image stack in Fig. 1. We consider the k th pixel value $y_k(i, j)$, which originates of the original pixel $x(i, j)$ and a white noise signal $n_k(i, j)$

$$y_k(i, j) = x(i, j) + n_k(i, j). \quad (1)$$

Averaging the observations, the mean value $y(i, j)$ has the following properties:

$$y(i, j) = \frac{1}{N} \sum_{k=1}^N y_k(i, j) = x(i, j) + \underbrace{\frac{1}{N} \sum_{k=1}^N n_k(i, j)}_{r(i, j)}. \quad (2)$$

With the noise variance σ_n^2 and the autocorrelation matrix

$$R_{nn} = \begin{pmatrix} \sigma_n^2 & 0 & \dots \\ 0 & \sigma_n^2 & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} \quad (3)$$

the variance of the average noise signal $r(m, n)$ is

$$\sigma_r^2 = E[R^2(i, j)] = \frac{1}{N^2} \sum_{l=1}^N \sigma_n^2 = \frac{\sigma_n^2}{N} \quad (4)$$

and the variance of the noise has been reduced by the factor N . Thus, averaging N quantized pixels y in the image stack in the temporal direction reduces the error variance by a factor of N , provided that the coding noise has the properties of white noise.

To achieve noise filtering in the above image stack, we have to align a number of frames for filtering one reference image. This is conducted using short-term and long-term motion estimation. In our theoretical analysis, we assume that pure translational motion occurs between frames

$$x_n(i, j) = x_{n-1}(i + t_i, j + t_j) \quad (5)$$

where t is the optimal motion estimation parameter. In practice, this operation does not match exactly. Therefore, an estimation error appears, which can be written for our theoretical case as

$$x_{n-1}(i + t_i + \Delta_i, j + t_j + \Delta_j) = x_n(i + \Delta_i, j + \Delta_j) \quad (6)$$

where Δ_x, Δ_y accounts for the estimation error. Thus, the resulting error signal is

$$e_n(i, j) = x_n(i, j) - x_n(i + \Delta_i, j + \Delta_j). \quad (7)$$

The term $x_n(i + \Delta_i, j + \Delta_j)$ is approximated using the Taylor expansion

$$x_n(i + \Delta_i, j + \Delta_j) \approx x_n(i, j) + \nabla x_n^T \cdot \begin{pmatrix} \Delta_i \\ \Delta_j \end{pmatrix}. \quad (8)$$

With this the error signal is given by (9). We can now calculate the error variance using the expected value with the assumption that $\frac{\partial x_n(i, j)}{\partial i}$, Δ_i , $\frac{\partial x_n(i, j)}{\partial j}$, and Δ_j are uncorrelated and statistically independent (10) and (11).

$$\begin{aligned} e_n(i, j) &= x_n(i, j) - \left(x_n(i, j) + \frac{\partial x_n(i, j)}{\partial i} \cdot \Delta_i + \frac{\partial x_n(i, j)}{\partial j} \cdot \Delta_j \right) \\ e_n(i, j) &= -\frac{\partial x_n(i, j)}{\partial x} \cdot \Delta_i - \frac{\partial x_n(i, j)}{\partial j} \cdot \Delta_j \end{aligned} \quad (9)$$

$$\sigma_{e_n}^2 = E[e_n^2] = E \left[\left(-\frac{\partial x_n(i, j)}{\partial i} \Delta_i - \frac{\partial x_n(i, j)}{\partial j} \Delta_j \right)^2 \right] \quad (10)$$

$$\begin{aligned}
\sigma_{e_n}^2 &= \sigma_\Delta^2 \cdot \left\{ E \left[\left(\frac{\partial x_n(i, j)}{\partial i} \right)^2 \right] + E \left[\left(\frac{\partial x_n(i, j)}{\partial j} \right)^2 \right] \right\} \\
&= \sigma_\Delta^2 \cdot \{ E[(x_n(i, j) - x_n(i-1, j))^2] + E[(x_n(i, j) - x_n(i, j-1))^2] \} \\
&= \sigma_\Delta^2 \cdot \{ E[x_n^2 - 2x_n x_n(i-1, j) + x_n^2(i-1, j)] + E[x_n^2 - 2x_n x_n(i, j-1) + x_n^2(i, j-1)] \} \\
&= \sigma_\Delta^2 \cdot \{ \sigma_x^2 - 2 \underbrace{E[x_n(i, j)x_n(i-1, j)]}_{ACF(AR(1))=\sigma_x^2 \cdot \alpha_1^{[1]}} + \sigma_x^2 + \sigma_x^2 - 2 \underbrace{E[x_n(i, j)x_n(i, j-1)]}_{ACF(AR(1))=\sigma_x^2 \cdot \alpha_2^{[1]}} + \sigma_x^2 \} \\
&= \sigma_\Delta^2 \sigma_x^2 \cdot (4 - 2(\alpha_1 + \alpha_2)) \\
&= \underline{2\sigma_\Delta^2 \sigma_x^2 (2 - \alpha_1 - \alpha_2)}
\end{aligned} \tag{13}$$

$$\begin{aligned}
\sigma_{e_n}^2 &= E \left[\left(-\frac{\partial x_n(i, j)}{\partial i} \cdot \Delta_i \right)^2 \right] + 2 \cdot \underbrace{E \left[\frac{\partial x_n(i, j)}{\partial i} \frac{\partial x_n(i, j)}{\partial j} \Delta_i \Delta_j \right]}_{=0} \\
&+ E \left[\left(-\frac{\partial x_n(i, j)}{\partial j} \cdot \Delta_j \right)^2 \right] \sigma_{e_n}^2 = E[\Delta_i^2] \cdot E \left[\left(\frac{\partial x_n(i, j)}{\partial i} \right)^2 \right] \\
&+ E[\Delta_j^2] \cdot E \left[\left(\frac{\partial x_n(i, j)}{\partial j} \right)^2 \right]
\end{aligned} \tag{11}$$

We approximate the derivative of x_n with the first numerical derivative in both the directions

$$\begin{aligned}
\frac{\partial x_n(i, j)}{\partial i} &\approx x_n(i, j) - x_n(i-1, j) \\
\frac{\partial x_n(i, j)}{\partial j} &\approx x_n(i, j) - x_n(i, j-1).
\end{aligned} \tag{12}$$

Using this result and the assumption $E[\Delta_i^2] = E[\Delta_j^2] = E[\Delta^2]$, (11) results in (13), shown at the top of the page, which is the motion prediction error variance due to the inaccurate estimation of one signal from a previous version. We assume that stationary pixel correlation properties on the basis of an AR(1) model with α_1, α_2 are the normalized correlation coefficients in horizontal and vertical directions.

