東海學報 49 卷(2012):34-42
中華民國一百零一年二月出版

The Architecture Design of Motion Adaptive De-interlacing with Multi-directional Motion Detection

Chishyan Liaw
Associate Professor, Department of Computer Science and Information Engineering, Tunghai University
liaw@thu.edu.tw

Chung-chi Lin
Associate Professor, Department of Computer Science and Information Engineering, Tunghai University
cclin@thu.edu.tw

Ching-tsorng Tsai
Professor, Department of Computer Science and Information Engineering, Tunghai University
cttsai@thu.edu.tw

Abstract

This paper presents a high-performance architecture of motion adaptive de-interlacing with multi-directional motion detection. Object movement happens quite often in film broadcasting and normally they move horizontally, vertically, or diagonally. The movements tend to destabilize the quality of performance such as jagged effect, blurred effect, and artifacts effect, while de-interlacing technique is utilized. In the proposed method, de-interlacing begins with object motion detection, which is to ensure that the interfield information is used precisely. To improve the quality of de-interlacing, the factors of horizontal, vertical, or diagonal motions are taken into account when de-interlacing techniques are applied. Our proposed technique provides a simple hardware architecture design, low computation cost and is easy to implement. Based on our technique, the high-speed VLSI architecture has been successfully designed and implemented with TSMC 0.13μm CMOS technology. The simulation results demonstrate that the high performance architecture of motion adaptive de-interlacing at 154MHz with 14278 gates in a 420×420μm^2 chip is able to process de-interlacing for Full HD display device in real-time.

Keywords: *de-interlacing, interpolation, horizontal and vertical motions, VLSI*

1. Introduction

The traditional NTSC system is broadcasted in an interlaced scan technique. It reduces both the bandwidth required and large area flicker but it also creates undesirable visual artifacts and makes the lines flicker and crawl. On the other hand, modern display devices like LCD displays, PC monitors, PDP displays and HDTV require a progressive scan format. Thus, video de-interlacing techniques, which convert the format of interlaced images to progressive images, are important today in improving the quality of display.

Numerous de-interlacing techniques have been proposed for interlaced to progressive scan conversion [1-18], which can be roughly classified into intrafield de-interlacing, interfield de-interlacing, motion adaptive de-interlacing, and motion compensated de-interlacing. Intrafield de-interlacing [1-6] uses a single field to reconstruct one complete frame. The edge-based line average, ELA [1], method is widely used. This method provides good results when edge can be correctly estimated. Nevertheless, it has shortcomings when wrong edge information is used, and it is sensitive to small pixel values. Park [2] proposed an edge dependent interpolation, EDI, algorithm based on a horizontal edge pattern. The EDI algorithm has a visually good performance in intrafield de-interlacing; however, it involves a more complex computation. Interfield de-interlacing [7-10] generates a full progressive frame by directly merging two consecutive fields. Normally the video quality is better than that of intrafield de-interlacing in static area, but a line-crawling effect occurs in the motion area. Motion adaptive de-interlacing [8-14] has the advantages of both intrafield de-interlacing and interfield de-interlacing. If non-motion is detected, interfield de-interlacing is able to present a pleasing resolution with low computational complexity; otherwise, intrafield de-interlacing is used. Motion compensated de-interlacing [8,9], [14-19] is to represent a macroblock searching for a most similar block in the two successive even or odd fields and calculates its motion vectors to form a new field. However, this approach is more complex to implement and it is difficult to obtain good results without reliable motion estimation.

The motion adaptive de-interlacing techniques are capable of improving the quality of the visual results. Most of them focus on the static or horizontal motion detection only; nevertheless, their performances are also affected by vertical motion. The vertical motion tends to destabilize the quality of performance while de-interlacing technique is utilized. It either generates a jagged effect, blurring effect, or artifacts effect. This paper presents the VLSI design of the proposed motion adaptive de-interlacing method which takes both horizontal and vertical motions into account. Our proposed technique provides a simple hardware architecture design with low computation cost that is easy to implement in real-time hardware. The core consists of intrafield de-interlacing by median edge dependent interpolation, Median EDI, and interfield de-interlacing with horizontal and vertical motions detection.

2. The Proposed Method

In the proposed method, the first stage of de-interlacing is motion detection, which detects horizontal motion, downward vertical motion, and upward vertical motion. If the motion is detected, then the interfield information is used for interpolation according to different directional motions; otherwise, intrafield de-interlacing is used for interpolation by Median EDI.

2.1 Multi-directional Motion Detection

Let $F_{n-2}(i,j)$, $F_{n-1}(i,j)$, $F_n(i,j)$, and $F_{n+1}(i,j)$ denote as the field before previous field, previous field, current field, and next field, respectively, where the

two dimensional spatial indices (i,j) are i=1, …, W and j=1, …, H; and W and H are the width and the height of frame, respectively. In horizontal motion detection, five directional temporal interpolations, as shown in Fig. 1, are used to achieve higher resolution than that of interfield interpolation. The method uses a 1×3 block to find the absolute difference value of block matching between F_{n-1} and F_{n+1} for five directional temporal interpolation. If the minimum difference of block matching is smaller than the threshold Th, the temporal prediction of horizontal motion will be adopted.

Normally, object moves horizontally, vertically, or diagonally. We separate vertical motion to be the upward, downward, and diagonal motions so that ten directional temporal interpolations are used to achieve higher resolution than that of interfield interpolation. In Fig. 2, five directional temporal interpolations of downward 90° motion and downward diagonal 30°, 45°, 135°, and 150° motions are used. The method uses a 1×3 block to find the absolute difference value of block matching between F_{n-1} and F_{n+1} for five directional temporal interpolation. If the minimum difference of block matching is smaller than the threshold Th, the temporal prediction of downward or downward diagonal motions will be adopted. The method for the temporal prediction of upward and upward diagonal motions is processed similar to that of downward motion and downward diagonal motions detection.

2.2 Intrafield De-interlacing using Median EDI

Edge dependent interpolation [2] is a technique of intrafield de-interlacing but with some artifacts occurring due to erroneous detection in a non-dominant directional edge region. The Median EDI can eliminate the artifacts by interpolating the missing pixels according to the classification of the edge region.

