Motion Adaptive Deinterlacing via Edge Pattern Recognition

Gwo Giun Lee, Hsin-Te Li,Ming-Jiun Wang, and He-Yuan Lin
MSoC Lab, Department of Electrical Engineering
National Cheng Kung University, Tainan, Taiwan

***Abstract*—In this paper, a novel edge pattern recognition (EPR) deinterlacing algorithm with successive 4-field enhanced motion detection is introduced. The EPR algorithm surpasses the performance of ELA-based and other conventional methods especially at textural scenes. In addition, the current 4-field enhanced motion detection scheme overcomes conventional motion missing artifacts by gaining good motion detection accuracies and suppression of "motion missing" detection errors efficiently. Furthermore, with the incorporation of our new successive 4-field enhanced motion detection, the interpolation technique of EPR algorithm is capable of flexible adaptation in achieving better performance on textural scenes in generic video sequences.**

I. Introduction

With recent advances of HDTV and multimedia personal computers, deinterlacing becomes an important technique which converts interlaced TV sequences into progressive frames for display on progressive devices such as LCD-TVs, Plasma Display Panels, Projective TVs, etc. However, visual defects such as edge flicker, line crawling, blur, and jaggedness introduced in the TV systems due to the inherent nature of interlaced sequences frequently exist and produce annoying artifacts to viewers if deinterlacing is not done properly.

Many techniques have been proposed for deinterlacing [1]-[7] in the past few decades. These techniques can be roughly categorized as either motion compensated (MC) or non motion compensated methods (Non-MC) [8]. Non-MC algorithms include spatial, temporal, and motion adaptive schemes. Spatial methods employ interpolation based on pixel values resided within the same field and are more cost-efficient among the deinterlacing techniques due to less memory requirements and hence less complexity. Temporal methods perform interpolation by exploiting the pixel values from several consecutive field sequences and they have high implementation cost due to multiple-field storage requirements. Spatial-temporal methods exploit the correlation in both time and spatial domain. Motion adaptive methods employ different filtering strategies for motion and non-motion cases by calculating the difference between two or more consecutive fields. Motion compensated methods involve motion estimation for filtering along the motion trajectories. However, motion compensated methods require much higher hardware complexity than other methods [9]. The visual quality of both motion adaptive and motion compensated methods highly relies on the correctness of motion information.

Among spatial methods, ELA algorithms [1][2] are widely used since they are capable of performing linear interpolation along the direction with smallest pixel difference and achieve relatively good performances with smaller computational load. However, ELA-based methods are sensitive to fine granularity features such as textures, noise and weak edges. They may also produce annoying artifacts when inaccurate edge information is used.

II. Motion Detection Methods

Conventional motion detection (MD) methods [10][11] for deinterlacing calculate the maximum of field luminance difference pairs at the same field parity (i.e. odd vs. odd, even vs. even), as shown in Fig. 1 If the maximum field difference exceeds a certain predefined threshold, the output of the deinterlacer is from the interpolator's moving path instead of being from the average of temporal neighbor pixels or the static path. However, these motion detection methods based on pixel differences are sensitive to noise and weak in detecting fine moving textures which leads to inaccurate motion judgment.

There are two types of motion detection errors, namely "motion missing" and "false detection". "Motion missing" refers to the of motion detector's incorrect recognition of moving pixels as stationary ones regardless of how high the PSNR is. The pixels from temporal neighboring fields will be applied to the output and this will produce severe artifacts when small objects moving in high velocity and when small objects located in a background of uniform intensity moving with high velocity (e.g. moving texts on an uniform background). "False detection" happens when the motion detection regards the stationary pixels as moving ones and the output of the interpolator at moving path will be applied. This however will not produce severe artifacts should there be a robust interpolator. Hence, it is essential that the "motion missing" detection error be reduced as much as possible.

Enlarging the detection window is one possible way to solve the motion missing problem. However, a larger detection window places stronger constraints on the detection of motion status for a

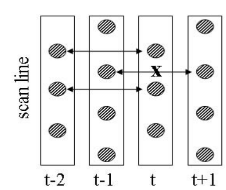


Figure 1. Conventional 4-field motion detection

This research was supported by the National Science Council of Taiwan, R.O.C., under Grant NSC 93-2215-E-006-024.

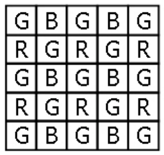


Figure 2. Bayer CFA Pattern

single pixel resulting in more "false detection" errors. This method could potentially solve the motion missing problem but the drawback is on the loss of details on small objects and textures. Reducing the threshold value is an alternative way to alleviate the motion missing problem but motion detection methods based on difference calculations are sensitive to noise.

III. PROPOSED METHOD

In order to achieve better performance when interpolating textural video sequences, we incorporate the concept of extended Color Filter Array (CFA). The commonly used Bayer Pattern [12] is shown in Fig. 2 for interpolating missing green pixels where R, G, B denote red, green and blue pixels respectively.

