

Motion Adaptive Compensation Approach for Deinterlacing of Video Sequences

S.Sreekanth
Asst. Professor
G.Pulla Reddy Engg college
Kurnool, India

Y.Chakrapani
H.O.D & Professor
G.Pulla Reddy Engg college,
Kurnool, India

K. Udaya Sri
Associate Professor
KG Reddy College of Engg & Tech.
Hyderabad, India

ABSTRACT

A de-interlacing algorithm using adaptive 4-field motion compensation approach is presented. It consists of block-based directional edge interpolation, same-parity 4-field motion detection, 4-field motion estimation and compensation.

The intra field methods are reconstructed the frame from the current field information .but this method introduce the edge flicker problems and jitter effect. The inter field methods are depends on the previous and future fields for reconstruction of the current frame. This method introduces feathering effect. The edges are sharper when the directional edge interpolation is adopted and jitter effect and the feathering effect eliminated.

The motion adaptive deinterlacing scheme is taking the advantages of both intra and inters field methods. First it finds the motion by using motion detection scheme if the field contain motion apply intra field interpolation method if the field contain stationary objects apply the inter field interpolation method. The 3-field motion detection can not detect the fast motion areas from field to field. The same parity 4-field motion adaptive deinterlacing and the 4-field motion compensation detect the static areas and fast motion by four reference fields. The Compensation recovers the interlaced videos to the progressive ones but the feathering effect is not recovered in this method.

The adaptive 4-field motion compensation method removes the feathering effect along with detecting fast motion areas by using four reference fields. Experimental results show that the peak signal-to-noise ratio of our adaptive 4-field motion compensation deinterlacing algorithm is 4 to 6 dB higher than that of 3-field motion adaptive deinterlacing and 2 to 3 dB higher than 4-field motion compensation deinterlacing and attain the best quality of video.

Key words:

Deinterlacing, motion estimation, motion compensation, SAD.

1. INTRODUCTION

A progressive scan sequence is one in which every line of the video is scanned in every frame. An interlaced sequence is one in which the display alternates between scanning the even lines and odd lines of the corresponding progressive frames is as shown in figure 1.1 [1]. The term field is used (rather than frame) to describe pictures scanned using interlaced scanning, with the even field or top field containing all the even lines of one frame and the odd field or bottom field containing the odd lines. The terms top field and bottom field are also used to denote the even and odd fields, respectively. While interlacing does succeed in reducing the transmission bandwidth, it also introduces a number of high frequency spatial-temporal artifacts that can be distracting to the human eye, such as line crawl and interline flicker. However interlacing benefits can only be experienced if the display

shows the individual fields in the same timely order they were shot in. At this time, only traditional cathode ray tube (CRT)-based television (TV) set is capable to do so. If correctly adjusted, they need not suffer from any interlacing artifacts, they even benefit from it. Yet when the fields of the original footage are displayed in a different order than they were shot in (i.e. when two fields taken at different points in time are re-combined to a full frame displayed at once), visual defects called interlace artifacts occur with moving objects in the image. This is true for most modern liquid crystal display (LCD) and PLASMA DISPLAYS, because they are currently not able to work in interlaced mode. The problem only exists with standard definition SD television and interlaced high definition television HDTV, and there only with material shot with a television or video camera, not with movies. Progressive high definition television HDTV video consists of full frames and does not exhibit interlace artifacts. Also computer displays don't suffer from it as their image isn't interlaced to begin with. To combat interlace artifacts, all liquid crystal display LCD and PLASMA consumer television sets have built-in circuitry that converts interlaced video into progressive video [1].

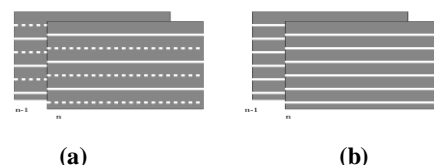


Figure 1.1: Scan modes for video sequences. (a) Interlaced scanning (b) progressive scanning.