In current frame, the value of interpolated pixel, $F_n(i,j)$, is decided by Median EDI method. Let $u(p)$ be the sum of the values of three consecutive pixels in row j-1, and be defined as

$$u(p) = sum\big(F_n(i+p-1, j-1), F_n(i+p, j-1), F_n(i+p+1, j-1)\big), \tag{1}$$

where p is -1, 0, or 1. Similarly, let $v(q)$ be the sum of the values of three consecutive pixels in row j+1 and be defined as

$$v(q) = sum\big(F_n(i+q-1, j+1), F_n(i+q, j+1), F_n(i+q+1, j+1)\big), \tag{2}$$

where q is -1, 0, or 1. The parameters p and q are the reference position with missing pixel. Thus, the absolute difference value of $u(p)$ and $v(q)$ pair, $diff(p,q)$, is defined as

$$diff\ (p, q) = \left|u(p) - v(q)\right| \tag{3}$$

Thus, the minimum absolute difference value, which is the pair of p' and q', can be obtained. It is defined as

$$(p', q') = \underset{-1 \le p, q \le 1}{arg\ min}\ diff\ (p, q)\ . \tag{4}$$

The estimated value for interpolated pixel is defined as

$$A = \frac{F_n(i+a, j-1) + F_n(i+b, j-1) + F_n(i+c, j+1) + F_n(i+d, j+1)}{4} \tag{5}$$

where a, b, c, and d have the relationship with p' and q' as

$$(a, b, c, d) = \begin{cases} (0,0,0,0) & if\ \ p' = q' \\ (p', p', q', q') & if\ \ p' = -q' \\ (-1,0,0,1) & if\ \ p' < q' \\ (0,1,-1,0) & if\ \ p' > q' \end{cases}. \tag{6}$$

At last, the value of interpolated pixel, $F_n(i,j)$, can be obtained by

(7)

$$F_n(i,j)=\begin{cases} A & if\ \ min\ diff(p,q)<Th \\ Median\left(F_n(i,j-1),A,F_n(i,j+1)\right) & otherwise \end{cases},$$

where a threshold parameter *Th* needs to be set in advance. This method could avoid artifacts while erroneous detection in a non-dominant directional edge region.

3. The Proposed Hardware Architecture

The proposed method utilizes the video de-interlacing technique to promote higher quality video sequences on progressive devices. Our hardware architecture provides a simple design with low computation cost and it is easy to implement in real-time. The block diagram of the hardware architecture for the proposed method is shown in Fig. 3. The images of the previous field $F_{n\text{-}1}$, the present field F_n, and the next field F_{n+1} are filed in the storages. While F_n is used to calculate the results of Median EDI, $F_{n\text{-}1}$, F_n, and F_{n+1} are applid to process the horizontal and vertical motions detection at the same time. If the motion is detected, then the interfield information is used for interpolation according to different directional motions. Otherwise, intrafield de-interlacing is used for interpolation by Median EDI.

3.1 The Hardware Architecture of Horizontal and Vertical Motions Detection

Vertical motion detection includes downward vertical detection and upward vertical detection. All of the horizontal motion detection circuit, downward vertical detection circuit, and upward vertical detection circuit are the same that except the detection positions are different. The block diagram of horizontal motion detection is depicted in Fig. 4. It uses a 1×3 block to find the absolute difference value of block matching between $F_{n\text{-}1}$ and F_{n+1} for five directional temporal interpolation. At first, fourteen pixel values which are used to obtain the absolute difference value of block matching are summed up. The adding operation sums the values of three consecutive pixels of a 1×3 block, which is shown in Fig. 1. The results are stored in 10-bit registers and they are used to find the minimum difference of block matching in the circuit of horizontal directional detection.

The circuit of horizontal direction detection is depicted in Fig. 5. The absolute difference is inputted to the comparator; then the directional difference in each direction is obtained by subtracting small number from large number. The minimum difference of block matching is found by the minimum value sorting circuit. The upper three bits of the data in the comparator in the circuit of sorting the minimum values is the decoded number and the rest of ten bits are the directional difference. After the minimum difference of block matching is found, the decoded number of the pair with the minimum difference of block matching is transferred to the decision switch 0 in Fig. 4 to choose the information of interfield interpolation which is calculated by the horizontal direction. The information is valid when the Horizontal Motion Signal is low. In other words, the information of horizontal direction is accurate when the value of minimum difference of block matching is smaller than the threshold *Th*.

The selection of the direction of vertical motion detection, shown in Fig. 6, includes downward vertical detection and upward vertical detection. The data output of vertical motion detection, which is the output of the decision switch 1, is selected by Downward Vertical Motion Signal. The interfield information of downward is applied when the signal is low; otherwise, the interfield information of upward is used. If both downward vertical detection and upward vertical detection obtain their interfield information, then the interfield information of downward is selected. The solution of whole vertical

motion detection is determined by either the Downward Vertical Motion Signal or Upward Vertical Motion Signal. The OR operation of both signals provides the decode information for the decision block of Fig. 3.

3.2 The Hardware Architecture of Median EDI

The circuit of Median EDI, shown in Fig. 7, consists of image information registers, 10-bit adders and registers, a Median EDI generator, and a decision switch 2. Images are read and saved in the information registers at first. Three consecutive pixels are added and stored in 10-bit registers. The interpolated information is obtained from current field. The results of $u(-1)$ to $u(1)$ and $v(-1)$ to $v(1)$, which shown in (1) and (2), can be found. The information is used to find intrafield information by Median EDI generator. If the current field is even, then line doubling technique is applied on the first row due to the insufficient information. If the current field is odd, the last row is duplicated. All others use the operation results of Median EDI generator.

The circuit of Median EDI generator is depicted in Fig. 8. The pair of (p', q') in (4) can be obtained from the circuit of absolute difference and minimum value sorting circuit. The upper three bits of the data format of the comparators used in minimum value sorting circuit is decoded number and the rest of ten bits are the directional difference. The decoded number, which is in upper 3-bit stands for the minimum value, will be transferred to decision switch 3 to process the operations of addition and division as indicated in (5) and (6). Thus, A of (5), can be obtained. The value will be compared to threshold Th as indicated in (7). It will go to the median value sorting circuit if A is greater than or equal to Th. The lower 10 bits of the output of the minimum value sorting circuit shown in Fig. 8 are used to compare with the threshold Th. The signal will be low if it is smaller than Th and decision switch 4 will choose the result of A. Otherwise, the signal is high and decision switch 4 will select the medium value, which is the output of median value sorting circuit. The Median EDI generator has to take all borders into account. Not only interpolating the first and last row, it also interpolates left and right borders by the bilinear method. The output of decision switch 5, Median EDI Data_out, is bilinear interpolation or the result of median value sorting depends upon whether the processing image is border.

3.3 The Decision Block

The decision block shown in Fig. 3 is depicted in Fig. 9. The output of the multiplexer, which is horizontal interfield information, vertical interfield information, or the intrafield information of Median EDI, is determined by Horizontal Motion Signal and Vertical Motion Signal, as show in Table 1. Those control signals are used to judge whether the best solution is found in horizontal or vertical direction. Horizontal interfield interpolation is adapted if Horizontal Motion Signal is low. If only Vertical Motion Signal is low, then vertical interfield interpolation is used. Otherwise, Median EDI interpolation is applied if both signals are high.

4. The Simulation Results and VLSI Implementation

To evaluate the performance of de-interlacing with horizontal and vertical motions, the threshold value Th used in the simulations is set as 48 to provide a simple and low-cost hardware implementation. The advantage of Th=48 is that in hardware implementation the comparison can be accomplished by an AND gate and an OR gate, instead of using a comparator. The software simulation of the proposed method with that of bilinear, ELA [1], EDI [2], and Lin [11] are analyzed. The test video sequences are illustrated in Table 2 and their PSNRs are shown in Table 3. It shows that the

proposed method has better performance than other methods. Fig. 10 to Fig. 12 demonstrates that our proposed method for horizontal and vertical motions detection present a more pleasing visual quality. The average PSNR of the results from our method and from different interpolation methods for various sequences are compared in Table 4. And Fig. 13 shows the PSNR performance of the various de-interlacing methods on some video sequences. The results also show that the proposed method has better average PSNR than other methods.