A. Edge Pattern Recognition

Edge pattern recognition method [13][14] is used for interpolating the missing green pixels in CFA. Due to the high complexity in many metrics in describing textural features, we tacitly introduce the concept of delta modulation as in communications systems for texture analysis. The four pixels, a, b, c, and d are analyzed and marked as "H" or "L" depending on their gray levels as shown in Fig. 3. Four unique types of edge patterns within a 3x3 window are depicted, namely the edge patterns labeled as edge pattern I with 3H1L, edge pattern II with 3L1H, the stripe pattern, and the corner pattern. Hence, we can obtain fourteen different edge patterns in four unique pattern types. In order to interpolate the center pixel X, all neighboring pixels around pixel X within the 3x3 window are required. Two functions for interpolation on stripe pattern and corner pattern are defined here:

$$add\_L = max(L,L)$$
$$add\_H = min(H,H) \qquad (1)$$

Fig. 3 also illustrates the interpolation scheme of four unique pattern types. Each pattern in the same pattern type uses similar interpolation scheme.

B. 4-field Enhanced Motion Detection

1) First Step (Coarse search)

A cross-shaped detection window is used for 4-field motion detection for coarse search, as shown in Fig. 4(a) Strong constraints aren't given in this step. Calculate the second largest value of the five luminance field difference pairs at the corresponding position in terms of temporal and spatial fields in Fig. 4(b).

2) Second Step Refinement

Since the false detection errors exist in the classified group of static pixels after first step coarse search, a further refining step is performed here. We calculate the summations in Fig. 5 for different combinations in the refining step for further threshold checking. If any summation of each combination is smaller than the predefined threshold TS, pixel X is detected as static.

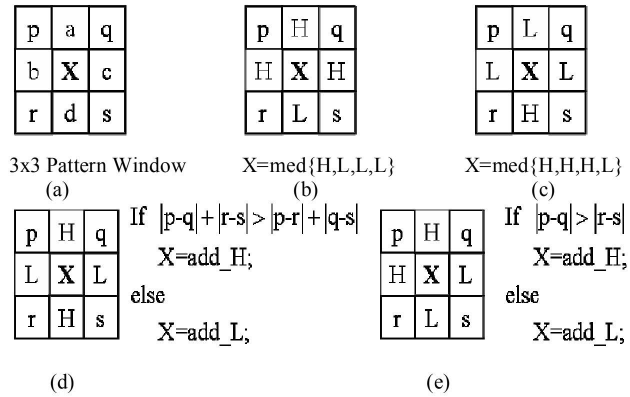


Figure 3. Interpolation scheme of each unique pattern type (a) original 3x3 pattern window (b) edge pattern I (c) edge pattern II (d)stripe pattern (e) corner pattern

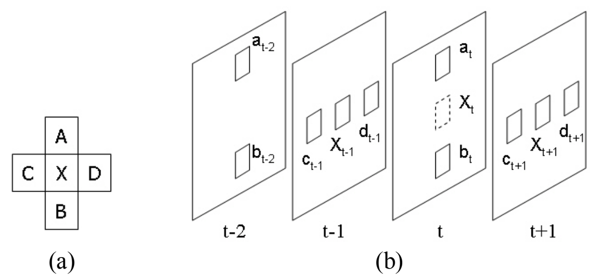


Figure 4. 4-field enhanced motion detection (a) cross-shaped window (b) corresponding position of each pixel

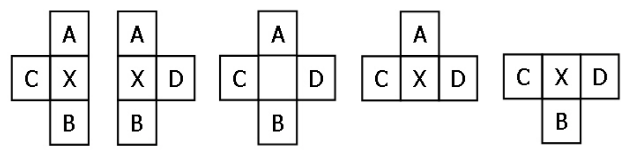


Figure 5. Different combinations of difference pairs for the refining step

C. Successive Motion Detection Assisted Edge Pattern Recognition Deinterlacing

When incorporating the edge patterns techniques to deinterlacing, the absence of missing lines should be considered (i.e. pixel b and c in Fig. 3(a)) in the interlaced video sequences. A motion adaptive interpolation scheme [15], as shown in Fig. 6, using 4-field enhanced motion detection is applied to our EPR algorithm. Fig. 7 illustrates the position of the pattern window and the motion detection window of successive motion detection method. Different interpolation schemes are used for pixels b and c.

$$b_t = med(\hat{X}_{t-1}, v\_avg_t, t\_avg_t) \qquad (2)$$
$$c_t = (1-\alpha)\times t\_avg_{t+1} + \alpha\times v\_avg_{t+1}$$

where $\hat{X}_{t-1}$ denotes the previous result of EPR at time t-1. v_avg_t and t_avg_t denotes the vertical and temporal average at time t respectively. Motion detection for C_t is performed before edge pattern recognition at X_t. The motion flag of C_t is exactly that of X_{t+1}. Hence, the motion flag of C_t can be used successively for adaptive interpolation.

