PARAMETRIC INTERPOLATION FILTER FOR MOTION COMPENSATED PREDICTION

Jie Dong and King Ngi Ngan

Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong SAR

ABSTRACT

Recently, adaptive interpolation filter (AIF) has received increasing attention for motion-compensated prediction (MCP). The existing methods code the filter coefficients individually and the accuracy of coefficients and the size of side information are conflicting. This paper studies the effect of making trade-off between the two conflicting aspects and proposes the parametric interpolation filter (PIF), which represents filters by five parameters instead of individual coefficients and approximates the optimal filter by tuning the parameters. The experimental results show that PIF approaches the efficiency of the optimal filter and outperforms the related work.

1. INTRODUCTION

Motion compensated prediction (MCP) is the key to the success of the modern video coding standards. With MCP, the video signal to be coded is predicted from the temporally neighboring signals, and only the prediction error and the motion vector (MV) are transmitted. However, due to the finite sampling rate, the actual position of the prediction in the neighboring frames may be out of the sampling grid, where the intensity is unknown, so the intensities of the positions in between the integer pixels, called sub positions, must be interpolated and the resolution of MV is increased accordingly.

In H.264, the resolution of MV is up to 1/4-pel and the reference frame is interpolated to be 16 times the size for MCP. As shown in Fig. 1 (a), the interpolation defined in H.264 includes two stages, interpolating the half-pel and 1/4-pel sub positions, respectively. The interpolation in the first stage is separable. For each direction, the sampling rate is doubled by zero-insertion and then a 1-D filter $h_1$ is applied. The second stage is non-separable and uses bilinear filtering supported by the integer pixel and interpolated half-pel values. The filter coefficients are fixed for all video sequences.

Recently, much research effort has been devoted to the design of adaptive interpolation filter (AIF) [1, 2, 3, 4], which considers the non-stationary statistical properties of video signals and allows filter coefficients changed every frame. The existing AIF techniques have the common methodology to derive the filter coefficients, but use different ways to reduce the side information, such as reducing the support region, imposing the symmetry constraints, and quantizing the filter coefficients. AIF in [1] follows the process in Fig. 1 (a). Only $h_1$ is modified for every frame and three coefficients are coded due to the even symmetry. Vatis et. al [2] developed a 2-D non-separable interpolation filter, of which the process is shown in Fig. 1 (b). The spatial sampling rate is increased 16 times at one time and each sub position can be interpolated directly by filtering the surrounding 6×6 integer pixels. $h$ is in circular symmetry, because the spatial statistical properties are assumed to be isotropic. A directional AIF (D-AIF) is proposed in [3], which also follows the process in Fig. 1 (b), but is much simpler than [2]. The support region is restricted to 1-D aligning integer pixels. To improve the performance of D-AIF, the enhanced AIF (E-AIF) [4] is proposed, which adds a 5×5 filter for integer pixels and a filter offset to each integer and sub position. The horizontal and vertical statistical properties are thought different, so $h$ is axisymmetric.

However, the side information is still significant at low bit-rates and the constraints and quantization on coefficients increase the prediction errors compared with the optimal filter. To solve the problem, parametric interpolation filter (PIF) is proposed, where the impulse response is a function determined by five parameters and the optimal $h$ for each frame is approximated by tuning the parameters. The criterion for the approximation is also proposed. With the parameters quantized to enough precision, the filter coefficients as the function values can be calculated with negligible loss, while the side information reduces to exactly 68 bits per frame.

2. PARAMETRIC INTERPOLATION FILTER

2.1. Effects of Coefficient Quantization

Supposing the resolution of the MV is 1/4-pel and each sub position is supported by the surrounding 6×6 integer pixels,
the size of $h$ in Fig. 1 (b) is $23 \times 23$. The optimal $h$, denoted as $h_{op}$, can achieve the minimum of (1)

$$MSE[h] = E\left[ \sum_{i,j} h(i,j)P(4x-i+d_x,4y-j+d_y)-S(x,y) \right]^2$$

where $P$ is upsampled from the reference frame by a factor 16 using zero-insertion method, $S$ is the current frame, and $d_x$ and $d_y$ are the two components of MV. The minimum can be achieved by the solution of Wiener-Hopf equations (2),

$$\sum_{i,j} h(i,j)R_{pp}(i-m,j-n) = R_{ps}(m,n)$$

where $R_{pp}$ and $R_{ps}$ represent the auto-correlation of $P$ and the cross-correlation of $P$ and $S$, respectively. These two statistical functions are based on the motion estimation (ME) before the current frame is coded. However, it is too expensive to code the $23^2$ coefficients in $h$, if no symmetry constraint and quantization are imposed.