The fig 1.1 (a) represents interlaced scanning in that the n^{th} interlaced field contains only even lines (solid lines) and the $(n-1)^{\text{th}}$ interlaced field contains only odd lines (solid lines), i.e. every field contains either even or the odd lines. The figure 1.1(b) represents progressive scanning in that every line is scanned (solid lines) in both n^{th} and $(n-1)^{\text{th}}$ frame.

1.1 Deinterlacing:

The process of converting interlaced video in to a progressive video called DEINTERLACING.

De-interlacing is important because it converts interlaced video sequences into progressive ones for displaying on progressive devices (e.g. Progressive cathode ray tube CRT, liquid crystal display LCD, Plasma Display, and Projection TV). Defects such as edge flicker, jagged effects, and 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.

1.2 Organization of paper:

An introduction to Deinterlacing algorithms is presented in section 2. Section 3 deals with the adaptive 4-field motion compensation method and Section 4 deals with the results. Section 5 deals with conclusions.

2. DEINTERLACING ALGORITHMS

1. Intra or Spatial Field Deinterlacing.
2. Inter or Temporal Field Deinterlacing.
3. Motion Adaptive Deinterlacing.
4. Motion Compensation Deinterlacing.

2.1 Intra field Methods:

Intra field or spatial methods only use pixels in the current field to reconstruct missing scan lines. Therefore, they do not require any additional frame storage. Initially, the memory required to store one video frame was expensive, so intra field methods were attractive. Since spatial methods consider only one frame at a time, their performance is independent of the amount of motion present in the sequence or the frame-recording rate. However, considering only the information in the present field greatly limits these algorithms due to the large temporal correlation that typically exists between successive fields.

2.1.1 Line Repetition:

Line repetition is one of the simplest deinterlacing algorithms, and thus, was one of the first to be considered. In this method, the missing lines are generated by repeating the line directly above or below the missing line. The top field is copied down to fill in the missing lines, and the bottom field is copied up to fill the missing lines.

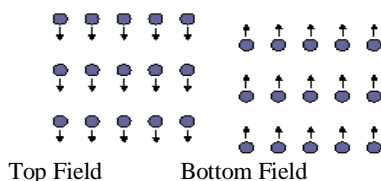


Figure 2.1 Illustration Of Line Repetition.

As shown in figure 2.1[5] the top field is copied down to fill in the missing bottom field. The bottom field is copied up to fill in the missing top field.

2.1.1.1. Jitter Effect:

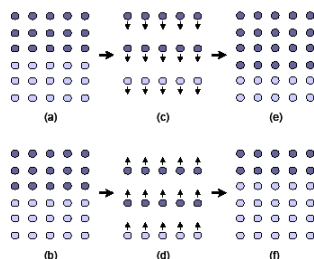


Figure 2.2 Jitter Caused By Line Repetition.

The Fig 2.2 (a),(b) Consecutive progressive frames of a stationary image with a horizontal edge. (c), (d) Corresponding interlaced fields. (e), (f) Result of reconstruction using line repetition. Comparing the deinterlaced frames, the incorrect reconstruction of the horizontal edge causes a 2 pixel shift in successive frames. This shift will make the output sequence appear to shake. [9].

2.2 Inter field methods:

While the performance of spatial methods is independent of the amount of motion present in an image, they essentially

ignore a significant amount of information in adjacent fields that may be useful. For instance, if a video sequence contains one perfectly stationary image, this sequence could be perfectly deinterlaced by simply combining the even and odd fields. Inter field or temporal methods consider previous and/or subsequent frames to exploit temporal correlation. These methods require storage for one or more frames in their implementation, which may have been a serious difficulty in the past. However, the cost of frame storage is not as serious an issue with the reduced cost of memory.

2.2.1 Field Repetition:

Field repetition refers to the generation of missing scan lines by copying lines from the previous frame at the same vertical position.

Specifically, field repetition defined as in equation 4.4

$$F_o[x, y, n] = \begin{cases} F_i[x, y, n] \text{ mod } (y, 2) = \text{mod } (n, 2) \\ F_i[x, y, n-1] \text{ otherwise} \end{cases} \quad (4.4)$$

This type of deinterlacing would provide essentially perfect reconstruction of stationary video. However, field repetition can cause severe blurring if the video contains motion.