The proposed VLSI architecture has been verified by the simulations of Verilog description and the simulations of MATLAB. The VLSI architecture implementation is synthesized and generated by using Synopsys Design Compiler with TSMC 0.13μm CMOS standard cell library. The specifications of the VLSI architecture are illustrated in Table 5 and the chip photomicrograph is shown in Fig. 14. The VLSI architecture has been characterized in terms of power consumption by gate-level simulations. The frequency of the chip is 154MHz, which could process de-interlacing for Full HD [20] display device in real-time. The hardware comparison of proposed design with de Haan [21] is shown in Table 6, and it indicates that the proposed implementation is more efficient than the other.

5. Conclusions

In this paper we proposed a high-performance architecture of motion adaptive de-interlacing with a new scheme of horizontal and vertical motions detection. The horizontal and vertical motions detection scheme ensures that the interfield information can be used more efficiently. In the proposed method, multiple directional temporal interpolations are used to achieve higher performance of de-interlacing. The results of our experiments show that the quality of picture can be improved by horizontal and vertical motions detection algorithm. Furthermore, the simulation results also show that our proposed method presents a higher quality of video sequences than other interpolation methods. Besides, based on our technique, the high-speed VLSI architecture has been designed and implemented and it is able to process de-interlacing for Full HD display device in real-time.

References

1. M. Lee, J. Kim, J. Lee, K. Ryu, and D. Song
May 1994 “A new algorithm for interlaced to progressive scan conversion based on directional correlations and its IC design,” IEEE Trans. Consumer Electronics, vol. 40, no. 2, pp. 119-129.

2. M.K. Park, M.G. Kang, K. Nam, and S.G. Oh
Nov. 2003 ”New edge dependent deinterlacing algorithm based on horizontal edge pattern,” IEEE Trans. Consumer Electronics, vol. 49, pp. 1508-1512.

3. Y. Kim
Feb. 1996 “Deinterlacing algorithm based on sparse wide vector correlations,” Visual Communications and Image Processing, Processing of SPIE, vol. 2727, pp. 89-99.

4. H.S. Oh, Y. Kim, Y.Y. Jung, A.W. Morales, and S.J. Ko
June 2000 “Spatio-temporal edge based median filtering for deinterlacing,” in Digest of the Int. Conference on Consumer Electronics, pp. 52-53.

5. H. Yoo and J. Jeong
Nov. 2003 “Direction-oriented interpolation and its application to de-interlacing,” IEEE Trans. Consumer Electronics, vol. 48, pp. 954-962.

6. H.Y. Lee, J.W. Park, T.M. Bae, S.U. Choi, and Y.H. Ha
Nov. 2000 “Adaptive scan rate up-conversion system based on human visual characteristics,” IEEE Trans. on Consumer Electronics, vol. 46, pp. 999-1006.

7. R. Li, B. Zheng, and M.L. Liou
Feb. 2000 “Reliable motion detection/ Compensation for interlaced sequences and its applications to deinterlacing,” IEEE Trans. Circuits and Systems for Video Technology, vol. 10, pp. 23-29.

8. J. Kovacevic, R.J. Safranek, and E.M. Yeh
Feb. 1997 “Deinterlacing by successive approximation,” IEEE Trans. Image Processing, vol. 6, pp. 339-344.

9. D. Han, C. Shin, S. Choi, and J. Park
Aug.1999 “A motion adaptive 3-D deinterlacing algorithm based on the brightness profile pattern difference,” IEEE Trans. Consumer Electronics, vol. 45, pp. 690-697.

10. C. Sun
May 2001 “De-interlacing of video images using a shortest path technique,” IEEE Trans. Consumer Electronics, vol. 47, pp. 225–230.

11. S.F. Lin, Y.L. Chang, and L.G. Chen
Nov. 2003 “Motion adaptive interpolation with horizontal motion detection for deinterlacing,” IEEE Trans. Consumer Electronics, vol. 49, pp. 1256-1265.

12. S.G. Lee and D.H. Lee
Nov. 2003 “A motion-adaptive de-interlacing method using an efficient spatial and temporal interpolation,” IEEE Trans. Consumer Electronics, vol. 49, pp. 1266-1271.

13. M.J. Chen, C.H. Huang, and C.T. Hsu
Nov. 2004 “Efficient de-interlacing technique by inter-field information,” IEEE Trans. Consumer Electronics, vol. 50, no. 4, pp. 1202-1207.

14. S.H. Keller, F. Lauze, and M. Nielsen
Nov. 2008 “Deinterlacing Using Variational Methods,” IEEE Trans. Image Processing, vol. 17, no. 11, pp. 2015-2028.

15. Y.Y. Jung, B.T. Choi, Y.J. Park, and S.J. Ko
Aug. 2000 “An effective de-interlacing technique using motion compensated interpolation,” IEEE Trans. Consumer Electronics, vol. 46, pp. 460-466.

16. Kwon, K. Sohn, and C. Lee,
Feb. 2003 “Deinterlacing using directional interpolation and motion compensation,” IEEE Trans. Consumer Electronics, vol. 49, pp. 198-203.

17. K. Sugiyama and H. Nakamura
Aug. 1999 “A method of de-interlacing with motion compensated interpolation,” IEEE Trans. Consumer Electronics, vol. 45, pp. 611-616.

18. X. Gao, J. Gu, J. Li
May. 2005 “De-interlacing Algorithms Base on Motion Compensation,” IEEE Trans. Consumer Electronics, vol. 51, no. 2, pp. 589-599.

19. Y.C. Fan, H.S. Lin, A. Chiang, H.W. Tsao, and C.C. Kuo
June. 2008 “Motion Compensated Deinterlacing with Efficient Artifact Detection for Digital Television Displays,” Journal of Display Technology, vol. 4, no. 2, pp. 218-228.

20. EIA Standard
May 2002 “A DTV Profile for Uncompressed High Speed Digital Interfaces”, EIA/CEA-861 -B, pp. 19.

21. G. de Haan
Aug. 1999 “IC for Motion-Compensated De-interlacing, Noise reduction, and Picture-Rate Conversion,” IEEE Trans. Consumer Electronics, vol. 45, no. 3, pp. 617 - 624.