A rough rate-distortion equation for the temporal filtering approach, which incorporates the motion estimation errors, is derived. For that, we consider the distortion-rate (D-R) equation for a Gaussian distributed memoryless signal

$$\sigma_{e_{xq}}^2 = 2^{-2R} \cdot \sigma_x^2 \tag{14}$$

wherein $\sigma_{e_{xq}}^2$ is the quantization error variance, R is the rate, and σ_x^2 is the variance of the signal. This equation stands for the upper bound of a D-R function. In the aligning process for our temporal filtering, we calculate short-term (frame-to-frame) motion parameters for the estimation between consecutive signals. Every aligned signal that represents a “new” version of the signal to be filtered is generated by applying long-term motion parameters according to the reference signal. The long-term motion parameters are calculated using accumulative multiplication of the short-term parameters. We assume that the model for the short-term estimation errors between two consecutive frames derived in (13) can serve for every estimation step. If these errors are now accumulated because of building the long-term motion parameters, the overall error caused by the motion estimation process increases. To model

that, the motion estimation error increases with increasing number of frames taken into account for the temporal filtering process. For our theoretical analysis, we assume that errors that occur due to short-term motion estimation can be simply added.

In summary, two error components of our model for temporal noise reduction apply: the temporally overlapped quantization error represented by its variance $\sigma_{e_{qf}}^2$ and the prediction error variance due to motion estimation $\sigma_{e_m}^2$

$$\begin{aligned}
\sigma_{e_q}^2 &= 2^{-2R} \frac{\sigma_x^2}{N} \\
\sigma_{e_m}^2 &= N \cdot 2\sigma_\Delta^2 \sigma_x^2 (2 - \alpha_1 - \alpha_2).
\end{aligned} \tag{15}$$

If these two error components are statistically independent, the D-R function results

$$\sigma_{e_{tf}}^2 = 2^{-2R} \frac{\sigma_x^2}{N} + N \cdot 2\sigma_\Delta^2 \sigma_x^2 (2 - \alpha_1 - \alpha_2). \tag{16}$$

Here $\sigma_{e_{tf}}^2$ is the filtered version of the quantization error variance. To estimate possible bit rate savings using the temporal filtering approach, the distortion values of (14) and (16) are set equal. The bit rate of the general quantization error shall be R_1 and the bit rate using temporal noise reduction shall be R_2 . An equation of the bit rate R_2 can now be given in

$$\begin{aligned}
\sigma_{e_{xq}}^2 &= \sigma_{e_{tf}}^2 \\
2^{-2R_1} \sigma_x^2 &= 2^{-2R_2} \frac{\sigma_x^2}{N} + N \cdot 2\sigma_\Delta^2 \sigma_x^2 (2 - \alpha_1 - \alpha_2) \\
2^{-2R_2} \frac{1}{N} &= 2^{-2R_1} - N \cdot 2\sigma_\Delta^2 (2 - \alpha_1 - \alpha_2) \\
-2R_2 - ld(N) &= ld\{2^{-2R_1} - N2\sigma_\Delta^2 (2 - \alpha_1 - \alpha_2)\} \\
-2R_2 &= ld\{2^{-2R_1} - N2\sigma_\Delta^2 (2 - \alpha_1 - \alpha_2)\} + ld(N) \\
R_2 &= -\frac{1}{2} \left\{ ld\{2^{-2R_1} - N2\sigma_\Delta^2 (2 - \alpha_1 - \alpha_2)\} + ld(N) \right\}.
\end{aligned} \tag{17}$$

For a real coding and filtering environment, the equation in (17) makes sense only if the term $2^{-2R_1} - N2\sigma_\Delta^2 (2 - \alpha_1 - \alpha_2)$ is greater than zero. This leads to

$$\begin{aligned}
0 &< 2^{-2R_1} - N2\sigma_\Delta^2 (2 - \alpha_1 - \alpha_2) \\
2^{-2R_1} &> N2\sigma_\Delta^2 (2 - \alpha_1 - \alpha_2) \\
-2R_1 &> ld(N2\sigma_\Delta^2 (2 - \alpha_1 - \alpha_2)) \\
R_1 &> -\frac{1}{2} ld(N2\sigma_\Delta^2 (2 - \alpha_1 - \alpha_2)).
\end{aligned} \tag{18}$$

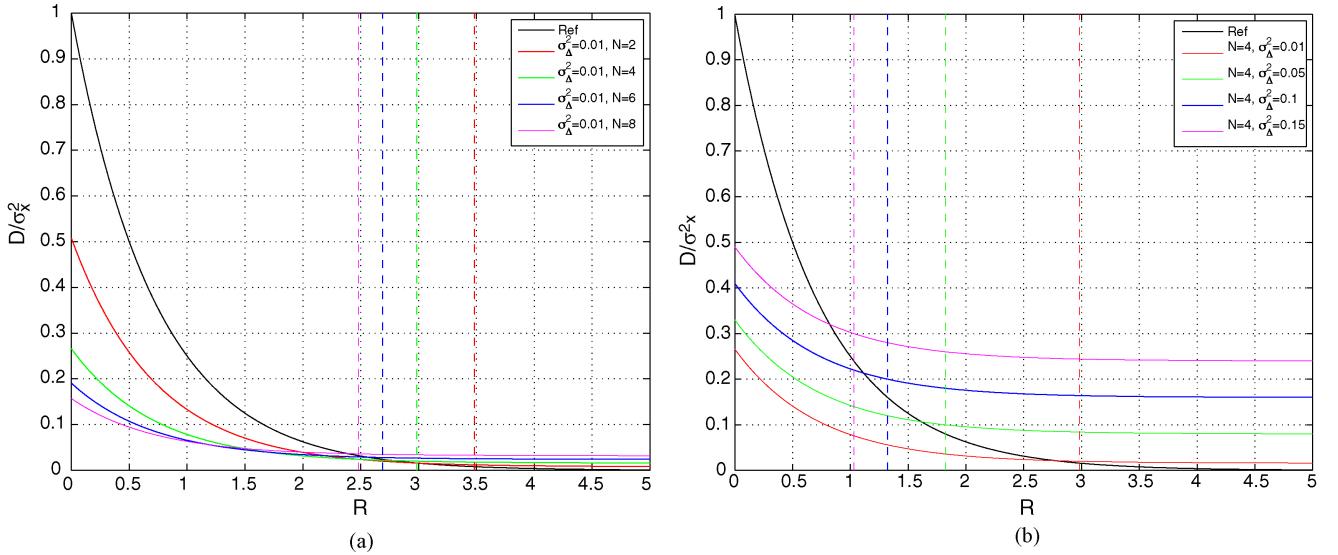


Fig. 2. Curves of theoretical rate-distortion function (16). (a) Variable frame number N . (b) Variable motion estimation error variance.

