CONCEALMENT OF INTERPOLATION ERRORS FOR LOW BIT-RATE MOTION-COMPENSATED INTERPOLATION

Yan Wu, M.N.S. Swamy, Fellow, IEEE, and M.O. Ahmad, Fellow, IEEE

Department of Electrical & Computer Engineering
Concordia University, Montreal, Quebec, Canada H3G 1MB
Email: [yan_wu, swamy, omair]@ece.concordia.ca

ABSTRACT

In this paper, we propose a low cost motion-compensated interpolation technique to improve the video quality for the low bit-rate video encoded in conjunction with frame dropping. The proposed approach exploits the block-based motion vector field available to the decoder to avoid the complex motion estimation at the receiver. An iterative refinement technique derived using the finite element method is employed to efficiently conceal the interpolation errors caused by unfilled and overlapped pixels in the predicted frames. Consequently, no pixel classification is needed in the proposed technique, thus substantially reducing the computational complexity. Simulation results show that this technique results in reconstructed frames with good visual quality.

1. INTRODUCTION

In low bit-rate video applications, frame dropping (frame skipping) associated with the existing video coding techniques is commonly used to increase the compression gain. In order to display the video at normal frame rate, frame interpolation is employed at the receiving end. For instance, given successive reconstructed frames F_{t-1} and F_t $(t \geq 1)$, we may need to interpolate the frame $F_{t-\tau}$ $(0 < \tau < 1)$ for a given τ. Early attempts such as frame repetition (FR) and frame averaging (FA) are straight-forward methods, but for video sequences containing fast motions, FR and FA will respectively result in jerky and blurred images. Therefore, a motion-compensated interpolation (MCI) scheme giving better results was first proposed in [1]. Most of the proposed MCI schemes are intended for sophisticated applications like offline video format conversion, which require the motion estimation and classification of each pixel into different classes to be performed at the receiver for the interpolated frame. These procedures are computationally demanding for a low bit-rate video application.

Without using the motion estimation at a low bit-rate video decoder, some block-based motion-compensated interpolation schemes [2]-[4] have been proposed. The scaled motion vector field (SMVF) technique utilizes the block-based motion information available at the decoder [2]. However, the interpolated frame shows some blurred and block artifacts due to the linear interpolation and averaging of the overlapped pixels. In [3], the authors have pointed out that even spatial interpolation to deal with these unfilled holes, which are left after performing the block-based prediction for the interpolated frame, is not simple; this is so since the neighborhood of a hole may still contain other holes. In view of this, the authors in [3] have proposed the non-deformable block-based frame MCI (NB-FMCI) scheme, wherein there are only those holes that follow some easy-to-handle patterns associated with the pixel classification. In [4], the unfilled pixels are predicted by frame repetition, but this method may produce stripe artifacts.

In this paper, we propose a technique to conceal the overlapped as well as the unfilled pixels by an iterative refinement procedure that is based on the finite element method (FEM) in order to avoid the need for any motion estimation or pixel classification at the decoder. This results in an inexpensive decoder to carry out the frame interpolation that meets the requirements of subjective quality of the interpolated frame.

2. CONCEALMENT OF INTERPOLATION ERRORS

Let F_{t-1} and F_t be successive reconstructed frames at the receiving end. Let $\vec{p}$ represent the pixel location and $\vec{v}_i$ the transmitted motion vector of a block B_i in the frame F_t. Obviously, $\vec{v}_i$ is assigned to any pixel at $\vec{p} \in B_i$. Let $\hat{I}(\vec{p},t)$ denote the pixel luminance at $\vec{p}$ in frame F_t. Considering a constant speed model, the motion vector between F_t and $F_{t-\tau}$ is given by $\vec{v}_i^{\tau} = \tau \cdot \vec{v}_i \quad 0 < \tau < 1$. For any $\vec{p} \in B_i$, the corresponding location in $F_{t-\tau}$ is found by using $\vec{p}_\tau = \vec{p} + round(\tau \cdot \vec{v}_i)$, where $round(\cdot)$ is an operator that

0-7803-8554-3/04/$20.00 ©2004 IEEE.

