ADAPTIVE 3D INTERPOLATION FILTER FOR MOTION AND ALIASING COMPENSATED PREDICTION

Thomas Wedi

Institut für Theoretische Nachrichtentechnik und Informationsverarbeitung
University of Hannover, Appelstr. 9a, 30167 Hannover, Germany
E-Mail: wedi@tnt.uni-hannover.de

ABSTRACT

Standardized hybrid video coding systems are based on motion compensated prediction with fractional-pel displacement vector resolution. In the recent video coding standard H.264 / MPEG-4 AVC, displacement vector resolutions of 1/4-pel are applied. In order to estimate and compensate these fractional-pel displacements, separable 2D interpolation filters are used. So far, these interpolation filters are invariant. The same filter coefficients are applied for all sequences and for all images of a sequence. Therefore, it is not possible to consider non-stationary statistical properties of video signals in the interpolation process. In the past, we already presented an adaptive 2D interpolation filter. This interpolation filter uses filter coefficients that are adapted once per image to the non-stationary statistical properties of the video signal. The filter-coefficients are coded and transmitted. In this paper, the adaptive 2D interpolation filter is extended to an adaptive 3D interpolation filter. Compared to the adaptive 2D interpolation filter, a gain up to 0.4 dB PSNR and compared to the original H.264 / MPEG-4 AVC a gain up to 1.2 dB PSNR is obtained with the 3D interpolation filter.

1. INTRODUCTION

Standardized hybrid video coding systems like the recent H.264 / MPEG-4 AVC coding scheme [1] are based on motion compensated prediction. Fig. 1 shows the generalized block diagram of such a hybrid video encoder. The current image s_t at time instance t is predicted by a motion compensated prediction (MCP) from already transmitted reference images. The result of the motion compensated prediction is image $\hat{s}_t$. Only the prediction error e_t and the motion information $\vec{d}_t$ are coded and transmitted.

For the motion compensated prediction, the current image is partitioned into blocks. A displacement vector $\vec{d}_t$ is assigned to each block that refers to the corresponding position of its image signal in an already transmitted reference image. The displacement vectors have a fractional-pel reso-
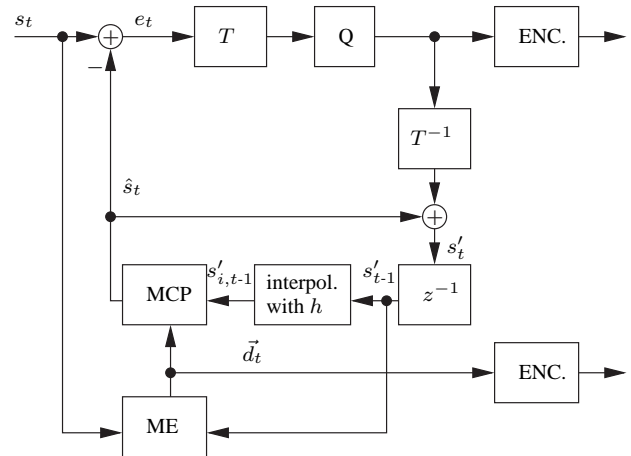

Fig. 1. Generalized block diagram of a hybrid video encoder based on motion compensated prediction.

lution. In H.264 / MPEG-4 AVC displacement vectors with 1/4-pel resolution are applied.

Displacement vectors with fractional-pel resolution may refer to positions in a reference image that are located between the sampled positions. In the following, these positions are called subpel positions. In order to estimate and compensate fractional-pel displacements, the image signal on subpel positions has to be generated by interpolation. Up to now, this interpolation is done with an invariant interpolation filter h. The same filter is used for all sequences and for all images of a sequence. Therefore it is not possible to consider non-stationary statistical properties of video signals in the interpolation process.

Due to non-ideal low-pass filters in the image acquisition process, the *Nyquist Sampling Theorem* is not fulfilled and aliasing disturbs the motion compensated prediction [2]. This leads to a prediction error that has to be coded. Thus, recent video coding standards like MPEG-4 Advanced Simple Profile and H.264 / MPEG-4 AVC apply Wiener interpolation filters [1, 3]. These filters were de-

signed to interpolate the image signal while reducing aliasing components, which deteriorate the motion compensated prediction [4]. These Wiener filters weight the image signal on the sampled positions of the image to interpolate in order to generate samples on subpel postions. Up to now, these filters are based on invariant filter coefficients. The same coefficients are used for all sequences and for all images of a sequence.

