Optical Engineering

SPIEDigitalLibrary.org/oe

Filter design and its application for scanning format conversion

Gwanggil Jeon
Jong-Kyu Lee

Filter design and its application for scanning format conversion

Gwanggil Jeon^a and Jong-Kyu Lee^b

^aUniversity of Incheon, Department of Embedded Systems Engineering, Incheon, Republic of Korea

^bHanyang University, Department of Computer Science and Engineering, Ansan, Republic of Korea
E-mail: 1900101@hanyang.ac.kr

Abstract. This letter presents the issue of interpolating missing line components at the interlaced signals, a procedure called deinterlacing. A filter design method is introduced in detail, utilizing a least-squares algorithm with training set for the requested deinterlacing filter. A thorough research of deinterlacing performance and system complexity is executed. The proposed filter design and its application on deinterlacing is compared with other conventional filter-based deinterlacing methods in terms of quality measures. It was concluded to show outstanding performance and speed trade-off among the methods compared. © 2012 Society of Photo-Optical Instrumentation Engineers (SPIE). [DOI: 10.1117/1.OE.51.4.040504]

Subject terms: intra-field deinterlacing; least-squares method; filter design.

Paper 111467L received Nov. 23, 2011; revised manuscript received Feb. 8, 2012; accepted for publication Feb. 14, 2012; published online Apr. 11, 2012.

1 Introduction

Interlaced video frame consists of two sub fields taken in sequence, each sequentially scanned at odd and even lines of the image sensor. Analog television employed this technique because it allowed for less transmission bandwidth and further eliminated the perceived flicker that a similar frame rate would give using progressive scan. The cathode-ray tube (CRT), based displays were able to display interlaced video correctly due to its complete analog nature. All of the newer displays are inherently digital, in that the display comprises discrete pixels. Therefore, the two fields need to be combined into a single frame, which leads to various visual defects, which the deinterlacing process should try to minimize. Deinterlacing has been researched for decades and employs complex processing algorithms. However, consistent results have been very hard to achieve.¹ There are various deinterlacing methods with different levels of quality and complexities, which are: (1) basic fixed methods such as the line average and field average; (2) adaptive methods that select one of the basic methods according to the motion information extracted from the image sequence; and (3) methods that utilize the motion estimation and compensation. In this letter, we focus on intra-field deinterlacing method, which belongs to category (1) and gives good performance with low complexity.

Most edge-based deinterlacing methods utilize the uniform weighted sum of two-tap filter [TTF: $h_{\text{TTF}} = 1/2$ (Ref. 2)] to estimate the current pixel to be interpolated in

the determined edge direction.³ The line average (LA) method⁴ is commonly used in the software approach, which adopts a single field to reconstruct a progressive frame in 90 deg edge direction. However, the frequency response of two-tap average filter looks like a bell shape as shown in Fig. 1(a). It is obviously known that LA shows high peak signal-to-noise ratio (PSNR) performance. However, it provides apparent jaggedness at edge area.

Meanwhile, the sinc function is known as an ideal filter, and its frequency response is rectangular-shape. By the sinc function, we can design the filter to have a more rectangular shaped frequency response than the TTF, so interlaced signals can be rebuilt more precisely. In Refs. 3 and 4, the authors adopted one of the sinc interpolation filters, which is a one-dimensional six-tap filter ($\text{STF}: h_{\text{STF}} = 1/128[3 \ -17 \ 78 \ 78 \ -17 \ 3]$), and coefficients of STF are determined by approximating the sinc function. This is same to the one in the video codec, such as MPEG-4, H.264/AVC, and high efficiency video coding (HEVC), to decrease residual errors.⁵ However, this method only regards the similarity of sinc function, and consciousness of the closeness or spatial locality is not taken into account.

In this letter, we proposed filter design algorithm, which minimizes the mean-squared deinterlacing error over a training set (TS). In Sec. 2, we propose filter design algorithm in detail. The simulation results are shown in Sec. 3 and the conclusions are made in Sec. 4.

