Lossy Parametric Motion Model Compression for Global Motion Temporal Filtering

Michael Tok, Andreas Krutz, Alexander Glantz, and Thomas Sikora
Technische Universität Berlin, Communication Systems Group
Sekr. EN 1, Einsteinufer 17, D-10587 Berlin, Germany
{tok, krutz, glantz, sikora}@nue.tu-berlin.de

***Abstract*— It has been shown that techniques using higher-order motion parameters outperform common translational motion compensated prediction for hybrid video coders. A critical issue is the transmission of accurate higher-order motion parameters with as little additional bits as possible to maximize the compression gain of the whole system. For that, we propose a compression scheme for perspective motion models using transformation before quantization and temporal redundancy reduction and integrate this scheme into a video coding environment using adaptive global motion temporal filtering. Experimental results show that using the proposed compression scheme for the perspective motion models, the BD-rate can be improved up to 8.8% in average in the higher bit rate range and up to 7.7% in average in the lower bit rate range compared to the latest version of the HEVC test model HM 4.0.**

I. Introduction

For temporal redundancy reduction in common video coding standards like MPEG-2 [1], MPEG-4 Visual [2] or H.264/AVC [3], block-based motion estimation followed by motion compensation is utilized. However, this kind of motion compensation can only describe translational motion and delivers suboptimal results for higher-order motion such as rotation, zoom, or perspective deformation. Additionally, for each translationally compensated block, the motion information has to be encoded and transmitted in the form of motion vectors as well.

In contrast to block-based translational motion models, higher-order motion models have the ability to describe the camera transformation between two frames more precisely. Thus, by using such models, the generation of better prediction signals is possible leading to less prediction error information to be encoded. Additionally, for all regions that are described by such a higher-order motion model, no additional motion vector information has to be transmitted.

In [4], Glantz et al. introduced a mode for H.264/AVC inter coding by using an 8 parameter perspective motion model for prediction signal generation and temporal filtering. Thereby, they save bits usually used for motion vectors and improve the prediction signal at the same time. Another way of increasing coding efficiency is to improve the quality of the decoded frames by filtering. For this purpose, the work in [5] describes how to use global, or parametric motion models (PMM) in more general terms, for temporal frame filtering to reduce block artifacts. The authors call their technique global motion temporal filtering (GMTF). Since highly precise parametric motion model estimation is not always possible at the decoder side, the required model parameters have to be transmitted in addition. This increases the bit rate. To reduce the amount of overhead caused by the additional parameters, motion model compression can be utilized.

For lossy compression of polynomial motion models with up to 12 parameters, Karczewicz et al. proposed to orthonormalize the coefficients of the models to obtain higher robustness to quantization [6]. Steinbach et al. [7] employ this technique to compress affine six parameter models. That way they extend the H.263 [8] inter prediction with a set of affine transformed reference frames. However, these polynomial models can only cover motion describable with linear combinations of basis functions while perspective motion models with 8 parameters can even follow perspective deformations which are also often introduced by camera motion.

We propose a method for compression of such perspective models to improve the performance of video coding methods that use PMMs. This compression scheme transforms a given model to global motion vectors at the corners of each frame. Subsequently, these vectors are quantized. As the motion between frames changes slowly in time, the temporal redundancy of the corner vectors is reduced by temporal difference coding. Then exponential Golomb coding is applied on these differences. With this proposed method, the amount of bits needed to transmit the PMMs can be reduced by a factor of up to 4 while the compensation quality decrease stays negligible. To demonstrate the benefit of this method in application to coding approaches based on PMMs, a block-adaptive version of the GMTF is incorporated into the latest HEVC test model HM 4.0 and extended by the motion model compression scheme.

The remainder of this paper is organized as follows. Section II shortly describes a robust estimation method based on the Helmholtz principle that is used for getting highly precise perspective 8 parameter motion models. In Section III, the steps of the presented method for compressing these models are described, the whole compression method is explained and the dependency of the quantization step size to the model quality is pointed out. The adaptive GMTF, which is extended by the model compression scheme is described in Section IV. Section V presents and discusses the results in terms of bit rate savings for adaptive GMTF solely and with the novel motion model compression. Finally, Section VI gives a summary of this paper.