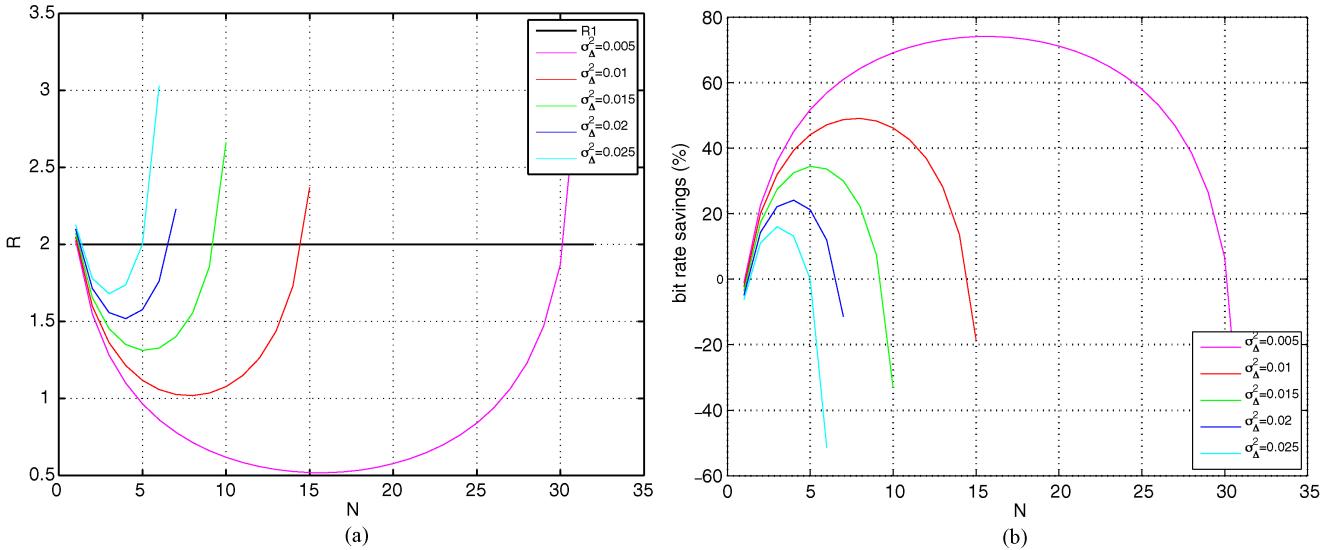


Fig. 3. Theoretical R - N function (17). (a) R - N -curve, variable motion estimation error variance. (b) Bit rate savings ΔR - N -curve, variable motion estimation error variance.

The number of frames for temporal filtering N is greater or equal to 1, thus $ld(N) \geq 0$. The upper limit for N and the error variance of the pixel difference due to motion estimation σ_{Δ}^2 are

$$\begin{aligned} 0 &\leqslant 2^{-2R_1} - N2\sigma_{\Delta}^2(2 - \alpha_1 - \alpha_2) \\ 2^{-2R_1} &\leqslant N2\sigma_{\Delta}^2(2 - \alpha_1 - \alpha_2) \\ N &\geqslant \frac{2^{-2R_1}}{2\sigma_{\Delta}^2(2 - \alpha_1 - \alpha_2)} \end{aligned} \quad (19)$$

$$\sigma_{\Delta}^2 \geqslant \frac{2^{-2R_1}}{2N(2 - \alpha_1 - \alpha_2)}. \quad (20)$$

Fig. 2 shows curves for the theoretical D-R-function developed above. It can be seen in Fig. 2(a) that by increasing the number of frames to be used for filtering, the distortion decreases at lower bit rates. An increased motion estimation

error has a higher impact at higher bit rates, which causes an increase of the distortion in that range. This is illustrated by the dashed vertical lines, where the model of our temporal filtering method crosses the reference model without filtering. Fig. 2(b) shows theoretical curves with variable motion estimation error variance and a fixed number of frames N . This case illustrates that, with an increase of motion estimation errors, the gain of the filtering approach decreases rapidly. Fig. 3 depicts important bounds on rate- N curves. It can be seen that there exist an optimal number of frames for filtering that yields the best improvement. As an important result of our theoretical analysis we predict that a maximum bitrate saving [see Fig. 3(b)] is achievable at the optimum number of frames N . This implies that at the encoder the optimum filter length N needs to be identified. In order to verify this important finding, we conducted experiments on

three test sequences: *Biathlon* (352×288 , 200 frames) taken from a German television broadcast, *Birds* (720×576 , 110 frames), and *Desert* (720×400 , 240 frames) from the BBC documentary “Planet Earth” (also included in the test data set in Section VI). Fig. 4 shows $\Delta R - N$ results for three test sequences. It can be seen that the experimental curves exhibit the postulated optimum saving as in Fig. 3(b). The curves have different shapes depending on the possible bit rate saving that can be achieved. The possible bit rate savings increase with a decreasing motion estimation error variance. The motion estimation performs very well with the test sequences *Birds* and *Desert*, but has more faults with *Biathlon*. Therefore, less bit rate savings are possible with “Biathlon.”

The above aspect motivates the design of an encoder-assisted postfiltering approach, where an optimal number of frames can be found at the encoder to get the best visual quality at the decoder after GMTF postprocessing. We will show in Section VI that our theoretical model approximates the real behavior of the GMTF approach very well using several versions of the High Efficiency Video Coding (HEVC) test model.

III. MOTION MODEL COMPRESSION

We employ higher order motion estimation to generate an image stack with globally aligned adjacent frames at the decoder. In an encoder-assisted video coding environment, we have thus to transmit motion parameters to the decoder. For that, we developed an efficient encoding scheme for the higher order motion parameters.

A single higher order motion parameter set, or parametric motion model (PMM), consists of eight parameters, each represented by a 32-bit single precision floating point value. Since the parameters m_0, \dots, m_7 are highly correlated and have different ranges of values, and as the two perspective parameters m_6 and m_7 are very sensitive to quantization, each PMM is transformed to a set of four corner motion vectors at the positions $(\pm x_{\text{res}}/2, \pm y_{\text{res}}/2)^T$ [7]

$$\begin{pmatrix} x' \cdot w' \\ y' \cdot w' \\ w' \end{pmatrix} = \mathbf{H} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (21)$$

$$\mathbf{V}_p = \mathbf{p}' - \mathbf{p}. \quad (22)$$

These vectors are more robust to quantization and can easily be transformed back to a perspective model at the decoder side. Additionally, each vector is highly correlated with its temporal predecessor so that differential coding in combination with exponential Golomb coding is used for redundancy reduction. The whole coding process for the PMMs is illustrated in Fig. 5.

As quantization step size for the corner motion vectors, $\frac{1}{32}$ was found to be a good trade-off between bit rate and model quality.

IV. ADAPTIVE GLOBAL MOTION TEMPORAL FILTERING USING THE HEVC TEST MODEL HM

It has been shown above that the exact amount of frames to be used for an optimal filtering is a critical issue. The

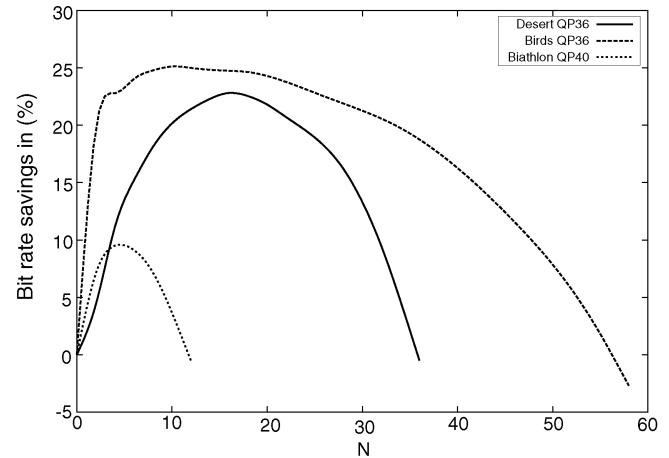


Fig. 4. Bit rate savings (ΔR) versus number of frames (N) with variable motion estimation error variance. Practical ΔR - N curves.

