AN ADAPTIVE DE-INTERLACING ALGORITHM BASED ON TEXTURE AND MOTION VECTOR ANALYSIS

Jianguo Du[1], Songnan Li[1], Debin Zhao[1], Qian Huang[2], Wen Gao[2]

[1]Department of Computer Science& Engineering, Harbin Institute of Technology, Harbin 150001,China
[2]Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China
{jgdu, snli, dbzhao, qhuang, wgao}@jdl.ac.cn

ABSTRACT

In this paper, we propose a novel hybrid de-interlacing algorithm, which effectively combines two motion-compensated (MC) de-interlacing techniques: MC Median Filtering (MCMF) and Adaptive Recursive (AR) with one spatial approach: Line Averaging (LA). Despite of its drawbacks, AR is one of the best methods nowadays. MCMF helps reduce flickers and LA is very robust to erroneous motion vectors. The interpolation switches among these methods based on the proposed measurement of texture smoothness and motion vector (MV) reliability. MCMF is adopted when MV is reliable and texture is rich. LA is used when MV is unreliable and texture is smooth. AR is applied to the remaining regions. Experimental results show that the proposed algorithm is superior to the compared algorithms in terms of peak signal-to-noise ratio (PSNR) and that the de-interlaced videos have very high subjective quality.

1. INTRODUCTION

De-interlacing is the conversion from interlaced videos to progressive ones. Traditional television systems adopted the interlaced format in order to save bandwidth [1]. However, interlaced scanning has many artifacts such as interline flicker, inter-field flicker, twitter, and serration. Hence interlaced programs need to be de-interlaced for display purpose. Additionally, interlaced scanning introduces a lot of inconvenience for video processing, e.g. motion estimation (ME) and video coding. Therefore a high quality de-interlacing algorithm is required to solve all the problems mentioned above.

De-interlacing algorithms can be classified into three categories, namely linear filtering, non-linear adaptive filtering, and motion-compensated filtering [2]. Linear algorithms are easy to implement, but they cause blur and alias in the spatial and temporal domains. Non-linear adaptive algorithms include edge adaptive, motion adaptive, and median filtering methods. Median filtering includes the edge adaptability in the spatial domain and motion adaptability in the time domain implicitly. Non-linear adaptive algorithms can't eliminate alias efficiently either. MC algorithms, making the most of correlation information in the time and spatial domains, may achieve the best results. However, they are very sensitive to erroneous MVs and may cause severe local distortions such as the block artifacts. Actually, due to the occlusion and aperture problems, MVs could not be very accurate.

So we have to use a hybrid algorithm that combines different methods to achieve perfect results. The AR [3] was ranked the best existing de-interlacing technique in [2]. Experiments reveal that AR has very good adaptability. However, it has the disadvantage of having blurs and flickers in the areas with rich texture and reliable ME, which can be removed by MCMF [4]. Additionally, when motion vectors are erroneous and texture is very smooth, there would be many noises caused by AR, which makes visual quality reduce greatly and looks very unpleasant. In this case, it is better to use LA.

The key problem is how to combine these three methods. In [5], they analyzed the reliability of MV and computed the weighted average of the results of a GST De-interlacer and a MC V-T Filter. We didn't follow this approach, as averaging different methods introduces blurring [2]. And it requires very high computational complexity, as the interpolation result of each method has to be calculated. In this paper, we propose an efficient way to switch among the three methods adaptively rather than compute the weighted average. By analyzing the smoothness of texture and reliability of MV, we classify the picture areas into four parts and apply different interpolation methods to them: MCMF is adopted where MV is reliable and texture is rich; LA is used where MV is unreliable and texture is smooth; AR is applied where MV is reliable and texture is smooth; in the region where MV is unreliable and texture is rich, the adaptive averaging mechanism in AR performs better than LA, so AR is still used. In this way, we take full advantage of each method and achieve perfect subjective and objective effects. Furthermore, due to the introduction of MCMF and LA, the computational load is reduced compared with using AR alone.