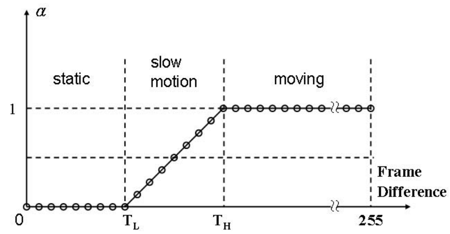


Figure 6. Adaptive Interpolation Scheme

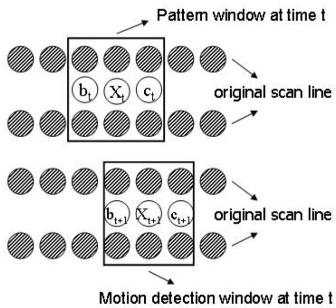


Figure 7. Position of pattern window and motion detection window at time t

In Fig. 4(b), the interpolation for static path output is similar to conventional ELA. The minimum of three temporal edges through pixel X at time t is calculated. Hence, the static path output is interpolated by the average of the temporal edge with smallest difference to achieve better performance.

$$F[x,y,n]=\begin{cases} F_i[x,y,n] & \text{, y mod 2= n mod 2} \\ (1-\alpha)\times F_{st}[x,y,n]+\alpha\times F_{mov}[x,y,n] & \text{,otherwise} \end{cases} \quad (3)$$

$F_{st}[x,y,n]$ denotes the output from static path

$F_{mov}[x,y,n]$ denotes the output from moving path

IV. Simulation Results

A. Texture Analysis without Motion Detection

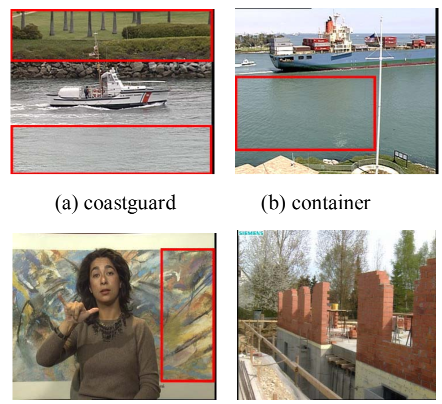
To evaluate the performance of our extended EPR algorithm on textures, some textural scenes are extracted such as waves (coastguard and container), grass (coastguard), painting (silent), trees and bricks (foreman) are extract from the corresponding video sequences. As shown in Fig. 8, these textures are static or mixed with little motion throughout the specific period of the sequence (e.g. frame 1 to 240 in "Container"). The average PSNR comparison on textures without motion detection is shown in Table I. EPR I and EPR II methods are listed to show the flexibility of EPR algorithm and pixel b is simply from the previous result and pixel c is from previous temporal data and from vertical average respectively. The subjective view of interpolation error in comparison with original progressive Foreman sequence is shown in Fig. 9. The higher luminance value indicates larger interpolation error.

B. Motion Detection Analysis and PSNR Comparison

The subjective view comparison between conventional 4-field and proposed motion detection are presented in Fig. 10. There are severe artifacts known as "motion holes" caused by motion missing in Fig. 10(a) on the fast-moving fingers when using conventional 3-field motion detection. In Fig. 10(b) the proposed enhanced 4-field motion detection method can suppress the motion missing errors efficiently and retain detection accuracy in comparison with other methods. Table II illustrates the average PSNR comparison between different deinterlacing algorithms. The proposed method shows better performance on PSNR than other methods.

TABLE I. Average PSNR (dB) Comparison on Different Textures without Motion Detection

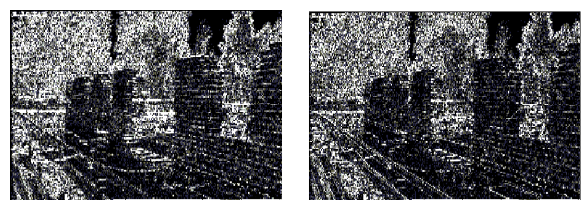
Name	Merge	Bilinear	ELA	EELA	EPR I	EPR II
grass (Coastguard)	26.83	32.04	30.67	30.64	31.51	30.90
waves (Coastguard)	33.37	33.50	32.91	32.85	34.49	33.15
painting (Silent)	48.83	34.00	33.40	33.86	34.58	33.79
waves (Container)	46.82	38.68	37.56	37.88	40.35	38.26
bricks & trees (Foreman)	34.85	29.27	28.44	28.70	30.96	29.84



(a) coastguard (b) container

(c) silent (d) foreman

Figure 8. Textures on different sequences



(a) EELA (b) EPR

Figure 9. Subjective view of interpolation error on Foreman sequence (a) EELA [5] (b) EPR method

V. Conclusion

A motion adaptive edge pattern recognition algorithm is presented in this paper. The EPR algorithm is shown to have better performance on textures as compared to conventional algorithms such as ELA. The enhanced cross-shaped 4-field motion method can suppress the motion missing problem caused by conventional motion detection methods and increase the motion detection accuracy. It can also adaptively change the interpolation scheme of EPR method. From the experiment results, the proposed technique reveals its advantages in having more accurate motion detection and better texture interpolation for deinterlacing.