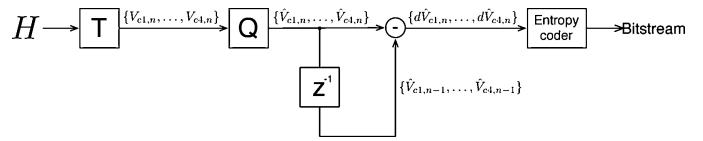


Fig. 5. Encoding process of PMMs.

theoretical model shows that for each sequence and noise level there exists an optimal number of aligned frames for filtering. To apply this to practice, we designed an encoder-assisted postfiltering scheme. Here, the higher order motion estimation is performed in combination with a quality optimization method to find the optimal number of frames for filtering. This optimal number is determined for each frame of the video sequence. Fig. 6 shows the encoder and decoder with the proposed adaptive GMTF using the HEVC Test Model HM. It can be seen that the GMTF approach is independent from the codec itself. The scheme operates as follows. At the encoder, global motion estimation (GME) and encoding of the video sequence is performed in parallel. Then, the resulting higher order motion parameters are compressed with the method described in the previous section. To simulate the same conditions of the decoder, the motion parameters and the encoded video sequence are locally decoded and stored in a buffer. Afterward, GMTF is performed using the decoded frames and decompressed motion parameters, starting with two frames ($n = 2$). Having the reference frame temporally filtered, a mean-square-error (MSE)-based block-wise decision is determined to evaluate if the spatially filtered block or the block filtered with GMTF results in a lower MSE, which is then used to reconstruct the final frame. Here, a fixed block size is used. For standard definition (SD) resolution and lower, the block size of 64×64 pixel is used. For high definition (HD) videos, the block size is set to 128×128 . Since the temporal filtering is only performed according to the camera motion, regions that do not correspond to higher order motion parameters, e.g. foreground objects, remain filtered spatially using the tools inside the reference encoder. The final spatially/GMTF-filtered reconstructed frame is compared with its original using the peak-signal-to-noise ratio (PSNR). This

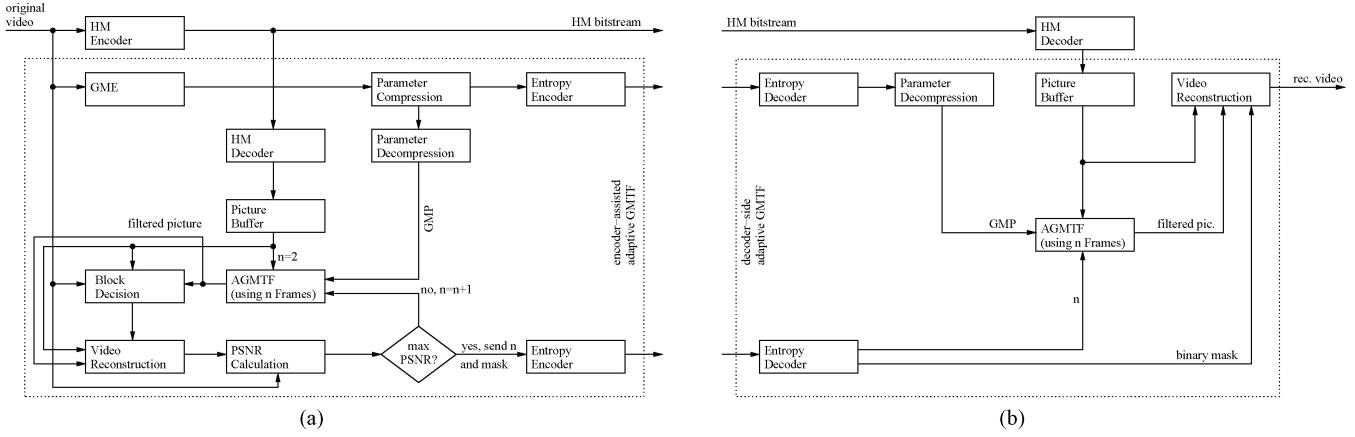


Fig. 6. HM with adaptive temporal filtering (GMTF). (a) Encoder. (b) Decoder.

TABLE I
RESULTS WITH HEVC TEST MODEL 1.0 (HM1.0) VERSUS HM1.0 + AGMTF

Sequence name	Size	FPS	Frames	BD-rate (%) QP _{high}	BD-PSNR (dB) QP _{high}	BD-rate (%) QP _{low}	BD-PSNR (dB) QP _{low}
<i>Biathlon</i>	352×288	25	200	0.1	0.0	0.8	0.0
<i>Birds</i>	720×576	25	110	-1.1	0.0	0.0	0.0
<i>Blue sky</i>	1920×1080	25	218	-2.0	0.1	-2.8	0.1
<i>BQSquare</i>	416×240	60	600	-1.0	0.0	-0.7	0.0
<i>BQTerrace</i>	1920×1080	60	600	-2.7	0.0	-1.2	0.0
<i>City</i>	1280×720	60	250	-6.0	0.2	-6.3	0.2
<i>Desert</i>	720×400	25	240	-0.4	0.0	0.0	0.0
<i>Jets1</i>	1280×720	60	300	-4.8	0.1	-2.5	0.1
<i>PartyScene</i>	832×480	50	500	-0.3	0.0	-0.5	0.0
<i>Station2</i>	1920×1080	25	313	-7.7	0.2	-6.5	0.2
<i>Sunflower</i>	1920×1080	25	500	-1.9	0.1	-2.9	0.1
<i>Waterfall</i>	704×480	25	260	-6.0	0.2	-4.6	0.2
Mean				-2.8	0.1	-2.3	0.1

procedure is performed until a predefined maximum number of frames for filtering is reached. In this scheme, we set the maximum number to 40. The number of frames for GMTF that results in the highest PSNR value is transmitted to the decoder along with a flag bit for each block to indicate whether the block is filtered using GMTF or with the deblocking filter. The numbers of frames for filtering are encoded using a simple Exp-Golomb code and the flag bit for each block is transmitted without any further processing. At the decoder, the side information is decompressed and the video data is decoded and stored in a frame buffer. Adaptive GMTF (AGMTF) is performed as a postprocessing step using the number of frames for each decoded frame, flag bits, and motion parameters to enhance the quality of the decoded frame.