2 Proposed Filter Design

Least mean squares method is a class of adaptive filter utilized to simulate a desired filter by obtaining the filter coefficients that relate to presenting the least mean squares of the error signal, e.g., difference between the desired and the real signal. A general approach is introduced in Refs. 6–8. Let us assume $x_O[i, j]$ is original image, that we reconstruct by filtering interlaced image $x_I[i, j]$ with the linear, shift-invariant filter having unit-sample response $h_{\text{TS}}: \hat{x}_O = x_I * h_{\text{TS}}$, where h_{TS} is desired filter obtained by TS. If we suppose the remainder between $\hat{x}_O$ and x_O can be designed as a stationary random field, then we may select $h_{\text{TS}} = \arg \min_h E[(x_O[i, j] - (x_I * h)[i, j])^2]$. Because a good random field model is hard to be modeled, we can minimize the substantial error on TS of representative images. Let us suppose that we have TS of original images. For every image in this TS, we can create (width $\times$ height)/2 sized interlaced image x_I from the width $\times$ height sized desired image x_O . Assume we segment the TS into S sub-images, where k 'th sub-image is specified on the partial block $P^{(k)}$. Assume that the desired filter h_{TS} is a two-dimensional (2-D) finite impulse response filter with area of support $Q \subset \Lambda$, where rectangular lattice $\Lambda = \mathbb{Z}^2$. Then the desired filter can be acquired as solution to

$$h_{\text{TS}} = \arg \min_h \sum_{\kappa=1}^S \sum_{(i,j) \in P^{(\kappa)}} \left\{ x_O^{(\kappa)}[i, j] - \sum_{(m,n) \in Q} h[m, n] x_I^{(\kappa)}[i - m, j - n] \right\}^2. \quad (1)$$

We assume $Q_N = \text{card}(Q)$ and $P_N = \text{card}[P^{(k)}]$ be the number of filter coefficients to be decided and the number of samples in the sub-images. We create an $Q_N \times 1$ column vector h from the filter coefficients by examining the area Q in some fixed command. Correspondingly, we create

Fig. 1 Perspective view of frequency responses of filters for deinterlacing. (a) h_{TTF} ; (b) h_{STF} ; (c) $h_{\text{LSF}}^{\text{S}}$ obtained by image#1; (d) $h_{\text{LSF}}^{\text{E}}$ obtained by image #2 to 24 for deinterlacing image #1; (e) $h_{\text{LSF}}^{\text{A}}$ obtained by image #1 to 24; and (f) $h_{\text{LSF}}^{\text{S}}$ obtained by image #2.

an $P_N \times 1$ column vector $x_O^{(\kappa)}$ from $x_O^{(\kappa)}[i, j]$ by examining the pixels of $P^{(\kappa)}$ in a fixed manner. Eventually, we create an $P_N \times Q_N$ matrix $M^{(\kappa)}$ from the components of $x_O^{(\kappa)}$ as each column of $M^{(\kappa)}$ matches an element $(m, n) \in Q$ examined in the same way as used to create h ; this column is obtained by scanning the elements of $x_O^{(\kappa)}[i - m, j - n]$ for $(i, j) \in P^{(\kappa)}$ in the identical manner exploited to create $x_O^{(\kappa)}$. Finally, Eq. (1) can be described in matrix form as

$$h_{\text{TS}} = \arg \min_h \sum_{\kappa=1}^S \|M^{(\kappa)}h - x_O^{(\kappa)}\|^2. \quad (2)$$

Equation (2) is a standard least-squares (LS) issue with solution⁹

$$h_{\text{TS}} = \left\{ \sum_{\kappa=1}^S \left[M^{(\kappa)} \right]^T M^{(\kappa)} \right\}^{-1} \left\{ \sum_{\kappa=1}^S \left[M^{(\kappa)} \right]^T x_O^{(\kappa)} \right\}. \quad (3)$$

Figure 1 shows the perspective view of frequency response of the (a) h_{TTF} ; (b) h_{STF} ; (c) $h_{\text{LSF}}^{\text{S}}$ (LS filter obtained by image #1); (d) $h_{\text{LSF}}^{\text{E}}$ (LS filter obtained by other 23 images to deinterlace a image); (e) $h_{\text{LSF}}^{\text{A}}$ (LS filter obtained by whole 24 images); and (f) $h_{\text{LSF}}^{\text{S}}$ (LS filter obtained by image #2). In this letter, 11×11 filters were designed.

3 Experimental Results

The proposed method under different filter condition was tested on the 24 Kodak images.¹⁰ We denote M_{TTF}^3 , M_{STF}^4 , $M_{\text{LSF}}^{\text{S}}$, $M_{\text{LSF}}^{\text{E}}$, and $M_{\text{LSF}}^{\text{A}}$ as the method that employed h_{TTF} , h_{STF} , $h_{\text{LSF}}^{\text{S}}$, $h_{\text{LSF}}^{\text{E}}$, and $h_{\text{LSF}}^{\text{A}}$, respectively.