Horizontal Motion Signal	Vertical Motion Signal	Interpolation
Low	Low	Horizontal Data_out
Low	High	Horizontal Data_out
High	Low	Vertical Data_out
High	High	Median EDI Interpolation

Table1 The truth table of Fig. 13

Video Sequences	Frame size	Number of frame
Akiyo	176x144	300
Coastguard	176x144	300
Container	176x144	300
Hall Monitor	176x144	325
Mother and Daughter	176x144	300
Weather	360x243	300
Table Tennis	352x288	300
News	176x144	300
Silent	176x144	450

Table2 The test video sequences

Name	Bilinear	ELA	EDI	Lin [11]	Proposed
Akiyo	36.25	34.86	36.30	49.61	56.79
Coastguard	27.12	26.66	26.93	29.46	37.07
Container	26.75	26.37	26.55	34.96	40.23
Hall monitor	28.39	27.65	28.27	37.79	44.71
Mother and Daughter	32.75	32.64	32.89	46.20	51.01
Weather	24.10	25.15	25.24	37.62	39.45
Table Tennis	26.39	24.87	26.07	36.58	40.53
News	30.50	27.65	29.94	39.46	44.45
Silent	32.01	31.30	31.97	44.73	51.85

Table3 The PSNRs of Figs. 10-12

Average	Bilinear	ELA	EDI	Lin[11]	Proposed
Akiyo	36.14	35.00	36.22	41.48	45.64
Coastguard	26.57	26.23	26.42	27.06	33.16
Container	26.11	25.81	25.92	33.20	40.19
Hall Monitor	27.75	27.06	27.64	34.62	40.94
Mother and Daughter	34.01	33.79	34.13	39.39	43.79
Weather	23.29	24.21	24.21	31.00	35.96
Table Tennis	27.44	26.40	27.26	32.06	32.92
News	30.02	27.55	29.54	35.73	38.17
Silent	32.28	31.58	32.27	36.51	39.66

Table4 The average PSNR of the various de-interlacing methods on some video sequences

Process	TSMC 0.13μm CMOS
Core Area	420 × 420μm^2
Gates Count	14278
Power Consumption	11.9mW
Frequency	154MHz

Table5 The chip specifications

	de Haan [21]	Proposed
Approach	Motion Compensated De-interlacing	Motion Adaptive De-interlacing
Function	ME/MC (range 64x24) Film detector (2-2 and 2-3 pulldown)	Scene change detection Background index counter 2D-ELA interpolation
Process	CMOS 0.35μm	TSMC 0.13μm CMOS
Area	72mm^2	0.1764mm^2
Transistors Count	4x10^6	
Gates Count		14278
Power	1.2W	11.9mW
Frequency	32/27 MHz	154MHz

Table6 Comparison with other architecture

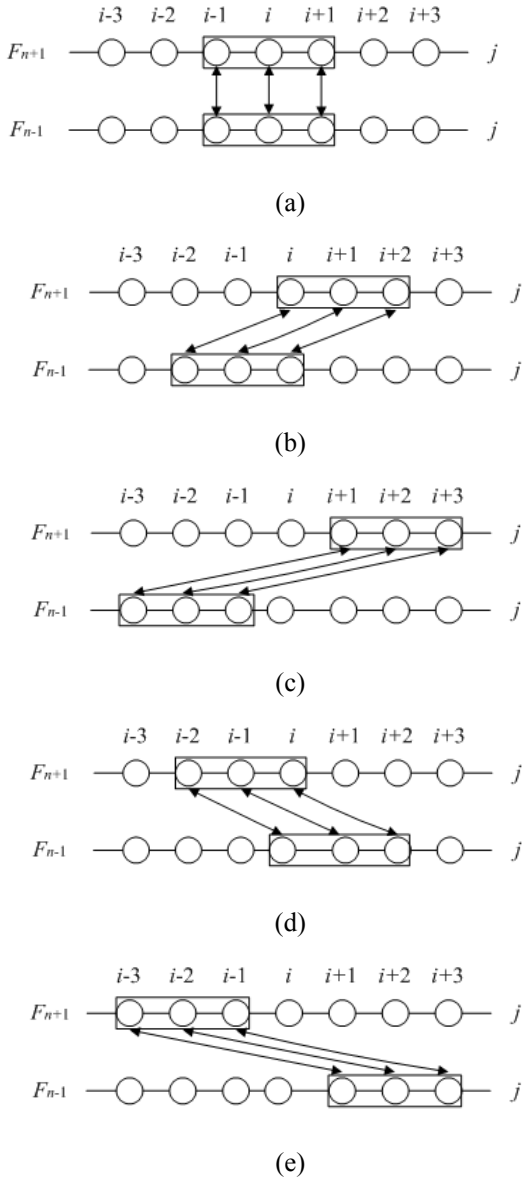


Fig. 1. Horizontal motion detection.

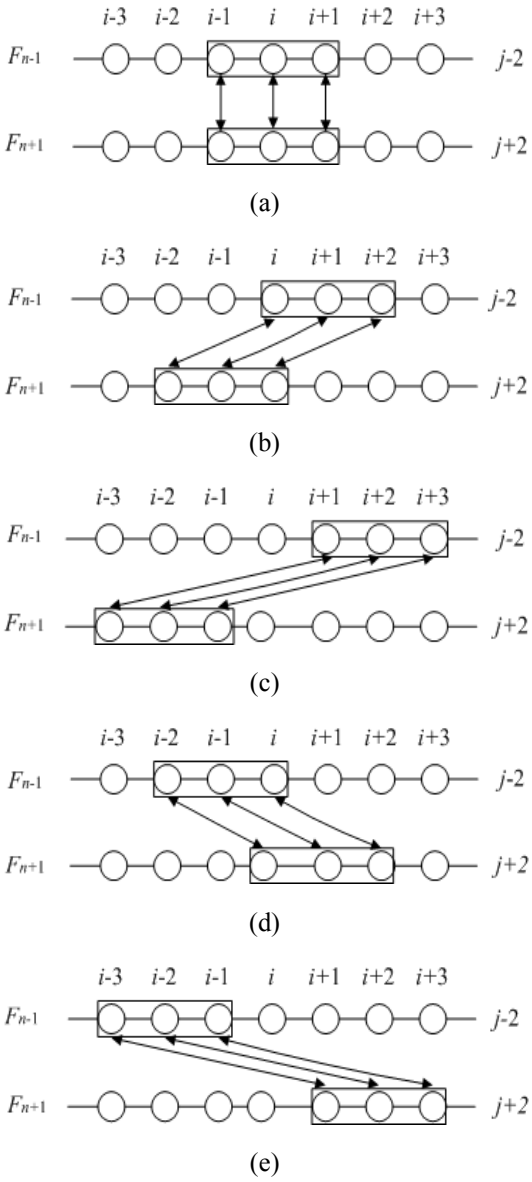


Fig. 2. Downward motion and Downward diagonal motions detection.