II. Motion Model Estimation

To get a PMM that describes the complex transformations induced by camera motion, the parametric motion estimation method presented in [9] is used. Thus, for each frame 400 features are selected and tracked. Then, a robust estimator based on the Helmholtz principle is applied on the set of feature correspondences to reject outliers resulting from foreground motion and mistracking and to derive a precise PMM. This estimator takes m randomly selected subsets of two correspondences to generate one simplified 4 parameter motion model $\mathbf{H}_k$ per subset

$$\mathbf{H}_k = \begin{pmatrix} \tilde{m}_{0,k} & \tilde{m}_{1,k} & \tilde{m}_{2,k} \\ -\tilde{m}_{1,k} & \tilde{m}_{0,k} & \tilde{m}_{3,k} \\ 0 & 0 & 1 \end{pmatrix}. \quad (1)$$

This model is then used to define whether a feature correspondence of the whole set is an inlier or an outlier regarding to $\mathbf{H}_k$. With the number of inliers N_k and the estimated error variance of these inliers σ_k, a rating per subset is defined by

$$\Phi_k = \frac{N_k}{\sigma_k} \quad (2)$$

Only for the inlier features $\mathbf{x}_k$ and their tracked correspondences $\check{\mathbf{x}}_k$ with the largest Φ_k, a final perspective PMM is calculated by Least Squares as

$$\mathbf{h} = \left(\mathbf{A}_k^T \mathbf{A}_k\right)^{-1} \mathbf{A}_k^T \check{\mathbf{x}}_k, \quad (3)$$

where $\mathbf{A}_k$ is the perspective design matrix for the feature correspondences of the kth consensus set and $\mathbf{h} = (m_0, \ldots, m_7)^T$ contains the final motion parameters.

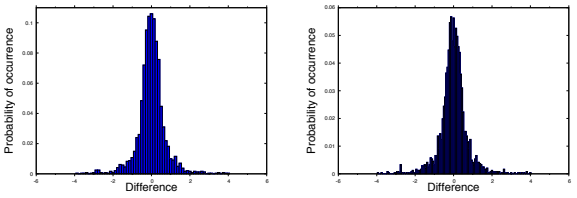
III. Motion Model Compression

As mentioned before, these perspective motion models $\mathbf{h}$ have the ability to describe each displacement $\mathbf{d}$ of a pixel at a position $\mathbf{p} = (x, y)^T$ to a new position $\mathbf{q} = (x', y')^T$ that follows a combination of translation, zoom, rotation, shearing, and perspective deformation:

$$\begin{pmatrix} x' \cdot w' \\ y' \cdot w' \\ w' \end{pmatrix} = \begin{pmatrix} m_0 & m_1 & m_2 \\ m_3 & m_4 & m_5 \\ m_6 & m_7 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \quad (4)$$

The transmission of such models with single precision floating point accuracy (32 bit) needs 8×32 bit $= 256$ bit. This e.g. means additional $6.4\,\mathrm{kbit/s}$ for a $25\,\mathrm{Hz}$ sequence or $15.4\,\mathrm{kbit/s}$ for a $60\,\mathrm{Hz}$ sequence. It would be possible to use lower model precision (24 or even 16 bit) or parameter quantization, to reduce this amount of data. But on the other hand such precision reduction would lead to extreme quality losses. Furthermore, for getting a set of motion parameters that are more robust to quantization, Karczewicz et al. propose to use orthonormalized versions of polynomial motion models. E.g., warping with affine models is done by the linear combination

$$\begin{aligned} x' &= m_0 \cdot x + m_1 \cdot y + m_2 \\ y' &= m_3 \cdot x + m_4 \cdot y + m_5 \end{aligned}. \quad (5)$$



(a) 1/8 pel (b) 1/16 pel

Fig. 1. Measured propability mass functions of global corner motion vector differences over all sequences for two different quantization step sizes.

Hence, orthonormalization of affine motion model parameters is possible. But when rewriting (4) as

$$\begin{aligned} x' &= \frac{m_0 \cdot x + m_1 \cdot y + m_2}{m_6 \cdot x + m_7 \cdot y + 1} \\ y' &= \frac{m_3 \cdot x + m_4 \cdot y + m_5}{m_6 \cdot x + m_7 \cdot y + 1} \end{aligned}, \quad (6)$$

it can be seen that warping with a perspective motion model needs an additional division step. Consequently, perspective motion parameters cannot be orthonormalized for higher quantization robustness, due to this additional, nonlinear step.