Table 1 shows the PSNR results of 24 Kodak images. The $M_{\text{LSF}}^{\text{S}}$ gave the highest PSNR (however, this is infeasible method), and $M_{\text{LSF}}^{\text{E}}$ and $M_{\text{LSF}}^{\text{A}}$ gave the same result. This implies that the proposed method is not sensitive to TS. In Fig. 2, reconstructed results of the image #5 are shown. It can be noticed that the proposed method with

Table 1 PSNR (DB) results of different filter-based methods for various kodak images.

Kodak images	M_{TTF}^3	M_{STF}^4	$M_{\text{LSF}}^{\text{S}}$	$M_{\text{LSF}}^{\text{E}}$	$M_{\text{LSF}}^{\text{A}}$
1	24.00	21.65	27.03	26.92	26.94
2	31.77	29.38	34.84	34.74	34.74
3	31.46	29.05	35.93	34.78	34.80
4	31.79	28.93	35.98	35.72	35.71
5	23.93	21.26	28.31	28.09	28.10
6	24.87	22.71	27.79	27.54	27.55
7	29.60	26.30	35.34	35.02	35.02
8	23.22	21.01	26.63	26.25	26.26
9	29.42	26.62	34.07	33.90	33.90
10	30.21	27.83	35.11	33.77	33.77
11	27.05	24.74	30.05	30.03	30.03
12	31.30	28.82	34.50	34.43	34.43
13	21.75	19.74	24.47	24.41	24.41
14	26.46	23.88	30.03	29.98	29.98
15	31.73	29.05	35.52	35.41	35.40
16	28.36	26.23	30.96	30.72	30.72
17	30.24	27.55	34.08	33.94	33.94
18	26.09	23.77	29.48	29.41	29.41
19	25.61	23.12	29.14	29.10	29.10
20	29.45	26.79	33.48	33.30	33.30
21	25.62	23.23	28.96	28.89	28.89
22	29.27	26.83	32.52	32.46	32.46
23	32.00	29.13	36.94	36.52	36.52
24	25.08	23.00	28.01	27.91	27.91
Avg.	27.93	25.44	31.63	31.39	31.39

Fig. 2 (a) Original Kodak image;¹⁰ Deinterlaced image #5 using: (b) h_{TTF} ; (c) h_{STF} ; (d) h_{LSF}^S obtained by image #5; (e) h_{LSF}^E obtained by image #1 to 4 and #6 to 24 for deinterlacing image #5; (f) h_{LSF}^A obtained by image #1 to 24; and (g) h_{LSF}^S obtained by image #6.

Table 2 PSNR (DB) results of conventional deinterlacing methods for various kodak images.

image	LCID ¹¹	FDD ²	EMD ¹²	FEPD ¹³	MCAD ¹⁴	CASA ¹⁵	DCAD ¹⁶	M_{LSF}^E
1	26.81	26.89	26.71	26.66	26.60	26.82	26.76	26.92
2	34.75	34.68	34.36	34.27	34.53	34.76	34.76	34.74
3	34.65	34.45	34.43	34.58	35.12	34.63	34.60	34.78
4	35.65	35.66	35.26	35.07	35.30	35.72	35.83	35.72
5	28.27	27.97	27.71	27.44	27.69	28.27	28.27	28.09
6	27.55	27.51	27.44	27.50	27.45	27.53	27.48	27.54
7	35.13	34.60	34.45	34.20	35.01	35.16	35.30	35.02
8	26.12	26.24	25.74	25.08	26.07	26.18	26.20	26.25
9	33.77	33.64	33.31	33.11	33.66	33.81	33.93	33.90
10	33.38	33.43	33.16	33.14	34.27	33.33	33.03	33.77
11	30.07	29.99	29.79	29.73	30.00	30.06	30.05	30.03
12	34.45	34.40	34.18	34.02	34.28	34.47	34.44	34.43
13	24.34	24.26	24.19	24.36	24.07	24.30	24.19	24.41
14	30.09	29.88	29.74	29.71	29.69	30.09	30.12	29.98
15	35.31	35.29	34.89	34.70	34.94	35.37	35.47	35.41
16	30.82	30.75	30.72	30.84	30.75	30.79	30.67	30.72
17	34.01	33.77	33.48	33.36	33.88	34.06	34.19	33.94
18	29.23	29.21	29.00	29.07	28.91	29.22	29.12	29.41
19	29.08	29.07	28.82	28.55	28.97	29.09	28.46	29.10
20	33.41	33.13	32.86	32.65	32.20	33.48	33.61	33.30
21	28.86	28.80	28.67	28.73	28.63	28.85	28.85	28.89
22	32.33	32.34	31.99	31.74	32.20	32.37	32.43	32.46
23	36.71	36.30	36.07	35.82	37.18	36.78	37.19	36.52
24	27.60	27.58	27.43	27.47	27.25	27.59	27.49	27.91
avg.	31.35	31.24	31.02	30.91	31.19	31.36	31.35	31.39