In [6, 7] we presented an adaptive 2D interpolation filter in order to improve coding efficiency. This interpolation filter is based on filter coefficients that are adapted once per image to the non-stationary statistical properties of the video signal. Similar to the non-adaptive Wiener filter, these adaptive coefficients weight the image signal on the sampled positions of the image to interpolate in order to generate the samples on subpel positions. It is shown in [6, 7] that an adaptive filter reduces prediction errors caused by displacement estimation errors and by aliasing. Since this adaptive filter did not use any information of previous images, its capability of aliasing compensation is limited.

In [5], we presented a motion compensated 3D interpolation filter that allows to interpolate an aliasing affected image more accurate then conventional 2D filters. This 3D filter is a combination of the Wiener filter and a motion compensated filter. It does not only use the signal of the image to be interpolated, but also the image signal of previous images for the interpolation purpose. It is shown in [5] that for certain features of the image signal, the motion compensated prediction with this filter could be done perfectly, even if the image signal is affected by aliasing. The motion compensated 3D interpolation is performed with displacement vectors that are already estimated and transmitted. One disadvantage of this scheme is the sensitivity concerning displacement estimation errors. Since the filter is a motion compensated filter that uses displacement vectors for the interpolation purpose, the efficiency depends on the accuracy of the displacement vectors.

In this paper, the adaptive 2D interpolation filter from [6, 7] is extended by the 3D interpolation concept from [5] in order to further reduce the prediction error and improve the coding efficiency. The adaptive 3D filter is based on filter coefficients that do not only use the samples of the image to interpolate in order to generate the image signal on subpel positions, but also the samples of one previous reference image.

In the following Sections 2 and 3, the adaptive 2D and the adaptive 3D filter are introduced in detail. Section 4 describes the integration of the adaptive interpolation filter scheme into the H.264 / MPEG-4 AVC standard. Experimental results are given in 5. The paper closes with a summary.

2. ADAPTIVE 2D INTERPOLATION FILTER

In this subsection, the adaptive 2D interpolation scheme is introduced. Fig. 2 illustrates the motion compensated prediction using an adaptive 2D interpolation filter. In order
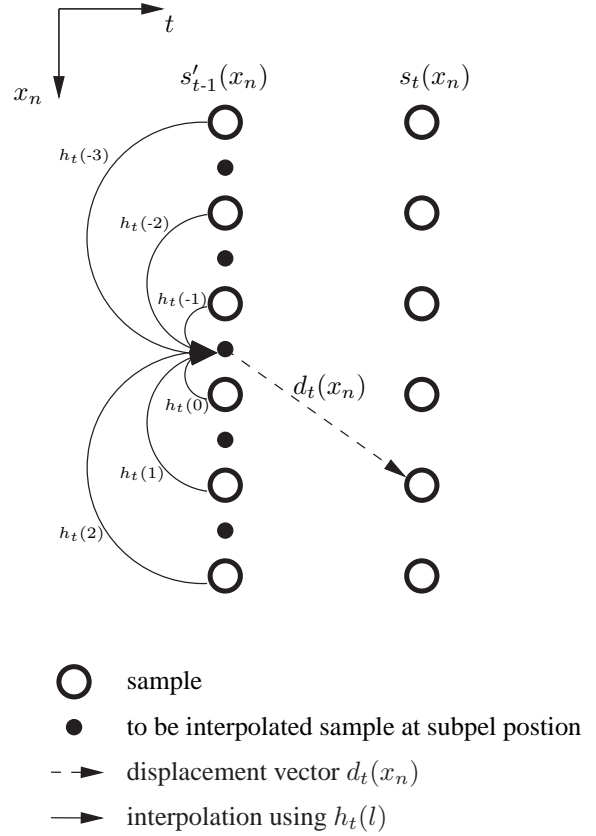
Fig. 2. Illustration of motion compensated prediction using an adaptive 2D interpolation filter. Corresponding lines of two consecutive images are shown: The image to be predicted $s_t(x_n)$ and the reconstructed reference image $s'_{t-1}(x_n)$ that has to be interpolated.