Nevertheless, an 8 parameter model can be transformed to a set of global motion vectors (GMV) at frame corners and then quantized as is similarily done in MPEG-4 Visual [2]. For a video with a resolution of $x_{\text{res}} \times y_{\text{res}}$, (7) shows the interdependency of the GMVs' end-points $(\hat{x}_{1\ldots4}, \hat{y}_{1\ldots4})^T$, their starting points at each frame's corners $\pm x_{\text{res}} \times \pm y_{\text{res}}$ and the corresponding PMM. These GMVs are highly robust to quantization in contrast to the GMM's original parameters. For the warping process in MPEG-4 Visual, the GMVs are interpolated bilinearly to obtain the transformed pixel positions at the decoder, which is a drawback concerning warping quality. We do an inverse transformation of the GMVs back to a perspective warping model $\mathbf{h}'$ at the decoder side. That way, more precise global motion compensation (GMC) and thus better coding or filtering results are possible.

Another advantage of the global corner motion vectors $\mathbf{V}_{1,n}$ to $\mathbf{V}_{4,n}$ of each frame n is their temporal correlation. This property is used for further compression gain after quantizing the GMVs to a new set $\hat{\mathbf{V}}_{1,n}$ to $\hat{\mathbf{V}}_{4,n}$. The differences of these vectors to their quantized predecessors $\hat{\mathbf{V}}_{1,n-1}$ to $\hat{\mathbf{V}}_{4,n-1}$ show a two-sided geometric probability mass function (PMF) like behavior. That is, why signed exponential Golomb coding is used to compress the quantized GMV differences. Figure 1 shows the measured PMFs of global motion vector differences for the used test sequences. To find a suitable quantization step size for the GMVs, so called rate-quality-curves for GMC with the compressed models are evaluated. Figure 2 shows one exemplary curve for the Blue Sky sequence with quantization step sizes $\{\frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}\}$. Like for all tested sequences, it can be seen, that the quality of a compressed model with $\frac{1}{32}$ quantization is close to the uncompressed model's quality. This step size is chosen for the model compression, in the following. The encoding process containing model transformation to GMVs, quantization, difference coding and Golomb coding is illustrated in Figure 3.

$$\begin{pmatrix} \hat{x}_1 \cdot h_1 & \hat{x}_2 \cdot h_2 & \hat{x}_3 \cdot h_3 & \hat{x}_4 \cdot h_4 \\ \hat{y}_1 \cdot h_1 & \hat{y}_2 \cdot h_2 & \hat{y}_3 \cdot h_3 & \hat{y}_4 \cdot h_4 \\ h_1 & h_2 & h_3 & h_4 \end{pmatrix} = \begin{pmatrix} m_0 & m_1 & m_2 \\ m_3 & m_4 & m_5 \\ m_6 & m_7 & 1 \end{pmatrix} \cdot \begin{pmatrix} -\frac{x_{\text{res}}}{2} & \frac{x_{\text{res}}}{2} & -\frac{x_{\text{res}}}{2} & \frac{x_{\text{res}}}{2} \\ -\frac{y_{\text{res}}}{2} & -\frac{y_{\text{res}}}{2} & \frac{y_{\text{res}}}{2} & \frac{y_{\text{res}}}{2} \\ 1 & 1 & 1 & 1 \end{pmatrix} \tag{7}$$

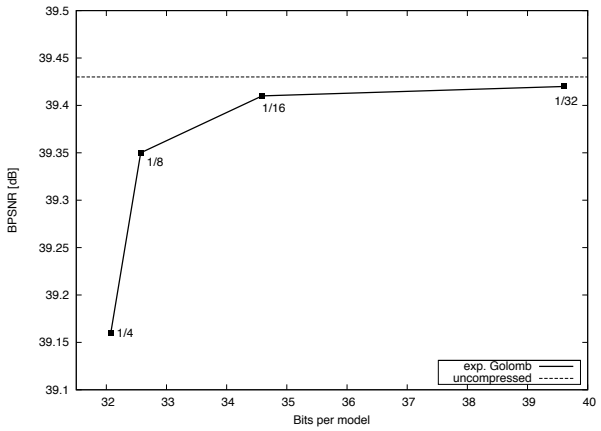


Fig. 2. The motion compensation quality in terms of background PSNR depending on the quantization step size of the model compression for BlueSky. The step sizes are $\{\frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}\}$. The dashed line shows the quality of the uncompressed, raw motion model.

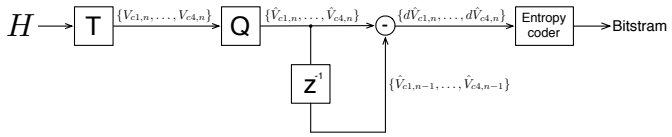


Fig. 3. The proposed motion model compression scheme.

