ADVANCED MOTION SEARCH AND ADAPTATION TECHNIQUES FOR DEINTERLACING

Kefei Ouyang[1], *Guobin Shen*[2], *Shipeng Li*[2], *Ming Gu*[1]

[1]Software School, Tsinghua University, Beijing, P.R. China, 100080
[2]Microsoft Research Asia, Beijing, P.R. China, 100080

ABSTRACT

Unlike video coding, video deinterlacing relies heavily on the correctness of motion. To obtain more reliable motion, we propose a new motion search criterion that imposes constraints on the motion diversity among neighboring blocks, and improve the symmetric ME method by dynamic block splitting and single direction ME. To further improve the visual quality, adaptive deinterlacing algorithm based on block variances is also proposed. Extensive experimental results demonstrate that the proposed techniques significantly improved the deinterlaced video quality, both PSNR-wise and visually.

1. INTRODUCTION

The interlaced format has been used overwhelmingly in the TV industry and consumer camcorders. The merit of interlaced format lies in that the refreshing rate is double without increasing the bandwidth. However, because of the inherent nature of the interlaced scanning process, annoying visual artifacts like edge flickering, interline flickering and line crawling will appear when displaying an interlaced content on a progressive device such as the PC monitor. Deinterlacing, which is a picture format conversion from an interlaced video to a progressive one, has been widely used to reduce those artifacts.

Numerous deinterlacing techniques have been proposed in the literature. A good review of all mainstream deinterlacing algorithms is provided in [1]. These methods can be generally classified two categories, i.e., spatial-domain methods [2]-[3] where only one field is involved in forming a deinterlaced frame, and temporal-domain methods [4]-[8] where multiple fields are used. The fields in the interlaced format are sampled at different time instants and there are usually object moving between subsequent fields. As a result, temporal-domain methods, which utilize the motion information between neighboring fields, often offer a significantly improved deinterlacing quality at the cost of more computational complexity.

Temporal-domain methods have the potential to increase the vertical resolution over spatial-domain methods. Therefore, it is a general rule that we should always try to use the more temporal information via either motion aligned filtering (typically motion compensation and median filtering) or motion adaptive processing. However, the performance depends heavily on the reliability of the motion information, which is in general not guaranteed. As a result, on the one hand, people are seeking more reliable motion search methods, and on the other hand, motion-adaptive deinterlacing methods are widely adopted.

One interesting observation is that, most algorithms presented in the literature can yield higher PNSR and better visual quality for small resolution (such as CIF) video. However, for standard-definition (SD) sequences, there are many spurious artifacts such as combing artifacts when the depth of field is shallow, especially at the areas that are out of focus, i.e., low detail areas, such as background and the areas that have strong vertical edges.

This paper presents an efficient motion adaptive deinterlacing method that try to improve along the both directions. Specifically, we try to improve the reliability of the motion obtained by incorporating constraints on motion differences between neighboring blocks and the texture pattern of the block. Based on the more reliable motion obtained, moving areas are processed adaptively using either spatial-domain methods or motion aligned temporal-domain methods.

The paper is organized as follows: in the next section the more reliable motion estimation (ME) method is discussed. The concept of block variance is presented followed by the adaptive deinterlacing method based on block variance in Section 3. Experimental results are presented in Section 4 with comparisons (both PSNR-wise and visually) against other methods. Section 5 concludes the paper.

2. RELIABLE MOTION ESTIMATION (RME)

Because of the intrinsic phase-shift among signals between neighboring fields with different parity, it is often more accurate to perform ME between neighboring fields with the *same* parity. One of the most frequently used ME algorithms is the so-called symmetric motion estimation (SYMME) [4] which finds the motion trajectory passing through a block in the current field. The reference blocks,

0-7803-9332-5/05/$20.00 ©2005 IEEE

pointed by the motion vectors (MV), in the previous and the next fields are then used in the motion compensation. Since there are two reference fields for SYMME, more temporal information can be taken into account. The SYMME algorithm is also a time recursive method in the sense that the deinterlaced previous frame is used as the reference for the current field [5], which helps to stop error propagation. As a result, we also choose to use SYMME in this paper.

