High Performance Error Concealment Algorithm By Motion Vector Refinement for MPEG-4 Video

Ming-Chieh Chi, Mei-Juan Chen, Jia-Hwa Liu and Ching-Ting Hsu
Department of Electrical Engineering
National Dong Hwa University, Taiwan

Abstract—In this paper, a new error concealment algorithm by recursive motion vector refinement is proposed. The proposed method utilizes the top/bottom motion vectors of lost MB in current and reference frames and refines motion vectors recursively. Simulation results based on MPEG-4 codec present the superior subjective and objective performance of the proposed technique compared with conventional temporal concealment techniques.

I. INTRODUCTION

The video compression techniques, such as MPEG-x and H.26x, are adopted in the remote video transmission. Compressed video over error prone networks may lead to objectionable visual distortion in the decoded video. Such distortion can propagate to succeeding frames because of the motion compensated prediction and the variable-length coding. Furthermore, traditional error control and recovery schemes for data communications have been extended for video transmission. These techniques aim at lossless recovery, such as forward error correction (FEC) and automatic retransmission request (ARQ) [1]. On the other hand, error concealment techniques have been proposed that strive to obtain a close approximation of the original signal. In the error concealment techniques [2-4], the decoder attempts to make the output signal least objectionable to human eyes by providing a subjectively acceptable approximation to the original video. The process is separated into spatial domain, temporal domain, frequency domain, or their combinations. The temporal error concealment processing with two major stages: “motion vector estimation” and “displacement compensation” attempts to estimate lost motion information and conceal lost macroblocks (MBs) by motion compensated temporal replacement. The simplest way to estimate lost motion is by replacing a damaged MB with the spatial corresponding MB in the previous frame, which is referred to as temporal replacement (TR) [5]. However, it will produce adverse visual artifacts in the presence of large motion. Significant improvement can be obtained by estimating lost motion from the motion vectors of the surrounding MBs. For example, one can estimate the lost motion information by either using the average, or the median of the motion vectors from spatially adjacent blocks [6].

J.Y. Pyun et. al. [7] proposed a temporal concealment method, which is called Bidirectional Motion Vector Tracking (BMVT). This technique extrapolates each motion vector of previously decoded frame, estimates the dependence of effect MB on motion extrapolation MB, and then, obtains the best estimation motion vector for the effect MB. This method can effectively improve the visual reconstruction quality especially when the loss of whole frame occurs.

The common problem with methods presented above is that the selected motion vector based on given algorithm is taken as the final estimation solution without further refinement. In this paper, we propose to add additional adjustment to the estimated motion vector by adjacent MBs. Simulation results show that significant improvement can be achieved by proposed motion vector refinement algorithm.

This paper is organized as follows. Section II presents the proposed error concealment algorithm. Simulation results are included in Section III, followed by the concluding remarks.

II. PROPOSED ERROR CONCEALMENT BY RECURSIVE MOTION VECTOR REFINEMENT

One error concealment method called co-location motion vector (MVC) uses the collocation motion vector in previous frame to conceal the lost MB. MVC can conceal large erroneous area by only utilizing the motion vectors in reference frame. Since MVC can only deal with the low motion video, we propose a modified collocation MV algorithm by using the information surrounding the corrupted MB to get a better quality in concealing the high motion video.

A. The Set of Candidates for MV

First of all, a set of combined MVs as the MV candidates is generated by the combination of the MVC and difference of top/bottom MVs as shown in Fig. 1(a)-(d). The collocation MV of lost MB, MVC, in the previous frame is
the base MV. The candidate set \( V_c \) includes the following motion vectors:

- \( MVC \)
- \( MVC + (MV_T^n - MV_T^{n-1}) \)
- \( MVC + (MV_B^n - MV_B^{n-1}) \)
- \( MVC + \text{avg}((MV_T^n - MV_T^{n-1}), (MV_B^n - MV_B^{n-1})) \)

\[ MV^{n+1} \quad MV^* \]
\[ MVC \quad \text{Lost MV} \]

(a)

\[ MV^{n+1} \quad MV^* \]
\[ MVC \quad \text{Lost MV} \]

(b)

\[ MVC \quad \text{Lost MV} \]
\[ MV^{n+1} \quad MV^* \]

(c)

\[ MVC \quad \text{Lost MV} \]
\[ MV^{n+1} \quad MV^* \]

(d)

![Diagram of motion vector candidates](image)

