

# Video De-Interlacing by Adaptive 4-Field Global/Local Motion Compensated Approach

Yu-Lin Chang, Shyh-Feng Lin, Ching-Yeh Chen, and Liang-Gee Chen, *Fellow, IEEE*

**Abstract**—A de-interlacing algorithm using adaptive 4-field global/local motion compensated approach is presented. It consists of block-based directional edge interpolation, same-parity 4-field motion detection, global/local motion estimation and compensation. The edges are sharper when the directional edge interpolation is adopted. The same parity 4-field motion detection and the 4-field local motion estimation detect the static areas and fast motion by four reference fields, and the global motion estimation detects the camera panning and zooming motions. The global and local motion compensation recover the interlaced videos to the progressive ones. Experimental results show that the peak signal-to-noise ratio of our proposed algorithm is 2~3 dB higher than that of previous studies and attain the best quality of subjective view.

**Index Terms**—Block-based directional edge interpolation, de-interlacing, global motion estimation, motion compensation.

## I. INTRODUCTION

DE-INTERLACING is important because it converts interlaced video sequences into progressive ones for displaying on progressive devices (e.g. Progressive CRT, LCD, Plasma Display, and Projection TV). Defects such as edge flicker, jagged effects, blurring, line-crawling and feathering cause uncomfortable visual artifacts if video sequences are not de-interlaced perfectly. For best visual effect on television, de-interlacing methods emphasize on the picture quality instead of the cost issue today. Thus the main problem for de-interlacing today is to develop an algorithm and architecture to recover all the defects appearing in the interlaced-to-progressive conversion.

The most common methods of intrafield de-interlacing are line doubling and BOB [1], which are used on small LCD panels. However, the vertical resolution is halved and there are jagged edges in the image. Many directional edge interpolation methods were proposed in [2]–[7]. The edge line average (ELA) algorithm [4], [5] was proposed to interpolate pixels along the



Fig. 1. Feathering effect appears while directly merging two fields in the motion area.

edges in the image. Many edge enhancement interpolation [2], [3] methods were proposed to improve the quality of the directional edge interpolation. Chang [6] proposed a generalized intelligent ELA to find the extent of the ELA algorithm with two reference scan line. Zhao [7] proposed an adaptive intrafield de-interlacing method to combine the benefit of the directional interpolation algorithms. Interfield de-interlacing is a de-interlacing method which utilizes the information between neighboring fields. Weave [1] is a simple interfield de-interlacing method directly combining two interlaced fields into one progressive frame. It needs one-field buffer. The video quality is better than that of intrafield interpolation in the static areas, but the feathering effect, as shown in Fig. 1, appears in the motion areas.

Motion adaptive de-interlacing is the most common de-interlacing method on digital television. Various kinds of motion adaptive (MA) de-interlacing methods [8]–[16] have been proposed to accomplish robust motion detection. Ville [12] proposed a MA technique using a fuzzy motion detector. Sun [13] proposed a shortest path technique of the motion information to re-align the fields of a video image. A motion detector with four reference fields was proposed by Lin [16] to obtain reliable motion information. However, the picture quality is still not high enough in large motion video sequences.

Some motion-compensated techniques [17]–[29] have been presented to preserve the picture resolution and improve image quality. Nguyen [17] proposed a same parity field motion compensated (MC) de-interlacing method by optical flow with one reference field, and it started the research of MC de-interlacing on motion vector (MV) accuracy and motion adaptation. A motion-compensated adaptive interpolation method was proposed by Sugiyama and Nakamura [19]. They used motion estimation

Manuscript received December 29, 2003; revised December 2, 2004. This work is supported by ActVision Technologies Inc. and AVermedia Technologies Inc., Taiwan, R.O.C. This paper was recommended by Associate Editor J.-N. Hwang.

Y.-L. Chang, C.-Y. Chen, and L.-G. Chen are with DSP/IC Design Laboratory, Graduate Institute of Electronics Engineering and Department of Electrical Engineering, National Taiwan University, Taipei 106, Taiwan, R.O.C. (e-mail: ylchang@video.ee.ntu.edu.tw; cychen@video.ee.ntu.edu.tw; lgchen@video.ee.ntu.edu.tw).

S.-F. Lin is with DSP/IC Design Laboratory, Graduate Institute of Electronics Engineering and Department of Electrical Engineering, National Taiwan University, Taipei 106, Taiwan, R.O.C. and also with the Research and Development Department of ActVision Technologies Inc., Taiwan 235, R.O.C. (e-mail: shyh@video.ee.ntu.edu.tw).

Digital Object Identifier 10.1109/TCSVT.2005.858746

and adaptive interpolation to reconstruct the missing field with the information obtained from the backward and forward fields. Ville [20] and Haan [21]–[23], proposed a motion-compensated de-interlacing algorithm with interpolation to reduce the interlaced-to-progressive artifacts. An object-based true-motion estimation algorithm was proposed by Schutten and Haan [23] to compensate the interlaced video sequences to the progressive ones according to the objects existence. Haan [24], [26] also proposed a system-on-chip MC de-interlacing architecture including three-dimensional recursive search (3DRS) true motion estimation with the consideration of global motion. Chang [28] proposed a same parity four field local MC de-interlacing method to utilize the characteristic of interlaced video on motion estimation. Yang, *et al.* [29] proposed a MA de-interlacing method with the assistance of motion estimation which considers and calculates the global motion by the distribution of the MVs. Nevertheless, camera rotation, and zooming may cause severe defects like radial feathering on the interlaced video sequences, and these problems cannot be solved by ordinary motion detection and motion estimation. With accurate global MVs, the global MC de-interlacing produces better results under these conditions. Thus, a real global motion estimation (GME) algorithm and global motion compensation (GMC) are employed in our design to solve these global motion problems.

In this paper, a 4-field adaptive global/local MC de-interlacing method with block-based directional edge interpolation, same parity 4-field motion adaptation, global/local motion estimation/compensation, and GMC/MC/MA mode decision are presented. The prior arts are listed in Section II. The fundamentals of the proposed method are described in Section III. Section IV shows the experimental results and comparison with previous methods. Section V gives the conclusion and remarks.

## II. PRIOR ARTS

De-interlacing methods can be characterized into four categories: intrafield de-interlacing, MA de-interlacing, motion-compensated de-interlacing, and global MC de-interlacing.

### A. Intra-Field De-Interlacing

The most cost-efficient method is intrafield de-interlacing. It is widely used in software implementations because it requires less computational power and needs only one delay line buffer. Line doubling repeats every line again in the current field, and BOB is an intrafield interpolation method which uses the current field to interpolate the missing field and reconstruct one progressive frame. However, by both of the methods, the vertical resolution is still the same, and the images are blurred. The jagged effect appears on the oblique line, and flickers are seen on the fine texture. Some interpolation methods use the pixels from the upper and lower line to reduce such flickers and aliasing.

Three pixels in the previous scan line and the next scan line are referenced to determine an obvious edge in the image shown in Fig. 2. This method reduces the blurring results of bilinear interpolation and produces sharp and straight edges. Although intrafield interpolation is a cost-efficient method, the resolution

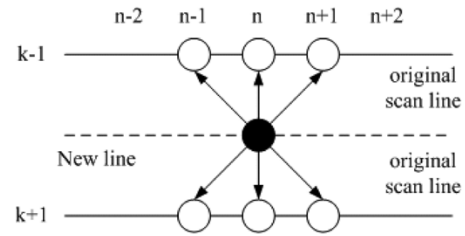


Fig. 2. Edge directional interpolation.

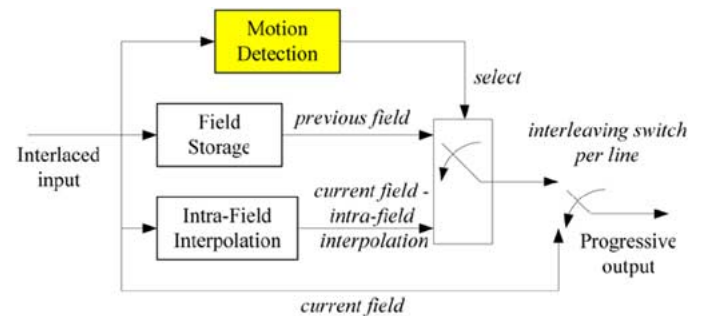


Fig. 3. Block diagram of MA de-interlacing.

of the de-interlaced frame is still the original field resolution. In addition, some defects occur when an object exists only in one-parity field, i.e. only even field or odd field.

### B. Motion Adaptive De-Interlacing

The MA de-interlacing combines the advantages of both intrafield de-interlacing and interfield de-interlacing. It detects the motion areas first. Then it adopts intrafield de-interlacing in the motion areas and adopts weave de-interlacing in the static areas. The block diagram is shown in Fig. 3. This method preserves video resolution in the static areas and eliminates the feathering effect in the motion areas. However, if the motion detection fails in some areas, the picture quality is lowered on those areas.

### C. Motion-Compensated Method

The motion-compensated methods utilize motion estimation to find the most similar blocks in the neighboring fields and calculate their MVs, as shown in Fig. 4. Then a new field is reconstructed from the neighboring field. Block matching algorithm needs extra internal buffer to store the data of the current macroblock and reference macroblock. It also needs large amount of computational power to calculate the matching criterion—sum of absolute difference (SAD). Although motion-compensated methods have greater potential to produce better result in panning and fast sequences, the video quality deteriorates in the video scenes with rotation and scaling.

### D. GMC De-Interlacing

While the camera is zooming in, the objects and background in the forward field are smaller than those in the current field. The forward field is scaled to a size matching with the current field, and the object is found by local motion information between the current field and the scaled forward field.

