

# CLASSIFICATION BASED DATA MIXING FOR HYBRID DE-INTERLACING TECHNIQUES

*M. Zhao\**, *C. Ciuhu†*, and *G. de Haan\*†*

\*Department of Electrical Engineering, Technische Universiteit Eindhoven  
Den Dolech 2, 5600 MB Eindhoven, the Netherlands  
phone: + (31) 40-2473614, fax: + (31)40-2433066, email: m.zhao@tue.nl

†Philips Research Laboratories Eindhoven  
Prof. Holstlaan 4, (WO02), 5656 AA Eindhoven, The Netherlands  
phone: + (31) 40-2745356, 2742555 fax: + (31) 40-2742630, email: {calina.ciuhu, g.de.haan}@philips.com

## ABSTRACT

De-interlacing is one of the key technologies in modern displays and multimedia personal computers. Various methods have been proposed including motion compensated (MC) methods and non motion compensated methods. Hybrid methods that combine different de-interlacing techniques are widely used to take advantages from individual algorithms. The combination is normally based on the quality criterion of individual de-interlacing algorithms. In this paper, we propose a classification based data mixing algorithm for hybrid de-interlacing. The algorithm first classifies the interpolated pixels from individual de-interlacing methods and then mix them to give the final output. The optimal mixing coefficients are obtained from an off-line training, which employs the Least Mean Squared (LMS) algorithm.

## 1. INTRODUCTION

Modern display principles and the introduction of video in personal computers (PC) require de-interlacing techniques to display the traditional interlaced video materials for progressive scanning [1]. Various de-interlacing techniques have been proposed in the last few decades and new methods are still being investigated. Previous overviews on de-interlacing [2, 3] categorize de-interlacing methods into non-motion compensated methods and motion compensated (MC) methods and hybrid methods. The non-MC methods includes linear techniques such as spatial filtering, temporal filtering, vertical-temporal filtering and non-linear techniques like motion adaptive filtering, edge-dependent interpolation, implicitly adapting methods. The MC category includes temporal backward projection, time-recursive de-interlacing, adaptive recursive de-interlacing, generalised sampling theorem based de-interlacing, etc. Non-MC methods are used in consumer products that require a reasonable performance at relatively low cost. The motion compensated methods provide better quality in high-end consumer and professional products.

Hybrid methods that combine different de-interlacing methods are designed to combine advantages of individual methods [4, 5, 6, 7]. In general, assume we have a number of  $N$  different de-interlacing results  $F_{i,j}(\vec{x}, n)$  ( $j = 1, 2, \dots, N$ ), the hybrid output  $F_o(\vec{x}, n)$  is a weighted sum of those  $N$  candidates:

$$F_o(\vec{x}, n) = \begin{cases} F(\vec{x}, n), & y \bmod 2 = n \bmod 2 \\ \sum_{j=1}^N k_j F_{i,j}(\vec{x}, n), & \text{otherwise} \end{cases} \quad (1)$$

Here,  $F(\vec{x}, n)$  is the luminance value of the pixel at position

$\vec{x}$  in the input field number  $n$  and  $k_j$  ( $j = 1, 2, \dots, N$ ) are mixing filter coefficients associated with the corresponding de-interlacing methods. The performance of the hybrid methods depends on the mixing coefficients  $k_j$  providing certain individual input de-interlacing methods. Usually,  $k_j$  are determined experimentally using quality metrics of the corresponding de-interlacing algorithms and are difficult to be optimised.

Classification based adaptive filter design methods have been proven to be successful for image up-scaling applications [8, 9]. These methods use an off-line training process in finding optimal interpolation parameters, which require a large amount of high resolution image and their corresponding down-scaled low resolution images. Specifically, in Kondo's method, the training is build on the Least Mean Square (LMS) algorithm [8] while Atkins uses the Expectation Maximization (EM) [9].