V. EXPERIMENTAL RESULTS

To compare the proposed method with the current state-of-the-art method, we used several versions of the HEVC Test Model HM (v1.0, v4.0, v5.1, v8.0) [29]–[32]. For the encoding settings, the random access high-efficiency (RAHE) case was considered as defined in [33]. We used two sets of QP values for encoding, i.e. $QP_{high} = \{22, 27, 32, 37\}$ and $QP_{low} = \{27, 32, 37, 42\}$, to evaluate a large bit rate range. For measuring the performance we used the widely known BD rate [34]. The results of the conducted experiments are summarized

in Tables I–IV. It is emphasized that only sequences containing complex camera motion are considered in this paper to see if the proposed concept still achieves a gain on top of HEVC. It is obvious that GME has no effect on sequences where no camera motion exists. In these cases, a flag can be provided to switch off the GME or other motion estimation methods may be used to set up temporal filtering. These are key issues for further work. AGMTF achieves a significant improvement over any of the considered HEVC test model versions with the test sequences considered. A bit rate reduction of up to 7.7% and 2.8% in average for HM 1.0, up to 8.9% and 3.0% an average for HM 4.0, up to 9.0% and 3.3% in average for HM 5.1, and up to 8.5% and 3.5% in average for HM 8.0 and are achieved. It is also interesting to note that the average bit rate reduction in a higher bit rate range is almost the same as in the lower bit rate range for every code base used. This means that despite the highly efficient spatial in-loop filters of HEVC there is an amount of noise that cannot be reduced by these filters. Adding the temporal filtering on top of the spatial filters results in a significant enhancement of the decoded frame quality. A number of parameters in the proposed scheme, including the fixed block size, can be improved and it is expected that further quality enhancements and bit rate reductions are possible.

Table V shows the average optimal number of frames for filtering (\bar{N}_{opt}) for each test sequence and the computational time of each part of the AGMTF-algorithm comparing to

TABLE II
RESULTS WITH HEVC TEST MODEL 4.0 (HM4.0) VERSUS HM4.0 + AGMTF

Sequence name	Size	FPS	Frames	BD-rate (%) QP _{high}	BD-PSNR (dB) QP _{high}	BD-rate (%) QP _{low}	BD-PSNR (dB) QP _{low}
<i>Biathlon</i>	352 × 288	25	200	0.2	0.0	0.9	0.0
<i>Birds</i>	720 × 576	25	110	-1.2	0.0	-0.2	0.0
<i>Blue sky</i>	1920 × 1080	25	218	-2.0	0.1	-2.7	0.1
<i>BQSquare</i>	416 × 240	60	600	-0.5	0.0	-0.4	0.0
<i>BQTerrace</i>	1920 × 1080	60	600	-2.3	0.0	-3.4	0.1
<i>City</i>	1280 × 720	60	250	-5.8	0.2	-7.1	0.3
<i>Desert</i>	720 × 400	25	240	-0.5	0.0	-0.7	0.0
<i>Jets1</i>	1280 × 720	60	300	-5.3	0.1	-3.7	0.1
<i>PartyScene</i>	832 × 480	50	500	-0.4	0.0	-0.6	0.0
<i>Station2</i>	1920 × 1080	25	313	-8.9	0.2	-7.8	0.3
<i>Sunflower</i>	1920 × 1080	25	500	-2.2	0.1	-3.2	0.1
<i>Waterfall</i>	704 × 480	25	260	-7.2	0.2	-6.5	0.2
Mean				-3.0	0.1	-3.0	0.1

TABLE III
RESULTS WITH HEVC TEST MODEL HM5.1 VERSUS HM5.1 + AGMTF

Sequence name	Size	FPS	Frames	BD-rate (%) QP _{high}	BD-PSNR (dB) QP _{high}	BD-rate (%) QP _{low}	BD-PSNR (dB) QP _{low}
<i>Biathlon</i>	352 × 288	25	200	0.2	0.0	0.9	0.0
<i>Birds</i>	720 × 576	25	110	-1.5	0.0	-0.6	0.0
<i>Blue sky</i>	1920 × 1080	25	218	-2.0	0.1	-2.7	0.1
<i>BQSquare</i>	416 × 240	60	600	-0.6	0.0	-0.5	0.0
<i>BQTerrace</i>	1920 × 1080	60	600	-2.3	0.0	-3.3	0.1
<i>City</i>	1280 × 720	60	250	-6.4	0.2	-7.6	0.3
<i>Desert</i>	720 × 400	25	240	-0.7	0.0	-0.8	0.0
<i>Jets1</i>	1280 × 720	60	300	-7.1	0.1	-4.7	0.2
<i>PartyScene</i>	832 × 480	50	500	-0.4	0.0	-0.6	0.0
<i>Station2</i>	1920 × 1080	25	313	-9.0	0.2	-7.8	0.3
<i>Sunflower</i>	1920 × 1080	25	500	-2.7	0.1	-3.9	0.1
<i>Waterfall</i>	704 × 480	25	260	-7.2	0.2	-6.4	0.2
Mean				-3.3	0.1	-3.2	0.1

TABLE IV
RESULTS WITH HEVC TEST MODEL HM8.0 VERSUS HM8.0 + AGMTF

Sequence name	Size	FPS	Frames	BD-rate (%) QP _{high}	BD-PSNR (dB) QP _{high}	BD-rate (%) QP _{low}	BD-PSNR (dB) QP _{low}
<i>Biathlon</i>	352 × 288	25	200	0.2	0.0	1.0	0.0
<i>Birds</i>	720 × 576	25	110	-2.0	0.1	-0.8	0.0
<i>Blue sky</i>	1920 × 1080	25	218	-2.3	0.1	-2.7	0.1
<i>BQSquare</i>	416 × 240	60	600	-1.0	0.0	-0.4	0.0
<i>BQTerrace</i>	1920 × 1080	60	600	-2.7	0.0	-3.3	0.1
<i>City</i>	1280 × 720	60	250	-6.3	0.2	-7.1	0.3
<i>Desert</i>	720 × 400	25	240	-0.9	0.0	-1.1	0.0
<i>Jets1</i>	1280 × 720	60	300	-7.0	0.1	-4.7	0.2
<i>PartyScene</i>	832 × 480	50	500	-0.6	0.0	-0.6	0.0
<i>Station2</i>	1920 × 1080	25	313	-8.5	0.2	-7.2	0.3
<i>Sunflower</i>	1920 × 1080	25	500	-3.6	0.1	-4.5	0.1
<i>Waterfall</i>	704 × 480	25	260	-6.9	0.2	-6.2	0.2
Mean				-3.5	0.1	-3.1	0.1