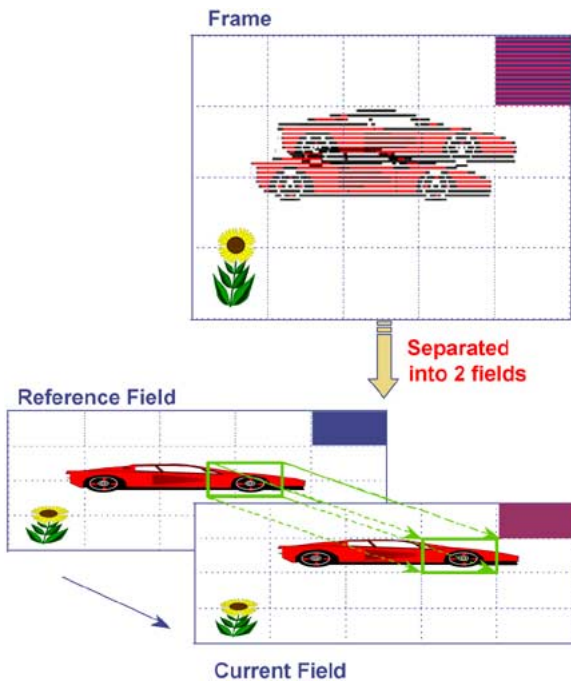


Fig. 4. Block matching by ME method.

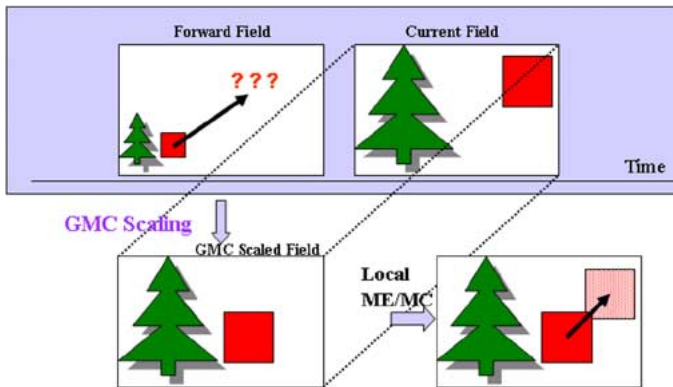


Fig. 5. Local ME/MC of the current field and the GMC field.

Take Fig. 5 for example, this is a video sequence with camera zooming-in. The tree and the object in the forward field are smaller than those in the current field. Common motion-compensated de-interlacing methods are not able to find the MV of the object, because the object is larger in the current field, they are regarded as different objects. But for our proposed method, the global motion estimation produces the global MV from the forward field to the current field. The global MC scaled field (GMC-scaled-field) is produced if the scaling parameter of the global motion estimation is enabled. The size of the GMC-scaled-field matches with that of the current field, so motion estimation between these two fields is done correctly. By compensating the current field with the GMC-scaled-field and the correct local MV, the radial feathering effect is eliminated.

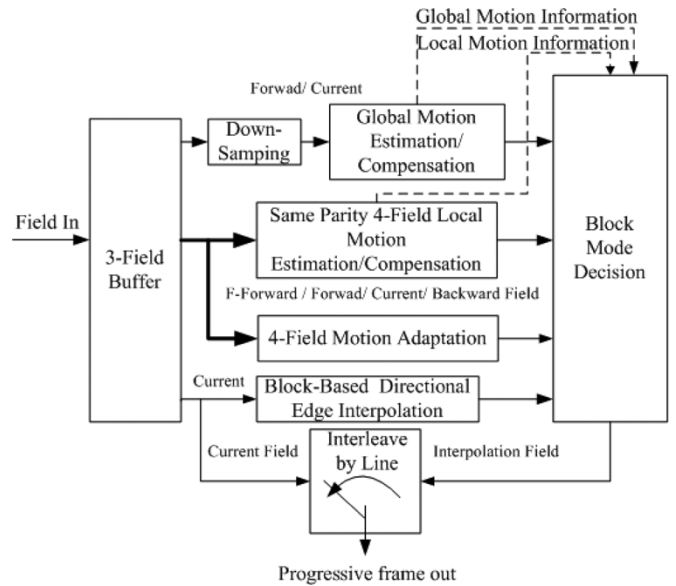


Fig. 6. Block diagram of proposed de-interlacing method.

### III. PROPOSED ADAPTIVE GLOBAL/LOCAL MOTION COMPENSATED DE-INTERLACING

The block diagram of the proposed method is shown in Fig. 6. It contains six major parts: the field buffer, the directional edge interpolation module, the global motion estimation/compensation, the 4-field motion adaptation, the 4-field local motion estimation/compensation, and GMC/MC/MA block mode decision module. The three-field buffers are used to store the reference data. The directional edge interpolation module interpolates an edge along its direction in accordance with the current-field information. The global motion information—rotation, zooming, and panning is obtained from the global motion estimation/compensation module. The local motion information is obtained by local motion estimation and compensation from backward to forward field, and from current field to forward-forward field, i.e. 4-field motion estimation. The 4-field motion adaptation module detects where the motion areas are in order to operate with the 4-field motion compensation. The mode decision block selects the results of global motion compensation, local motion compensation, motion adaptation or block-based directional edge interpolation according to the motion information. Finally, the current field and the interpolated field are merged into a progressive frame.

Fig. 7 is the flowchart of our proposed block mode decision scheme. The motion information is retrieved from the global motion estimation and local motion estimation. The global motion estimation takes the down-sampled video as its source and produces several global motion parameters (vectors). The global MVs contain the scaling, and rotation information. If global motion exists, the reference field of the local motion estimation is the global MC field. Otherwise, there are three reference fields—the forward-forward field, the forward field and the backward field—for local motion estimation.

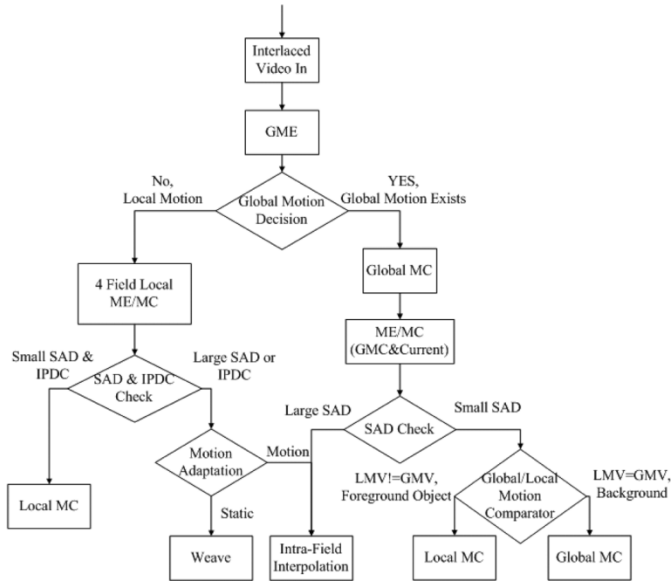


Fig. 7. Data flowchart of the proposed method.

The block mode decision module works with following rules: Under the existence of the global motion, the SAD is used to determine whether to use the output of the intrafield de-interlacing or the output of the MC de-interlacing. If the SAD is larger than a certain threshold, it means that the object is moving too fast, motion estimation doesn't find a matching block, and the block-based edge interpolation is adopted. The local MVs and the global MVs are compared if the SAD of a block is small. Once the local MV of a block is the same with the translation part of the global MV, the block is regarded as a background block and the output is the GMC result. Or else the local MC results are outputted.

If there is no global motion, the left part—the 4-field local MC de-interlacing with motion adaptation is invoked. The 4-field local motion estimation is adopted to calculate MVs. After the 4-field local motion estimation, the SAD and the interlaced pixel distortion classification (IPDC) of a block is checked. If the SAD and IPDC of a block are small, the local MC result is used. If the SAD or IPDC of a block is larger than a certain threshold, the motion adaptation is invoked, and it produces the MA results.

The details of the global motion estimation/compensation, the 4-field local motion estimation/compensation, the 4-field motion adaptation and the block-based directional edge interpolation are described in this section.

#### A. Global Motion Estimation/Compensation

The GME algorithm is adopted from the MPEG-4 Verification Model. The algorithm is a differential method, which is called Levenberg–Marquardt iterative minimization algorithm, and it supports four global motion models, namely the translation, isotropic, affine, and perspective models.

The flowchart of the global motion estimation/compensation of our de-interlacing method is shown in Fig. 8. This algorithm

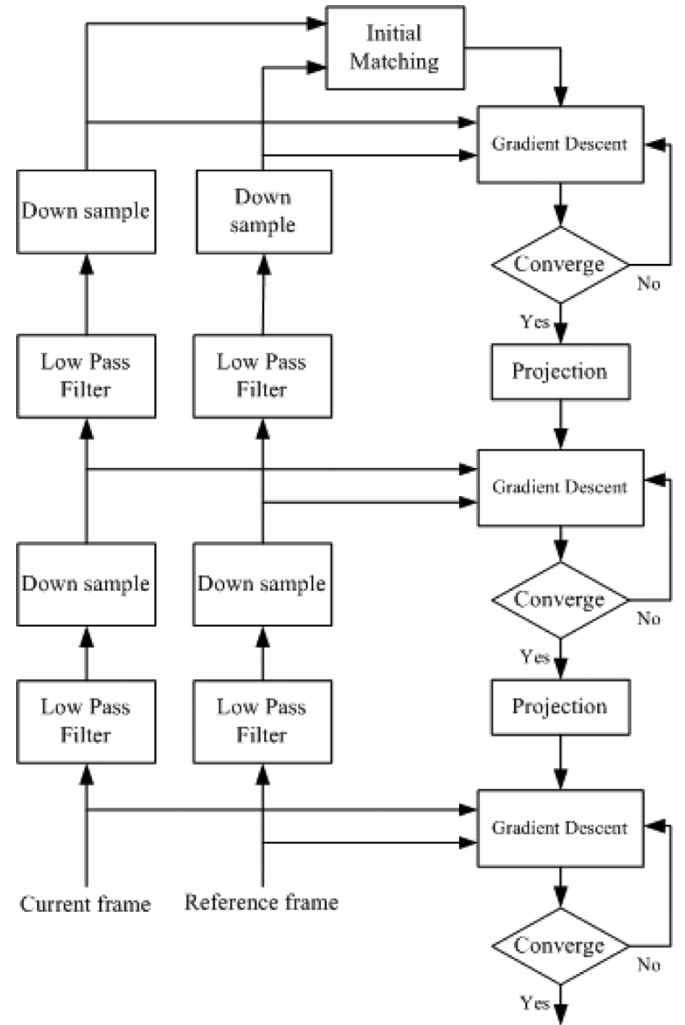


Fig. 8. Flowchart of GME algorithm in MPEG-4 VM.

is applied with a three-level pyramid, and a three-tap filter is used for this purpose. The cost function to be minimized is the mean square error between the current frame and the GMC frame. At the top level of the pyramid, the current and reference frames are processed in the initial matching to get the prediction of translation first. After initial matching, it is processed by gradient descent, and more accurate global motion parameters are then derived. When the global motion parameters converged, the next level proceeds.