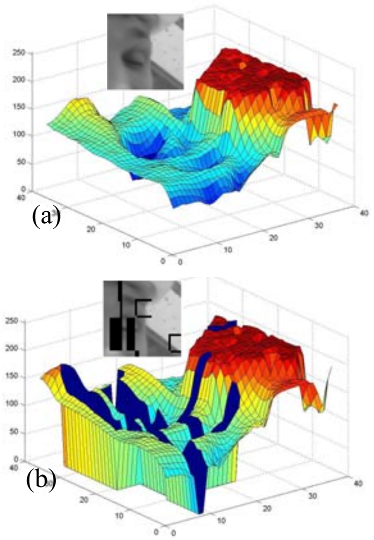

Fig. 1 (a) A sub-image of the original frame and its 3D surface plot; (b) A sub-image of the interpolated frame, with interpolation errors, and its 3D surface plot.

rounds to the nearest integer within the image dimension. Then, the pixel luminance at $\vec{p}_\tau$ is given by $\hat{l}(\vec{p}_\tau, t-\tau) = \hat{l}(\vec{p}, t)$, which we refer to as the backward motion prediction that yields the predicted frame $\hat{F}_{t-\tau}$. This interpolation may cause the occurrence of overlapped as well as unfilled pixels. The values of such pixels cannot be predicted correctly using the current frame F_t, giving rise to errors that we refer to as the interpolation errors. As shown in Fig. 1, we mark these pixels, and assign a zero value to these pixels. We shall refer, thereafter, to such pixels as the unknown pixels. The remaining pixels are referred to as known pixels.

Usually, such interpolation errors occur in regions sensitive to the human eyes within a frame, such as the area containing the head and shoulders, where the motion is likely to be relatively high and nonuniform. We shall refer to this area as the region of interest (ROI). If these errors are not concealed properly, the reconstructed frame has an annoying stripe effect in the ROI, leading to an unsatisfactory subjective quality. A variety of error concealment techniques [5] have been widely employed for video communication in an error-prone environment. However, none of these proposed error concealment schemes is suitable to be applied to the problem at hand.

To conceal the interpolation errors, we now propose an iterative refinement technique. From the 3D surface plot of the sub-image shown in Fig. 1(b), it is seen that the surface of the predicted frame is not continuous due to the unknown pixels. Under certain constraints, we can find a suitable luminance function, which will assign a luminance value to each of the unknown pixels. Thus, the 3D surface plot of the sub-image of the post-processed predicted frame will have a continuous surface similar to that shown in Fig. 1(a). We refer to such an assigned luminance value as a restored pixel luminance value, and the pixel itself as a restored pixel. Since we do not alter the values of known pixels, and at the same time, the restored pixels are intended to show a smooth effect with immediate-neighboring pixels, we apply the following constraints to the luminance function. A restored pixel should be fit smoothly with its immediate-neighboring pixels, which are either known pixels or themselves restored pixels.

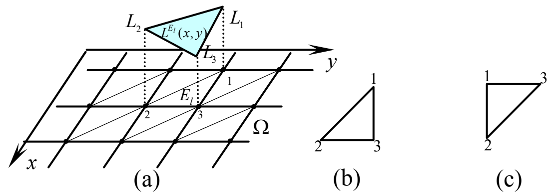

Fig. 2 (a) The triangular elements partitioned in the domain Ω of $L(x,y)$. Shaded area is the restored luminance function in the element E_l; (b) Type I local triangular element; (c) Type II local triangular element.

The proposed iterative refinement technique is based on the FEM, which offers the simplicity of piecewise approximation of a function given its values at discrete points. Let us apply the FEM to construct a continuous luminance function $L(x,y)$, the domain Ω of this function being a given frame. Therefore, for a predicted frame $\hat{F}_{t-\tau}$ of size $(W \times H)$ pixels, Ω is

$$\Omega = [0, W-1] \times [0, H-1]. \quad (1)$$

Now, we employ triangular segmentation and partition Ω into $m\,(m \in Z)$ non-overlapping triangular elements, the nodes of which correspond to the pixel locations in an image, see Fig. 2(a). These elements may be classified into two types of local triangular elements, the first, second, and third nodes of which are ordered counterclockwise (see Figs. 2(b) and 2(c)). D is a finite set containing all the nodes in Ω. That is $D \subset \Omega$,

$$D = (0,1,\ldots,W-1) \times (0,1,\ldots,H-1). \quad (2)$$

After the triangular segmentation of $\hat{F}_{t-\tau}$, the nodes corresponding to the known pixels are referred to as the known nodes, whereas the remaining nodes as unknown nodes. Let K $(K \subset D)$ denote the set of known nodes and U $(U \subset D)$ the set of unknown nodes, $U \cup K = D$ and $U \cap K = \varnothing$. Under the constraint

$$L(x,y) = \hat{l}(\vec{p}, t-\tau),\ (\vec{p} = (x,y)^T),\ (x,y) \in K, \quad (3)$$