IV. Adaptive Global Motion Temporal Filtering

HEVC utilizes two kinds of deblocking filters, both only working in spatial domain. In [5], a deblocking filter for the temporal domain, called global motion temporal filtering has been presented. This new type of filter uses parametric motion models to compensate a set of already decoded frames for building a so called image stack and fuses them to reduce blocking artifacts appearing in the H.264/AVC compression process. The optimal amount m_{opt} of images used for the filtering is determined at encoder side. One drawback of that filtering method is, that arbitrarily moving foreground objects are vanishing when the image stack is fused. To overcome this issue, Glantz et al. generate foreground segmentation masks at encoder-side and transmit them in addition to the motion models, m_{opt} and the original H.264/AVC bit stream. We use the GMTF approach to improve the image quality of encoded HEVC streams. Instead of segmentation, an adaptive block-based decision between temporal filtering and spatial filtering is made at encoder-side. Therefore, blocks with a size of 128×128 are taken and for each block a flag for using this adaptive GMTF or not is sent. Figure 4 overviews the whole encoder and decoder setting consisting of HM 4.0 and adaptive GMTF with model compression.

TABLE I
HEVC CODING SETTINGS USED FOR EXPERIMENTAL EVALUATION

HEVC test software	HM 4.0
Profile	High efficiency
Picture order / GOP settings	hierarchical B, random access
QP_{low}	$\{27, 32, 37, 42\}$
QP_{high}	$\{22, 27, 32, 37\}$
Largest CU size	64
Smallest CU size	8
Number of reference frames	4
Motion search range	64×64

V. Experimental Evaluation

Six test sequences with differing resolution, frame rate and camera motion have been selected for evaluation. All sequences are encoded with the HEVC test model HM 4.0 [10], and by HM 4.0 with adaptive GMTF to show, how adaptive GMTF performs without model compression. Eventually, the model compression is incorporated into HM 4.0 in addition to adaptive GMTF. Table I shortly overviews the settings of the used HEVC reference coder.

As evaluation criterion for comparing the coding efficiency of the reference coder and the novel framework, the Bjøntegaard metric (BD-rate and BD-PSNR) is chosen [11]. Table II shows properties as resolution, frame rate and frame amount of the used test sequences and the coding results for adaptive GMTF with and without motion model compression in terms of BD-rate and BD-PSNR. For BQSquare, adaptive GMTF has an increased average BD-rate at higher QP ranges (2.3%) as well as at lower QP ranges (5.2%).

By using model compression in addition, it turns out, that bit rate savings of 0.5% for the high QP range and 0.4% for the low QP range are achievable for the same sequence. This example demonstrates the influence of an uncompressed models bit rate on coding efficiency. For Sunflower, the motion estimation itself is difficult due to the video content. Faulty parameters result in reduced filtering performance as well as in reduced model compression efficiency. Thus, losses in higher quality ranges appear. A reduction in coding gain when using model compression is only recognized for the Blue Sky sequence. Nevertheless, for all other sequences, in which adaptive GMTF provides coding gain, additional motion model compression further increases the coding efficiency.

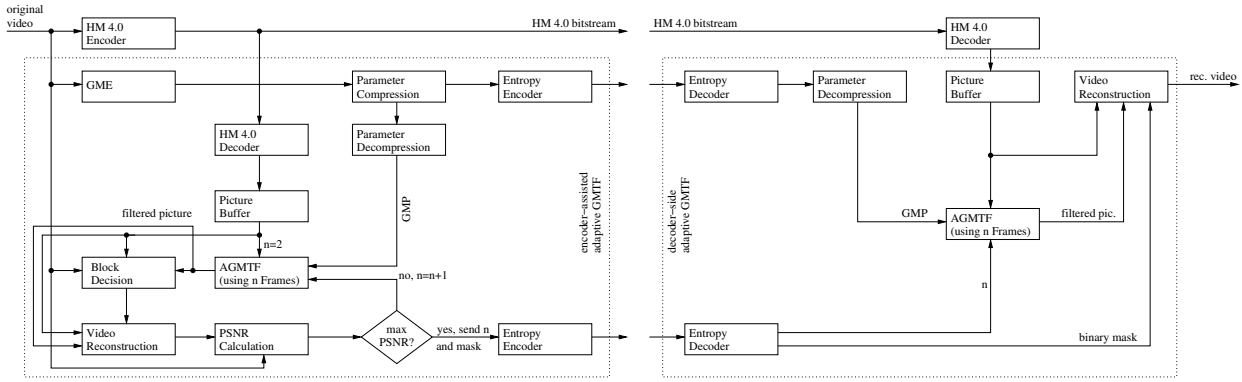


Fig. 4. The motion model compression scheme incorporated into the adaptive GMTF framework.