The initial matching stage resembles the local motion estimation in common video coding. However, it uses the whole frame, not a macroblock. The algorithm of this stage is full search and its search range is  $-8 \sim +7$ . Initial matching iterates until the result of initial matching converges or the number of iterations is above five with the estimated translation MV obtained as the initial prediction for gradient descent.

In the gradient descent stage, the global motion parameters are computed by minimizing the mean square error  $E$

$$E = \frac{1}{N} \sum_{i=1}^N |e(i)|^2$$

$$e(i) = I(x, y) - I'(x', y') \quad (1)$$

where  $I(x, y)$  is the luminance of pixel at position  $(x, y)$  in the current frame,  $I'(x', y')$  is the luminance of the corresponding position  $(x', y')$  in the reference frame and  $N$  is the total number of effective pixels. The iterative procedure is shown as follows:

$$\begin{aligned} M^{t+1} &= M^t + A^{-1}B \\ M &= (m_0 \ m_1 \ \cdots \ \cdots \ m_{n-2} \ m_{n-1})^T \end{aligned} \quad (2)$$

where

$M^t$  global motion parameters at iteration  $t$ ;

$A$   $n \times 1$  matrix;

$B$   $n \times 1$  matrix;

$n$  number of global motion parameters.

The coefficients of matrices  $A$  and  $B$  are given by

$$\begin{aligned} A_{kj} &= \sum_{i=1}^N \frac{\partial e(i)}{\partial m_k} \frac{\partial e(i)}{\partial m_j} \\ B_k &= \sum_{i=1}^N e(i) \frac{\partial e(i)}{\partial m_k}. \end{aligned} \quad (3)$$

The gradient descent procedure starts after the Initial Matching at the top level of the pyramid and repeats at the subsequent levels in a top-down approach. At each level, this process iterates until the change of each parameter is small enough or the count of iterations is above 32. Here we use the translation, scaling and rotation vectors as the global motion parameters, so the camera zooming and rotation are detected by the global motion estimation module.

Finally, the global MVs are calculated, and the global motion compensation is adopted to compensate the camera panning/rotation/zooming of the video sequence. And this global motion estimation method is implemented on the hardware through several decomposition methods, the implemented hardware considerations are listed in the Results section of this paper.

### B. Same Parity 4-Field Local Motion Estimation

1) *Same-Parity Field Characteristic*: For interlaced video, the sampling data of the even field and the odd field with the same vertical position are slightly different. Same-parity 4-field motion detection [15] detects the motion areas by the even-to-even-field or odd-to-odd-field difference. The benefits are seen in the static area. Traditional de-interlacing using successive-field difference for motion detection leads to some problems. The motion detection by successive-field difference makes wrong decisions in the horizontal edge of the static region and produces the line-crawling effect. With same-parity field characteristic, the even-to-even-field difference and odd-to-odd-field difference are very small in the static area. The correct decisions are made, and the static regions are reconstructed perfectly.

2) *4-Field Local Motion Estimation/Compensation*: As seen in Fig. 9, the 4-field local motion estimation involves two more fields than the same-parity 2-field motion estimation [17]—the forward field and the backward field. It also utilizes same-parity field characteristic, which solves the detection error

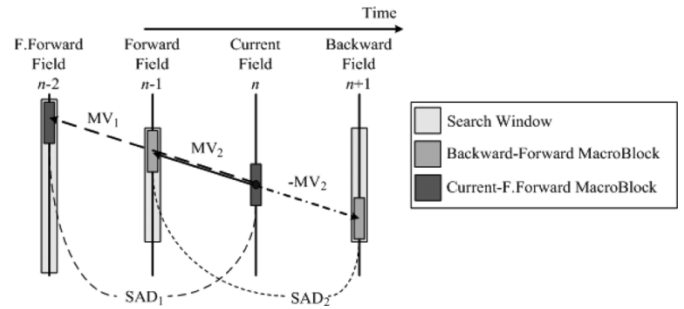


Fig. 9. Same Parity 4-field local motion estimation.

caused by the different sampling data with the same vertical positions in the even field and odd field. The extra field data between the current field and forward-forward field provides more motion information than the same-parity 2-field motion estimation. Therefore the feathering effect is eliminated by this method.

Traditional motion estimation method for video coding is to find the best matching block which produces the minimal residue. However, the motion estimation we need in de-interlacing is to find the best-matching block which suits with its neighboring block. That is to say, true MVs on the true images are needed for the de-interlacing application, not the MVs that produce the minimal residues. The proposed 4-field motion estimation is shown in Fig. 9. The difference between this motion estimation and traditional motion estimation is the matching criterion. The matching criterion of the 4-field motion estimation calculates two same parity field SAD values. The  $SAD_1$  in Fig. 9 denotes the block matching SAD value of the current field to the forward-forward field motion estimation.  $SAD_1$  is calculated by accumulating the absolute difference of the current macroblock and the macroblock candidate in the search window of the forward-forward field, which is written as following equation:

$$SAD_{1k} = \sum_{i,j \in MB} |f(i + MV_{kx}, j + MV_{ky}, n-2) - f(i, j, n)| \quad (4)$$

where  $f(i, j, n)$  denotes the luminance intensity at the  $(i, j)$  location of the  $n$ th field, and the  $MV_{kx}$  and  $MV_{ky}$  means the  $x$  component and the  $y$  component of the  $k$ th MV candidate  $MV_k$  in the search window of the forward-forward field.

The second SAD value is named  $SAD_2$ . It is calculated by accumulating the absolute difference between the macroblock where the  $MV_2$  indicated to in the forward field and the macroblock where  $-MV_2$  indicated to in the backward field.  $SAD_2$  is shown by

$$SAD_{2k} = \sum_{i,j \in MB} \left| f \left( i + \frac{1}{2}MV_{kx}, j + \frac{1}{2}MV_{ky}, n-1 \right) - f \left( i - \frac{1}{2}MV_{kx}, j - \frac{1}{2}MV_{ky}, n+1 \right) \right| \quad (5)$$

where the meanings of  $f(i, j, n)$ ,  $MV_{kx}$  and  $MV_{ky}$  are the same as  $SAD_{1k}$  in (4).

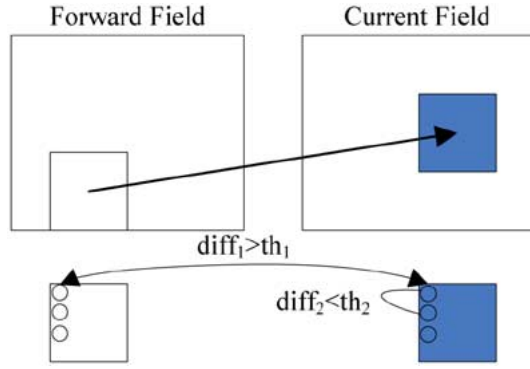


Fig. 10. Interlaced pixel distortion classification.

In order to find a MV, the matching criterion of the 4-field local motion estimation is minimizing the value of  $SAD_1 + SAD_2$ , as described in (6). And  $MV_k$  is the found MV

$$k = \text{Arg} \left( \min_{k \in SW} (SAD_{1k} + SAD_{2k}) \right). \quad (6)$$

If only  $SAD_1$  is considered to be the matching criterion, the pixel vales of the forward field and the backward field are not considered. In other methods, there are lots of errors if we divide the found MV by two and produce the MC field according to the divided MV. If only  $SAD_2$  is considered to be the matching criterion, the image data of the current field is not considered. While a complex-textured macroblock is on a uniform background, the MV is prone to be wrong. Hence  $SAD_1$  and  $SAD_2$  must be considered together to obtain the correct motion information. Because the  $SAD_1 + SAD_2$  matching criterion involves four field image data, this motion estimation is called “4-field local motion estimation.”

As the MVs of the macroblocks are gotten, the motion compensation starts to compensate the current field from the forward field and backward field. In Fig. 9, if  $MV_1$  is the found MV and  $MV_2 = (1/2)MV_1$ , the macroblock data indicated by  $MV_2$  in the forward field and the macroblock data indicated by  $-MV_2$  in the backward field are averaged and compensated to the position of the current macroblock. In order to compensate the real sampled data, the motion estimation and compensation here is an integer-pixel motion estimation, not the quarter-pixel one used in coding. After the compensation of every macroblock is done, a new factor for the block mode decision named IPDC is calculated.

The most suffering defect on video without de-interlacing is the feathering effect. We proposed a method to accumulate the pixel count with the feathering effect within a block. If there are many pixels with feathering effect in a compensated block, it means the MC result may be wrong. Thus the pixels in that block are the results of MA de-interlacing. And the proposed method shown in Fig. 10 is called IPDC. The feathering effect is detected through IPDC. By IPDC and SAD, the block mode

decision module determines the block mode more accurately. As shown in Fig. 10, there is a matching block in the forward field. And the IPDC calculation in a block is presented as the following pseudo-code:

```
diff1 = abs(fn-1(i + MVx, j + MVy) - fn(i, j));
diff2 = abs(fn(i, j) - fn(i, j + 1));
if (diff1 > th1 && diff2 < th2) IPDC ++;
```

$MV_x$  and  $MV_y$  are the found MV. The value of  $diff_1$  represents whether a pixel in the compensated block is similar to its current block or not, and the value of  $diff_2$  shows if the pixel is placed in between two similar pixels with current block. If the pixel in the compensated block is placed in between two different pixels, the possibility for the viewer to suffer the feathering effect is increased, and the IPDC increased. Hence the IPDC detects the feathering effect and propagates the information to the block mode decision module.