to simplify the explanations, this illustration is restricted to one spatial coordinate x_n. Thus, Fig. 2 shows one dimensional image lines instead of two dimensional images. The image line $s_t(x_n)$ is predicted from the reference image line $s'_{t-1}(x_n)$ using the displacement vector $d_t(x_n)$. It is assumed that the displacement vector resolution is 1/2-pel. In the example of Fig. 2, the displacement vector refers to a subpel position of image line $s'_{t-1}(x_n)$ that has to be generated by interpolation. For the interpolation purpose a 2D interpolation filter $h_t(l)$ is used. This filter generates the sample on a subpel position that is used to predict the corresponding sample of image line $s_t(x_n)$. Due to the restriction to one dimensional image lines in Fig. 2, the 2D filter in this figure is also restricted to a 1D filter.

2

In the following, an adaptive filter $h_t(l)$ of finite impulse response described by $2L$ coefficients is assumed:

$$h_t(l) = 0 \quad \text{,for} \quad (l > L-1) \wedge (l < -L) \qquad (1)$$

In case of a fractional pel displacement (e.g. $d_t(x_n) = \pm 0.5$ pel, ± 1.5 pel, ...), the prediction signal $\hat{s}_t(x_n)$ is given by

$$\hat{s}_t(x_n) = \sum_{l=-L}^{L-1} h_t(l) \cdot s'_{t-1}(x_n - \tilde{d}_t(x_n) + l) \qquad (2)$$

where

$$\tilde{d}_t(x_n) = \lfloor d_t(x_n) \rfloor \qquad (3)$$

is the displacement that is rounded to the nearest integer towards minus infinity. This rounding is necessary, because $d_t(x_n)$ refers to a subpel position in image $s'_{t-1}(x_n)$ (see Fig. 2). Since the filter $h_t(l)$ has to weight the samples of $s'_{t-1}(x_n)$ in order to calculate the prediction image $\hat{s}_t(x_n)$, the rounded displacement from (3) has to be used in equation (2).

In case of a fullpel displacement (e.g. $d_t(x_n) = \pm 1$ pel, ± 2 pel, ...) no interpolation has to be performed. Thus, the prediction signal for fullpel displacements is given by

$$\hat{s}_t(x_n) = s'_{t-1}(x_n - d_t(x_n)). \qquad (4)$$

The coefficients of the adaptive interpolation filter are estimated by minimizing the prediction error

$$e_t(x_n) = s_t(x_n) - \hat{s}_t(x_n) \qquad (5)$$

that has to be coded (Fig. 1). Thus, the adaptive interpolation filter $h_t(l)$ can be interpreted as an adaptive prediction filter and the filter coefficients can be determined by solving the Wiener-Hopf equation [8]. With the autocorrelation function of the reference signal $s'_{t-1}(x_n)$

$$R_{t-1,t-1}(n) = E[s'_{t-1}(x_n) \cdot s'_{t-1}(x_n - n)] \qquad (6)$$

and the crosscorrelation function of the signal to be predicted $s_t(x_n)$ and the displaced reconstructed reference signal $s'_{t-1}(x_n - \tilde{d}(x_n))$

$$R_{t-1,t}(n) = E[s'_{t-1}(x_n - \tilde{d}(x_n)) \cdot s_t(x_n - n)] \qquad (7)$$

the Wiener-Hopf equation is given by

$$\sum_{l=-L}^{(L-1)} h_t(l) \cdot R_{t-1,t-1}(i - l) = R_{t-1,t}(i),$$
$$\text{for } i = -L, \ldots, (L-1). \qquad (8)$$

For stationary signals $s_t(x_n)$ and $s'_{t-1}(x_n)$, it could be shown that due to the symmetry of the autocorrelation and crosscorrelation function, the coefficients of the filter are also symmetric

$$h_t(l) = h_t(-l-1), \quad \text{for} \quad 0 \le l \le L-1. \qquad (9)$$

Thus, the Wiener-Hopf equation from (8) can be expressed by

$$\sum_{l=0}^{(L-1)} h_t(l) \left[R_{t-1,t-1}(i\text{-}l) + R_{t-1,t-1}(i{+}l{+}1) \right] = R_{t-1,t}(i),$$
$$\text{for } i = 0, \ldots, (L-1). \qquad (10)$$

With (9), the number of coefficients of $h_t(l)$ that have to be determined by solving the Wiener-Hopf equation is reduced from $2L$ to L.