Similar methodology can be applied for hybrid de-interlacing when we interpret the mixing problem as finding the optimal filter coefficients  $k_j$ , given certain input de-interlaced pixels  $F_{i,j}(\vec{x}, n)$  and the corresponding quality metrics, or error indicators. The LMS algorithm can be used to find the optimal mixing coefficients since it has been proven that, in the context of de-interlacing, the subjective image quality is in good correlation with objective metrics like MSE [13].

The remainder of this paper is organized as follows. In Section 2, the general principle of this classification based data mixing method is presented. In Section 3, this classification based mixing principle is used to combine four de-interlacing methods. The results are shown in Section 4. In Section 5, we draw our conclusions.

## 2. CLASSIFICATION BASED DATA MIXING

In our classification based data mixing method, the equation to interpolate the output de-interlaced pixel is slightly different from Equation (1):

$$F_{co}(\vec{x}, n) = \begin{cases} F(\vec{x}, n), & y \bmod 2 = n \bmod 2 \\ \sum_{j=1}^N k_{j,c} F_{i,j}(\vec{x}, n), & \text{otherwise} \end{cases} \quad (2)$$

where  $k_{j,c}$  are weights for class  $c$ .

There are innumerable ways to classify the input data, i.e.,  $F_{i,j}(\vec{x}, n)$  ( $j = 1, 2, \dots, N$ ). Kondo proposed adaptive dynamic range coding (ADRC) [10] for classification, which

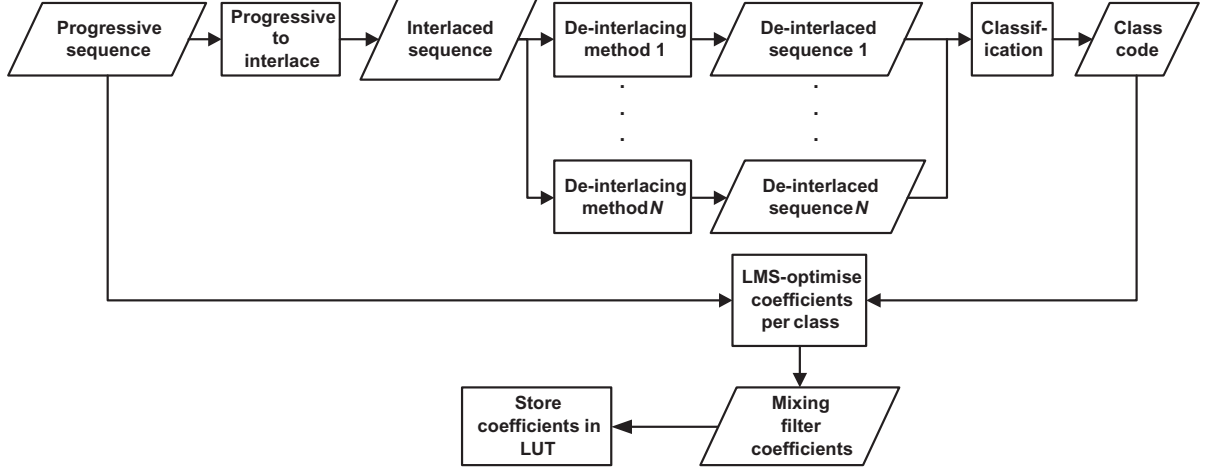


Figure 1: Training process of the adaptive data mixing method.

encode each pixel  $F$  into  $n$ -bit  $Q$ :

$$DR = F_{MAX} - F_{MIN} + 1 \quad (3)$$

$$Q = \left\lceil \frac{(F - F_{MIN} + 0.5) \times 2^n}{DR} \right\rceil \quad (4)$$

$DR$  is the dynamic range of the input pixels to be encoded,  $F_{MAX}$  and  $F_{MIN}$  corresponding to the maximum and minimum pixel values of the current input pixel group and  $\lceil \cdot \rceil$  is the rounding operator. The concatenation of the encoded  $Q$  from each input pixel  $F$  generates the class code  $c$ , which can be used to address the LUT.