The rest of this paper is organized as follows. In Section 2, we first give a brief introduction to the proposed de-interlacing system and then describe the ME, Texture and MV Reliability Analysis, and Adaptive MC Interpolation separately. Section 3 presents and discusses the experimental results. Finally our conclusions are given in Section 4.

2. ALGORITHM DESCRIPTION

As shown in Fig.1, let f_n and f_{n+1} denote the current and next field, respectively. And let F_{n-1} and F_n represent the previously and currently de-interlaced frame, respectively. The de-interlacing system is comprised of three modules: Bidirectional ME, Texture and MV Reliability Analysis, and Adaptive MC Interpolation.
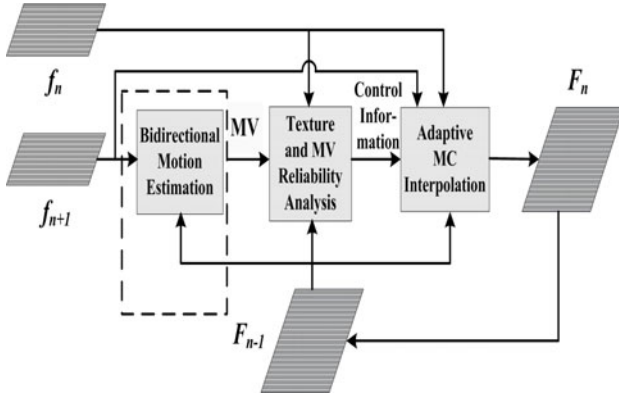

Fig.1 System Overview

The ME module adopts the bidirectional ME. Then MVs as well as F_{n-1} and f_n are fed to the Texture and MV Reliability Analysis module. After analyzing, the control information is sent to the Adaptive MC Interpolation module to select the interpolation method.

2.1. Bidirectional ME

As is shown in Fig.2, ME in de-interlacing includes unidirectional ME and bidirectional ME.

The matching error formulas are as follows respectively:

$$\varepsilon_1(\overline{D}, \overline{X}, n) = \sum_{\overline{x} \in B(\overline{X})\,\&\&\,y\%2=n\%2} | f_n(\overline{x}) - F_{n-1}(\overline{x} - \overline{D})|, \quad (1)$$

$$\varepsilon_2(\overline{D}, \overline{X}, n) = \sum_{\overline{x} \in B(\overline{X})\,\&\&\,(y+1)\%2=n\%2} | f_{n+1}(\overline{x} + \overline{D}) - f_{n-1}(\overline{x} - \overline{D})|,$$

$$(2)$$

where $\overline{X}$ represents the center of the block $B(\overline{X})$, and $\overline{x} = (x, y)^T$ denotes the horizontal/vertical coordinates.

As bidirectional ME won't be influenced by the interpolation error propagation, we adopt bidirectional ME. The 3-D Recursive Search (3DRS) [6], which shows its superiority at true ME and complexity, is used in our system.

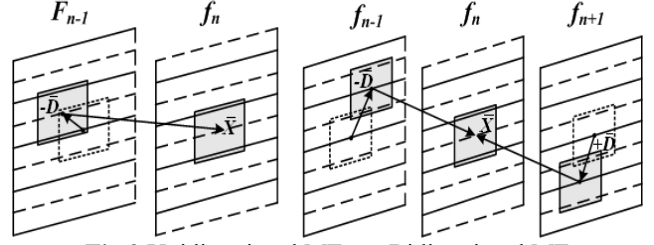The ME block size is 8 by 8. The MV filtering is applied according to [6] to obtain the MV of every 4 by 4 block.