2.3 Motion Adaptive Deinterlacing:

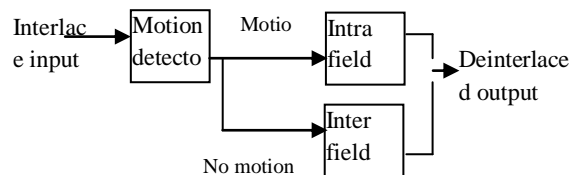


Figure 2.3 Block Diagram of Motion Adaptive Interlacing

The block diagram of the motion adaptive deinterlacing is as shown in figure 2.3 [10]. The goal of motion detection is to detect changes in a video sequence from one field to the next. If a significant change is detected, that region is declared to be moving. If no significant changes are detected the region is assumed to be stationary. This task is straightforward for a progressive sequence. A simple frame difference can be used to detect changes and a threshold can be applied to indicate the presence or absence of motion. Frame differencing cannot be used for interlaced sequences. In fact, it is impossible since consecutive fields do not have the same pixel locations, i.e. an even field cannot be subtracted from an odd field since the corresponding pixels do not exist. By using some techniques we can find motion detection in interlaced sequences [2].

2.4 Motion Compensation Deinterlacing:

The motion-compensated methods utilize Motion Estimation to find the most similar blocks in the neighboring fields and calculate their Motion Vectors. Then a new field is reconstructed from the neighboring field. Block matching algorithm needs extra internal buffer to store the data of the current macro block and reference macro block [7]. It also needs large amount of computational power to calculate the matching criterion Sum of Absolute Difference (SAD).

2.4.1 Motion Estimation:

Motion estimation attempts to find a region called a Macro block in a previously encoded frame (called a reference frame) by taking some search area that closely matches each macro block in the current frame [7]. For each macro block, motion estimation results in a Motion Vector. The motion vector is comprised of the horizontal and vertical offsets from the location of the macro block in the current frame to the location in the reference frame of the selected 16-pixel by 16-pixel region as shown in Fig 2.4.

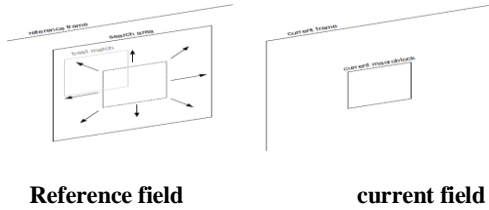


Fig 2.4 Object is moving from current field to reference field.

2.5 Motion Compensation:

Motion compensation is the process of reconstructing of original Field based on the information provided with the motion vector.

2.5.1 Motion Compensation Block Size:

The energy in the residual is reduced by motion compensating each 16×16 macro block .Motion compensating each 8×8 block (instead of each 16×16 macro block) reduces the residual energy further and motion compensating each 4×4 block gives the smallest residual energy of all .From these smaller motion compensation block sizes can produce better motion compensation results. However, a smaller block size leads to increased complexity (more search operations must be carried out) and an increase in the number of motion vectors. An effective compromise is to adapt the block size to the picture characteristics, for example choosing a large block size in flat, homogeneous regions of a frame and choosing a small block size around areas of high detail and complex [11].

2.5.2. Two-Field Motion Estimation/Compensated Deinterlacing:

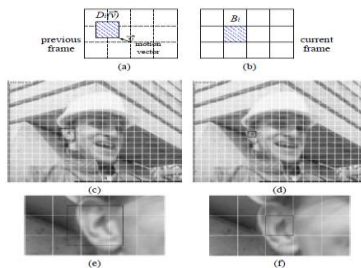


Figure 2.5 Block-matching motion estimation algorithm

In figure 2.5 [11] Find the motion vector (V) of the current block (B_i) by finding the best-matching displaced block D_i(v) in the previous frame. For example, figure (c) and figure (d) show the two consecutive frames in the video sequence.