### C. 4-Field Motion Adaptation

The motion adaptation part is similar to the 4-field MA de-interlacing [16]. It uses three reference field and the current field for the motion detection in the MA de-interlacing. The reference fields are the same as those of the same parity 4-field local motion estimation. The motion adaptation also utilizes the characteristics of the interlaced video. It uses even-to-even and odd-to-odd field difference for its motion detection. As shown in Fig. 11, there are two field differences calculated at the first time—the forward-forward field to current field difference and the forward field to backward field difference. After the two field differences calculation, an “OR” operation is done to produce an accurate motion information field. With the motion information field, the motion areas are processed by intrafield de-interlacing, and the static areas are processed by weave de-interlacing.

The architecture of the proposed local motion estimation method is shown in Fig. 12. Since the operation and reference fields of the motion adaptation are similar to the 4-field local motion estimation. The logic circuits of the two modules are combined. The areas and memory bandwidth are reduced after the combination.

### D. Block-Based Directional Edge Interpolation

The directional edge interpolation algorithm uses the pixel-difference in  $9 \times 4$  block size. This method generates straighter lines with higher contrast than the ELA [4], because it references more pixels than ELA. Pixel differences are calculated along the 17 directions

$$\{d_l(1), d_l(3), d_l(5), d_l(7), d_l(9), d_r(3), d_r(5), d_r(7), d_r(9), \\ d_{lu}(3), d_{lu}(5), d_{lu}(7), d_{lu}(9), d_{ru}(3), d_{ru}(5), d_{ru}(7), d_{ru}(9)\}$$

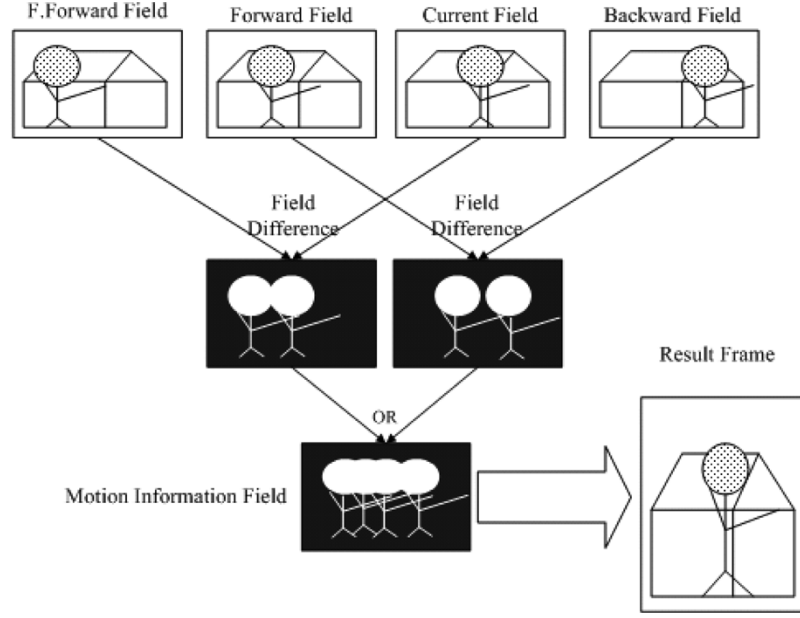


Fig. 11. The 4-field motion detection.

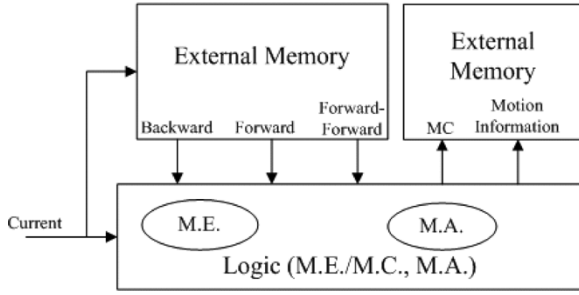


Fig. 12. ME/MC/MA architecture.

as shown in Fig. 13 and (7). The output pixel is produced depending on the minimum difference of pixel in the 17 directions as shown in (8)

$$d_l(n) = \left| x \left( i - \frac{n-1}{2}, j-1 \right) - x \left( i + \frac{n-1}{2}, j+1 \right) \right|$$

$$n \in \{1, 3, 5, 7, 9\}$$

$$d_r(n) = \left| x \left( i + \frac{n-1}{2}, j-1 \right) - x \left( i - \frac{n-1}{2}, j+1 \right) \right|$$

$$n \in \{3, 5, 7, 9\}$$

$$d_{lu}(n) = \left| x \left( i + \frac{n-1}{2}, j-3 \right) - x \left( i - \frac{n-1}{2}, j+3 \right) \right|$$

$$n \in \{3, 5, 7, 9\}$$

$$d_{ru}(n) = \left| x \left( i + \frac{n-1}{2}, j-3 \right) - x \left( i - \frac{n-1}{2}, j+3 \right) \right|$$

$$n \in \{3, 5, 7, 9\} \quad (7)$$

$$(n', p) = \text{Arg}(\min(d_l(n), d_r(n), d_{lu}(n), d_{ru}(n)))$$

$$n \in \{1, 3, 5, 7, 9\} \quad (8)$$

where  $n'$  denotes the found direction with a smallest difference, and  $p$  means whether the left, right, left up, or right up part the difference is in. The next thing we have to do is to check

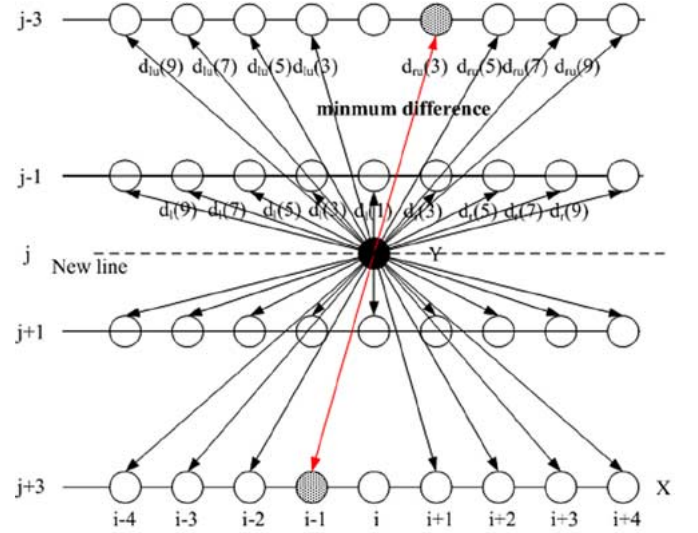


Fig. 13. Proposed block-based directional edge interpolation.

whether the chosen direction is the dominant edge or not. If the chosen direction is on the right side, the difference of the chosen direction is  $d_r(n_0)$ , and it conforms to the following equation

$$|d_r(n_0) - d_l(n)| < \theta, \quad n \in \{1, 3, 5, 7, 9\} \quad (9)$$

where  $\theta$  is a given threshold which help to check the dominant edge, usually in the range of 10~30. Finally, if the chosen direction is the dominant edge, the final output is the bilinear interpolation along the dominant edge, else it is the bilinear interpolation of  $x(i, j-1)$  and  $x(i, j+1)$ .

#### IV. EXPERIMENTAL RESULTS AND PERFORMANCE ANALYSIS

This section discusses the instruction analysis and experimental results of the proposed de-interlacing methods. For the de-interlacing applications in the digital TV, the quality issue is

TABLE I  
INSTRUCTION PROFILE FOR THE PROPOSED METHOD

Arithmetic Instructions	30.20%
Data Instructions	52.93%
Logic Instructions	1.12%
Rotate and Shift Instructions	1.31%
Jump, Test and Compare Instructions	12.15%
Stack Instructions	1.24%
Other Instructions	1.05%
Total	100%
Total Instructions (CIF 30FPS)	87,399,298,641
	87 GIPS

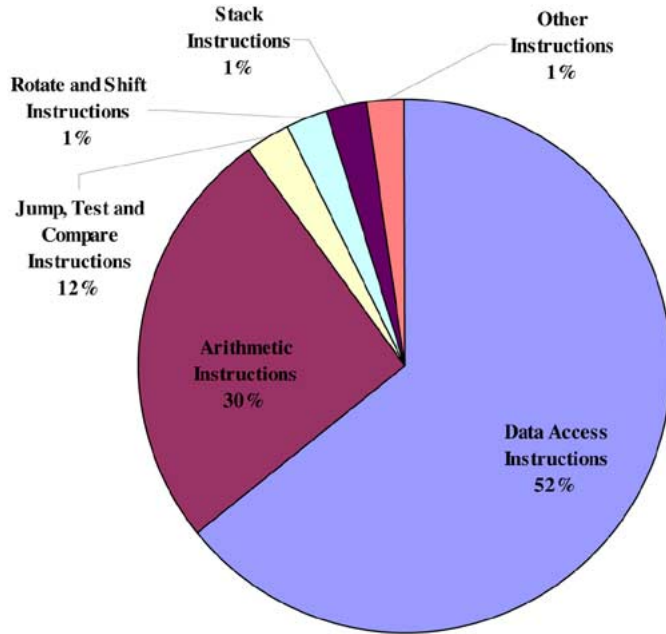


Fig. 14. Percentage of instruction sets.

the main problem. Thus the hardware analysis of this paper emphasizes on the operation reduction of the design to achieve the real-time issue. And the quality comparison shows the picture quality of various de-interlacing methods.

