

# A Novel Deinterlacing Algorithm Based on Adaptive Polynomial Interpolation

Seung-Won Jung, Hye-Soo Kim, Le Thanh Ha, Seung-Jin Baek, and Sung-Jea Ko

**Abstract**—In this paper, a novel deinterlacing algorithm is proposed. The proposed algorithm approximates the distribution of the luminance into a polynomial function. Instead of using one polynomial function for all pixels, different polynomial functions are used for the uniform, texture, and directional edge regions. The function coefficients for each region are computed by matrix multiplications. Experimental results demonstrate that the proposed method performs better than the conventional algorithms.

**Keywords**—Deinterlacing, polynomial interpolation.

## I. INTRODUCTION

**D**EINTERLACING is a technique that converts an interlaced frame into a progressive frame. Even though high definition television (HDTV) adopts a progressive scanning, an interlaced scanning is still necessary in the viewpoint of compatibility with the existing TV and camera systems. In order to reduce the artifacts caused by the inherent nature of the interlaced image, a number of techniques have been presented in the literature [1-4].

Among many techniques, the edge-based line average (ELA) [1-2] algorithm is most widely used due to its reliable performance and small computational complexity. ELA interpolates an interlaced line by selecting the direction of the highest correlation. However, this simple edge-based algorithm often produces artifacts when edge direction is not correctly estimated. In order to avoid this failure, the edge patterns are considered. In [3], various horizontal edge patterns are considered and the interpolation is performed according to the estimated pattern. This algorithm interpolates edge regions more reliably, but it still suffers from the misclassification of the edge pattern. In [4], an edge preserving deinterlacing is performed by using a block-based region classification. Though this technique preserves a strong edge of the image, a texture region of the image is not effectively interpolated by the block based filtering method.

In this paper, a novel deinterlacing algorithm that interpolates the interlace line by using a polynomial approximation is proposed. Unlike applying an interpolation filter with the fixed coefficients for every pixel or edge pattern in the conventional algorithm [1-4], the proposed technique adaptively computes the filter coefficients. Since the

The authors are with the electrical engineering department at Korea University. (corresponding author: Seung-Won Jung, e-mail: jungsw@dali.korea.ac.kr).

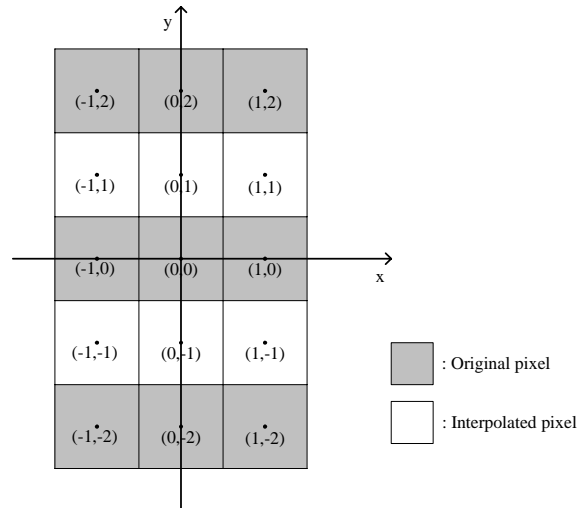


Fig. 1  $5 \times 3$  window for the proposed algorithm

coefficients are obtained by simple matrix multiplications, the computational burden is not severe. The experimental results demonstrate that the proposed technique efficiently interpolates texture and edge regions of the image.

## II. PROPOSED DEINTERLACING ALGORITHM

Fig. 1 shows a  $5 \times 3$  window for the proposed algorithm. Since the luminance of an image inherently has a smooth-varying characteristic, a luminance distribution of this window is modeled as the following function,

$$f(x, y) = c_1x^2y^2 + c_2x^2y + c_3xy^2 + c_4xy + c_5x^2 + c_6x + c_7y^2 + c_8y + c_9. \quad (1)$$