Fig.2 Unidirectional ME and Bidirectional ME

2.2. Texture and MV Reliability Analysis

Let variable *Texture_smooth* indicate the texture smoothness of the current block. *Texture_smooth* equals one means the texture is smooth.

$$Texture\_smooth = \begin{cases} 1, \text{if } SD < C * SAD \\ 0, \text{otherwise} \end{cases}, \quad (3)$$

where $C=1.5$, *SAD* means the forward matching error of the current block, that is to say:

$$SAD = \varepsilon_1(\overline{D}, \overline{X}, n), \quad (4)$$

SD is the standard difference multiplied the number of pixels N in the current block. Fig.3 shows a block in an even field, in which N equals 8.
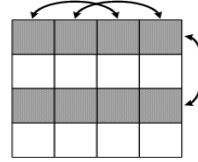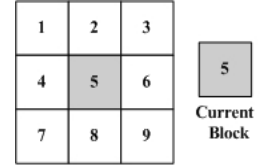

Fig.3 A block being processed **Fig.4** Eight neighborhoods of current block

Motion vector analysis mainly involves vector field analysis and matching error analysis. For the translational motion, it is assumed that ME is very accurate when MVs are consistent. However, it is not very suitable for the rotational and zooming motions. In these situations, we can analyze the matching error.

SAD couldn't be simply used as the criterion to judge the MV reliability, since the matching error also depends on the texture. For example, *SAD* is generally very small in smooth areas. Therefore, according to the analysis in [7], a linear function *Function(VAR)* is defined as follows to determine whether the *SAD* is in the reasonable range:

$$Function(VAR) = a * VAR + b, \quad (5)$$

where a and b are set to 0.75 and 4.0, respectively. The texture measurement *VAR* is the sum of the absolute difference between pixels at one interval, as shown in Fig.4. The computation of *VAR* is similar to [7]. If *SAD* is not larger than *Function(VAR)*, it is considered to be reasonable.

Then we consider the eight neighborhoods of the current block shown in Fig.4. The MV reliability of the current block can be formalized as:

$$Mv\_right = f(MV_1,\dots,MV_9,SAD_1,\dots,SAD_9,VAR_1,\dots,VAR_9), \quad (6)$$

where MV_i, SAD_i, and VAR_i are the MV, the SAD, and the texture measurement of the ith block (i=1~9), respectively. Function f is defined as follows:

$$f = \begin{cases} 1, \text{if } MV_1 = MV_2 = \dots = MV_9 \text{ or } N_{sad} <= 3, \\ 0, \text{otherwise} \end{cases} \quad (7)$$

where N_{sad} means the number of blocks that have unreasonable SAD. As the MVs got by 3DRS are very smooth, the condition on local consistency of MVs is very strict.

The major difference between the method in this paper and that in [7] is that we utilize the number of unreasonable blocks to determine the reliability of MV, which reduces the possibility of misjudgment caused by considering the current block alone.

2.3. Adaptive MC Interpolation

LA is applied to the first field and the proposed adaptive interpolation is applied to the other fields.

The detailed algorithm is shown in Fig.5. The positions of A, B, C, D, E, F, and I are depicted in Fig.6.
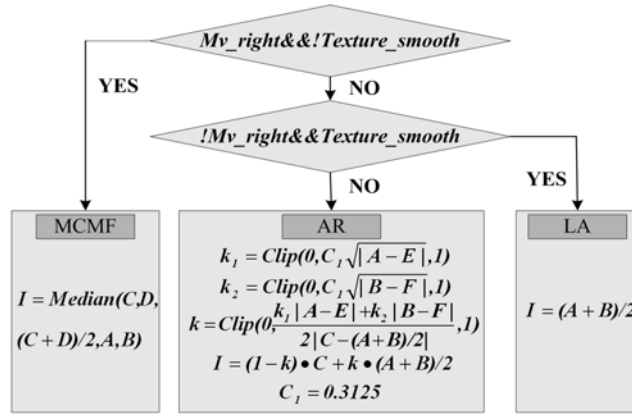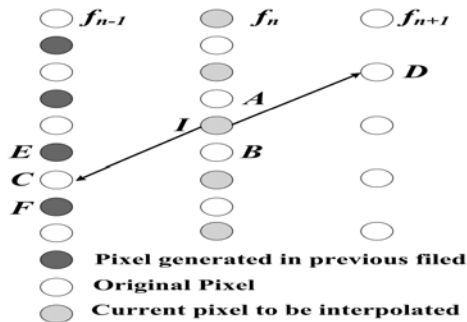