The training process of the adaptive data mixing method is illustrated in Figure 1. A large amount of progressive input video sequences are first interlaced, and then de-interlaced with a number of  $N$  de-interlacing methods to obtain input pixels  $F_{i,j}(\vec{x}, n)$  ( $j = 1, 2, \dots, N$ ) for training. The input pixels are classified to generate the class code. The LMS optimisation is performed within each class to obtain the optimal mixing filter coefficients, which are stored in a look-up table (LUT) and can be addressed with the class code  $c$  after training.

To clarify the LMS algorithm used for training, let  $F_p(\vec{x}, n)$  be the luminance value of the original progressive pixels that were dropped out in the interlace process and  $F_d(\vec{x}, n)$  be the hybrid de-interlaced output on the corresponding position. Suppose one class contains in total a number of  $t$  samples in the training process. The error of the  $m^{th}$  interpolation sample is:

$$e_{m,c} = F_{p,m} - F_{d,m} = F_{p,m} - \sum_{j=1}^N k_{j,c} F_{i,j,m}(\vec{x}, n) \quad (m = 1, 2, \dots, t) \quad (5)$$

Consequently, the total squared error of this class can be expressed as:

$$e_c^2 = \sum_{m=1}^t e_{m,c}^2 \quad (6)$$

To find the minimum, we calculate the first derivative of  $e_c^2$  to each  $k$

$$\frac{\partial e_c^2}{\partial k_{j,c}} = - \sum_{m=1}^t 2F_{i,j,m}(\vec{x}, n) e_{m,c} \quad (7)$$

The minimum occurs when the first derivative is zero, which lead to the following equation for each class:

$$\begin{bmatrix} X_{0,0} & X_{0,1} & \cdots & X_{0,2} \\ X_{1,0} & X_{1,1} & \cdots & X_{1,2} \\ X_{2,0} & X_{2,1} & \cdots & X_{2,2} \\ \vdots & \vdots & \vdots & \vdots \\ X_{N,0} & X_{N,1} & \cdots & X_{N,N} \end{bmatrix} \begin{bmatrix} k_{0,c} \\ k_{1,c} \\ k_{2,c} \\ \vdots \\ k_{N,c} \end{bmatrix} = \begin{bmatrix} Y_0 \\ Y_1 \\ Y_2 \\ \vdots \\ Y_N \end{bmatrix} \quad (8)$$

The coefficients  $k_{j,c}$  can be obtained by solving Equation (8) for each class, where  $X_{l,r}$  and  $Y_l$  are:

$$X_{l,r} = \sum_{m=1}^t F_{i,l,m}(\vec{x}, n) \cdot F_{i,r,m}(\vec{x}, n) \quad (l, r = 0, 1, \dots, N) \quad (9)$$

$$Y_l = \sum_{m=1}^t F_{i,l,m}(\vec{x}, n) \cdot F_{p,m}(\vec{x}, n) \quad (l = 0, 1, \dots, N) \quad (10)$$

### 3. IMPLEMENTATION

The hybrid de-interlacing algorithm proposed by Nguyen [6] and Kovacevic [7] that mixes four methods is used to benchmark the classification based mixing. The four individual de-interlacing methods are: line averaging (LA), edge-dependent interpolation, field averaging (FA) and MC field averaging. We use EDDI [11] for edge-dependent interpolation method and 2D GST [12] for the MC method.

The interpolated pixels from the individual methods are calculated as follows:

$$F_{iLA}(\vec{x}, n) = \frac{F(\vec{x} - \vec{u}_y, n) + F(\vec{x} + \vec{u}_y, n)}{2} \quad (11)$$

$$F_{iFA}(\vec{x}, n) = \frac{F(\vec{x}, n-1) + F(\vec{x}, n+1)}{2} \quad (12)$$

$$F_{iEDDI}(\vec{x}, n) = \frac{F(\vec{x} - \vec{u}_y + l\vec{u}_x, n) + F(\vec{x} + \vec{u}_y - l\vec{u}_x, n)}{2} \quad (13)$$

$$F_{iGST}(\vec{x}, n) = \frac{F_i^{n,n-1}(\vec{x}, n) + F_i^{n,n+1}(\vec{x}, n)}{2} \quad (14)$$

with  $\vec{u}_x = (1, 0)^T$ ,  $\vec{u}_y = (0, 1)^T$  and  $l$  be the edge orientation.  $F_i^{n,n-1}(\vec{x}, n)$  is the result from GST de-interlacing method