The related work [1, 2, 3, 4] makes some effort to reduce the side information as introduced in Section 1, leading to an approximation of $h_{op}$, denoted as $\tilde{h}$. The difference between $h_{op}$ and $\tilde{h}$ is denoted as $h_{\Delta}$, i.e., $h_{\Delta} = h_{op} - \tilde{h}$, and the increased energy of prediction error, $\Delta \text{err}$, is given in (3).

$$\Delta \text{err} = MSE[\tilde{h}] - MSE[h_{op}]$$

$$= \sum_{i,j} \sum_{m,n} h_{\Delta}(i,j)h_{\Delta}(m,n)R_{pp}(i-m,j-n)$$

$$- 2 \sum_{m,n} h_{\Delta}(m,n) \sum_{i,j} h_{op}(i,j)R_{pp}(i-m,j-n) - R_{ps}(m,n))$$

Since $h_{op}$ is the solution of (2), the latter part in (3) is equal to zero and $\Delta \text{err}$ can be expressed by (4).

$$\Delta \text{err} = \sum_{i,j} \sum_{m,n} h_{\Delta}(i,j)h_{\Delta}(m,n)R_{pp}(i-m,j-n)$$

$\Delta \text{err}$ introduced by different interpolation techniques is studied, based on the first ten frames (except I frame) of ten HD sequences, and the typical results on city and raven are given in Fig. 2. Generally, $\Delta \text{err}$ introduced by 2-D non-separable AIF and D-AIF is close to or even larger than that introduced by the standard filter in H.264. Especially, 2-D non-separable AIF, which has larger support region than D-AIF and is expected to perform better, is more likely to have particularly large $\Delta \text{err}$. That is because $R_{pp}$ in (2) is ill-conditioned, so any slight change in $h_{op}$, having been denoted as $h_{\Delta}$, will influence $\Delta \text{err}$ significantly. Furthermore, the value of $\Delta \text{err}$ is more influenced by the quantization error than by the support region, and 2-D non-separable AIF has much more coefficients involved in quantization. This study implies that the accuracy of the coefficients is as important as the size of side information for the efficiency of AIF.

In the related work, the accuracy of the coefficients and the size of the side information are conflicting, because the coefficients are coded individually. In this paper, the impulse response of a filter is represented by a function $h_f$, determined by five parameters, and $h_{op}$ is approximated by tuning the parameters. The coefficients can be calculated as the function values, when the parameters are determined. Obviously, the side information for coding five parameters is very small. On the other hand, the accuracy of the coefficients can also be guaranteed, if the parameters are quantized in enough precision. How to find the form of $h_f$ is presented as below.

Let $P(e^{j\omega_x}, e^{j\omega_y})$ be the Fourier transform of the original reference frame. After the zero-insertion upsampling shown in Fig. 1 (b), the Fourier transform of the intermediate frame, $P_{16}(e^{j\omega_x}, e^{j\omega_y})$, is given by (5).

$$P_{16}(e^{j\omega_x}, e^{j\omega_y}) = P(e^{4j\omega_x}, e^{4j\omega_y})$$

According to (5), $P_{16}$ is a frequency-scaled version of $P$. Fig. 3 (a) and (b) give an example of $P$ and its corresponding $P_{16}$, respectively. In Fig. 3 (b), the undesired spectra centered at integer multiples of $(\pi/2, \pi/2)$, i.e., the original sampling rate, are introduced by the zero-insertion upsampling and should be removed. This requires a lowpass filter $h$ (see Fig. 1 (b)) with a gain of 16 and a cutoff frequency $\pi/4$, and Fig. 3 (c) shows the ideal frequency response. It is similar to the frequency response of the standard interpolation filter in H.264 and the information in $P$ is completely preserved.

In this paper, the desired filter is proposed to have a diamond-shaped passband (see Fig. 3 (d)) and some high-frequency components neither in the horizontal direction nor in the vertical direction are filtered out. That is because interpolation in the context of video coding is for a better MCP, and the high-frequency components are more likely to introduce large prediction error, thus exerting negative influence on MCP. This passband shape also accords with our observation on a number of optimal filters. Theoretically, the cutoff frequency should be $\pi/4$, such as the one in Fig. 3 (d). In practice, they may vary around $\pi/4$. As shown in Fig. 4, two parameters, $\omega_1$ and $\omega_2$, are used to denote the cutoff frequencies at two axes and the shadowed diamond-shape area, $\sigma$, represents the passband. The corresponding impulse
response, \( h_d \), can be obtained by inverse Fourier transform,

\[
h_d(m, n) = \frac{1}{4\pi^2} \int_{[-\pi, \pi]^2} H_d(e^{ju}, e^{jv}) e^{jmu+jnv} dudv \tag{6}
\]

where \( H_d \) is given as below.

\[
H_d(e^{ju}, e^{jv}) = \begin{cases} 
16, & \text{if } (u, v) \in \sigma \\
0, & \text{otherwise}
\end{cases} \tag{7}
\]

Substituting (7) into (6), we will get the result of \( h_d \) in (8).