$L(x,y)$ is given by [6],

$$\begin{aligned} L(x,y) &= L^{E_l}(x,y) \quad (x,y) \in E_l \\ L^{E_l}(x,y) &= L_1\varphi_1(x,y) + L_2\varphi_2(x,y) + L_3\varphi_3(x,y) \end{aligned} \quad (4)$$

where $L^{E_l}(x,y)$ $\left((x,y)\in E_l, l=(0,1,\dots,m-1)\right)$ is a continuous function as shown in the shaded area of Fig. 2(a), $L_i = L(x_i, y_i)$, $(i=1,2,3)$, and $(x_i, y_i)\in D$, (x_i, y_i) being the location of the ith node in E_l. The shape functions are given by

$$\varphi_1(x,y)=\frac{1}{2S}\begin{vmatrix}1 & x & y\\ 1 & x_2 & y_2\\ 1 & x_3 & y_3\end{vmatrix},\ \varphi_2(x,y)=\frac{1}{2S}\begin{vmatrix}1 & x_1 & y_1\\ 1 & x & y\\ 1 & x_3 & y_3\end{vmatrix},\ \varphi_3(x,y)=\frac{1}{2S}\begin{vmatrix}1 & x_1 & y_1\\ 1 & x_2 & y_2\\ 1 & x & y\end{vmatrix}, \quad (5)$$

with S denoting the area of the element E_l.

To satisfy the requirements set earlier regarding the smooth fit of a restored pixel, we apply the optimization constraint that the surface represented by $L(x,y)$ has the smallest area. Then, we can find suitable luminance values for the unknown nodes to minimize

$$\iint_\Omega \sqrt{1+\left(\frac{\partial L(x,y)}{\partial x}\right)^2+\left(\frac{\partial L(x,y)}{\partial y}\right)^2}\,dxdy\,. \quad (6)$$

That is equivalent to finding a set containing $L(i,j)$ at each node $(i,j)\in U$, which minimizes

$$\sum_l \iint_{E_l}\left[\left(\frac{\partial L^{E_l}(x,y)}{\partial x}\right)^2+\left(\frac{\partial L^{E_l}(x,y)}{\partial y}\right)^2\right]dxdy \quad (7)$$

It can be shown that the optimal luminance value at an unknown node $(i,j)\in U$, in a raster scan order, using the following equations in an iterative manner.

$$L^{(k+1)}(i,j) = S_{NS}^{(k)}(i,j) + S_{WE}^{(k)}(i,j)\,,\ (k\ge 0) \quad (8)$$

with

$$S_{NS}^{(k)}(i,j)=\begin{cases}\frac{1}{2}L^{(k)}(i+1,j) & i=0\\ \frac{1}{2}L^{(k+1)}(i-1,j) & i=H-1\\ \frac{1}{4}\left[L^{(k+1)}(i-1,j)+L^{(k)}(i+1,j)\right] & i\neq 0\ or\ H-1\end{cases}\quad S_{WE}^{(k)}(i,j)=\begin{cases}\frac{1}{2}L^{(k)}(i,j+1) & j=0\\ \frac{1}{2}L^{(k+1)}(i,j-1) & j=W-1\\ \frac{1}{4}\left[L^{(k+1)}(i,j-1)+L^{(k)}(i,j+1)\right] & j\neq 0\ or\ W-1\end{cases}.$$

It can be shown that convergence is guaranteed [6] with such an iterative refinement procedure for any unknown node in Ω (i.e., any unknown pixel in the predicted frame $\hat{F}_{t-\tau}$). The procedure will not alter the values of the known nodes involved in (8), with $L^{(k+1)}(x,y)=L^{(k)}(x,y)$, $(x,y)\in K$, whereas the unknown nodes will be refined gradually. When the stop criterion is met, we obtain $\hat{L}(i,j)$, $(i,j)\in U$, as the luminance of a restored pixel. When all the unknown nodes in Ω are assigned such values, we have the interpolated frame $F_{t-\tau}$. In the regions, where interpolation errors have occurred, the restored pixels will exhibit a smooth variation without any discontinuity.