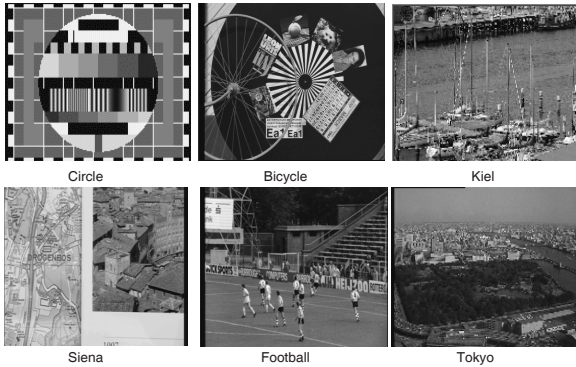


Figure 2: Screen shots from the progressive test sequences used for evaluation.

based on the previous and current field and  $F_i^{n,n+1}(\vec{x}, n)$  is the result based on current and next field. According to this mixing method, four error indicators ( $\epsilon_{LA}$ ,  $\epsilon_{EDDI}$ ,  $\epsilon_{FA}$  and  $\epsilon_{GST}$ ) are computed by calculating the absolute pixel (or group of pixels) difference along the interpolation direction.

$$\epsilon_{LA} = |F(\vec{x} - \vec{u}_y, n) - F(\vec{x} + \vec{u}_y, n)| \quad (15)$$

$$\epsilon_{FA} = |F(\vec{x}, n-1) - F(\vec{x}, n+1)| \quad (16)$$

$$\epsilon_{EDDI} = |F(\vec{x} - \vec{u}_y + l\vec{u}_x, n) - F(\vec{x} + \vec{u}_y - l\vec{u}_x, n)| \quad (17)$$

$$\epsilon_{GST} = |F_i^{n,n-1}(\vec{x}, n) - F_i^{n,n+1}(\vec{x}, n)| \quad (18)$$

The corresponding weighting factors  $k_q$ , according to Kovacevic [7], are defined as:

$$k_q = \frac{\epsilon_{LA}\epsilon_{EDDI}\epsilon_{FA}\epsilon_{GST}}{\epsilon_q \cdot SUM}, \quad q \in \{LA, EDDI, FA, GST\} \quad (19)$$

with

$$SUM = \epsilon_{LA}\epsilon_{EDDI}\epsilon_{FA} + \epsilon_{LA}\epsilon_{EDDI}\epsilon_{GST} + \epsilon_{LA}\epsilon_{FA}\epsilon_{GST} + \epsilon_{EDDI}\epsilon_{FA}\epsilon_{GST} \quad (20)$$

We implemented the classification based mixing algorithm with this hybrid de-interlacing problem. Instead of calculating the mixing coefficients using Equation (19), an off-line training was performed to obtain the optimal mixing coefficients, which employs about 2000 frames of video sequences with large variety of features. A 2-bit ADRC per input pixel or error indicator was used in the classification to obtain sufficient precision. The classification generates an 8-bit class code that corresponding to 256 classes in total.

#### 4. RESULTS

Six video sequences that differ in many aspects were used to evaluate the performance of the classification based mixing method. Each sequence contains about 30 frames. In those sequences, the *stationary* sequence Circle, the *global* horizontal moving sequence Tokyo, the *global* vertical moving sequence Siena and the *zooming* sequence Kiel are favorable for the MC de-interlacing method GST. The less accurately estimated *local* motion in Football and Bicycle requires protections obtained from the intra-field de-interlacing methods. The long distinct edges in all directions in the Bicycle sequence are best de-interlaced with edge dependent methods.