#### A. Analysis of Instruction Counts and Hardware Cost

1) *Instruction Profiling:* The instruction profile is shown in Table I and Fig. 14. The average instructions to be executed for video sequences (CIF format, 30 fps) are 87GIPS. Because of the huge amount of data iterative operations in ME and GME, most of the computational instructions are data instructions. If hardware implementation issue is concerned, the data iterative frequency of GME must be enormously reduced.

2) *Hardware Analysis for Global Motion Estimation:* The gate count of the proposed global motion estimation [30] is 130 K, and it operates in the 100 MHz by 0.18- $\mu$ m technology.

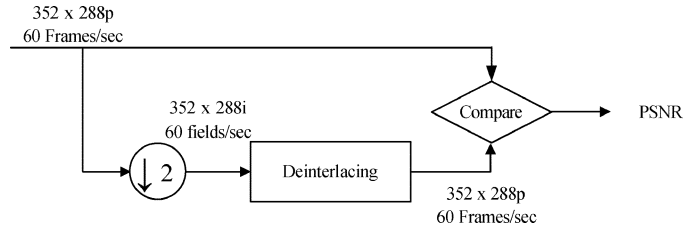


Fig. 15. Performance measurement method.

TABLE II  
PSNR COMPARISON

Name	Merged	Bilinear	ELA [4]	2-Field + Morphology [15]	4-Field MA[16]	3DRS MC[27]	Proposed 4-Field Adaptive MC	Proposed 4-Field Adaptive GMC/MC
Silent	29.52	29.15	26.72	39.22	39.82	38.06	40.37	40.37
Weather	31.01	26.93	25.55	37.49	40.49	37.86	43.72	43.72
Mobile	17.83	24.92	23.99	25.17	25.83	26.14	27.29	25.93
Mother Daughter	33.59	32.89	30.82	38.05	42.65	41.83	44.83	44.83
Container	31.01	28.14	25.04	35.33	37.81	31.47	40.17	40.17
Stefan	11.86	26.91	25.49	27.38	26.12	27.02	26.99	26.12
Dancer	23.19	36.15	33.41	32.95	34.06	35.68	35.33	33.23
Foreman	23.30	29.96	27.41	30.83	33.21	32.54	34.77	34.37
Hall Monitor	30.61	30.68	27.75	35.31	39.14	33.18	40.04	40.04
Table Tennis	23.89	27.32	25.85	32.46	35.11	29.81	36.23	35.20
Coastguard	22.59	27.84	26.32	29.00	30.45	29.34	33.30	28.83

The memory bandwidth of the global motion estimation with  $720 \times 480i$  60 fps format is 183.2 Mb/s in the worst case. In average cases, the memory bandwidth of the global motion estimation is 131 Mb/s.

3) *Hardware Analysis for Local Motion Estimation:* The macroblock size of the local motion estimation is  $16 \times 8$ , and the search range of horizontal/vertical directions is from  $-32$  to  $+31$ . The motion estimation is an integer-pixel motion estimation to get the real sampled data in the fields. The average memory bandwidth of the proposed local motion estimation with CCIR601  $720 \times 240$ , 60 fps format is 46 Mb/s, it includes the memory access of the three reference fields. It is possible for this method to be implemented on the hardware using the 0.18- $\mu$ m technology today.

#### B. Objective Performance Comparison

Several CIF standard video sequences have been converted from progressive into interlaced format according to the algorithm shown in Fig. 15. The original 60 progressive frames/s video sequence is decimated into the 60 interlaced fields/s one. After de-interlacing, the 60 interlaced fields/s video sequence is reconstructed into the 60 progressive frames/s one. Then we compare the original progressive sequence and the output de-interlaced sequence.

The comparisons of peak signal-to-noise ratio (PSNR) and mean square error (MSE) are presented in Table II, Fig. 16, and Fig. 17. There are eleven sequences and eight methods



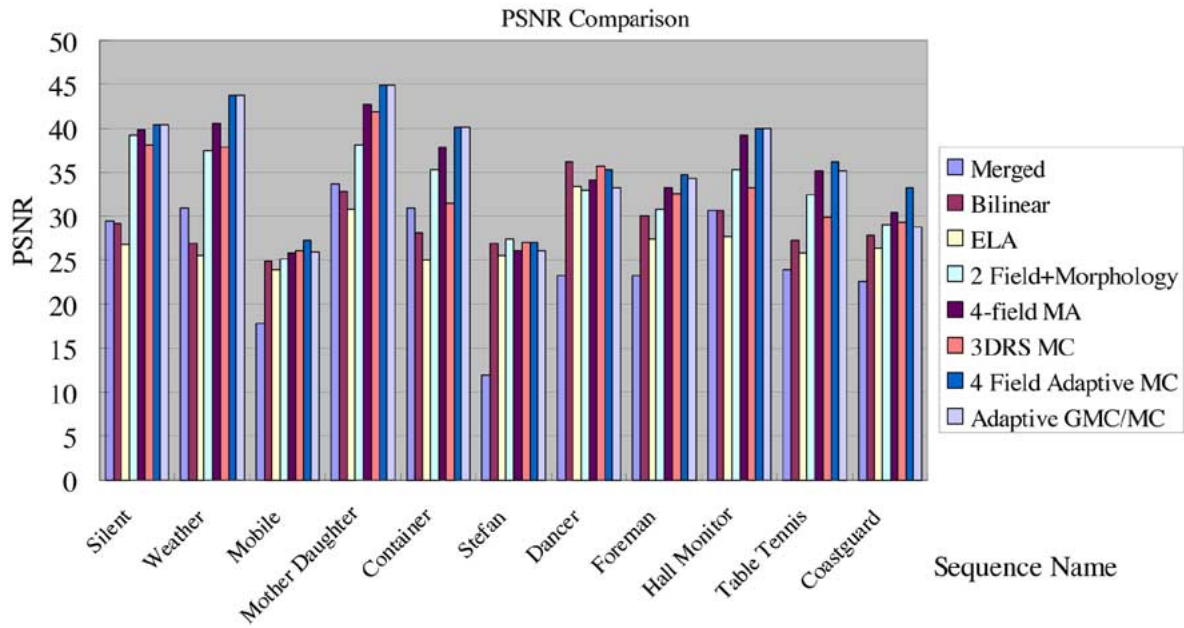


Fig. 16. Bar chart of PSNR performance comparison.

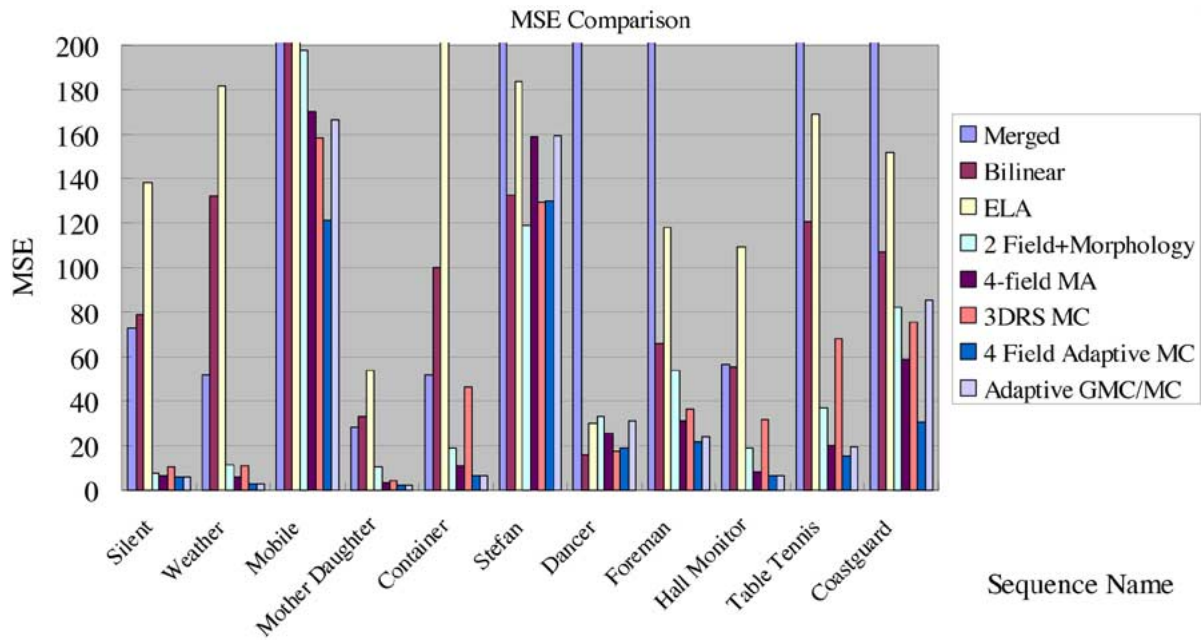


Fig. 17. Bar chart of MSE performance comparison.

compared. The methods are directly merged frame (Merged), bilinear interpolation (Bilinear), ELA [4], 2-field morphological MA de-interlacing (2-field + morphology) [15], 4-field MA de-interlacing (4-field MA) [16], 3DRS MC de-interlacing (3DRS MC) [26], the proposed 4-field local MC de-interlacing with MA (4-field adaptive MC), and the proposed adaptive GMC/MC de-interlacing (Adaptive GMC/MC). The proposed 4-field local MC de-interlacing with motion adaptation is the part with no global motion in Fig. 7.

The results show that the proposed 4-field local MC de-interlacing with motion adaptation has best objective performance than the other de-interlacing methods. As for the adaptive

GMC/MC de-interlacing, the objective results are the same as the 4-field local MC with motion adaptation one while there is no global motion. Otherwise, if global motion exists, the objective results degraded because the reference field of the local motion estimation here is the global MC field, and the global MC field is produced by bilinear interpolation. If better filter is used for the global motion compensation, the results of the adaptive GMC/MC de-interlacing are better. Although the PSNR is lower than the local MC de-interlacing methods, the subjective view is the best one especially for video sequences with large panning, zooming and rotation such as Stefan, Table Tennis, etc.