TABLE V

AVERAGE OPTIMAL NUMBER OF FRAMES FOR FILTERING PER QP VALUE AND COMPLEXITY ANALYSIS USING HM 5.1 WITH AGMTF

Sequence	QP	\bar{N}_{opt}	Enc (s)	Dec [s]	FT (ms)	Est. (ms)	Warp (ms)	Enc _{total} [s]	Dec _{total} [s]	Factor _{enc}	Factor _{dec}
<i>Biathlon</i>	22	6.2	1032.7	2.3	70.6	0.6	42.0	1383.1	54.4	1.3	24.0
	27	7.4	900.2	1.9	70.6	0.6	42.0	1250.7	64.1	1.4	33.7
	32	7.1	796.4	1.6	70.6	0.6	42.0	1146.8	61.2	1.4	39.0
	37	7.3	719.2	1.3	70.6	0.6	42.0	1069.7	62.7	1.5	47.1
	42	7.7	665.7	1.2	70.6	0.6	42.0	1016.2	66.0	1.5	53.2
<i>Birds</i>	22	6.9	1901.5	4.1	355.9	3.0	175.6	2713.5	137.4	1.4	33.6
	27	17.9	1622.3	3.1	355.9	3.0	175.6	2434.3	348.8	1.5	111.8
	32	23.3	1485.1	2.8	355.9	3.0	175.6	2297.1	452.8	1.5	159.5
	37	25.4	1403.9	2.5	355.9	3.0	175.6	2215.9	493.1	1.6	197.2
	42	24.4	1327.4	2.4	355.9	3.0	175.6	2139.4	473.6	1.6	200.7
<i>Blue sky</i>	22	6.3	18440.6	47.8	2325.3	40.5	913.6	26923.3	1302.6	1.5	27.2
	27	4.5	16172.6	38.9	2325.3	40.5	913.6	24655.3	935.2	1.5	24.0
	32	6.2	15031.7	35.8	2325.3	40.5	913.6	23514.4	1270.7	1.6	35.5
	37	12.9	14447.9	35.0	2325.3	40.5	913.6	22930.7	2604.3	1.6	74.5
	42	20.2	14019.7	32.5	2325.3	40.5	913.6	22502.4	4055.8	1.6	124.9
<i>BQSquare</i>	22	3.7	2872.5	6.8	60.5	0.4	41.6	3908.1	99.2	1.4	14.5
	27	7.0	2370.0	5.3	60.5	0.4	41.6	3405.5	180.1	1.4	33.9
	32	11.7	2093.6	4.6	60.5	0.4	41.6	3129.1	296.8	1.5	64.5
	37	16.1	1941.9	4.1	60.5	0.4	41.6	2977.5	406.2	1.5	98.6
	42	19.2	1851.9	3.3	60.5	0.4	41.6	2887.4	482.8	1.6	146.8
<i>BQTerrace</i>	22	11.5	64746.8	163.0	2259.7	40.2	895.8	87627.0	6344.4	1.4	38.9
	27	11.9	47324.5	102.9	2259.7	40.2	895.8	70204.7	6499.2	1.5	63.2
	32	16.0	41168.0	83.8	2259.7	40.2	895.8	64048.2	8683.9	1.6	103.7
	37	22.6	38893.2	82.4	2259.7	40.2	895.8	61773.5	12230.0	1.6	148.5
	42	25.9	37453.7	66.4	2259.7	40.2	895.8	60334.0	13987.9	1.6	210.5
<i>City</i>	22	13.1	26828.9	55.1	935.2	10.4	399.2	31057.3	1362.5	1.2	24.7
	27	15.3	22073.6	43.4	935.2	10.4	399.2	26302.0	1570.3	1.2	36.2
	32	22.9	20015.8	36.7	935.2	10.4	399.2	24244.2	2322.1	1.2	63.3
	37	25.7	18630.0	31.8	935.2	10.4	399.2	22858.3	2596.7	1.2	81.6
	42	27.8	17660.5	23.8	935.2	10.4	399.2	21888.9	2798.2	1.2	117.8
<i>Desert</i>	22	7.1	3433.1	7.5	218.6	1.8	120.7	4645.2	213.2	1.4	28.5
	27	12.1	2873.2	5.7	218.6	1.8	120.7	4085.3	356.3	1.4	62.4
	32	16.4	2549.1	4.5	218.6	1.8	120.7	3761.1	479.7	1.5	107.3
	37	18.8	2350.0	3.7	218.6	1.8	120.7	3562.0	548.5	1.5	149.0
	42	20.4	2191.1	3.1	218.6	1.8	120.7	3403.1	594.2	1.6	192.9
<i>Jets1</i>	22	21.2	8837.7	17.6	977.3	9.7	393.5	13855.4	2520.0	1.6	143.5
	27	27.5	7844.6	14.3	977.3	9.7	393.5	12862.3	3260.4	1.6	228.3
	32	29.5	7526.2	12.8	977.3	9.7	393.5	12543.9	3495.0	1.7	272.6
	37	31.0	7300.8	11.5	977.3	9.7	393.5	12318.4	3670.7	1.7	320.6
	42	32.2	7122.7	9.4	977.3	9.7	393.5	12140.4	3810.3	1.7	405.8
<i>PartyScene</i>	22	3.8	10640.9	24.4	332.7	2.8	170.3	14214.0	347.9	1.3	14.3
	27	5.4	9020.7	20.1	332.7	2.8	170.3	12593.9	479.8	1.4	23.9
	32	10.8	7915.3	17.7	332.7	2.8	170.3	11488.5	937.2	1.5	52.8
	37	18.8	7180.9	15.7	332.7	2.8	170.3	10754.1	1616.3	1.5	102.9
	42	24.9	6642.9	11.1	332.7	2.8	170.3	10216.1	2131.0	1.5	192.5
<i>Station2</i>	22	17.8	22979.3	52.6	2208.3	44.3	886.7	34785.7	4992.7	1.5	94.8
	27	22.4	20395.8	40.4	2208.3	44.3	886.7	32202.1	6257.1	1.6	154.9
	32	26.3	19339.5	36.3	2208.3	44.3	886.7	31145.8	7335.4	1.6	202.1
	37	28.1	18754.6	36.2	2208.3	44.3	886.7	30560.9	7834.9	1.6	216.4
	42	30.0	18250.0	30.2	2208.3	44.3	886.7	30056.4	8356.1	1.6	277.2
<i>Sunflower</i>	22	7.8	43397.5	92.0	2261.2	43.4	899.8	62546.0	3601.2	1.4	39.2
	27	10.6	39966.1	85.1	2261.2	43.4	899.8	59114.6	4854.1	1.5	57.1
	32	15.9	37543.3	76.5	2261.2	43.4	899.8	56691.8	7230.0	1.5	94.5
	37	22.4	35514.5	68.8	2261.2	43.4	899.8	54663.0	10146.6	1.5	147.6
	42	27.2	33608.7	54.6	2261.2	43.4	899.8	52757.2	12292.0	1.6	225.3
<i>Waterfall</i>	22	24.2	3681.8	7.6	255.6	2.2	144.1	5247.8	914.4	1.4	120.8
	27	27.5	3073.0	6.0	255.6	2.2	144.1	4639.0	1036.6	1.5	171.9
	32	28.4	2809.1	5.0	255.6	2.2	144.1	4375.1	1069.3	1.6	213.4
	37	29.4	2663.3	4.5	255.6	2.2	144.1	4229.3	1106.2	1.6	246.4
	42	31.2	2535.2	3.7	255.6	2.2	144.1	4101.2	1172.9	1.6	314.5

TABLE VI
AVERAGE BD-RATES AND AVERAGE NUMBER OF FRAMES
FOR FILTERING (AVNoF) COMPARING SEVERAL HM
VERSIONS VERSUS HM WITH AGMTF

HM versus HM+AGMTF	HM1.0+ AGMTF	HM4.0+ AGMTF	HM5.1+ AGMTF	HM8.0 AGMTF
BD rate in % (high)	-2.8	-3.0	-3.3	-3.5
BD rate in % (low)	-2.3	-3.0	-3.2	-3.1
Average number of frames for GMTF	16.2	16.4	17.7	17.6