Table 1: MSE scores on individual sequences with various data mixing methods for hybrid de-interlacing algorithm. A. the mixing method using coefficients calculated from individual error indicators, B. classification based mixing method that uses input pixels for classification, C. classification based mixing method that uses error indicators for classification

MSE	A	B	C
Bicycle	47.35	36.21	26.69
Circle	0	0.32	0.38
Football	37.66	30.58	27.41
Kiel	100.68	85.65	82.20
Siena	21.44	7.82	8.57
Tokyo	9.93	8.22	9.25

This total set of sequences is believed to cover all strengths of the individual candidate de-interlacing algorithms for mixing. Figure 2 gives a screen shots from each of the progressive sequences.

The hybrid de-interlacing was performed on the interlaced test sequences with Equation (1) (with coefficients determined by Equation (19)) and (2) respectively. The Mean Square Error (MSE) was calculated between the de-interlaced video sequences and progressive ones, for which results are given in Table 1. Column A gives the result from mixing algorithm proposed by Nguyen [6] and Kovacevic [7], using error indicators to determine the mixing coefficients. Column B and C show results from our proposed classification based method, using input pixels or error indicators for classification respectively. We conclude that using interpolated pixels from individual de-interlacing methods for classification gives a reduction in the MSE score compared to the mixing method proposed by Nguyen and Kovacevic. Using error indicators for classification will further improve the overall performance, however, more calculations for obtaining the error indicators are required.

To enable a subjective comparison of those methods, screen shots of the sequence Bicycle are given in Figure 3. The top row shows that the classification based mixing algorithm successfully removes artifacts in the text areas. In the area that contains distinct long edges in all directions and areas that contain complex foreground and background (Bottom row), the method that perform classification based on interpolated pixels generates severe artifacts while the other two give better results. Clearly, the method that perform classification based on error indicators gives the optimal balance between the MC method and the intra-field methods.

#### 5. CONCLUSIONS

De-interlacing is the key technology in merging traditional interlaced video format with modern progressive, high definition display requirements. Hybrid de-interlacing methods are widely used to combine advantages from individual algorithms. The classification based data mixing method is able to find the optimal coefficients for combining outputs from individual de-interlacing algorithms based on the Minimum Mean Square Error (MMSE) criterion. The classification based mixing method gives both significant reduction in MSE of the de-interlaced video sequences and a clear improvement of the subjective image quality.