2.5.3 Three -field motion estimation/ compensated deinterlacing:

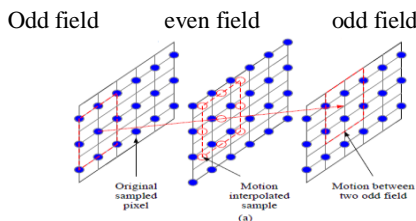


Figure 2.6 Three field motion estimation/compensation

The missing information in the even field is obtained by doing motion estimation between the two odd fields as shown in figure 2.6 [11]. We do not use odd fields for even field motion estimation or vice versa. Most of the previous methods perform the motion estimation using the previous fields related to the current field, no matter if the previous field is even or odd. But, we use previous or next odd fields for the motion estimation of the current odd field. We use previous or next even fields for the motion estimation of the current even

field. Most of the time, pixels in the odd field will stay at the odd field (e.g., non-motion background, horizontal panning regions). Only when there is an odd-pixel vertical movement, will a pixel in the odd field move to the even field. However, when there is an odd-pixel vertical movement, the lost information in the current odd field is also lost in the previous even field. This means that it is unnecessary to track the motion of the pixels that move from an odd field to an even field [7].

Therefore, using previous or next odd fields for the motion estimation of the current odd field or using previous or next even fields for the motion estimation of the current even field is good enough. This method is incorrect because we are not considering the even field information. We have to consider the even field information then only we get the correct results.

3. THE ADAPTIVE 4-FIELD MOTION COMPENSATION METHOD

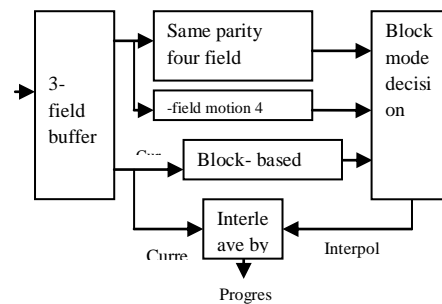


Fig3.1. Block diagram

The block diagram of the Adaptive 4-Field Motion Compensation Method is shown in figure 3.1 [12]. It contains five major parts: the field buffer, the directional edge interpolation module, the 4-field motion adaptation, the 4-field local motion estimation/compensation, and block mode decision module. The three-field buffers are used to store the reference data. (i.e. forward forward field, forward field, backward field).

3.1 Block Based Directional Edge Interpolation:

The directional edge interpolation algorithm uses the pixel difference in 9× 4 block size. Pixel differences are calculated along the 17 directions .[12].

$$d_l(n) = |x(i - n-1, j-1) - x(i + n-1, j+1)| \quad n \in \{1,3,5,7,9\}$$

$$d_{lu}(n) = |x(i + n-1, j-1) - x(i - n-1, j+1)| \quad n \in \{3,5,7,9\}$$

The output pixel is produced depending on the minimum difference of pixel in the 17 directions, the differences are calculated using below [12].

$$d_l(n) = \frac{|x(i - n-1, j-1) - x(i + n-1, j+1)|}{2} \quad n \in \{1,3,5,7,9\}$$

$$d_{lu}(n) = \frac{|x(i + n-1, j-1) - x(i - n-1, j+1)|}{2} \quad n \in \{3,5,7,9\}$$

$$d_{lu}(n) = \frac{|x(i + n-1, j-3) - x(i - n-1, j+3)|}{2} \quad n \in \{3,5,7,9\}$$

$$d_{ru}(n) = \frac{|x(i + n-1, j-3) - x(i - n-1, j+3)|}{2} \quad n \in \{3,5,7,9\}$$

$$(n', p) = \text{Arg}(\min(d_l(n), d_r(n), d_{lu}(n), d_{ru}(n))) \quad n \in \{1,3,5,7,9\}$$

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 equation. The next thing we have to do is to check 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 $d_r(n_o)$, and it conforms to the following equation

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

Where ‘ θ ’ 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 $x(i, j+1)$. The input and output for the edge interpolation method is as shown in figure 3.2.



Figure 3.2 Out put of block based edge interpolation method.

3.2 Same Parity 4-Field Local Motion Estimation/Compensation Method:

3.2.1. 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 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 [11] & [13].