\[
h_d(m, n) = \frac{1}{4\pi^2} \left( \int_{-\omega_1}^{0} \int_{-\omega_2}^{\omega_2} 16 e^{jmu+jnv} dudv \\
+ \int_{0}^{\omega_1} \int_{-\omega_2}^{\omega_2} 16 e^{jmu+jnv} dudv \right)
\]

\[
= \frac{8m_1\omega_2}{\pi^2} \sin(\frac{\omega_1 m + \omega_2 n}{2}) \sin(\frac{\omega_1 m - \omega_2 n}{2}) \tag{8}
\]

\( h_d \) has to be truncated by an appropriate window function \( w \) before used for interpolation. In this paper, \( w \) is proposed in (9), as it also has a diamond-shaped cross section.

\[
w(m, n) = a + \sin(b|m| + c|n|) \tag{9}
\]

Then, the approximation of the ideal desired filter, denoted as \( h_f \), is obtained as the product of \( h_d \) and \( w \).

\[
h_f(m, n) = h_d(m, n)w(m, n) = N \sin(\frac{\omega_1 m + \omega_2 n}{2}) \sin(\frac{\omega_1 m - \omega_2 n}{2})(a + \sin(b|m| + c|n|)) \tag{10}
\]

As shown in (10), \( h_f \) is determined by a parameter set, \( C = \{\omega_1, \omega_2, a, b, c\} \), and the factor \( N \) guarantees the filter gain is 16. \( h_f \) is the interpolation filter proposed in this paper, which represents filters by parameters rather than individual coefficients and is named as parametric interpolation filter (PIF).

### 3. EXPERIMENTAL RESULTS

The proposed PIF is integrated into the VCEG’s reference software KTA2.0 and Table 1 shows the test conditions. In the encoder, the MVs estimated in the first pass are reused in the second pass in order to show PIF’s performance with one-pass encoding strategy. All the filters in comparison, including the optimal filter, PIF, and the benchmarks, are implemented using 32-bit integer arithmetic. The AIF mode can be turned off based on the criterion of frame-level R-D distortion [6].

Table 2 compares the average percentage of bit-rate reduction [7], denoted as \( \Delta BR \), and the frequency of occurrence,
Table 1. Test conditions

<table>
<thead>
<tr>
<th>Test sequence</th>
<th>Frame number</th>
<th>Reference frame</th>
<th>Search range</th>
<th>Adaptive rounding</th>
<th>QP</th>
<th>FME</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequences</td>
<td>1280x720</td>
<td>4</td>
<td>±64</td>
<td>on</td>
<td>I(22, 27, 32, 37) P(23, 28, 33, 38) B(24, 29, 34, 39)</td>
<td>on</td>
</tr>
<tr>
<td>Intra frame period</td>
<td>Only the first frame</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Entropy coding</td>
<td>CABAC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Bit-rate reduction and frequency of occurrence

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>[1]</td>
<td>ΔBR (%)</td>
<td>freq. (%)</td>
<td>ΔBR (%)</td>
<td>freq. (%)</td>
</tr>
<tr>
<td>BigShip</td>
<td>-1.07</td>
<td>23.0</td>
<td>-1.53</td>
<td>55.3</td>
</tr>
<tr>
<td>City</td>
<td>-4.63</td>
<td>41.7</td>
<td>-3.90</td>
<td>65.2</td>
</tr>
<tr>
<td>Crew</td>
<td>-4.77</td>
<td>28.5</td>
<td>-4.65</td>
<td>48.2</td>
</tr>
<tr>
<td>Harbour</td>
<td>-4.65</td>
<td>53.1</td>
<td>-3.38</td>
<td>74.1</td>
</tr>
<tr>
<td>Jet</td>
<td>-0.38</td>
<td>15.4</td>
<td>-1.63</td>
<td>51.3</td>
</tr>
<tr>
<td>Optus</td>
<td>-1.12</td>
<td>39.0</td>
<td>-1.16</td>
<td>63.6</td>
</tr>
<tr>
<td>Raven</td>
<td>-2.03</td>
<td>33.8</td>
<td>-4.72</td>
<td>78.9</td>
</tr>
<tr>
<td>Sailormen</td>
<td>-0.80</td>
<td>36.0</td>
<td>-0.49</td>
<td>50.9</td>
</tr>
<tr>
<td>Sheriff</td>
<td>-1.97</td>
<td>29.8</td>
<td>-1.53</td>
<td>50.4</td>
</tr>
<tr>
<td>ShuttleStart</td>
<td>-3.13</td>
<td>2.7</td>
<td>-3.21</td>
<td>44.3</td>
</tr>
<tr>
<td>Average</td>
<td>-2.66</td>
<td>30.5</td>
<td>-2.46</td>
<td>58.0</td>
</tr>
</tbody>
</table>

Fig. 5. The impulse response and frequency response of different interpolation filters

4. CONCLUSION

In this paper, PIF is proposed for MCP, which determines interpolation filters by five parameters instead of individual coefficients, thus solving the conflict of the accuracy of coefficients and the size of side information. The experimental results show that PIF outperforms the existing AIF techniques and approaches the efficiency of the optimal filter.

5. REFERENCES