TABLE II

USED TEST SEQUENCES AND EXPERIMENTAL RESULTS IN TERMS OF BD-RATE [%] AND BD-PSNR [dB] FOR HM 4.0 VS. HM 4.0 + AGMTF.

Test sequence	Size	FPS	Frames	QP_{high}				QP_{low}			
				no model compression		with model compression		no model compression		with model compression	
				BD-rate	BD-PSNR	BD-rate	BD-PSNR	BD-rate	BD-PSNR	BD-rate	BD-PSNR
BQSquare	416 × 240	60	600	2.3	−0.1	−0.5	0.0	5.2	−0.2	−0.4	0.0
BQTerrace	1920 × 1080	60	600	−2.2	0.2	−2.3	0.0	−3.2	0.1	−3.4	0.1
BlueSky	1920 × 1080	25	217	−2.1	0.1	−2.1	0.1	−3.0	0.1	−2.7	0.1
Jets1	1280 × 720	60	300	−2.8	0.1	−6.3	0.1	2.0	−0.1	−4.3	0.2
Station2	1920 × 1080	25	313	−8.0	0.2	−8.8	0.2	−6.1	0.2	−7.7	0.3
Sunflower	1920 × 1080	25	500	1.4	0.0	1.3	0.0	−0.4	0.0	−1.1	0.0
Waterfall	704 × 480	25	260	−5.3	0.2	−7.0	0.2	−2.8	0.1	−6.3	0.3
mean				−2.4	0.1	−3.7	0.1	−1.2	0.0	−3.7	0.1

VI. SUMMARY

A new method for lossy perspective motion model compression has been presented and explained. It uses transform, difference, and signed exponential Golomb coding. To show the performance of this approach, an adaptive version of the GMTF post filter is incorporated into the HM 4.0 and enhanced by our motion model compression method. Bit rate savings of up to 8.8% in comparison to the HM 3.2 reference can be observed, when switching from lossless model transmission to model compression. Furthermore, cases where GMTF without model compression results in bit rate increases. Then additional model compression can still result in coding gain.

For further improvements of the motion model compression scheme, spatial redundancy reduction and more efficient prediction techniques as polynomial prediction will be considered in further work. Increased coding gain can be achieved by using arithmetic encoding after Golomb coding as well.

REFERENCES

[1] MPEG-2 *Video,* ISO/IEC IS 11172-2, 2000.

[2] T. Sikora, "The MPEG-4 video standard verification model," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 19 –31, Feb 1997.

[3] D. Marpe, T. Wiegand, and G. Sullivan, "The H.264/MPEG4 advanced video coding standard and its applications," *Communications Magazine, IEEE*, vol. 44, no. 8, pp. 134 –143, Aug 2006.

[4] A. Glantz, A. Krutz, and T. Sikora, "Adaptive Global Motion Temporal Prediction for Video Coding," in *Proceedings of the 28th IEEE Picture Coding Symposium*, Nagoya, Japan, Dec 2010.

[5] A. Glantz, A. Krutz, M. Haller, and T. Sikora, "Video coding using global motion temporal filtering," in *16th IEEE International Conference on Image Processing*, Nov 2009, pp. 1053 –1056.

[6] M. Karczewicz, J. Nieweglowski, J. Lainema, and O. Kalevo, "Video coding using motion compensation with polynomial motion vector fields," in *First International Workshop on Wireless Image/Video Communications*, Sep 1996, pp. 26 –31.

[7] E. Steinbach, T. Wiegand, and B. Girod, "Using multiple global motion models for improved block-based video coding," in *Proceedings of the 6th IEEE International Conference on Image Processing*, Sep 1999, pp. 56–60 vol.2.

[8] K. Rijkse, "H.263: video coding for low-bit-rate communication," *Communications Magazine, IEEE*, vol. 34, no. 12, pp. 42 –45, Dec. 1996.

[9] M. Tok, A. Glantz, A. Krutz, and T. Sikora, "Feature-Based Global Motion Estimation Using the Helmholtz Principle," in *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, Prague, Czech Republic, May 2011.

[10] K. McCann, T. Wiegand, B. Bross, W.-J. Han, J.-R-Ohm, S. Sekiguchi, and G. J. Sullivan, "Hevc draft and test model editing," *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 document JCTVC-F002.doc*, Jun 2011.

[11] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," *ITU-T SG16/Q.6 VCEG document VCEG-M33*, Mar 2001.