3. CONCEALMENT ALGORITHM

Based on the discussion in the previous section, we may formulate an algorithm for the concealment of the interpolation errors in the interpolated frame $\hat{F}_{t-\tau}$, which is an image of size $(W\times H)$ pixels. Let $\vec{L}$ be a vector, each component of which corresponds to the calculated luminance value of an unknown pixel $(i,j)\in U$, and $L^{(0)}(i,j)$ is an initial value of such a pixel. Let $MAPD(\cdot,\cdot)$ represent an operator that calculates the maximum absolute pixel difference and ε a pre-specified tolerance. Then the algorithm can be formulated as follows.

1. Initialize $\vec{L}^{(0)}$ with $L^{(0)}(i,j)$, and set $k=1$.
2. For each pixel at $(i,j)\in U$, compute using (8),
   $L^{(k+1)}(i,j) = S_{NS}^{(k)}(i,j) + S_{WE}^{(k)}(i,j)$
3. If $MAPD(\vec{L}^{(k+1)}, \vec{L}^{(k)}) < \varepsilon$, set $\hat{\vec{L}} = \vec{L}^{(k+1)}$, and stop.
4. Otherwise, increment k and go to step 2.

In the above, $\hat{\vec{L}}$ represents the vector of luminance values of the restored pixels that conceals the interpolation errors. The operation of $MAPD(\cdot,\cdot)$ is restricted to integer accuracy so that we can set the tolerance $\varepsilon=0$. The initialization value $L^{(0)}(i,j)$ has an influence on the speed of convergence. We choose for $L^{(0)}(i,j)$ three different values, namely, 0, the minimum luminance and the mean value of the known pixels in the image. Simulation results show that initialization with the mean value of the known pixels in the image results in the fastest convergence.

Since the algorithm above can be applied to an image of size $N\times N$ $(N>1)$, we can partition the interpolated frame into image blocks BLK_i, $i=1,2,\dots,Q$, $Q=(W\times H)/N^2$, of equal size so that the computational effort to conceal the interpolation errors gets distributed among the various blocks having interpolation errors. A value of $N=16$ is selected for our algorithm, since simulation studies show that it results in a better performance than when $N=8$. At the same time, it coincides with the macro-block size used in most of the video standardization and conventional coding schemes.

4. SIMULATION RESULTS

In order to evaluate the proposed algorithm, several original QCIF 30 fps test sequences are sub-sampled with a dropping factor of 2, which means every other frame in the sequence is skipped. All these skipped frames are interpolated using the sub-sampled test sequences and the motion vector field generated through the block matching algorithm applied to the sequences. Four algorithms are implemented: (a) FR, SMVF [2], and MCI+FR algorithm, which predicts the unfilled pixels with FR [4], and (b) the proposed IEC-MCI (Interpolation Error Concealment-MCI) algorithm, which conceals the interpolation errors using an iterative refinement procedure. The algorithms were implemented on a Pentium III 800 Hz PC, and four standard QCIF-format video sequences employed: frame 90 to frame 180 of Foreman, frame 20 to frame 110 of

Suzie, frame 20 to frame 110 of Miss America, and frame 40 to frame 130 of Mother Daughter.

The results of objective test are presented in Table 3. Note that the average PSNR is obtained over all the skipped frames. The motion-compensated interpolation schemes show a better performance than the simple temporal interpolation technique such as the FR technique. It is observed that the proposed IEC-MCI algorithm exhibits a performance comparable to that of MCI + FR algorithm, but both these algorithms are superior to the SMVF algorithm.