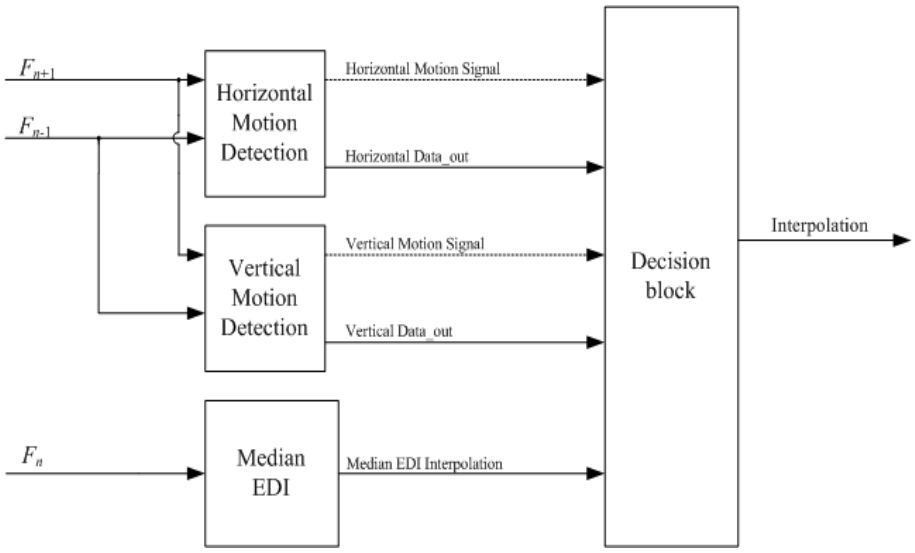


Fig. 3. Block diagram of the proposed hardware architecture. (solid-line: data flow; dash-line: control line)

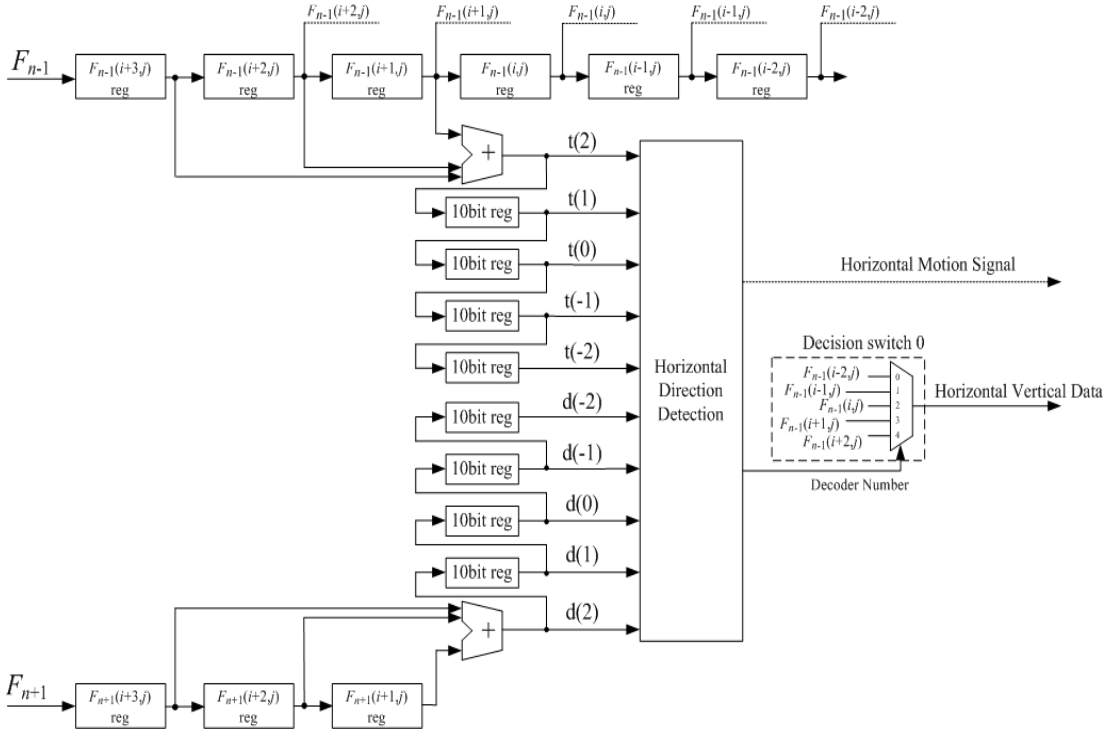


Fig. 4. The block diagram of Horizontal Motion Detection.

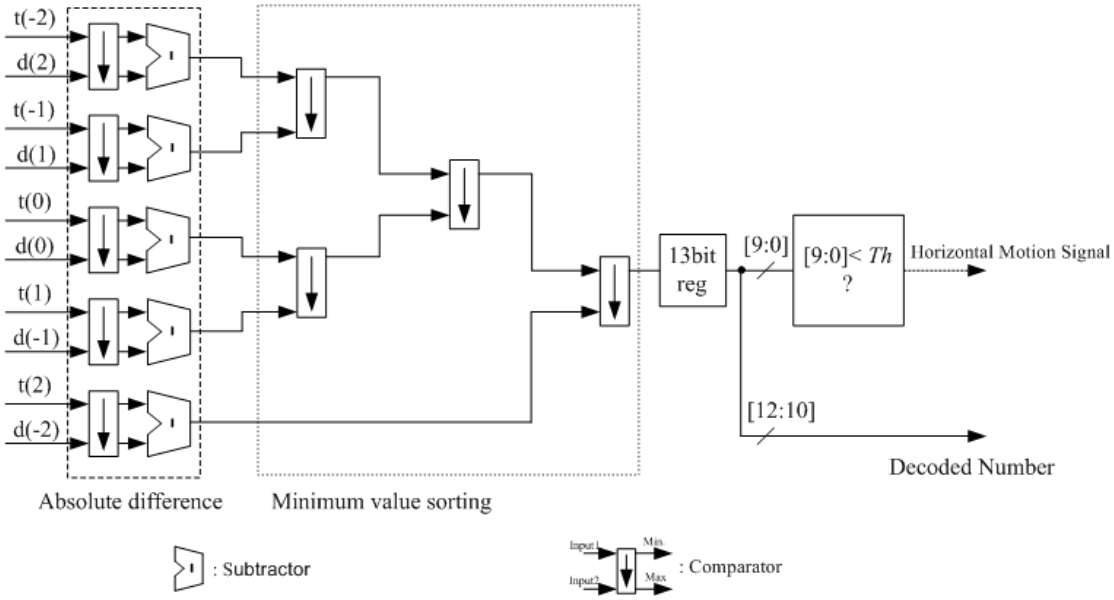


Fig. 5. The circuit of Horizontal Direction Detection.

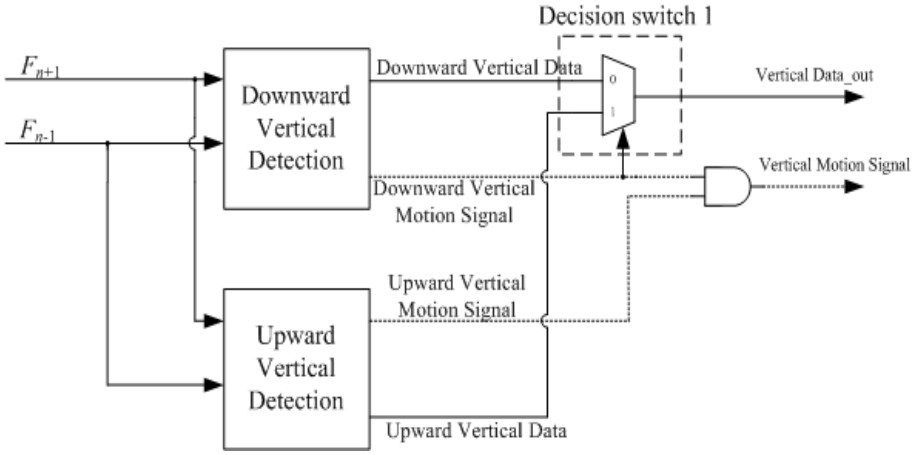


Fig. 6. The block diagram of Vertical Motion Detection.