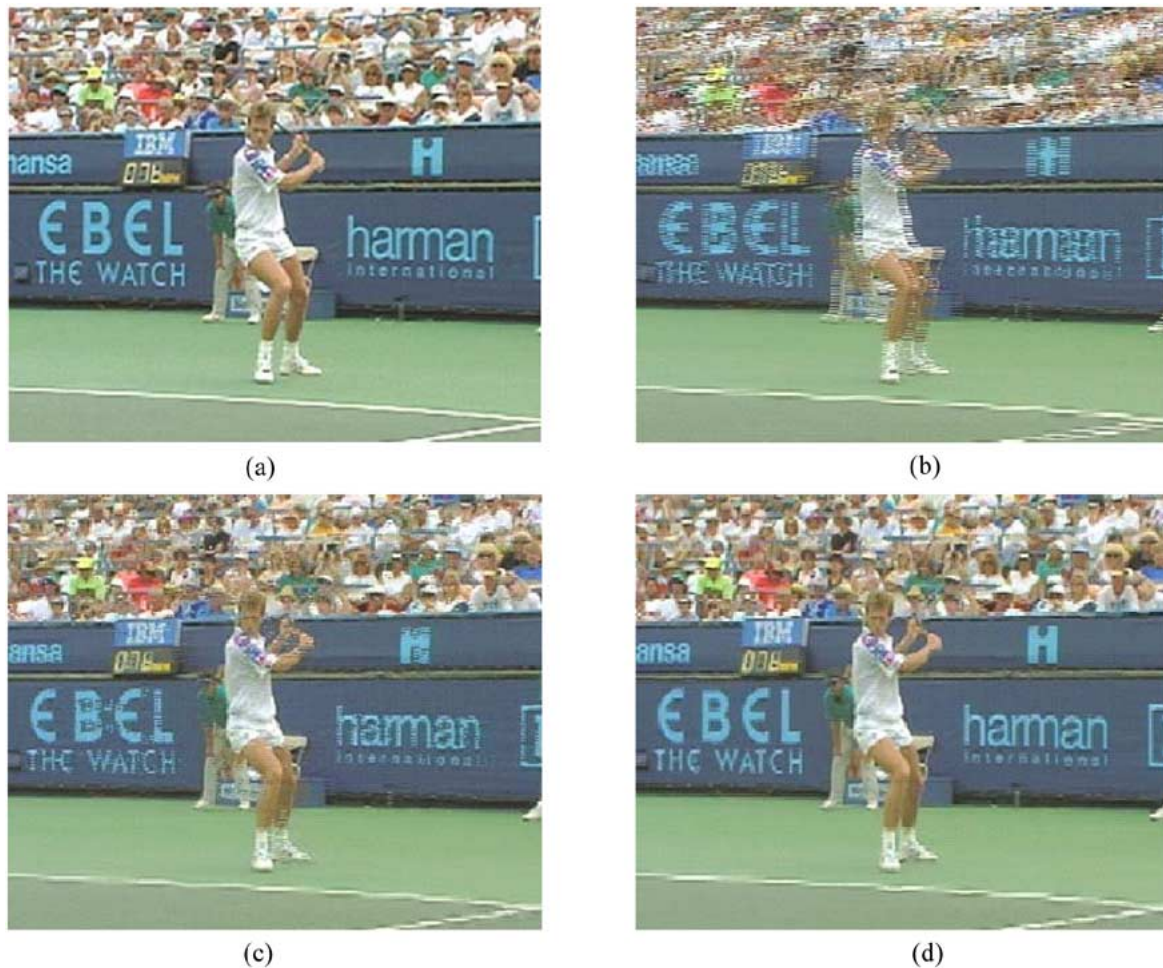


Fig. 18. Video sequence “Stefan” CIF 30 FPS. (a) Original frame. (b) Directly merged frame. (c) MA method. (d) Proposed method.

### C. Comparisons of Subject View

Three subject views of video sequences are shown in Fig. 18, Fig. 19, and Fig. 20. The comparison of the MA de-interlacing methods to the global MC ones is shown in Fig. 18. The comparison of the local MC de-interlacing and the adaptive global/local MC de-interlacing is shown in Fig. 19. And the comparison of various local MC de-interlacing methods is shown in Fig. 20.

The original frame, interlaced frame, 2-field MA de-interlaced frame, and the proposed de-interlaced frame of “Stefan” are shown in Fig. 18(a)–(d), respectively. The camera is panning in the Stefan sequence, and there are severe feathering effects over the whole interlaced frame. Most parts of the picture are de-interlaced with common MA de-interlacing methods, but there are still many feathering defects on Stefan’s body and the words on the wall. But for the proposed adaptive GMC/MC method, feathering defect is no more seen in the picture. Because the camera panning problem has been solved by global motion compensation, the processed video sequences possess high-quality subject view.

The original frame, directly merged frame, MA de-interlaced frame, the proposed 4-field local motion compensation with motion adaptation de-interlaced frame, and the proposed adaptive

GMC/MC de-interlaced frame in “Table Tennis” are shown in Fig. 19(a)–(e), respectively. The camera is zooming-out in this video sequence and there are lots of radial feathering defects on the wall. After the process of common MA methods, there are still many radial feathering defects in the picture. As for the proposed 4-field local MC de-interlacing with motion adaptation, because the scale of the adjacent fields are different, the motion estimation cannot find a suitable MV for the ball and the hand, and the block mode decision doesn’t recognize these blocks as feathering blocks. Therefore the subjective result of the proposed 4-field local MC de-interlacing with motion adaptation does not perform well. In the result of the proposed adaptive GMC/MC de-interlacing method, the reference field is scaled first for de-interlacing, so the proposed method possesses higher quality subject view in this kind of camera-zooming video sequences than conventional MA or motion-compensated methods.

In Fig. 20, the “flag” video sequence is used to test the local motion estimation/compensation in the de-interlacing methods. The flag is breezing in the air, and the feathering effect appears on the color boundaries of the flag. The directly-merged frame, the ELA de-interlaced frame, the 3DRS local MC de-interlaced frame, and the 4-field local MC de-interlaced frame are shown

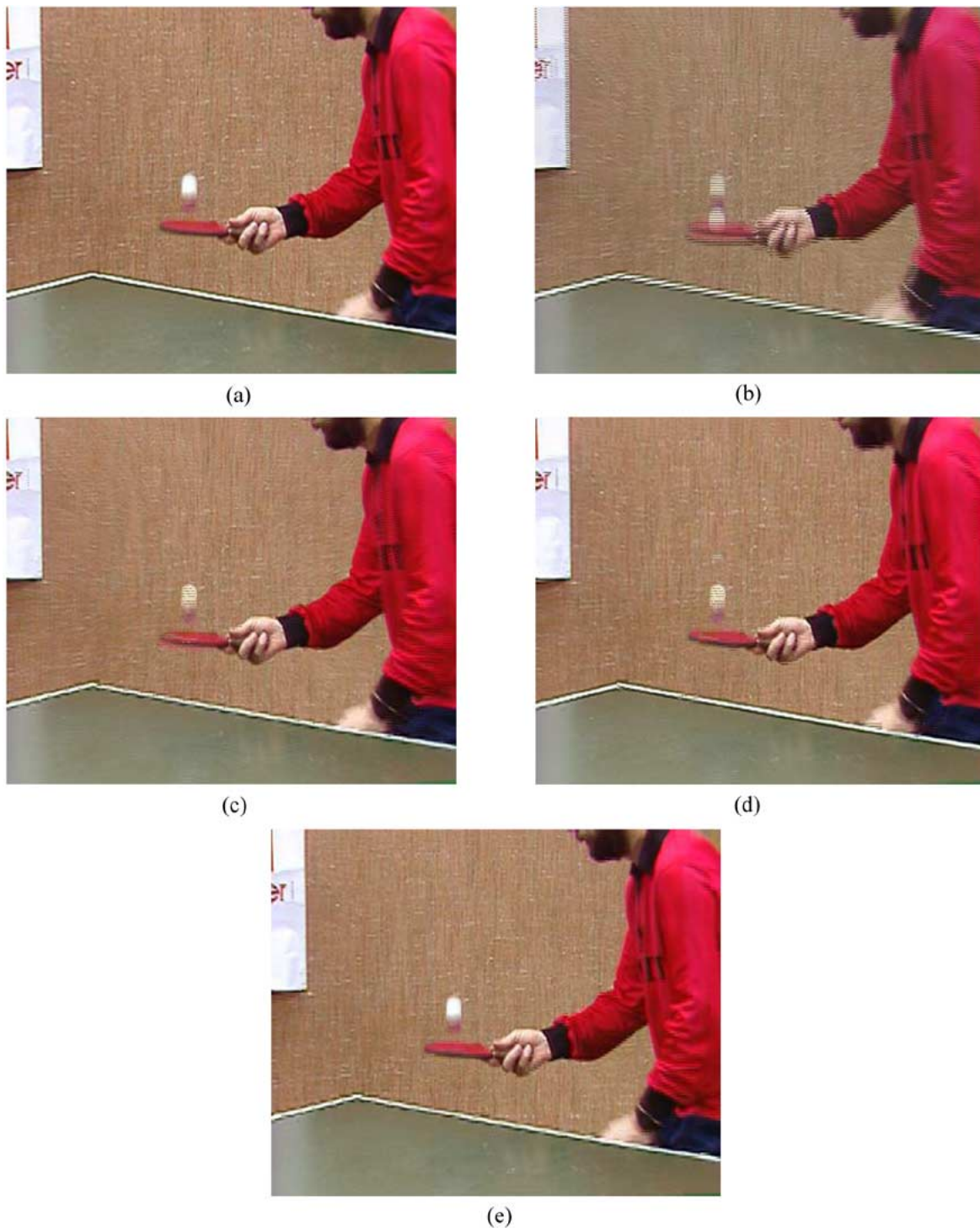


Fig. 19. Video sequence “Table Tennis” CIF 30 FPS. (a) Original frame. (b) Directly merged frame. (c) MA method. (d) Local MC method. (e) Proposed adaptive GMC/MC method.

in Fig. 20(a)–(d). There’s no intrafield de-interlacing combined with the MC de-interlacing methods in Fig. 20. The subjective view tells that the 4-field local motion estimation estimates smooth MVs for de-interlacing. Even without intrafield de-interlacing, the result of the 4-field local MC de-interlacing is suitable for de-interlaced output.