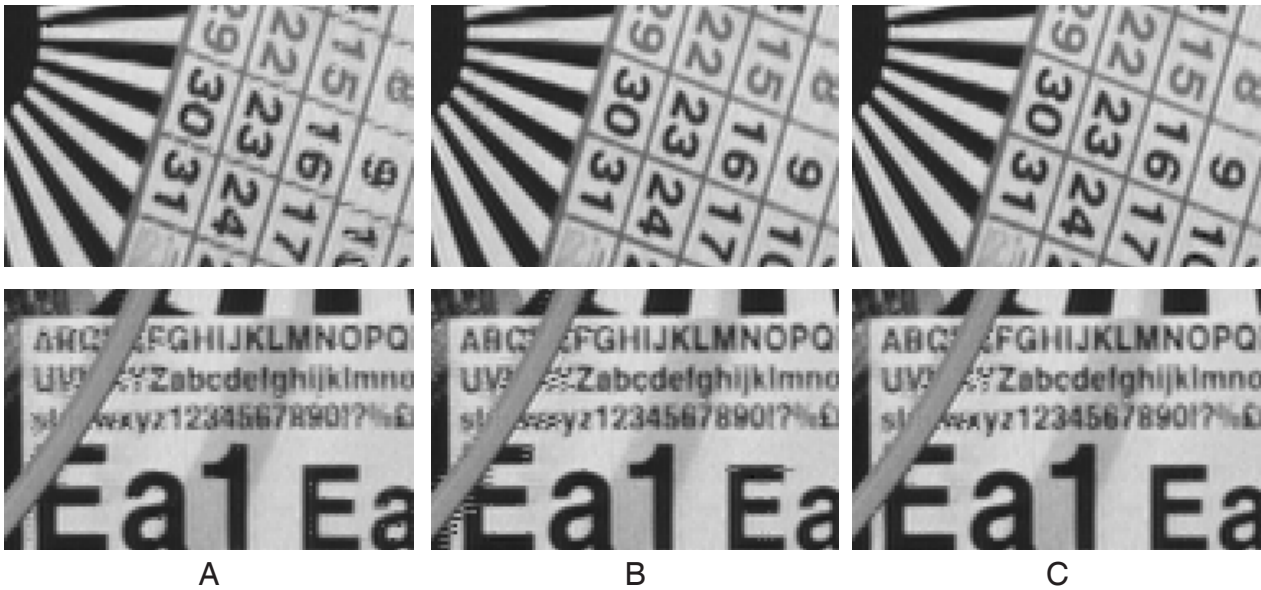


Figure 3: Image portions from de-interlaced images using: A. mixing method using coefficients calculated from individual error indicators, B. classification based mixing method that uses input pixels for classification, C. classification based mixing method that uses error indicators for classification.

The traditional methods perform the mixing based on a convex linear combination of the various de-interlacing candidates, i.e., all coefficients are positive numbers, which prevents it from finding the optimal solution. The classification based mixing method obtains the coefficients from training with LMS criterion will remove this restriction, thus extending the solution space.

Classification using error indicators, from our experiment, gives overall better result than classification using input pixels for mixing. We conclude that using error indicators for classification will better reflect the characteristic of the current mixing problem.

## REFERENCES

- [1] E. B. Bellers and G. de Haan, "De-interlacing: A key technology for scan rate conversion", *Advances in Image communications*, Elsevier, Vol. 9, ISBN: 0-444-50594-6, 2000.
- [2] G. de Haan and E. B. Bellers, "De-interlacing - an overview", *Proceedings of the IEEE*, Vol. 86, Issue 9, pp. 1839-1857, Sep. 1998.
- [3] G. de Haan, *Video Processing for Multimedia Systems, 2nd Edition*, ISBN: 90-9014015-8, pp. 145-174, 2000.
- [4] J. Salo, Y. Nuevo and V. Hameenaho, "Improving TV picture quality with linear-median type operations", *IEEE Transactions on Consumer Electronics*, Vol. 34, no 3, pp. 373-379, Aug. 1988.
- [5] J. S. Kown, K. Seo, J. Kim and Y. Kim, "A motion adaptive de-interlacing method", *IEEE Transactions on Consumer Electronics*, Vol. 38, no 3, pp. 145-150, Aug. 1992.
- [6] A. Nguyen and E. Dubois, "Spatial-temporal adaptive interlaced-to-progressive conversion", in *Signal Processing of HDTV, IV*, E. Dubois and L. Chiariglione, Eds., Elsevier Science Publishers, pp. 749-756, 1993.
- [7] J. Kovacevic, R. J. Safranek and E. M. Yeh, "De-interlacing by successive approximation", *IEEE Transactions on Image Processing*, Vol. 6, no 2, pp. 339-344, Feb. 1997.
- [8] T. Kondo, T. Fujiwara, Y. Okumura, and Y. Node, "Picture conversion apparatus, picture conversion method, learning apparatus and learning method", US-patent 6,323,905, Nov. 2001.
- [9] C. B. Atkins C. A. Bouman, and J. P. Allebach, "Optimal image scaling using pixel classification", *Proceedings of 2001 International Conference on Image Processing*, vol.3, pp. 864-867, 2001.
- [10] T. Kondo *et. al.*, "Adaptive dynamic range encoding method and apparatus", US-patent 5,444,487, Aug. 1995.
- [11] G. de Haan and R. Lodder, "De-interlacing of video data using motion vectors and edge information", *Digest of the ICCE'02*, pp. 70-71, Jun. 2002.
- [12] C. Ciuhu and G. de Haan, "A two-dimensional generalized sampling theory and application to de-interlacing", *SPIE, Proceedings of VCIP*, pp. 700-711, Jan. 2004.
- [13] M. Zhao and G. de Haan, "Subjective evaluation of de-interlacing techniques", to be presented at IS&TSPIE's 17th Annual Symposium, Jan. 2005.