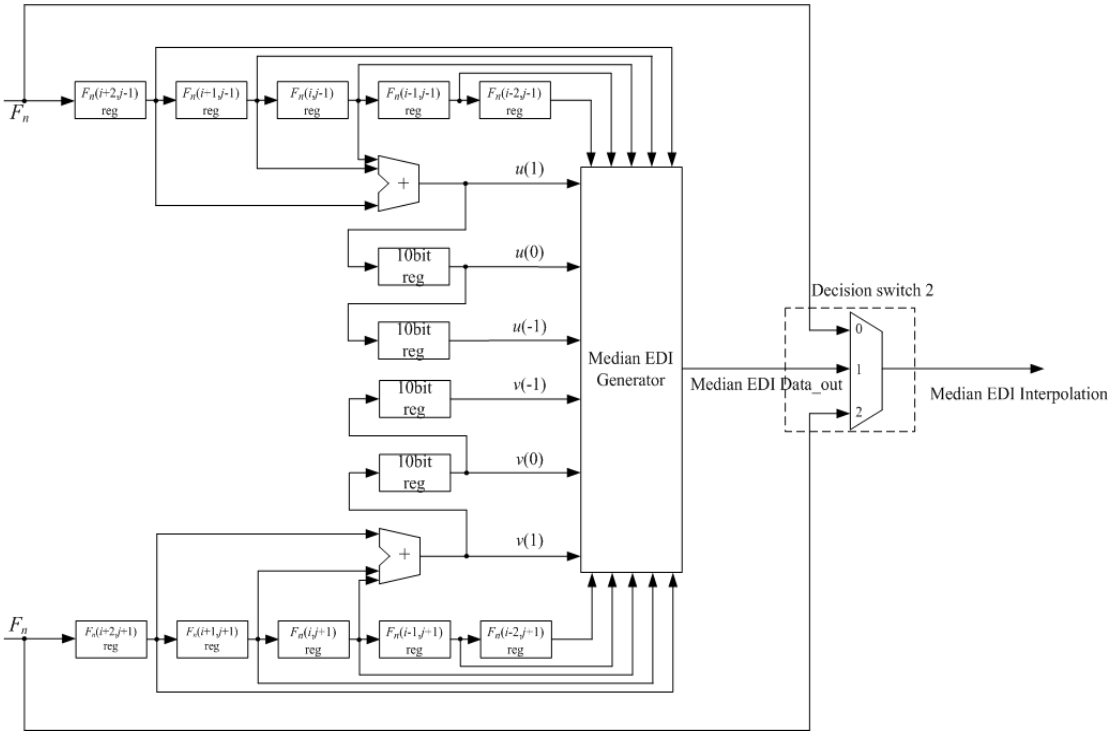


Fig. 7. The block diagram of Median EDI.

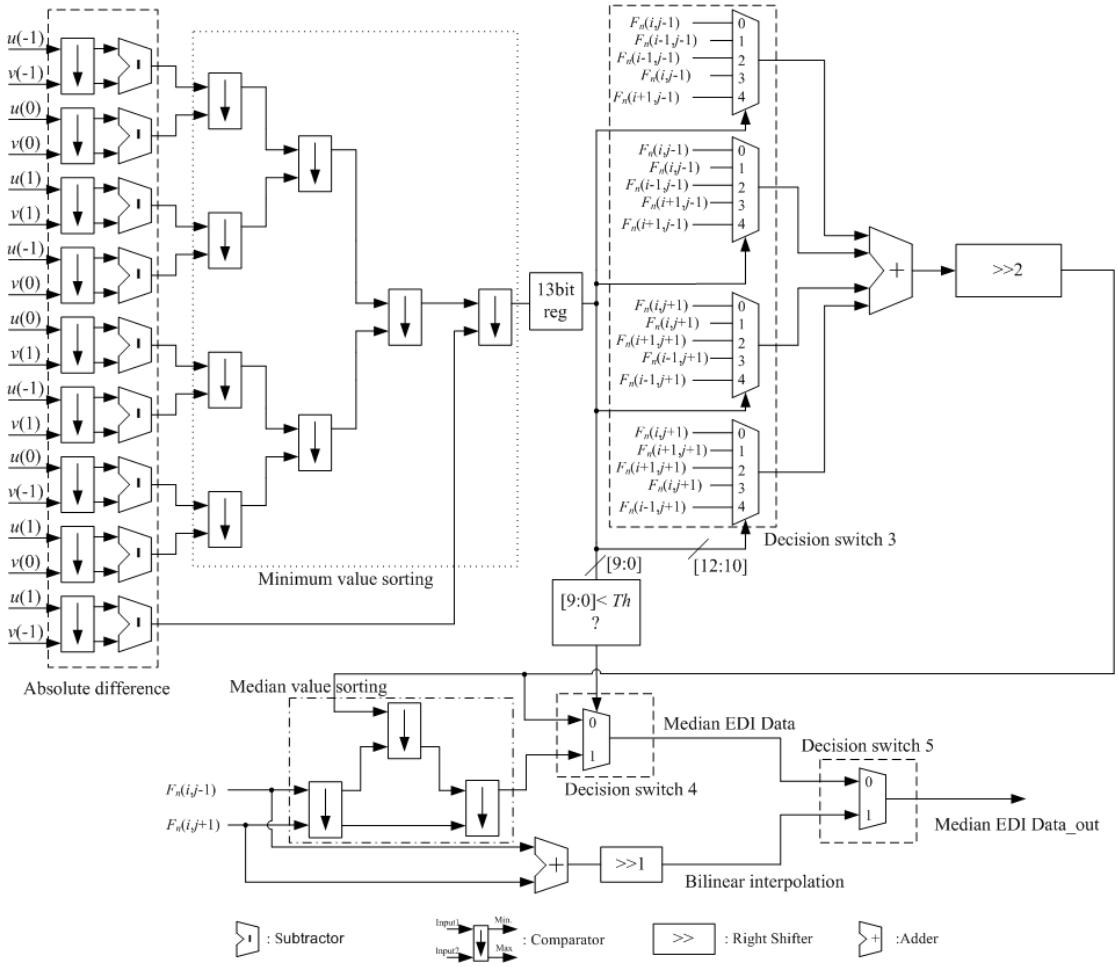


Fig. 8. The VLSI architecture of Median EDI Generator.