The sequence “Pendulum” emphasizes texture preservation and edge sharpness after de-interlacing. The original sequence,

as shown in Fig. 21(a)–(b), is divided into three parts. The most important is the oscillating pendulum: it swings back and forth between the left and right margin of the screen. In the directly-merged frame, the feathering effect appears at the motion area of the pendulum. The second part is the title “OK,” which is composed of an “O” existing only in the even fields and a “K” existing only in the odd fields. The third part of the sequence is the logo rotating at the lower-right-hand corner of the screen.

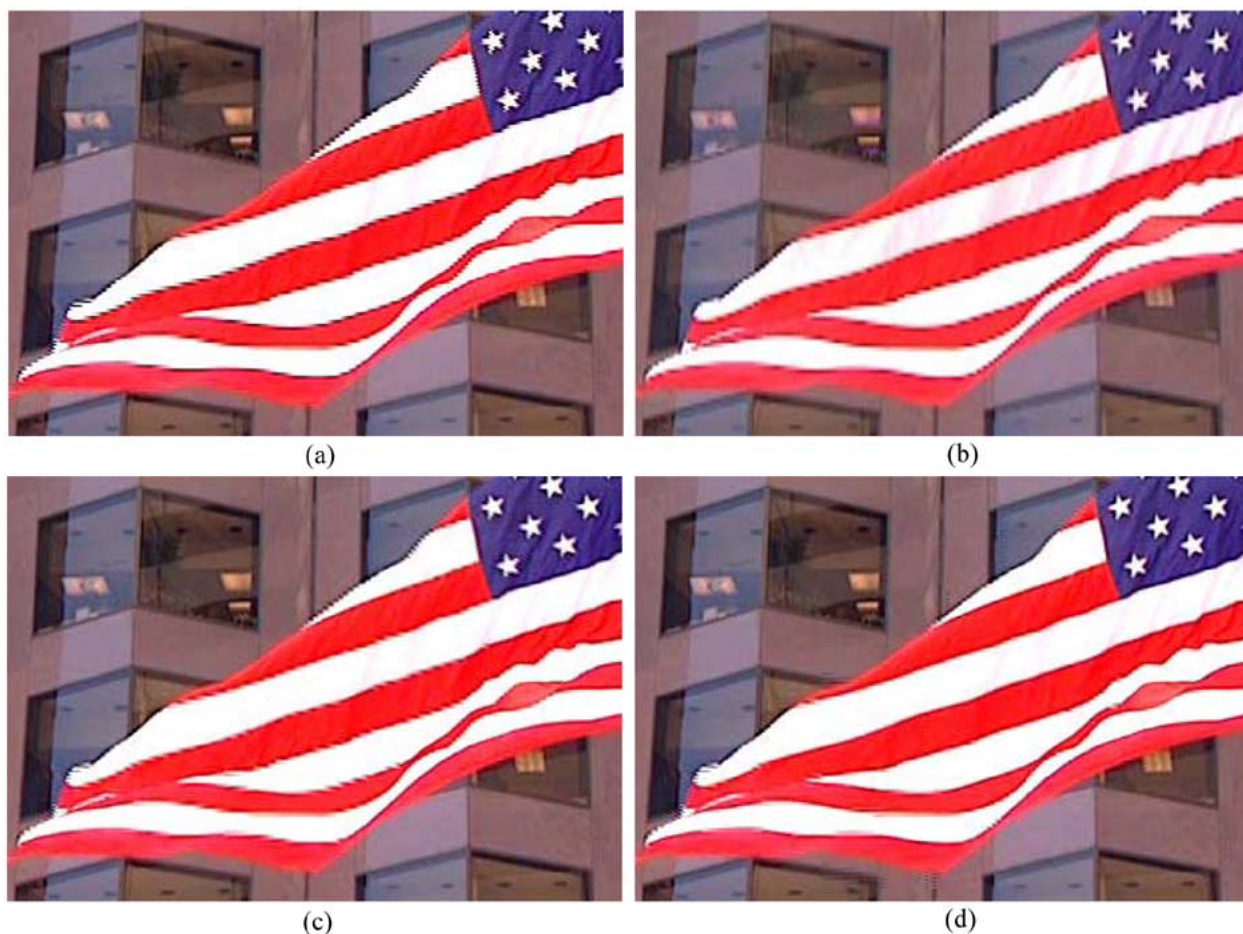


Fig. 20. Video sequence “Flag” CCIR601 60 FPS. (a) Directly-merged frame. (b) ELA de-interlaced frame. (c) 3DRS local MC de-interlaced frame. (d) Proposed 4-field local compensated de-interlaced frame.

The rotational logo is a nonrigid body, and the real appearance of nonrigid bodies are often lost in motion-compensated de-interlacing while the MA de-interlacing reconstructs the nonrigid bodies very well. The original two fields and the directly combined frame are shown in Fig. 21(a)–(b), and Fig. 22(a), respectively.

The edges on the pendulum are able to be recognized after intrafield de-interlacing is adopted. But the words “O” and “K” disappear alternatively because they exist only in different parity fields, as shown in Fig. 21(c) and Fig. 21(d), respectively.

The result of the two-field same-parity motion detection [15] is shown in Fig. 22(b). Although the words “OK” are successfully preserved, the feathering effect still exists in the shaft of the pendulum.

For intrafield de-interlacing, the bilinear interpolation and the proposed block-based directional edge interpolation show different sharpness. The result of bilinear interpolation is shown in Fig. 22(c), where alias edges are seen. The result of the proposed method is shown in Fig. 22(d). The edges are much sharper and straighter than that produced by the bilinear interpolation.

The result of the proposed method is shown in Fig. 22(e). The edges of the pendulum and the logo are sharp and straight, and the letters “O” and “K” are displayed correctly in the even field and odd field, respectively. The proposed

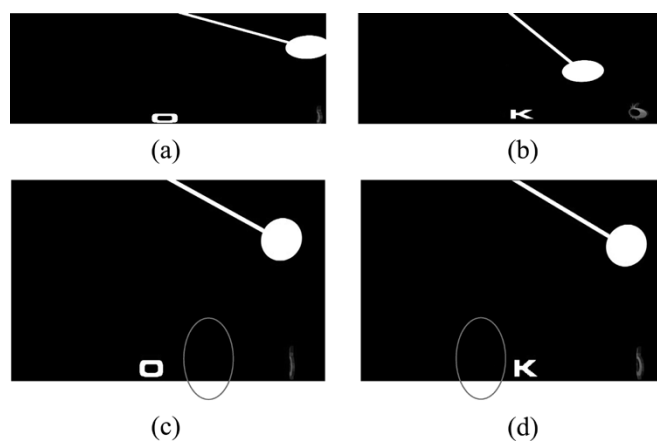


Fig. 21. (a) Original even field. (b) Original odd field. (c) Letter K disappears by intrafield interpolation in the even field. (d) Letter O disappears by intrafield interpolation in the odd field.

method has overcome the problems of all other de-interlacing methods.

## V. CONCLUSION

A de-interlacing algorithm using adaptive global and local motion estimation/compensation is presented in this paper.

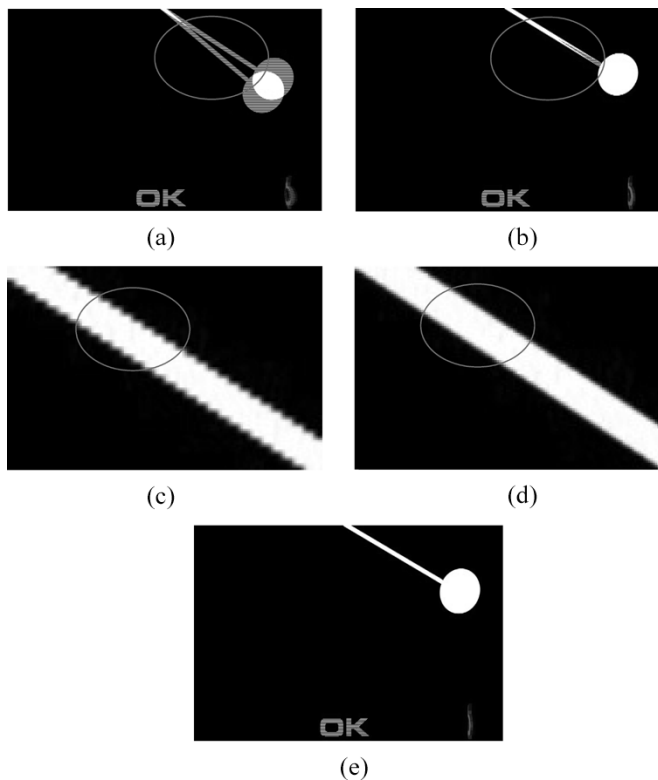


Fig. 22. (a) Feathering effect occurs when merged directly. (b) Feathering effect is introduced by same-parity 2-field detection. (c) Jagged effect caused by bilinear interpolation. (d) Straight line obtained by proposed directional edge interpolation. (e) Result obtained by the proposed adaptive GMC/MC de-interlacing method.

It consists of the global and local motion estimation/compensation, 4-field motion adaptation, the block-based directional edge interpolation, and the GMC/MC/MA block mode decision module. All defects such as jagged effects, blurring, line-crawling, and feathering are suppressed lower than traditional methods. Moreover, the true motion information is extracted accurately by the 4-field motion estimation and global motion information. The experimental results show that the proposed algorithm achieves higher PSNR than previous works, and attain the best quality of subject view.