various filters present less artifacts with respect to the other filter based methods.

Table 2 shows PSNR performance of the proposed M_{LSF}^E and other conventional deinterlacing algorithm systems from the literature, they are LCID,¹¹ FDD,² EMD,¹² FEPD,¹³ MCAD,¹⁴ CASA,¹⁵ and DCAD.¹⁶ At first observation, it is noticed that our proposed method outperforms all other methods over the entire images in terms of PSNR.

4 Conclusion

The proposed filter based deinterlacing has shown the lowest reconstruction error among published filter-based techniques on the reference Kodak dataset. This letter has demonstrated that the feasible h_{LSF}^E filter is sufficient to achieve the full benefits of the technique, improving over the M_{TTF} method by 3.46 dB. Moreover, the proposed h_{LSF}^E filter is superior to CASA by 0.21 dB, which is the best edge-based method in literature.

Acknowledgments

This work was supported in part by the National Science Foundation of China (NSFC) under Grant 61001100, 61077009, and the Provincial Science Foundation of Shaanxi, China, under Grant 2010K06-15.

References

1. K. Jack, "Video demystified—a handbook for the digital engineer," Elsevier, Eagle Rock, VA (2005).
2. S. Jin, W. Kim, and J. Jeong, "Fine directional de-interlacing algorithm using modified sobel operation," *IEEE Trans. Cons. Elect.* **54**(2), 857–862 (May. 2008).
3. E. B. Bellars and G. D. Haan, "Deinterlacing: a key technology for scan rate conversion," Elsevier, Amsterdam (2000).
4. S. M. Hong, S.-J. Park, and J. Jeong, "Deinterlacing algorithm using fixed directional interpolation filter and adaptive distance weighting scheme," *Opt. Eng.* **50**(6), 067008 (June 2011).
5. A. Fuldseth et al., "Low complexity directional interpolation filter," ITU-T SG16/Q.6 Doc. VCEG-A112, pp. 16–18 (2008).
6. T. Kondo et al., "Picture conversion apparatus picture conversion method learning apparatus and learning method," U.S. Patent 6 323 905 (Nov. 2001).
7. X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process.* **10**(10), 1521–1527 (Oct. 2001).
8. M. Zhao, "Video enhancement using content-adaptive least mean square filters," PhD Thesis, Technische Universiteit Eindhoven (2006).
9. S. Haykin, "Adaptive filter theory," Prentice Hall, (2002).
10. [Online]. Available:
11. P. Y. Chen and Y. H. Lai, "A low-complexity interpolation method for deinterlacing," *IEICE Trans. Inf. Syst.* **E90-D**(2), 606–608 (2007).
12. K. Kang, G. Jeon, and J. Jeong, "A single field interlaced to progressive format conversion using edge map in the image block," in *IASTED Signal and Image Processing (SIP) 2009*, pp. 80–85 (2009).
13. S. Yang, D. Kim, and J. Jeong, "Fine edge-preserving deinterlacing algorithm for progressive display," *IEEE Trans. Cons. Elect.* **55**(3), 1654–1662 (Aug. 2009).
14. S. J. Park, G. Jeon, and J. Jeong, "Covariance-based adaptive deinterlacing method using edge map," in *International Conference on Image Processing Theory Tools and Applications (IPTA) 2010*, pp. 166–171 (2010).
15. S. J. Park, G. Jeon, and J. Jeong, "Computation-aware algorithm selection approach for interlaced-to-progressive conversion," *Opt. Eng.* **49**(5), 057005 (2010).
16. S. J. Park, S. M. Hong, and J. Jeong, "Deinterlacing algorithm using direction-oriented inverse-free wiener filtering," *Opt. Eng.* **50**(6), 067014 (2011).