**Fig. 1** The candidate set of motion vector

**B. Side Matching Distortion (SMD) for Candidate MV**

Since side matching distortion (SMD) could obtain the matching motion vector for the lost MB and it is easy to implement in most video decoders, after generating the candidate motion vectors, these motion vectors are applied to SMD algorithm. For each MV candidate, the valid surrounding pixels around the lost MB to be concealed by motion-compensated and side match function is evaluated between the known valid pixels and the motion-compensated ones in the extended area. The criterion only involves pixels with known values, and can be seen as motion estimation on the available pixels in the frame. Although no motion vector is estimated, the criterion allows the validation of the MV candidates to preserve the spatial coherence of the image after motion compensation.

It is a tradeoff between the quality of the concealment and the size of the surrounding area injected in the side match function. On the one hand, with only one pixel extension, the support of the cost function dose not span pixels significant enough to remove any ambiguity between several motion vectors. On the other hand, when many pixels are included in the cost function (for instance 8 layers of surrounding pixels), the criterion may be disturbed by the motion of other object in the neighborhood and the best motion vector may not be selected. In our experiments as Table I, a width of 1 or 2 pixels for the surrounding layer is more precise for tested sequences. Fig. 2 shows the side matching area for candidate MVs. In Fig. 2 and Fig. 3, \( X_T, X_B, X_R \) and \( X_c \) denote the top, bottom, right and left MBs of the lost MB in the current frame \( n \), respectively. \( C_T, C_B, C_R \), \( C_L \) is the top, bottom, right and left MBs of the motion compensated MB in the reference frame \( n-1 \), respectively. The formation of side match distortion in Fig. 3 is defined as

\[
d_{\text{smT}} = \sum_{j=0}^{N-1} \left| X_T^{n-1} \right|_{0,j} - \left[ X_T^n \right]_{0,j}^r
\]

\[
d_{\text{smB}} = \sum_{j=0}^{N-1} \left| X_B^{n-1} \right|_{0,j} - \left[ X_B^n \right]_{0,j}^r
\]

\[
d_{\text{smR}} = \sum_{j=0}^{N-1} \left| X_R^{n-1} \right|_{0,j} - \left[ X_R^n \right]_{0,j}^r
\]

\[
d_{\text{smL}} = \sum_{j=0}^{N-1} \left| X_L^{n-1} \right|_{0,j} - \left[ X_L^n \right]_{0,j}^c
\]

(1)

**TABLE I** AVERAGE PSNR (dB) OBSERVATION WITH VARIOUS PIXEL WIDTHS FOR SIDE MATCH

<table>
<thead>
<tr>
<th>Layer</th>
<th>Football</th>
<th>Foreman</th>
<th>Stefan</th>
<th>Table Tennis</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>34.62</td>
<td>45.90</td>
<td>38.98</td>
<td>41.03</td>
</tr>
<tr>
<td>2</td>
<td>34.54</td>
<td>45.87</td>
<td>39.01</td>
<td>40.98</td>
</tr>
<tr>
<td>3</td>
<td>34.47</td>
<td>45.83</td>
<td>38.87</td>
<td>40.91</td>
</tr>
<tr>
<td>4</td>
<td>34.46</td>
<td>45.79</td>
<td>38.72</td>
<td>40.84</td>
</tr>
<tr>
<td>5</td>
<td>34.36</td>
<td>45.72</td>
<td>38.72</td>
<td>40.78</td>
</tr>
<tr>
<td>6</td>
<td>34.30</td>
<td>45.62</td>
<td>38.63</td>
<td>40.85</td>
</tr>
<tr>
<td>7</td>
<td>34.25</td>
<td>45.59</td>
<td>38.60</td>
<td>40.80</td>
</tr>
<tr>
<td>8</td>
<td>34.18</td>
<td>45.52</td>
<td>38.54</td>
<td>40.79</td>
</tr>
</tbody>
</table>

To reduce the computation of side match distortion, the absolution error rather than squared error is adopted in our method. The total side matching distortion across the MB boundaries is denoted as \( d_{\text{sm}} \), where

\[
d_{\text{sm}} = d_{\text{smT}} + d_{\text{smB}} + d_{\text{smR}} + d_{\text{smL}}
\]

(2)