## REFERENCES

- [1] P. L. Swan, "Method and Apparatus for Providing Interlaced Video on a Progressive Display," U.S. Patent 5 864 369, Jan. 26, 1999.
- [2] H. Jiang and C. Moloney, "A new direction adaptive scheme for image interpolation," in *IEEE Int. Conf. Image Processing*, 2002, pp. 369–372.
- [3] T. Blu, P. Thevenaz, and M. Unser, "How a simple shift can significantly improve the performance of linear interpolation," in *IEEE Int. Conf. Image Processing*, 2002, pp. 377–380.
- [4] C. J. Kuo, C. Liao, and C. C. Lin, "Adaptive interpolation technique for scanning rate conversion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 317–321, Jun. 1996.
- [5] T. Doyle, "Interlaced to sequential conversion for EDTV applications," in *Proc. 2nd Int. Workshop Signal Processing for HDTV*, Feb. 1988, pp. 412–430.
- [6] Y.-L. Chang, S.-F. Lin, and L.-G. Chen, "Extended intelligent edge-based line average with its implementation and test method," in *Proc. 2004 IEEE Int. Symp. Circuits and Systems (ISCAS 2004)*, Vancouver, Canada, May 2004, pp. 341–344.

- [7] M. Zhao and G. de Haan, "Intra-field de-interlacing with advanced up-scaling methods," in *Proc. ISCE*, Reading, U.K., Sep. 2004, pp. 315–319.
- [8] Y.-T. Kim, S.-H. Kim, and S.-W. Park, "Motion decision feedback de-interlacing algorithms," in *IEEE Int. Conf. Image Processing*, 2002, pp. 397–400.
- [9] S.-K. Kwon, K.-S. Seo, J.-K. Kim, and Y.-G. Kim, "A motion-adaptive de-interlacing method," *IEEE Trans. Consumer Elec.*, vol. 38, no. 3, pp. 145–150, Aug. 1992.
- [10] J. Kovacevic, R. J. Safranek, and E. M. Yeh, "De-interlacing by successive approximation," *IEEE Trans. Image Process.*, vol. 6, no. 2, pp. 339–344, Feb. 1997.
- [11] A. J. Patti, M. I. Sezan, and A. M. Tekalp, "Robust methods for high quality stills from interlaced video in the presence of dominant motion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 2, pp. 328–342, Apr. 1997.
- [12] D. Van De Ville, B. Rogge, W. Philips, and I. Lemahieu, "De-interlacing using fuzzy-based motion detection," in *Proc. 3rd Int. Conf. Knowledge-Based Intelligent Information Engineering Systems*, 1999, pp. 263–267.
- [13] D. Han, C.-Y. Shin, S.-J. Choi, and J.-S. Park, "A motion adaptive 3-D de-interlacing algorithm based on the brightness profile pattern difference," *IEEE Trans. Consum. Electron.*, vol. 45, no. 3, pp. 690–697, Aug. 1999.
- [14] C. Sun, "De-interlacing of video images using a shortest path technique," *IEEE Trans. Consum. Electron.*, vol. 47, no. 2, pp. 225–230, May 2001.
- [15] S.-F. Lin, Y.-L. Chang, and L.-G. Chen, "Motion adaptive interpolation with morphological operation and 3:2 pull-down recovery for de-interlacing," in *IEEE Int. Conf. Multimedia Expo*, Lausanne, Switzerland, Aug. 2002, CD-ROM.
- [16] —, "Motion adaptive de-interlacing with horizontal motion detection and ELA with median," in *Proc. 2003 IEEE Int. Symp. Circuits and Systems (ISCAS 2003)*, Bangkok, Thailand, May 2003, pp. 696–699.
- [17] A. Nguyen and E. Dubois, "Spatio-temporal adaptive interlaced-to-progressive conversion," in *Signal Processing of HDTV*, E. Dubois and L. Chiariglione, Eds. New York: Elsevier, 1993, pp. 749–756.
- [18] C. Ryu and S. P. Kim, "De-interlacing using motion compensated local spectra," in *Proc. Record 29th Asilomar Conf. Signals, Systems and Computers*, vol. 2, 1996, pp. 1394–1397.
- [19] K. Sugiyama and H. Nakamura, "A method of de-interlacing with motion compensated interpolation," *IEEE Trans. Consum. Electron.*, vol. 45, no. 3, pp. 611–616, August 1999.
- [20] D. Van de Ville, W. Philips, and I. Lemahieu, "Motion compensated de-interlacing for both real time video and still images," in *Proc. Int. Conf. Image Processing*, vol. 2, 2000, pp. 680–683.
- [21] G. De Haan, "Motion compensated de-interlacing, noise reduction, and picture rate conversion," *Trans. Consum. Electron.*, vol. 45, no. 3, pp. 617–624, Aug. 1999.
- [22] —, "De-interlacing of video data," *IEEE Trans. Consum. Electron.*, vol. 43, no. 3, pp. 819–825, Aug. 1997.
- [23] R. J. Schutten and G. de Haan, "Real-time 2–3 pull-down elimination applying motion estimation/compensation in a programmable device," *IEEE Trans. Consum. Electron.*, vol. 44, no. 3, pp. 930–938, Aug. 1998.
- [24] G. de Haan and P. W. A. C. Biezen, "An efficient true-motion estimator using candidate vectors from a parametric motion model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 1, pp. 85–91, Feb. 1998.
- [25] Y.-Y. Jung, B.-T. Choi, Y.-J. Park, and S.-J. Ko, "An effective de-interlacing technique using motion compensated interpolation," *IEEE Trans. Consum. Electron.*, vol. 46, no. 3, pp. 460–466, Aug. 2000.
- [26] G. de Haan and J. Kettenis, "'System-on-Silicon' for high quality display format conversion and video enhancement," in *Proc. ISCE'02*, Sep. 2002, pp. E1–E6.
- [27] B. Martins and S. Forchhammer, "A unified approach to restoration, de-interlacing and resolution enhancement in decoding MPEG-2 video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 9, pp. 803–811, Sep. 2002.
- [28] Y.-L. Chang, P.-H. Wu, S.-F. Lin, and L.-G. Chen, "Four field local motion compensated de-interlacing," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP 2004)*, Montreal, Canada, May 2004, pp. 253–256.
- [29] 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, no. 9, pp. 1138–1140, Sep. 2004.
- [30] C.-Y. Chen, S.-Y. Chien, W.-M. Chao, Y.-W. Huang, and L.-G. Chen, "Hardware architecture for global motion estimation for MPEG-4 advanced simple profile," in *Proc. IEEE Int. Symp. Circuits Systems*, vol. 2, May 2004, pp. 301–304.



**Yu-Lin Chang** was born in Taipei, Taiwan, R.O.C., in 1981. He received the B.S. degree from the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C., in 2002. He is currently working toward the Ph.D. degree at the Graduate Institute of Electrical Engineering, National Taiwan University.

His research interests include video coding technology, digital TV, and associated VLSI architectures.



**Shyh-Feng Lin** was born in I-Lan, Taiwan, R.O.C., in 1974. He received the B.S. degree from the Department of Electrical Engineering, Tatung Institute of Technology, Taipei, Taiwan, R.O.C., in 1996. He received the M.S. and Ph.D. degrees from the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C., in 1998, and in 2004, respectively.

He joined AVermedia Inc., Taiwan, R.O.C., in 2004. His research interests include video coding technology, video processing, digital TV, and associated VLSI architectures.

associated VLSI architectures.



**Ching-Yeh Chen** was born in Taipei, Taiwan, R.O.C., in 1980. He received the B.S. degree from the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C., in 2002. He is currently working toward the Ph.D. degree at the Graduate Institute of Electronics Engineering, National Taiwan University.

His research interests include intelligent video signal processing, motion estimation, scalable video coding, and associated VLSI architectures.



**Liang-Gee Chen** (S'84-M'86-SM'94-F'01) received the B.S., M.S., and Ph.D. degrees in electrical engineering from National Cheng Kung University, Tainan, Taiwan, R.O.C., in 1979, 1981, and 1986, respectively.

In 1988, he joined the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C. During 1993–1994, he was a Visiting Consultant in the DSP Research Department, AT&T Bell Laboratories, Murray Hill, NJ. In 1997, he was a Visiting Scholar in the Department of Electrical

Engineering, University of Washington, Seattle. Currently, he is a Professor at National Taiwan University. His current research interests are DSP architecture design, video processor design, and video coding systems.

Dr. Chen has served as an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY since 1996, as Associate Editor of the IEEE TRANSACTIONS ON VLSI SYSTEMS since 1999, and as Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART II: EXPRESS BRIEFS since 2000. He has been the Associate Editor of the *Journal of Circuits, Systems, and Signal Processing* since 1999, and a Guest Editor for the *Journal of Video Signal Processing Systems*. He is also the Associate Editor of the PROCEEDINGS OF THE IEEE. He was the General Chairman of the 7th VLSI Design/CAD Symposium in 1995 and of the 1999 IEEE Workshop on Signal Processing Systems: Design and Implementation. He is the Past-Chair of Taipei Chapter of IEEE Circuits and Systems (CAS) Society, and is a Member of the IEEE CAS Technical Committee of VLSI Systems and Applications, the Technical Committee of Visual Signal Processing and Communications, and the IEEE Signal Processing Technical Committee of Design and Implementation of SP Systems. He is the Chair-Elect of the IEEE CAS Technical Committee on Multimedia Systems and Applications. During 2001–2002, he served as a Distinguished Lecturer of the IEEE CAS Society. He received Best Paper Awards from the R.O.C. Computer Society in 1990 and 1994. Annually from 1991 to 1999, he received Long-Term (Acer) Paper Awards. In 1992, he received the Best Paper Award of the 1992 Asia-Pacific Conference on Circuits and Systems in the VLSI design track. In 1993, he received the Annual Paper Award of the Chinese Engineers Society. In 1996 and 2000, he received the Outstanding Research Award from the National Science Council, and in 2000, the Dragon Excellence Award from Acer. He is a Member of Phi Tan Phi.