The underlining assumption of SYMME is that the motion is linear among neighboring fields. The reference blocks are obtained via a searching process and the metric is usually the Sum of Absolute Differences (SAD). The blocks with the smallest SAD is chosen to be the reference. Let $B(\vec{p},t)$ denote the given block in the current field t centering at position $\vec{p}$ ($\vec{p}=\{x,y\}$), then the motion vector of the block, $d(\vec{p},t)$ is obtained via:

$$d(\vec{p},t)=\arg\min_{c_i\in CS}(\varepsilon(c_i,\vec{p},t)) \tag{1}$$

where $c_i=\{c_{ix},c_{iy}\}$ is the displacement vector under testing, CS is the search window and $\varepsilon(c_i,\vec{p},t)$ is the SAD, defined as:

$$\varepsilon(c_i,\vec{p},t)=\left|B(\vec{p}+c_i,t+1)-B(\vec{p}-c_i,t-1)\right| \tag{2}$$

The resulting motion vector field using SAD criterion is plotted in Fig.2. Evidently, the motion vectors found are not very reliable in the sense that the blocks obviously belonging to the same object (say the hat in Fig. 2) have quite different motion vectors which contradicts human visual perception. To improve the reliability of resulting motion vectors, we impose a constraint such that blocks belong to the same object will have the similar motion vectors. The constraint is exerted via a modified SAD metric, as shown in Eq. 3 where λ is the weighting factor that determines strength of the constraint and c_p is the predicted motion vector from those of neighboring blocks.

$$\varepsilon(c_i,\vec{p},t)=\left|B(\vec{p}+c_i,t+1)-B(\vec{p}-c_i,t-1)\right|+\lambda\left|c_i-c_p\right| \tag{3}$$

Clearly, the more a motion vector deviates from its prediction, the heavy penalty is imposed. Similar ideas have been widely adopted in video coding. In video coding, the motion vectors are obtained in favor of high coding efficiency instead of true motion. As a result, the weighting factor is uniform throughout the whole picture since it corresponds to an optimal rate-distortion tradeoff. However, in deinterlacing applications, true motion is most desired. It was also observed [10] that the motion discontinuity usually correlates to image singularity (e.g., edges), therefore, the weighting factor should be adaptive to local statistics. Specifically, in this work, blocks with less texture or edges (i.e., prone to have wrong motion vectors) will be assigned a heavier weighting while blocks with more textures or edges (i.e., resilient to wrong motion vectors) will be assigned a lighter weighting factor. The resulting motion vector field with modified SAD criterion is shown in Fig.3.

Another improvement we made to original SYMME is dynamic block splitting and single direction ME. That is, we start to search MVs on a 16x16 macroblock basis. If the resulting SAD is too large, we continue to split the macroblock into four 8x8 blocks and continue to perform MV search for each 8x8 block. If the result is still not satisfactory, we will conduct ME using only one reference, either forward or backward. This effectively handles object occlusion and scene changes in the sequence. As a last resort, the block will be spatially processed since no good match can be found within the search window.

Fig.2. the vector field produced by full-search

Fig3 the vector field produced by our method

3. BLOCK VARIANCE ADAPTIVE DEINTERLACING (BVAD)

Traditional motion adaptive deinterlacing methods detect the motion areas first. Temporal interpolation is applied only in static areas and spatial interpolation for all motion areas to avoid possible artifacts. The block diagram of these methods is shown in Fig.4. However, it is an overkill to motion areas especially when we have a means to obtain reliable motion vectors.