The advantage of the proposed method can be better appreciated through a visual comparison. For this purpose, the original and the interpolated frames of the video sequence Foreman (frame 179) are shown in Fig. 3. It is evident that the proposed method outperforms the other three methods. Compared to the MCI + FR technique, the proposed method provides a finer texture in those areas where interpolation errors occur. In contrast, the MCI + FR method reveals unpleasant stripe-like artifacts due to the simple FR strategy used to interpolate the unfilled pixels. As mentioned earlier, since these annoying artifacts appear in the ROI, the subjective quality of the interpolated image seriously deteriorates. It is noted that the proposed algorithm does not have such unpleasant artifacts.

Table 3. Average PSNR values (dB) using various algorithms

Sequence	FR	SMVF	MCI+FR	IEC-MCI
Foreman	28.62	28.63	30.02	29.98
Suzie	30.18	30.53	32.42	32.56
Miss America	38.48	37.93	39.54	39.53
Mother Daughter	39.11	39.01	39.12	39.11

Table 4. Average number of basic operations and unknown pixels for the various test sequences

Sequences	Forman	Suzie	Miss America	Mother Daughter
Unknown Pixels	1063	1041	252	232
Basic Operations	28407	28933	3000	3795

Table 4 shows the average number of basic operations (including those of the addition, multiplication, and absolute difference) needed for the concealment of the interpolation errors over all the skipped frames. These values suggest that the proposed method has much a lower computational complexity than that of any other block-based MCI schemes employing pixel classification, since it introduces the operations required for the detection of scene changes for each pixel using F_{t-1}, F_t and F_{t+1}, as well as the analysis of motion vector fields at the decoder. The advantages of the proposed method allow it to be applied to low bit-rate video decoders.

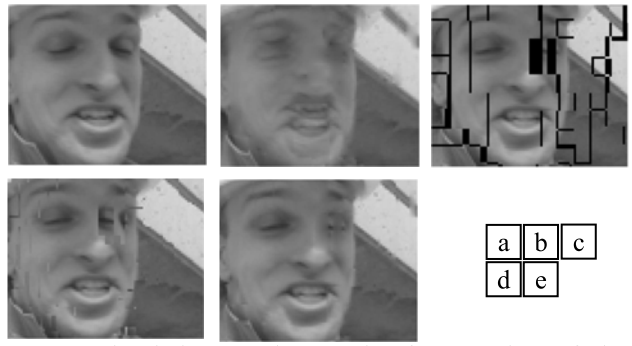

Fig. 3 Simulation results on the face portion of the Foreman frame 179. (a) Original frame; (b) Reconstructed frame using SMVF; (c) Predicted frame with interpolation errors; (d) Reconstructed frame using FR-MCI; (e) Reconstructed frame using IEC-MCI

5. CONCLUSION

A motion-compensated interpolation algorithm has been presented for low bit-rate video applications in order to reconstruct the dropped frames. The proposed approach is characterized by a low computational complexity and a good visual quality. These have been achieved by introducing in the algorithm a technique that conceals the interpolation errors making use of the finite element method. Simulation results support these features of the proposed technique.

6. REFERENCES

[1] R.Thoma and M.Bierling, "Motion compensated interpolation considering covered and uncovered background," *Signal Processing:Image Communication*, vol.1, pp.1191 –212, 1989.

[2] J.K.Su and R.M.Mersereau, "Motion-compensated interpolation of untransmitted frames in compressed video," *Conference Record of the Thirtieth Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, USA, pp.100 – 104,1996.

[3] T.Y. Kuo and C.C.J. Kuo, "Motion-compensated interpolation for low-bit-rate video quality enhancement," *SPIE Conf. on Application of Digital Image Processing XXI*, San Diego, California, vol.3460, pp.277-288, 1998.

[4] T. Liu, et al., "Frame interpolation scheme using inertia motion prediction," *Signal Processing: Image Communication*, vol. 18, pp.221 –229, 2003.

[5] Y. Wang, et al., " *Error resilient video coding techniques*", IEEE Signal Processing Magazine, PP. 61-82, July 2000.

[6] O.C. Zienkiewicz and R.L. Taylor, *The Finite Element Method*, Oxford; Boston: Butterworth-Heinemann, 2000.

[7] Mark A. Robertson and Robert L. Stevenson, "Temporal Resolution Enhancement in Compressed Video Sequences," *EURASIP Journal on Applied Signal Processing: Special Issue on Nonlinear Signal Processing*, pages 230-238, December 2001