3.2.2: Four -Field Local Motion Estimation /Compensation:

As seen in Fig 3.3 [12], the 4-field local motion estimation involves four fields. The fields are forward forward field and forward field and the current field and the backward field. It also utilizes same-parity field characteristic, which solves the detection error 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 motion vectors on the true images are needed for the de-interlacing application, not the motion vectors that produce the minimal residues. The proposed 4-field motion estimation is shown in Fig 3.3. 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 figure 3.3 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 macro block and the macro block candidate in the search window of the forward-forward field, which is written as following equation.

$$SAD_{1K} = \sum |f(i + MV_{kx}, j + MV_{ky}, n - 2) - f(i, j, n)|$$

Where $f(i, j, n)$ denotes the luminance intensity at the (i, j) location of the n^{th} field, and the MV_{kx} and the 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 macro block where the MV_2 indicated to in the forward field and the macro block where $-MV_2$ indicated to in the backward field.

SAD_2 is calculated as in equation

$$SAD_{2k} = \sum |f(i + \frac{1}{2}MV_{kx}, j + \frac{1}{2}MV_{ky}, n-1) - f(i - \frac{1}{2}MV_{kx}, j - \frac{1}{2}MV_{ky}, n+1)|$$

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 eq and MV_k is the found MV in equation

$$K = \text{Arg}(\min(SAD_{1K} + SAD_{2k}))$$

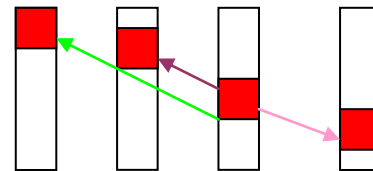


Figure 3.3 four field local motion estimation/compensation

If only SAD_1 is considered to be the matching criterion, the pixel values of the forward field and the backward field are not considered. In other methods, there are lots of errors if we divide the found motion vector by two and produce the Motion Compensation Field according to the divided motion vector.

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 macro block is on a uniform background, the motion vector 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 fields image data, this motion estimation is called “4-field local motion estimation.”

As the motion vectors of the macro blocks are gotten, the motion compensation starts to compensate the current field from the forward field and backward field. If MV_1 is the found motion vector and $MV_2 = (1/2)MV_1$, the macro block data indicated by MV_2 in the forward field and the macro block data indicated by $-MV_2$ in the backward field are averaged and compensated to the position of the current macro block.

In order to compensate the real sampled data, the motion estimation and compensation here is integer-pixel motion estimation, not the quarter-pixel one used in coding. After the compensation of every macro block is done, a new factor for the block mode decision named IPDC is calculated [13] & [14].

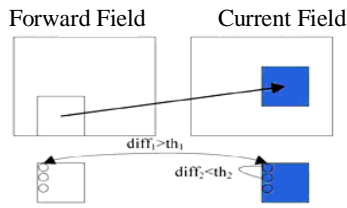


Figure 3.4 IPDC calculation

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 figure 3.4 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. The IPDC calculation in a block is presented as the following code:

$$Diff_1 = \text{abs}(f_{n-1}(i + MV_x, j + MV_y) - f_n(i, j));$$

$$Diff_2 = \text{abs}(f_n(i, j) - f_n(i, j+1));$$

If $(Diff_1 > th1 \& \& Diff_2 < th2)$ IPDC++;

MV_x and MV_y are the found motion vectors. 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.

3.3 Four Field Motion Adaptation:

3.3.1: Four-Field Motion Detection:

To improve the three-field motion detection by reducing the number of missed detections, we use a four-field motion detection algorithm. The idea behind four-field motion detection is presented in figure 3.5 [11].

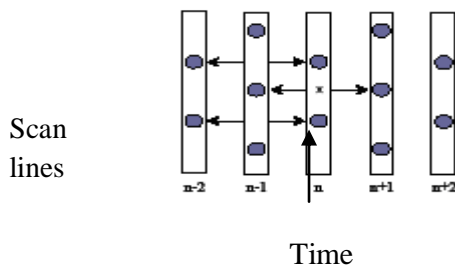


Figure 3.5 Four-field motion detection scheme

The figure 3.5 is used for determining the presence or absence of motion at pixel location 'X'. Three absolute differences are computed. This results in a better detection of fast motion. Only pixels on corresponding scan lines are compared, so no interpolation is needed.