Similarly to [5], six unknown pixel values in the window can be estimated. From the nine known values, (1) can be represented as a matrix form,

$$\begin{bmatrix} f(-1,2) \\ f(0,2) \\ f(1,2) \\ f(-1,0) \\ f(0,0) \\ f(1,0) \\ f(-1,-2) \\ f(0,-2) \\ f(1,-2) \end{bmatrix} = \begin{bmatrix} 4 & 2 & -4 & -2 & 1 & -1 & 4 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 & 2 & 1 \\ 4 & 2 & 4 & 2 & 1 & 1 & 4 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 4 & -2 & -4 & 2 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 & -2 & 1 \\ 4 & -2 & 4 & -2 & 1 & 1 & 4 & -2 & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \\ c_8 \\ c_9 \end{bmatrix} \quad (2)$$

or briefly  $\mathbf{f} = \mathbf{A}\mathbf{c}$ . Then,  $\mathbf{c}$  can be obtained by multiplying the inverse matrix of  $\mathbf{A}$  to the both sides of (3) as follows,

$$\mathbf{c} = \mathbf{A}^{-1}\mathbf{f}. \quad (3)$$

Since  $\mathbf{A}^{-1}$  is fixed, only calculation of (3) is required to approximate the luminance distribution of (1). Finally, six pixels to be interpolated can be determined by

$$\begin{bmatrix} f(-1,1) \\ f(0,1) \\ f(1,1) \\ f(-1,-1) \\ f(0,-1) \\ f(1,-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \\ c_8 \\ c_9 \end{bmatrix} \quad (4)$$

or simply  $\hat{\mathbf{f}} = \mathbf{B}\mathbf{c}$ , where  $\hat{\mathbf{f}}$  represents a  $6 \times 1$  vector of unknown pixels. By locating a center position of the window for every original pixel except boundary pixels, the interpolated pixels become overlapped. Therefore, the interpolated pixel values are accumulated and divided by the number of accumulations. By applying the proposed method, pixel values can be smoothly interpolated. However, the polynomial approximation of (1) for every luminance distribution does not consider the local characteristics of the image. In order to preserve image details, the luminance distribution is modeled based on the local characteristic of the pixels in the window.

From nine known pixels in the  $5 \times 3$  window, a center pixel is classified into six regions. Based on the variance of the nine pixels, the center pixel is firstly classified into the uniform, texture, and edge regions [6]. Then, if the center pixel corresponds to the edge region, its direction is determined out of four directions including the horizontal, vertical, and two diagonal directions. Specifically, among four directions, the one with the minimum directional variance is selected to the edge direction.

Given the classification of the pixel, the best luminance model for each class is investigated. In order to find a suitable luminance model for each class, several luminance models are examined from (1). Among the 9 terms in (1), each term is discarded independently, so that 511 combinations are possible. Then, several interlaced images are generated from the progressive images, and interpolate every interlaced line by using 511 models. By comparing 511 models in terms of the peak signal-to-noise ratio (PSNR) of the deinterlaced image, the most suitable model for each class can be found. Note that for each model,  $\mathbf{A}$  and  $\mathbf{B}$  need to be modified according to a function model. Throughout the simulations, the most suitable luminance distribution model for each class is determined as follows:

$$\begin{aligned} f_u(x, y) = & c_1x^2y^2 + c_2xy^2 + c_3xy \\ & + c_4x + c_5y^2 + c_6y + c_7, \end{aligned} \quad (5)$$

$$\begin{aligned} f_t(x, y) = & c_1x^2y^2 + c_2xy^2 + c_3xy \\ & + c_4x^2 + c_5x + c_6y^2 + c_7y + c_8, \end{aligned} \quad (6)$$

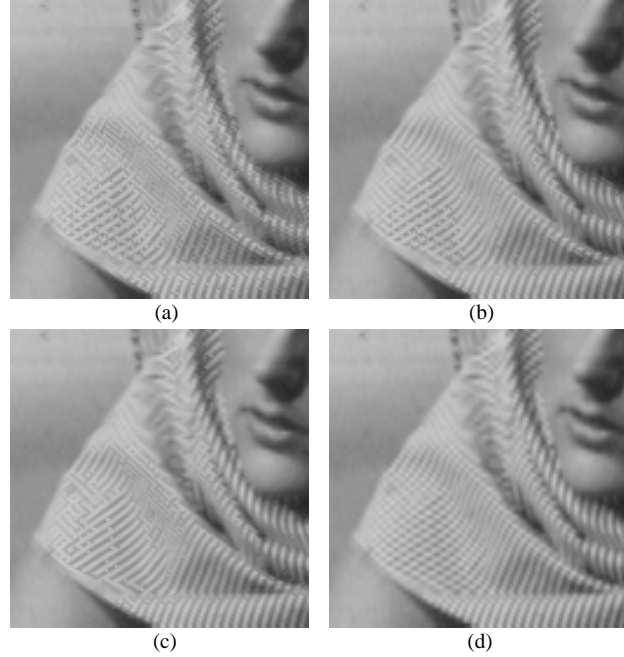


Fig. 2 Experimental results on *Barbara* image using (a) ELA, (b) CM1, (c) CM2, and (d) Proposed methods