Fig.5 Adaptive Interpolation Algorithm

Fig.6 Positions of pixels for interpolation

3. EXPERIMENTAL RESULTS

3.1. Subjective Visual Quality

Fig.7 is a frame from the sequence Basketball. When MCMF is applied directly without reliability analysis of MVs, it can be seen that local distortions are very obvious in the areas where MV is inaccurate. The result of AR seems close to that of our adaptive method. However, the processed sequences have obvious flickers in the areas, such as auditoria and stairs, where no flicker occurs in our adaptive algorithm by using MCMF. Therefore, Fig.7 shows the necessity of combining MCMF and AR by MV reliability analysis.
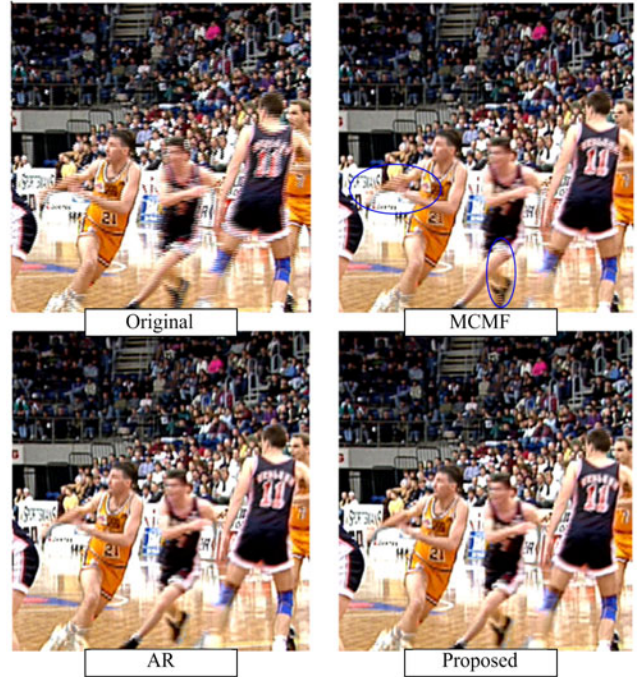

Fig.7 Subjective comparisons of Basketball

Fig.8 shows the necessity of judging texture smoothness and adding intra frame interpolation. Fig.8 is a de-interlaced frame from sequence Football. It can be seen that there are block artifacts or noises in some areas by both MCMF and AR methods. However, subjective effect is improved greatly by applying LA to this region in our adaptive method.
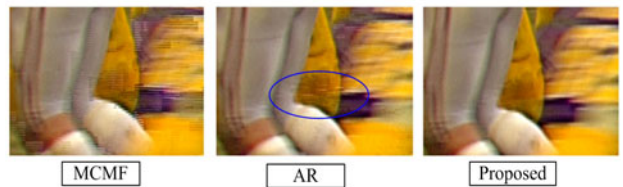

Fig.8 Subjective comparisons of Football

3.2. Objective Results

3.2.1. Adaptability of the Algorithm

Table 1 shows the results of our proposed adaptive selection mechanism. In the sequence Interview, which is basically still, very accurate ME can be performed. So the MVs are basically correct and MCMF is applied in the majority of the picture. The proportion of AR is high because of the smooth texture of the background. Basketball and Flowergarden mainly contain the global motion caused by panning of the camera and have some other types of motions in some local areas. For the global motion, ME can be well performed. So MCMF is mainly applied to these two sequences. AR and LA are applied in some local areas due to the inaccurate ME. The proportions of AR and LA are much higher in Horseriding and Football. Because there are many acute and complex movements in these sequences and the MEs are inaccurate. Especially in Football the moving objects are very blur because of the acute motion and the smooth texture. In this situation, LA can give the best results and indeed LA has a very large proportion. Therefore it can be concluded that the adaptive mechanism proposed possesses good adaptability from its reasonable performances on different sequences.