The motion vector with minimum side matching distortion is selected as the concealed motion vector of lost MB. Finally, the lost MB is concealed by using motion vector compensation. Then the recovered motion vector \( \hat{V} \) is obtained from following equation

\[
\hat{V} = \arg \min_{v \in V_c} d_{\text{sm}}
\]

where \( V_c \) is the candidate set.
III. EXPERIMENTAL RESULTS

To simulate video transmission, MPEG-4 encoder is modified to generate desired erroneous video data which encoded by the MoMuSys reference software [8]. In the experiments, Foreman, Stefan and Mobile in CIF format are encoded for 100 frames. The MPEG-4 encodes these three sequences into 128Kbps, 256Kbps and 512Kbps with packet loss rates (PLR) of 1%, 5% and 10%. Parameters in encoder configuration file are set as the following: 30 frames per second, quantization parameter is 5, there are 500 bytes per video packet, no data partitioning and no RVLCs.

MPEG-4 MoMuSys decoder is revised by adding different error concealment algorithms: 1) Temporal replacement (TR); 2) Median MV of top/bottom MBs (Median MV); 3) Average MV of top/bottom MBs (AV); 4) Bidirectional Motion Vector Tracking (BMVT)[5]; 5) Proposed method, to decode these erroneous and video data.

Table II-IV demonstrate the average PSNR comparison of various algorithms under various packet loss rates. As shown in the tables, the performance of our proposed algorithm always performs better than other methods. In most simulation environments, our proposed algorithm has benefit about 1–2 dB in PSNR. Fig. 5 and Fig. 6 show the subjective comparison of Stefan and Foreman sequences when the packet rate is 10% at 128Kbps and 256Kbps, respectively. The erroneous data are concealed by TR, Median MV, AV, BMVT and our proposed algorithm. In Fig. 5, our proposed method has significant quality improvement, especially at Stefan’s legs. Our proposed method also has better subjective quality in Foreman’s face in Fig. 6.

IV. CONCLUSION

A new error concealment method with recursive motion vector refinement is proposed. Side matching distortion is utilized to estimate and refine the MV for the corrupted MB. Simulation results show that the recursive motion vector refinement can significantly improve the reconstruction quality of the lost MBs.

### TABLE II: AVERAGE PSNR (dB) COMPARISON AT DIFFERENT PACKET LOSS RATES FOR FOREMAN SEQUENCE

<table>
<thead>
<tr>
<th>Bitrate</th>
<th>PLR</th>
<th>TR</th>
<th>Median MV</th>
<th>AV</th>
<th>BMVT</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>256Kbps</td>
<td>1%</td>
<td>32.26</td>
<td>34.17</td>
<td>34.16</td>
<td>34.44</td>
<td>36.07</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>29.13</td>
<td>29.83</td>
<td>29.75</td>
<td>30.64</td>
<td>32.46</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>27.60</td>
<td>27.82</td>
<td>27.76</td>
<td>28.54</td>
<td>30.64</td>
</tr>
</tbody>
</table>

### TABLE III: AVERAGE PSNR (dB) COMPARISON AT DIFFERENT PACKET LOSS RATES FOR STEFAN SEQUENCE

<table>
<thead>
<tr>
<th>Bitrate</th>
<th>PLR</th>
<th>TR</th>
<th>Median MV</th>
<th>AV</th>
<th>BMVT</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>128Kbps</td>
<td>1%</td>
<td>28.48</td>
<td>30.45</td>
<td>30.75</td>
<td>31.61</td>
<td>32.10</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>25.39</td>
<td>26.59</td>
<td>26.67</td>
<td>28.22</td>
<td>28.32</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>23.68</td>
<td>24.73</td>
<td>24.73</td>
<td>26.52</td>
<td>26.66</td>
</tr>
</tbody>
</table>
TABLE IV  AVERAGE PSNR (dB) COMPARISON AT DIFFERENT PACKET LOSS RATES FOR MOBILE SEQUENCE

<table>
<thead>
<tr>
<th>Bitrate</th>
<th>PLR</th>
<th>Mobile</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>TR</td>
<td>Median</td>
</tr>
<tr>
<td>512Kbps</td>
<td>1%</td>
<td>27.51</td>
<td>31.71</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>24.53</td>
<td>28.34</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>22.89</td>
<td>26.70</td>
</tr>
</tbody>
</table>

REFERENCES


Fig. 5 Subjective video quality comparison for sequence “Stefan” with PLR is 10% at 128Kbps (Frame #7)

Fig. 6 Subjective video quality comparison for sequence “Foreman” with PLR is 10% at 256Kbps (Frame #33)