HM 5.1. For the complexity analysis, computational times are measured for Enc = Encoding time HM encoder only, Dec = Decoding decoder time HM only, FT = Feature tracking time, Est = Estimation time, Warp = Time for the Warping process of one image, Enc_{total} = Encoding time HM encoder with AGMTF, and Dec_{total} = Decoding time HM decoder with AGMTF (2.8 GHz AMD Opteron 8439 SE with 64 GB of RAM). Having this and N_{opt}, we can calculate the additional computational time in average for using the proposed AGMTF method and show the Factor of the complexity increase of AGMTF at the encoder (Factor_{enc}) (when calculating the optimal number of frames for filtering) and at the decoder (Factor_{dec}) when the filtering is performed as a postprocessing step. Having a look at Factor_{enc} it can be seen that the complexity of the encoder only increases by a factor of 1.5 in average. However, the complexity of the decoder is drastically increased due to the warping process. Of course, as described above, we only would like to introduce the concept of long temporal filtering based on higher order motion models and our code is not optimized. On the other hand, this analysis brings us to a critical aspect for further work, i.e. reducing the complexity of the warping process with multiple frames. One approach could be that our algorithm can be implemented in a GPU so that the multiple warping process can be parallelized.

Considering only N_{opt}, the theoretical analysis can be confirmed that with increasing coding noise more frames for temporal filtering are needed. This effect can also be confirmed when the bit rate reductions and average optimal number of frames for GMTF are compared directly as shown in Table VI. It can be seen that the combination with the latest versions of the HEVC test model (HM5.1 and HM8.0) even leads to increased bit rate reductions when using the proposed long-term temporal filtering approach. Additionally, the amount of gain is also proportional to the amount of optimal frame number for filtering. That means that the more frames can be used for temporal filtering the more coding gain is possible. This motivates further work especially in an exact modeling of the motion between consecutive frames.

VI. CONCLUSION AND FURTHER WORK

We proposed a long-term motion temporal filtering approach as a postprocessing step. First, we introduced an approach for theoretically modeling the proposed filtering scheme. The motivation was to evaluate the potential of the filtering method and to derive practical implementation. It was assumed that the quantization noise can be treated as white noise so that temporal overlapping of several noisy versions of the same

signal can be applied for noise reduction. As an important result we showed that an optimal number of frames for filtering needs to be derived at the encoder. The resulting equation predicted the behavior of the filtering approach in a coding environment very well. We showed that the coding performance of several versions of the current emerging HEVC video coding standard can be improved by applying the proposed long-term temporal filtering as postprocessing. A critical issue is the complexity of the proposed method. To reduce it at the encoder, a scheme for a prediction of the optimal number of frames can be developed. Having this, it is expected that the complexity of the encoder is only slightly increased. A more challenging issue is the reduction of the complexity at the decoder. Here, the analysis has been shown that the proposed method increases the reference decoder drastically. Therefore, a method has to be developed, e.g. bringing the warping process on a GPU, to tackle this problem.

REFERENCES

- [1] E. Dubois and S. Sabri, "Noise reduction in image sequences using motion-compensated temporal filtering," *IEEE Trans. Commun.*, vol. 32, no. 7, pp. 826–831, Jul. 1984.
- [2] J.-R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 559–571, Sep. 1994.
- [3] K. Ramchandran, M. Vetterli, M., and C. Herley, "Wavelets, subband coding, and best bases," *Proc. IEEE*, vol. 84, no. 4, pp. 541–560, Apr. 1996.
- [4] S.-T. Hsiang, J. Woods, and J.-R. Ohm, "Invertible temporal subband/wavelet filter banks with half-pixel-accurate motion compensation," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1018–1028, Aug. 2004.
- [5] V. Seran and L. Kondi, "New temporal filtering scheme to reduce delay in wavelet-based video coding," *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2927–2935, Dec. 2007.
- [6] L. Jovanov, A. Pizurica, S. Schulte, P. S. A. Munteanu, E. Kerre, and W. Philips, "Combined wavelet-domain and motion-compensated video denoising based on video codec motion estimation methods," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 3, pp. 417–421, Mar. 2009.
- [7] T. Sikora, "The MPEG-4 video standard verification model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 19–31, Feb. 1997.
- [8] A. Smolic, T. Sikora, and J.-R. Ohm, "Long-term global motion estimation and its application for sprite coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 8, pp. 1227–1242, Dec. 1999.
- [9] G. Ye, M. Pickering, M. Frater, and J. Arnold, "A robust approach to super-resolution sprite generation," in *Proc. ICIP*, Sep. 2005, pp. 897–900.
- [10] T. Sikora, "Trends and perspectives in image and video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 6–17, Jan. 2005.
- [11] A. Krutz, A. Glantz, M. Frater, and T. Sikora, "Rate-distortion optimized video coding using automatic sprites," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 7, pp. 1309–1321, Nov. 2011.
- [12] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 302–311, Jul. 2003.
- [13] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical B pictures and MCTF," in *Proc. IEEE ICME*, Jul. 2006, pp. 1929–1932.
- [14] X. Wang, M. Karczewicz, J. Ridge, and Y. Bao, "Simplified update step of motion-compensated temporal filtering for video coding," in *Proc. Picture Coding Symp.*, Apr. 2006.
- [15] D. T. Vo and T. Q. Nguyen, "Quality enhancement for motion JPEG using temporal redundancies," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 5, pp. 609–619, May 2008.
- [16] F. Dufaux and J. Konrad, "Efficient, robust, and fast global motion estimation for video coding," *IEEE Trans. Image Process.*, vol. 9, no. 3, pp. 497–501, Mar. 2000.
- [17] A. Smolic, Y. Vatis, and T. Wiegand, "Long-term global motion compensation applying super-resolution mosaics," in *Proc. IEEE ISCE*, Sep. 2002.