As for the computational complexity, the computational load of AR is very high. By greatly reducing the percentage of AR, our hybrid algorithm has lower computational load than AR.

Table1. Percentage of the three methods on several sequences from field 2 to field 12 (%)

Sequence	Format	MCMF	AR	LA
Interview		74.55	24.80	0.65
Basketball	720x576	83.34	14.54	2.12
Flowergarden		83.02	15.88	1.10
Horseriding		21.72	36.81	41.47
Football	704x480	18.94	38.88	42.18

3.2.2. PSNR Comparison

We get the interlaced sequences by dropping one field in each frame from progressive sequences. The average PSNR between the first 100 fields dropped and the corresponding de-interlaced fields are depicted in Table 2. The resolutions of CIF and HD are 352x288 and 1280x720, respectively. It can be seen that the results of our adaptive method are better than those of the other three methods on these sequences.

Table2. PSNR (dB) Comparison

Sequence		LA	MCMF	AR	Proposed
bus		25.19	25.84	27.29	28.36
football	CIF	31.86	24.06	30.25	31.64
news		31.15	39.19	39.24	40.49
mobile		22.47	26.77	24.70	26.68
Crew		35.39	36.47	37.06	37.85
Night	HD	28.97	32.27	33.12	34.64
City		28.92	29.92	30.08	30.63

The tested sequences in this section cover most scenes we often encounter. We make improvements not only on subjective quality but also objective quality. These prove that the adaptive switching mechanism is very effective. It makes every method utilize others's strong points to offset its weakness.

4. CONCLUSION

In this paper, a novel adaptive MC de-interlacing algorithm has been proposed. Based on analyzing the smoothness of texture and reliability of MV, we present an adaptive mechanism to interpolate the missing pixels by adaptively switching among the three methods. It adopts MCMF in the areas with reliable motion information and rich texture, LA in the areas with unreliable motion information and smooth texture, and AR in other areas. Subjective visual and objective experimental results show the validity of the adaptive mechanism and the feasibility of the algorithm for high quality de-interlacing applications.

5. ACKNOWLEDGEMENT

6. REFERENCES

[1] T. Fukinuki, "Television: Past, Present, and Future," Proceedings of the IEEE, vol. 86, no. 5, pp.998–1004, 1998.

[2] G. de Haan and E.B. Bellers, "Deinterlacing – An overview," Proceedings of the IEEE, vol. 86, no. 9, pp. 1839-1857,Sep. 1998.

[3] G. de Haan and E. B. Bellers, "De-interlacing of video data," IEEE Trans. on Cons. Elecn., vol. 43, pp. 819–825, Aug. 1997.

[4] Y. -Y. Jung, B. -T. Choi, Y. -J. Park, and S. -J. Ko, "An Effective De-interlacing Technique using Motion Compensated Interpolation," IEEE Trans. on Cons. Elecn., vol. 46, no. 3, pp. 460-465, Aug. 2000.

[5] Demin Wang, André Vincent, and Philip Blanchfield, "Hybrid De-Interlacing Algorithm Based on Motion Vector Reliability," IEEE Trans. on Circ. and Syst. for Video Techn., vol.15, no.8, pp.1019–1025, Aug. 2005.

[6] G. de Haan, P. W. A. C. Biezen, H. Huijgen, and O. A. Ojo, "True Motion Estimation with 3-D recursive search block-matching," IEEE Trans. on Circ. and Syst. for Video Techn.,vol3,no 5, pp. 368–388, 1993.

[7] G.A. Lunter, "Occlusion-insensitive motion estimation for segmentation," Proceedings of the SPIE: Visual Communications and Image Processing, San Jose, CA, pp. 573-584, January 2002.