The assumption

$$\sum_{l=-L}^{(L-1)} h_t(l) = 1 \qquad (11)$$

is not made. Thus, video signals with lightness changes or fading sequences where $E[s_{t-1}] \neq E[s_t]$ may also be predicted with the Adaptive Filter .

Due to equations (6) and (7) the correlation functions of the Wiener-Hopf equation (10) are calculated from the image to be coded s_t, the reference image s'_{t-1}, and the estimated displacement vectors $\vec{d}_t$. Thus, influences that deteriorate the accuracy of the prediction like

- *Aliasing* in s_t and s'_{t-1},
- *Quantization errors* in s'_{t-1}, and
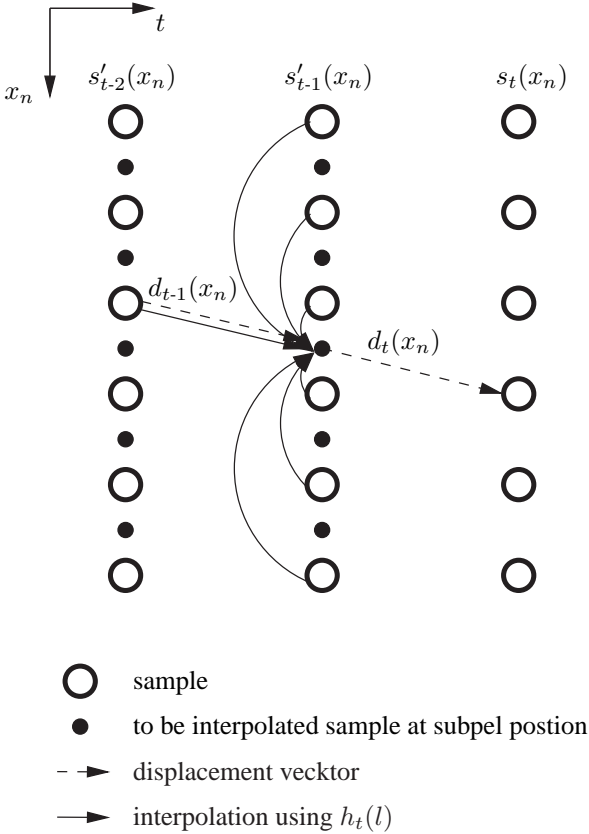- *Displacement estimation errors* in $\vec{d}_t$

are implicitly considered in the estimation of the filter coefficients. Due to these considerations, the prediction error is reduced and the coding efficiency is improved by using the adaptive interpolation filter.

3. ADAPTIVE 3D INTERPOLATION FILTER

In the illustration of the following Fig. 3, the adaptive 2D interpolation approach of Fig. 2 is extended by one dimension.

For this purpose, the displacement vector $d_{t-1}(x_n)$ describing the motion between the image to interpolate $s'_{t-1}(x_n)$ and the previously interpolated image $s'_{t-2}(x_n)$ is used. This displacement vector refers to a sample in image $s'_{t-1}(x_n)$ that is also weighted with an additional filter coefficient. Due to the restriction to one dimensional image lines in Fig. 2, the 3D filter in this figure is restricted to a 2D filter.
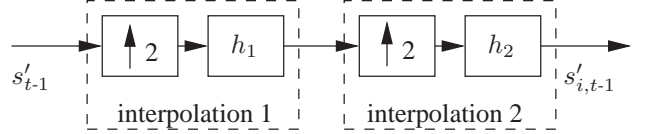
In order to estimate the filter coefficients of the adaptive 3D interpolation filter, the Wiener-Hopf equation of (8) is extended by the additional 3D coefficient.