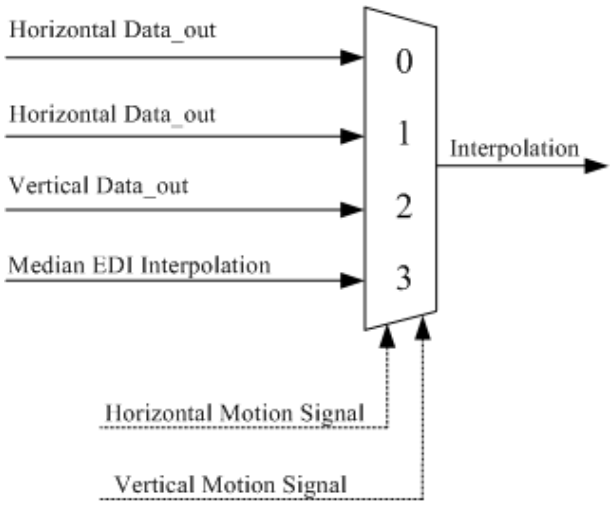
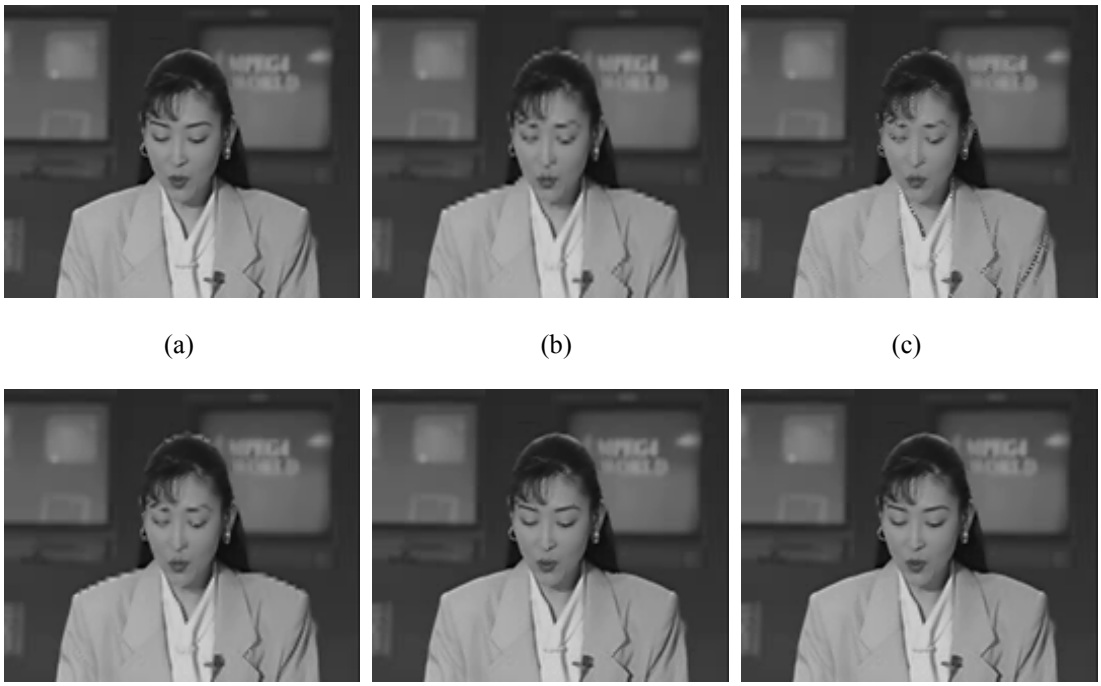


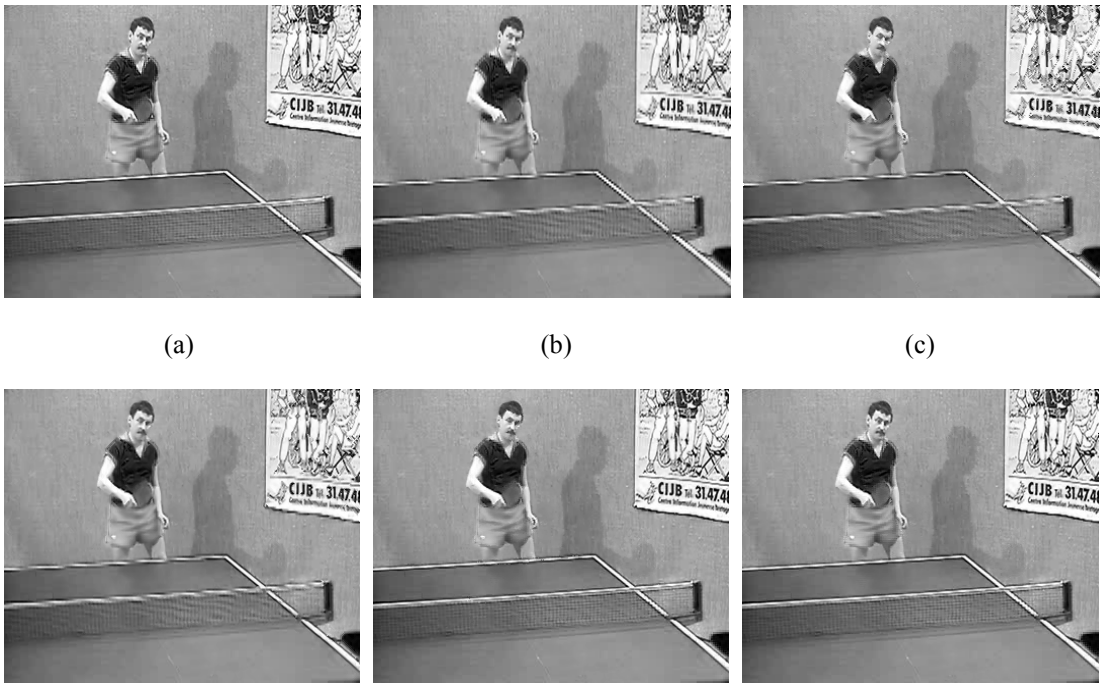
Fig. 9. The Decision Block shown in Fig. 3.



(a) (b) (c)

(d) (e) (f)

Fig. 10. Akiyo frame #50 (a)Original (b)Bilinear (c)ELA (d)EDI (e)Lin[11] (f)Poposed.