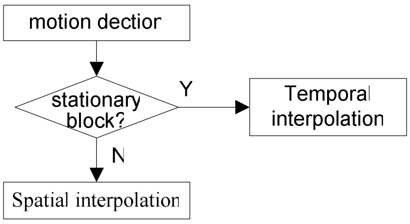

Fig.4. block diagram of motion adaptive deinterlacing

To improve image quality, motion aligned temporal filtering (via motion compensation and median filtering) is used in this paper so as to exploit more temporal information. Even with the motion estimation method presented in the previous section, true motion is not guaranteed due to, for example, the missing information in field pictures or the sampling noise when capturing. In our careful observation, flat areas (i.e., low detail regions) and areas with strong vertical edges are most likely to have artifacts when perform temporal filtering, as illustrated in Fig. 8 and Fig. 9. Therefore, these areas must be handled gracefully. In this paper, we propose a simple yet effective method based on block variance model to identify such error prone areas.

The so-called block variance model is to classify each image block into one of the following four classes, namely flat block, high texture block, strong vertical edge block, and strong horizontal edge block, according to their block level variances. Two block level variances are computed: vertical variance (V_v) and horizontal variance (V_h) of the block, which are calculated as the variance of mean values of all horizontal and vertical lines in a block, respectively. V_v and V_h are utilized jointly to classify each block according to the algorithm shown in Fig. 5.

IF (V_v < TH_f && V_h < TH_f)
  This block is a flat block;
ELSE IF (V_h/V_v > TH_v)
  This block is a strong vertical edge block;
ELSE IF (V_v/V_h > TH_h)
  This block is a strong horizontal edge block;
ELSE
  This block is a high texture block;

Fig.5. Variance model

We notice that for flat blocks and strong vertical edge blocks, temporal domain filtering does not provide much more texture information which it is extremely prone to artifacts resulting from wrong motion vector or subtle luminance intensity change. As a result, we propose to apply spatial interpolation for these blocks.

Combining this idea with the block variance model, a new motion adaptive method is presented as shown in Fig. 6. Compared with Fig. 4, the new adaptive model tries to incorporate as much as possible temporal information while avoiding possible artifacts of incorrect or improper use of temporal information. Specifically, instead of performing spatial interpolation for all motion blocks, we classify motion blocks into several categories and apply most appropriate processing accordingly. Note that the classification also leads to complexity reduction since some blocks are pruned away from complex motion estimation and motion aligned temporal filtering.

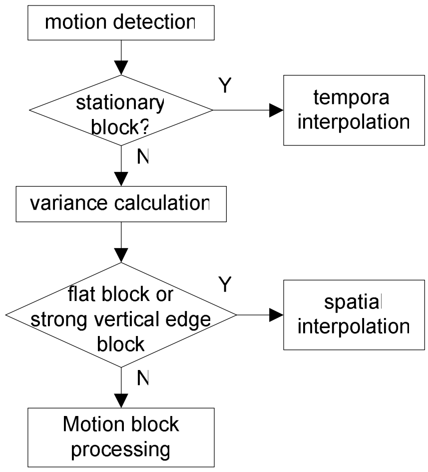

Fig.6 variance adaptive deinterlacing

4. EXPERIMENTAL RESULTS AND DISCUSSIONS

We have implemented an advanced deinterlacing system (ADS) that contains several known technologies reported in the literature to evaluate the performance of the proposed method. The technologies we adopted include edge-based line averaging [2], time recursive symmetric ME [4] [5], motion detection with co-located pixels [6] [7], and 5-tap median filtering [8]. Specifically, we performed motion detection on a macroblock block basis, by testing the sum of absolute differences between all the co-located pixels in the macro blocks from two composite input frames (i.e., four fields). The 5-tap median filtering is used to effectively stop the temporal interpolation error propagation.