Fig. 4. Interpolation process for 1/4-pel displacement vector resolution that is used in H.264 / MPEG-4 AVC. It is based on two steps with two interpolation filters h_1 and h_2.

In the adaptive interpolation scheme, an adaptive symmetric filter $h_1=h_{1,t}$ is used in the first interpolation step. In the second interpolation step, an invariant bilinear interpolation filter is applied. In case of an adaptive 2D filter, a symmetric 6-tap filter is used. Thus, only 3 coefficients have to be estimated and transmitted. In case of a 3D filter, only one more coefficient is used.

The coefficient estimation and the motion compensated prediction are performed in the following steps:

1. Displacement vectors $\vec{d}_t$ are estimated for the image to be coded.

2. Coefficients of the adaptive interpolation filter are estimated by minimizing the energy of the prediction error $e_t=s_t-\hat{s}_t$ when performing the motion compensated prediction using the displacement vectors $\vec{d}_t$ from step 1 and the reconstructed image s'_{t-1}.

3. The reference image s'_{t-1} is interpolated by using the estimated coefficients of step 2

4. The current image is predicted by motion compensated prediction. For this purpose, the interpolated reference image of step 3 and the displacement vectors of step 1 are applied.

The estimated coefficients of the adaptive interpolation filter are coded and transmitted. For this purpose, a differential coding scheme is applied. The coefficients are quantized with 8 bit precision and the differences to the filter coefficients of the preceding image are transmitted.

Due to this coefficient estimation process, influences that deteriorate the accuracy of the prediction like

- *Aliasing* in s_t and s'_{t-1},

- *Quantization errors* in s'_{t-1}, and

- *Displacement estimation errors* in $\vec{d}_t$

are implicitly considered in the estimation of the filter coefficients. Due to these considerations, the prediction error is reduced and the coding efficiency is improved by using the adaptive interpolation filter. A more detailed analysis of the consideration of these influences is given in [7].

- ○ sample
- ● to be interpolated sample at subpel postion
- - ➤ displacement vecktor
- ➤ interpolation using $h_t(l)$

Fig. 3. Illustration of motion compensated prediction using an adaptive 3D interpolation filter. Corresponding lines of two consecutive images are shown: The image to be predicted $s_t(x_n)$, the reconstructed reference image $s'_{t-1}(x_n)$ that has to be interpolated, and the previously reconstructed and already interpolated image $s'_{t-2}(x_n)$.

4. ADAPTIVE INTERPOLATION FILTER IN H.264 / MPEG-4 AVC

Fig. 4 shows the block diagram of the interpolation process that is applied in H.264 / MPEG-4 AVC for 1/4-pel displacement vector resolution. A similar interpolation process is applied in MPEG-4 Advanced Simple Profile. The image is interpolated in two steps. In the first interpolation step the resolution of the reconstructed image s'_{t-1} is increased by a factor of 2 and filtered by an interpolation filter h_1. In H.264 / MPEG-4 AVC, filter h_1 is an invariant 6-tap Wiener filter with the following coefficients: [1,-5,20,20,-5,1]/32 [1]. Thus, the corresponding filter h_1 of Fig. 4 is h_1=[1,0,-5,0,20,32,20,0,-5,0,1]/32. In the second interpolation step, the resulting image is again sampled up by a factor of 2 and filtered by filter h_2. In H.264 / MPEG-4 AVC h_2 is a simple bilinear interpolation filter.