(a) (b) (c)

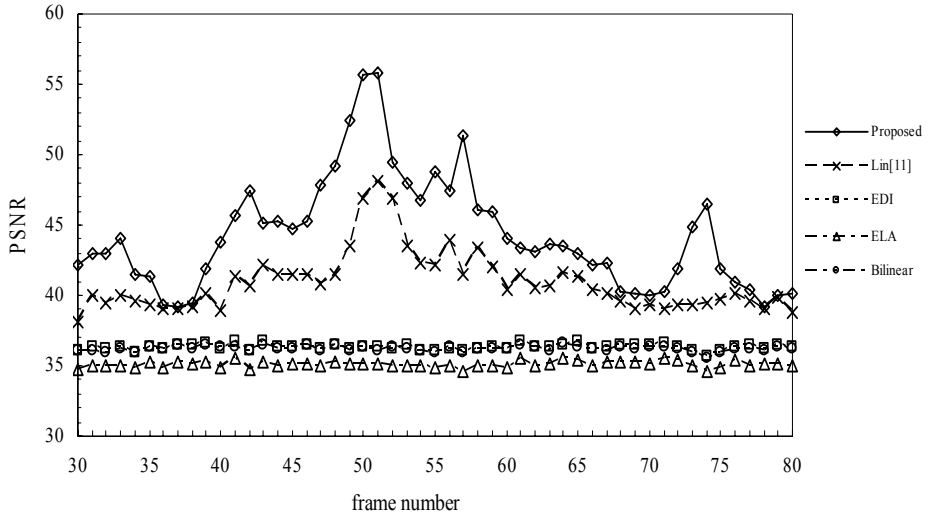
(d) (e) (f)

Fig. 11. Table Tennis frame #272 (a)Original (b)Bilinear (c)ELA (d)EDI (e)Lin[11] (f)Proposed.

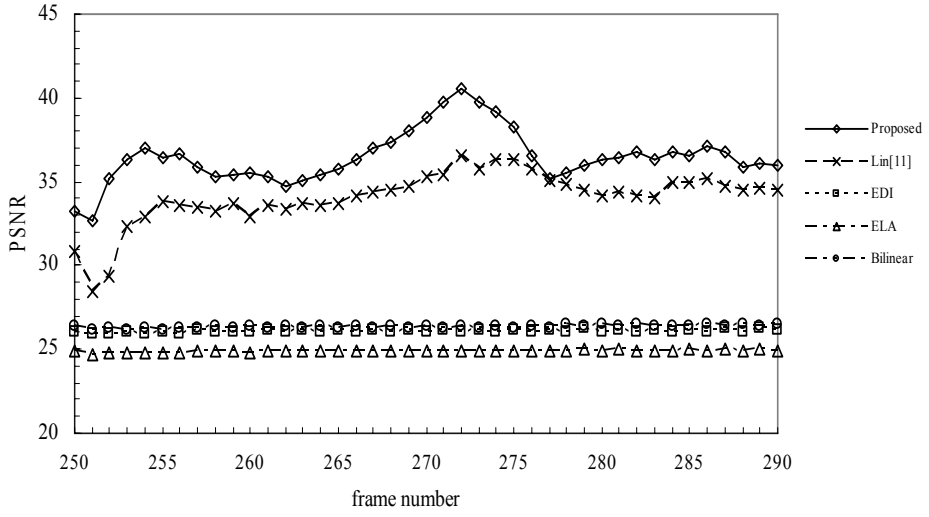
(a) (b) (c)

(d) (e) (f)

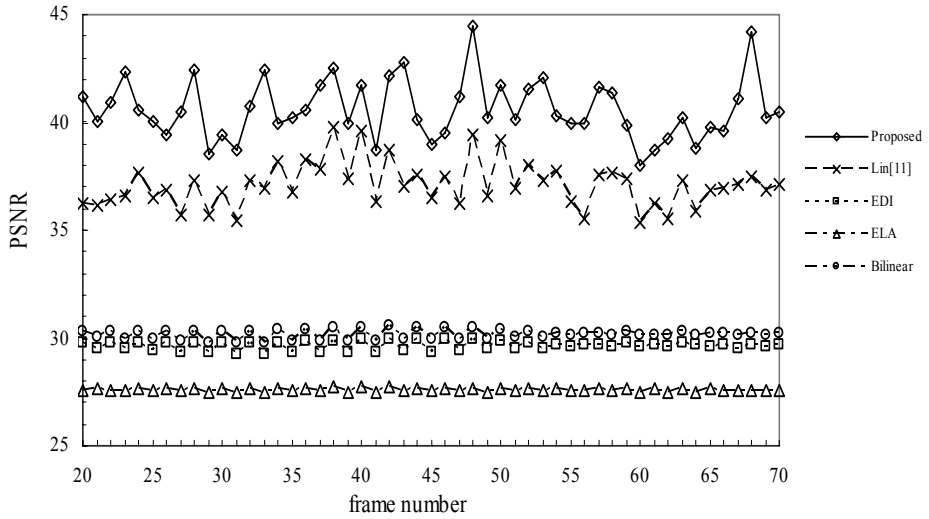
Fig. 12. News frame #48 (a)Original (b)Bilinear (c)ELA (d)EDI (e)Lin[11] (f)Proposed.



(a)



(b)



(c)

Fig. 13. PSNR performance of the various de-interlacing methods on some video sequences (a)Akiyo (b)Table Tennis (c)News.

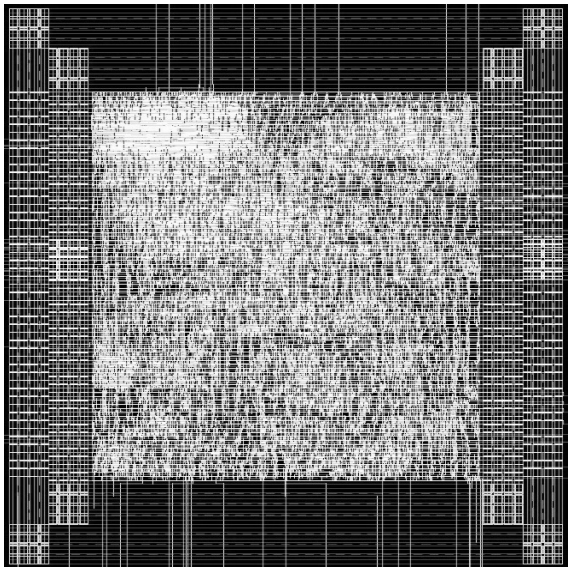


Fig. 14. Chip photomicrograph.

東海學報 49 卷(2012):34-42
中華民國一百零一年二月出版

適應多方向移動偵測的解交錯硬體架構

廖啓賢
東海大學資訊工程學系 副教授
liaw@thu.edu.tw

林正基
東海大學資訊工程學系 副教授
cclin@thu.edu.tw

蔡清檥
東海大學資訊工程學系 教授
cttsai@thu.edu.tw

摘要

本文討論一高性能能適應多方向移動偵測的解交錯硬體架構。影片中常發生物體以水平、垂直或斜向移動，這種移動在使用解交錯技術時常導致影像品質不穩，例如呈現鋸齒、模糊、和疊影等現象。本文爲了改善解交錯的品質，將針對各種不同移動方向來進行移動偵測，如此便可使用時間解交錯技術來提高所插補的影像品質。而且我們的方法具有低運算與硬體架構簡單容易實做的優點，根據此技術，一高速超大型積體電路的硬體架構被成功地完成 VLSI 晶片設計，實作在 TSMC 0.13μm CMOS 上。實驗結果顯示此一高性能的解交錯硬體架構在 420×420μm^2 積體電路上包括 14278 的邏輯閘在 154MHz 下可以即時處理高解析顯像裝置的解交錯。

關鍵字：解交錯，插補，水平與垂直移動，超大型積體電路