Figure 10. Subjective view of "motion missing" detection error

TABLE II. AVERAGE PSNR COMPARISON

Name	Bilinear	ELA	EELA	EPR	4-field EPR	Proposed
Silent	33.77	33.09	33.37	34.58	37.99	41.69
Coastguard	28.65	27.89	27.82	29.40	31.23	30.62
Foreman	32.32	32.46	32.64	32.08	32.75	34.00
Akiyo	39.69	37.83	38.42	37.86	41.43	42.94
News	34.02	31.75	31.83	32.09	36.98	40.07
Stefan	27.40	26.12	25.85	25.26	27.17	28.36
M&D	39.17	38.25	38.37	39.53	42.42	44.26
Hall Monitor	31.73	30.53	30.64	32.69	35.55	36.18
Mobile	26.89	25.10	25.38	24.25	24.28	26.22
Container	28.74	27.77	27.81	30.13	31.08	35.14

REFERENCES

[1] T. Doyle and M. Looymans, "Progressive scan conversion using edge information," in Signal Processing of HDTV II, pp. 711-721. 1990.

[2] Chung J. Kuo, Ching Liao, and Ching C. Lin, "Adaptive interpolation technique for scanning rate conversion," IEEE Trans. Circuits Syst. Video Technol., Vol. 6, No. 3, 1996.

[3] Shyh-Feng Lin, Yu-Lin Chang, and Liang Gee Chen, "Mo-tion adaptive de-interlacing by horizontal motion detection and enhanced ELA processing," IEEE International Sympo-sium on Circuits and Systems, 2003.

[4] D. Han, C. Y. Shin, S. J. Choi, and J. S. Park, "A motion adaptive 3-D de-interlacing algorithm based on the bright-ness profile pattern difference", IEEE Transactions on Con-sumer Electronics, Vol. 45, Issue: 3, pp. 690-697, 1999.

[5] R. Li, B. Zeng, and M. L. Liou, "Reliable Motion Detection/Compensation for Interlaced Sequences and Its Application to Deinterlacing," IEEE Trans. Circuits Syst. Video Technol., Vol. 10, No. 1, Feb 2000.

[6] S. Yang, Y. Y. Jung, Y. H. Lee, and R. H. Park, "Motion Compensation Assisted Motion Adaptive Interlaced-to-Progressive Conversion", IEEE Trans. Circuits Syst. Video Technol., Vol. 14, Issue: 9, pages: 1138-1148, 2004.

[7] G. G. Lee, Drew Wei-Chi Su, He-Yuan Lin and Ming-Jiun Wang, "Multiresolution-based Texture Adaptive Motion Detection for De-interlacing," IEEE International Symposium on Circuits and Systems (ISCAS2006).

[8] G. de Haan and E. B. Bellers, "Deinterlacing – an overview," Proceedings of the IEEE, Vol. 86, Issue 9, pp.1839-1857, 1998.

[9] G. G. Lee, Kees A. Vissers, and Bin-Da Liu, "On A 3D Recursive Motion Estimation Algorithm and Architecture for Digital Video," IEEE Midwest Symposium on Circuits and Systems (MWCAS), Japan, 2004.

[10] T. Koivunen, "Motion Detection of an Interlaced Video Signal," IEEE Transactions on Consumer Electronics, Volume: 40 Issue: 3, pp.753-760, 1994.

[11] B.Bhatt, F. Templin, B. Hogstrom, H. Derovanessian, S. Lamadrid, and J. Mailhot, "Grand-alliance HDTV multi-format scan converter," IEEE Transactions on Consumer Electronics, vol. 41, pp. 1020-1031, Nov. 1995.

[12] B. E. Bayer, "Color imaging array," U.S Patent 3,971,065, 1976.

[13] J. E. James Jr., "Interactions between color plane inter-polation and other image processing functions in electronic photography", Proceedings of SPIE, Vol. 2416 pp.144-151, 1995.

[14] Cok, David R., "Signal Processing method and apparatus for sampled image signals," U.S Patent 4,630,307, 1986.

[15] A.M. Bock, "Motion-adaptive standards conversion between formats of similar field rates," Signal Processing : Image Commun., vol. 6, no. 3,pp. 275-280, June 1994.