TABLE I  
PSNR COMPARISON OF CONVENTIONAL AND PROPOSED ALGORITHMS

Methods	<i>Lena</i>	<i>Barbara</i>	<i>Bridge</i>	<i>Pepper</i>
ELA	31.94	32.08	25.90	33.23
CM1	32.99	33.77	26.30	33.55
CM2	33.43	34.21	26.42	33.68
Proposed	33.38	35.07	26.83	33.55

$$f_{e-hor}(x, y) = c_1x^2y^2 + c_2xy + c_3x + c_4y + c_5, \quad (7)$$

$$\begin{aligned} f_{e-ver}(x, y) = & c_1x^2y^2 + c_2xy^2 + c_3xy \\ & + c_4x^2 + c_5x + c_6y + c_7, \end{aligned} \quad (8)$$

$$f_{e-diag}(x, y) = c_1xy^2 + c_2xy + c_3x + c_4y^2 + c_5y + c_6. \quad (9)$$

where  $f_u$ ,  $f_t$ ,  $f_{e-hor}$ ,  $f_{e-ver}$ , and  $f_{e-diag}$  represent the obtained model for the uniform, texture, horizontal edge, vertical edge, and diagonal edge region, respectively. By adopting this adaptive luminance model, the local characteristics of the image can be utilized, so that the interlaced line can be more effectively interpolated.

### III. EXPERIMENTAL RESULTS

In order to evaluate the performance, four progressive images including *Lena*, *Barbara*, *Bridge*, and *Pepper* are firstly converted to the interlaced images by skipping every even line. Then, ELA, the edge pattern based algorithm (CM1) [3], and region classification based algorithm (CM2) [4] are compared with the proposed algorithm. Fig. 2 shows the magnified region of deinterlaced *Barbara* image obtained by various algorithms.

As can be seen, annoying artifacts exist in the texture region in Figs. 2 (a), (b), and (c). Though CM1 and CM2 perform better than ELA, these artifacts are still noticeable. However, the proposed algorithm preserves the texture and edge of the image without generating annoying artifacts as shown in Fig. 2(d).

Table I shows the quantitative comparison of the deinterlacing methods. As can be seen, the PSNR of the deinterlaced image obtained by the proposed method is comparable or higher than that of the conventional algorithms.

#### IV. CONCLUSION

In this paper, novel deinterlacing algorithm that interpolates the pixel by applying the pattern based polynomial approximation is proposed. Since a polynomial model is derived for each class by considering the local characteristics of the image, the local image details can be preserved without producing annoying artifacts in the resultant image.

#### ACKNOWLEDGMENT

This research was supported by Seoul Future Contents Convergence (SFCC) Cluster established by Seoul R&BD Program.

#### REFERENCES

- [1] T. Doyle, "Interlaced to sequential conversion for EDTV applications," in *Proc. Second Int. Workshop Signal Processing of HDTV*, pp. 412-430, Feb. 1998.
- [2] M. H. Lee, J. H. Kim, J. S. Lee, K. K. Ryu, and D. I. Song, "A new algorithm for interlaced to progressive scan conversion based on directional correlations and its IC design," *IEEE Trans. Consumer Electron.*, vol. 40, no. 2, pp. 119-129, May. 1994.
- [3] M. K. Park, M. G. Kang, and S. G. Oh, "New edge dependent deinterlacing algorithm based on horizontal edge pattern," *IEEE Trans. on Consumer Electron.*, vol. 49, no. 4, Nov. 2003.
- [4] S.-H Hong, R.-H Park, S. Yang, and J.-Y Kim, "Edge-preserving spatial deinterlacing for still images using block-based region classification," in *Proc. Second Int. Workshop Signal Processing of HDTV*, pp. 412-430, Feb. 2006.
- [5] J. W. Suh and J. Jeong, "Fast sub-pixel motion estimation techniques having lower computational complexity," *IEEE Trans. on Consumer Electron.*, vol. 50, no. 3, pp. 968-973, Aug. 2004.
- [6] M. I. Sezan and A. M. Tekalp, "Adaptive image restoration with artifact suppression using the theory of convex projections," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 38, no. 1, Jan. 1990.