4

5. EXPERIMENTAL RESULTS

For experimental investigations, the JM-2 (Joint Model 2) of H.264 / MPEG-4 AVC is used. The applied test-sequences are *Mobile & Calendar* and the VQEG test-sequence *Waterfall* each at CIF format and 30 Hz. In Fig. 5 and 6, operational rate distortion (RD) curves for both sequences are given.
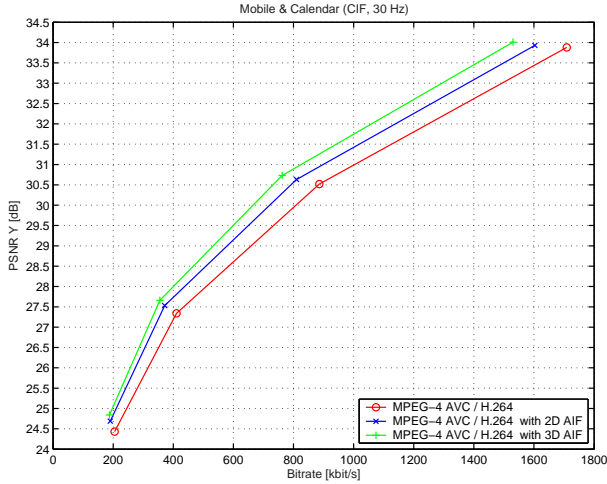
Fig. 5. Operational rate distortion curves for test-sequence *Mobile & Calendar*.
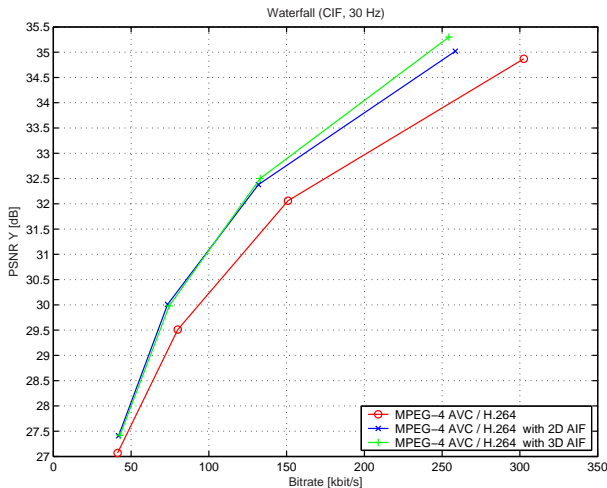
Fig. 6. Operational rate distortion curves for VQEG test-sequence *Waterfall*.

In each graph, the RD-curve of the reference H.264 / MPEG-4 AVC with the invariant interpolation filter and H.264 / MPEG-4 AVC with the Adaptive 2D and 3D Interpolation Filter (AIF) are shown. Compared to the adaptive 2D in-terpolation filter, a gain up to 0.4 dB PSNR and compared to the original H.264 / MPEG-4 AVC a gain up to 1.2 dB PSNR is obtained with the adaptive 3D interpolation filter.

6. SUMMARY

In this paper, an adaptive 3D interpolation filter for motion compensated prediction in an hybrid video coding scheme is presented. The adaptive 3D interpolation filter is an extension of the adaptive 2D interpolation filter. The adaptive interpolation filters are based on coefficients that are adapted once per image to the non-stationary statistical properties of the image signal. Compared to the adaptive 2D interpolation filter, a gain up to 0.4 dB PSNR and compared to the original H.264 / MPEG-4 AVC a gain up to 1.2 dB PSNR is obtained with the adaptive 3D interpolation filter.

7. REFERENCES

[1] Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, "Joint Final Committee Draft (JFCD) of Joint Video Specification (ITU-T Rec. H.264 — ISO/IEC 14496-10 AVC)," 4th Meeting: Klagenfurt, Austria, July 2002.

[2] R.W. Schaefer A.V. Oppenheim, *Discrete-Time Signal Processing*, Prentice Hall, 1989.

[3] MPEG-4: ISO/IEC JTC1/SC29/WG11, "ISO/IEC 14496:2000-2: Information on technology - coding of audio-visual objects - part 2: Visual," ISO/IEC, Genf, Dec. 2000.

[4] O. Werner, "Drift analysis and drift reduction for multiresolution hybrid video coding," *Signal Processing: Image Communication*, vol. 8, no. 5, July 1996.

[5] T. Wedi, "A time-recursive interpolation filter for motion compensated prediction considering aliasing," in *Proc. IEEE Int. Conference on Image Processing (ICIP)*, Kobe, Japan, Oct. 1999.

[6] T. Wedi, "Adaptive interpolation filter for motion compensated hybrid video coding," in *Proc. Picture Coding Symposium (PCS)*, Seoul, Korea, Jan. 2001.

[7] T. Wedi, "Adaptive interpolation filter for motion compensated prediction," in *Proc. IEEE Int. Conference on Image Processing (ICIP)*, Rochester, New York USA, Sept. 2002.

[8] A. Papooulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, 1991.