- [18] A. Smolic, Y. Vatis, H. Schwarz, and T. Wiegand, "Long-term global motion compensation for advanced video coding," in *Proc. 10th ITG-Fachtagung Dortmunder Fernsehseminar*, Sep. 2003, pp. 213–216.
- [19] A. Smolic, Y. Vatis, H. Schwarz, and T. Wiegand, "Improved H.264/AVC coding using long-term global motion compensation," in *Proc. IS&T/SPIE Symp. VCIP*, Jan. 2004, pp. 343–354.
- [20] A. Jain, "Advances in mathematical models for image processing," *Proc. IEEE*, vol. 69, no. 5, pp. 502–534, May 1981.
- [21] R. Mester, "A system-theoretical view on local motion estimation," in *Proc. 5th IEEE Southwest Symp. Image Anal. Interpretation*, 2002, pp. 201–205.
- [22] M. Agostini and M. Antonini, "Theoretical model of the coding error in MCWT video coders," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2006, pp. 1885–1888.
- [23] G. Dane and T. Nguyen, "The effect of global motion parameter accuracies on the efficiency of video coding," in *Proc. ICIP*, vol. 5. Oct. 2004, pp. 3359–3362.
- [24] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz, "Adaptive deblocking filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 614–619, Jul. 2003.
- [25] A. Glantz, A. K. M. Haller, and T. Sikora, "Video coding using global motion temporal filtering," in *Proc. IEEE ICIP*, Nov. 2009, pp. 1053–1056.
- [26] A. Krutz, A. Glantz, and T. Sikora, "Theoretical consideration of global motion temporal filtering," in *Proc. IEEE ICIP*, Sep. 2011, pp. 3473–3476.
- [27] A. Krutz, A. Glantz, M. Tok, and T. Sikora, "Theoretical consideration of global motion temporal filtering," in *Proc. PCS2011*, May 2012, pp. 473–476.
- [28] M. Tok, A. Glantz, A. Krutz, and T. Sikora, "Feature-based global motion estimation using the Helmholtz principle," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Process.*, May 2011, pp. 1561–1564.
- [29] K. McCann, B. Bross, and S. Sekiguchi, "High efficiency video coding (HEVC) test model 1 (HM 1) encoder description," JCT-VC of ITU-T VCEG and ISO/IEC MPEG, Guangzhou, China, Tech. Rep. JCTVC-C402, Oct. 2010.
- [30] K. McCann, S. Sekiguchi, B. Bross, and W.-J. Han, Eds., "HM4: HEVC test model 4 encoder description," JCT-VC of ITU-T VCEG and ISO/IEC MPEG, Torino, Italy, Tech. Rep. JCTVC-F802, Jul. 2011.
- [31] K. McCann, B. Bross, W.-J. Han, S. Sekiguchi, and G. J. Sullivan, "High-efficiency video coding (HEVC) test model 5 (HM 5) encoder description," JCT-VC of ITU-T VCEG and ISO/IEC MPEG, Geneva, Switzerland, Tech. Rep. JCTVC-G1102, Nov. 2011.
- [32] K. McCann, B. Bross, W.-J. Han, I. K. Kim, K. Sugimoto, and G. J. Sullivan, "High-efficiency video coding (HEVC) test model 8 (HM 8) encoder description," JCT-VC of ITU-T VCEG and ISO/IEC MPEG, Stockholm, Sweden, Tech. Rep. JCTVC-J1002, Jul. 2012.
- [33] F. Bossen, "Common test conditions and software reference configurations," JCT-VC of ITU-T VCEG and ISO/IEC MPEG, Torino, Italy, Tech. Rep. JCTVC-F900, Jul. 2011.
- [34] G. Bjøntegaard, "Calculation of average psnr differences between RF-curves," Tech. Rep. ITU-T SG16/Q.6 VCEG document VCEG-M33, Mar. 2003.



Andreas Krutz (M'10) received the Dipl.-Ing. and Dr.-Ing. degrees in electrical engineering from Technische Universität Berlin, Berlin, Germany, in 2006 and 2010, respectively.

He is currently a Post-Doctoral Researcher with the Communication Systems Group, Technische Universität Berlin. He was a Student Research Assistant with the University College, University of New South Wales, Canberra, Australia, where he started to research in fields of image processing and video coding. He has been engaged in four European

network of excellences, such as 3DTV, K-space, VISNET II, and PetaMedia. Within the K-Space Project, he has worked on video analysis techniques, especially on object segmentation. Within the remaining three projects, he has worked on new techniques for hybrid video coding based on analysis or synthesis approaches. In 2008, he was a Guest Researcher with the University of New South Wales and a Co-Chair of a special session at MMSP, Cairns, Australia.

Dr. Krutz was an active contributor to the JCT-VC, a joint standardization activity of ITU and ISO/IEC MPEG leading to the next video coding standard HEVC. He is a member of VDE/ITG.



Alexander Glantz (S'08) received the Dipl.-Ing. degree in computer engineering from Technische Universität Berlin, Berlin, Germany, where he is currently pursuing the Dr.-Ing. degree with the Communication Systems Group.

He has been involved in the European Networks of Excellence (NoE) VISNET II and PetaMedia and currently takes part in international ITU and ISO standardization activities. His main research interests include multimedia signal processing, hybrid video coding, rate-distortion theory, GME, and object seg-

mentation in video sequences.

He is a member of the German Society for Information Technology.



Michael Tok (S'10) received the Dipl.-Ing. degree in computer engineering from Technische Universität Berlin, Berlin, Germany, where he is currently pursuing the Dr.-Ing. degree with the Communication Systems Group.

He is currently involved in international ITU and ISO/IOC standardization activities leading to the next video coding standard, HEVC. His main research interests include multidimensional signal processing, hybrid video coding, object-based video coding, GME, and motion modeling.



Marko Esche (S'10) received the B.S. and M.S. degrees in electrical engineering from Technische Universität Berlin, Berlin, Germany, in 2008 and 2009, respectively, where he is currently pursuing the Dr.-Ing. degree.

Since 2009, he has been a Research and Teaching Assistant with the Communication Systems Group, Technische Universität Berlin. His current research interests include hybrid video coding, video denoising, and rate-distortion theory.



Thomas Sikora (SM'96) received the Dipl.-Ing. and Dr.-Ing. degrees in electrical engineering from Bremen University, Bremen, Germany, in 1985 and 1989, respectively.

He is currently a Professor and the Director of the Communication Systems Group (Nachrichtenübertragung), Technische Universität Berlin, Berlin, Germany. In 1990, he joined Siemens Ltd. and Monash University, Melbourne, Australia, as a Project Leader responsible for video compression research activities in the Australian Universal Broadband Video Codec Consortium.

From 1994 to 2001, he was the Director of the Interactive Media Department, Heinrich Hertz Institute, Berlin. In 2011, he was elected as a member of the Berlin-Brandenburg Academy of Science. He frequently works as an Industry Consultant and Patent Reviewer on issues related to interactive digital audio and video. He has published more than 300 journal and conference papers related to audio and video processing. He is a frequent Invited Speaker at international conferences. He has co-authored three books, *Introduction to MPEG-7: Multimedia Content Description Interface* (Wiley, 2002), *MPEG-7 Audio and Beyond: Audio Content Indexing and Retrieval* (Wiley, 2005), and *3D Videocommunication: Algorithms, Concepts and Real-Time Systems in User-Centered Communications* (Wiley, 2005).

Dr. Sikora is a co-founder of Vis-a-Pix GmbH (www.visapix.com) and imcube media GmbH (www.imcube.com), two Berlin-based start-up companies involved in research and development of audio and video signal processing and compression technology. He is an Appointed Member of the Advisory and Supervisory Board of a number of German companies and international research organizations. He has been involved in international ITU and ISO standardization activities, and in several European research activities for more than 15 years. As the Chairman of the ISO-MPEG (Moving Picture Experts Group) Video Group, he has been responsible for the development and

standardization of the MPEG-4 and MPEG-7 video algorithms. He has also served as the Chairman of the European COST 211th Video Compression Research Group. He was appointed as the Research Chair for the VISNET and 3DTV European Networks of Excellence. He is a member of the German Society for Information Technology and a recipient of the 1996 ITG Award. He received the 1996 Engineering Emmy Award of the U.S. He is a past Editor-in-Chief of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. From 1998 to 2002, he was an Associate Editor of the IEEE SIGNAL PROCESSING MAGAZINE. He is an Advisory Editor for the *EURASIP Signal Processing: Image Communication Journal* and was an Associate Editor of the *EURASIP Signal Processing Journal*.