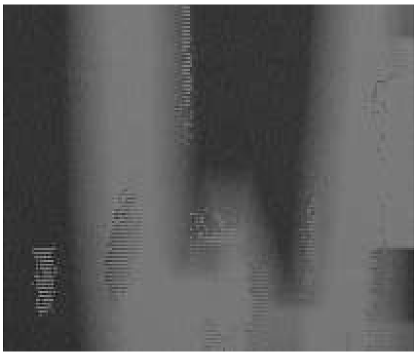

Fig.8 VirtualDub deinterlacing result

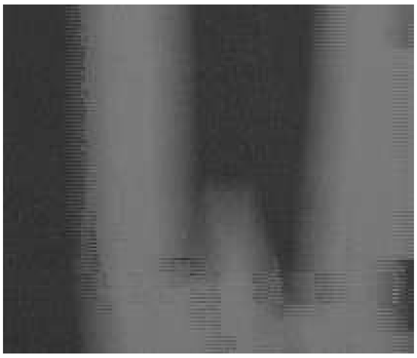

Fig.9 ADS deinterlacing result

Fig.10 ADS + BVAD deinterlacing result

We compare ADS and the deinterlacing filter in the free software Virtual Dub [9] using several standard test sequences. Table 1 shows the PSNR results of ADS and those from Virtual Dub. Since ADS contains many other techniques, we also present the PSNR contributions of the techniques proposed in this paper in the last two columns of the table. Our system significantly outperforms the Virtual Dub in both PSNR and visual quality. While the proposed reliable motion search method leads to PSNR improvement, the PSNR of the block variance adaptive deinterlacing method actually decreases slightly because of more spatial information is utilized for artifact prone areas. On the other hand, the visual quality, which is the ultimate metric of a deinterlacing system, is significantly improved with the block variance adaptive method. The visual quality comparison is shown in Fig. 8 through Fig. 10.

Table 1. PSNR comparison (unit is dB).

Sequence	Virtual Dub	ADS	ADS +RME	ADS+RME +BVAD
Foreman	32.48	34.76	34.92	34.42
Funfair	28.52	30.03	30.23	30.13
Mobile	24.71	30.88	31.21	30.78
Tab.-Tennis	32.63	34.56	35.57	35.41
Bus	27.81	29.49	29.74	29.69

5. CONCLUSION

In this paper, we proposed a new motion search criterion and improved the symmetric ME method. Much more reliable motion field was obtained. While trying to take into account more temporal information, the deinterlacing algorithm was made adaptive using block variances so that the errors due to temporal filtering can be suppressed and more pleasant visual quality result. Extensive experiments were performed. Both PSNR-wise and visual quality comparisons demonstrated the superiority of the proposed techniques.

6. REFERENCES

[1] G. De Haan, E.B. Bellers, "Deinterlacing – an Overview," in Proc. IEEE, Vol.86, No.9, pp.1839-1857, Sept.1998

[2] T. Doyle, "Interlaced to sequential conversion for EDTV applications," in Proc.2nd International Workshop Signal on Processing for HDTV, pp. 412-430, Feb.1998.

[3] M.K. Park, M.G Kang, K. Nam, S.G. Oh, "New Edge Dependent Deinterlacing Algorithm Based on Horizontal Edge Pattern," IEEE Trans. Consumer Electronics, Vol.49, No.4, Nov.2003.

[4] A. Tekalp, "Digital Video Processing, Prentice Hall," 1995.

[5] F.M. Wang, D. Anastassiou, and A.N. Netravali, "Time-recursive deinterlacing for IDTV and pyramid coding," Signal Processing: Image Commun., 2, pp. 365-374, 1990

[6] D. Han, C.Y. Shin, S.J Choi and J.S. Park, "A motion adaptive 3-D deinterlacing algorithm based on brightness profile pattern difference," IEEE Trans. Consumer Electronics, Vol. 45, No.3, pp.690-696, Aug.1999

[7] Tero Koivunen, "Motion detection of an interlaced video signal," IEEE Trans. Consumer Electronics, pp.753-759, 1994.

[8] Y.Y. Jung, B.T. Choi, Y.J. Park and S.J. Ko, "An effective deinterlacing technique using motion compensated interpolation," IEEE Trans. Consumer Electronics, 46(3),pp. 460-466, Aug. 2000.

[9] VirtualDub, http://www.virtualdub.org.

[10] Xin Li, "New directions in video coding," VCIP 2003, pp. 1428-1438.