In this scheme, three pixel comparisons are used rather than the one set of pixels that are compared in the previous two algorithms. The additional two pixel differences help to protect against the error that was demonstrated in previous methods [11] & [13].

3.3.2 Four Field Motion Adaptive Deinterlacing:

(n - 2) (n - 1) (n) (n + 1)

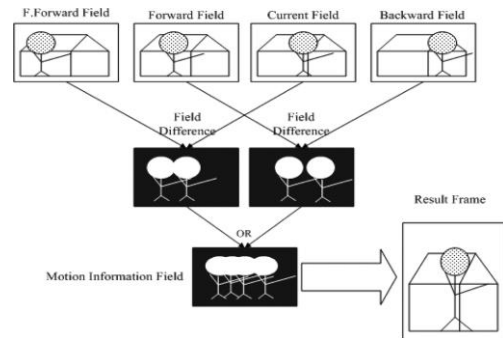


Figure 3.6 Four-field motion adaptive deinterlacing

$n - 2$ is forward forward field, $n - 1$ is forward field, n is current field, $n + 1$ are back ward field.

It uses three reference field and the current field for the motion detection in the MA de-interlacing. 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 figure 3.6, 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 intra field de-interlacing, and the static areas are processed by inter field de-interlacing.

3.4 Flow Chart of Adaptive 4-Field Motion Compensation (proposed) method:

The above figure 3.7 [12] is the flowchart of our adaptive 4-field motion compensation method. There are three reference Fields the forward-forward field, the forward field and the backward field for local motion estimation. The interlaced video first come to the 4-field local motion estimation/compensation block. The 4-field local motion estimation is adopted to calculate MVs and SAD values. 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.

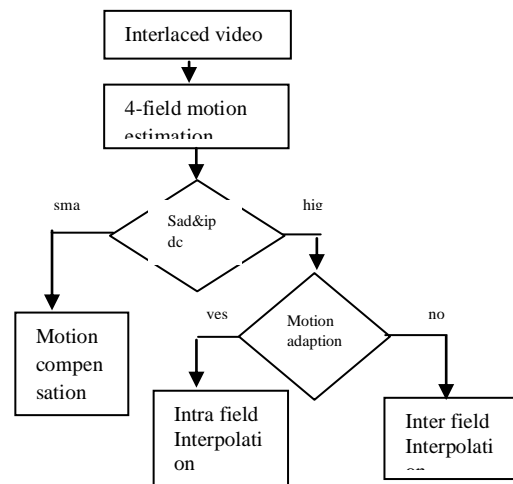


Fig 3.7 flow chart of proposed method

The above figure 3.7 [12] is the flowchart of our adaptive 4-field motion compensation method. There are three reference fields the forward-forward field, the forward field and the backward field for local motion estimation. The interlaced video first come to the 4-field local motion estimation/compensation block. The 4-field local motion estimation is adopted to calculate MVs and SAD values. 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.

4. PSNR

PSNR is a quantitative measure of the video quality is calculated in equation 4.1. Peak Signal to Noise Ratio (PSNR) is measured on a logarithmic scale and depends on the mean squared error (MSE) of between an original and an impaired image or video frame, relative to $(2^n - 1)^2$ (the square of the highest-possible signal value in the image, where n is the number of bits per image sample).

$$PSNR_{db} = 10 \log_{10}((2^n - 1)^2 / MSE). \quad (4.1)$$

And the Mean Square Error (MSE) is defined in equation 5.11 to be the average squared difference between the original and the resulting video.

$$MSE = \frac{1}{M} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (F_o[x, y, n] - F_i[x, y, n])^2$$

4.1 Results:

Several standard definition (SD) video sequences have been tested. These standard definition video sequences are used for television broadcasting and video streaming (internet) applications. The SD video sequences are having a resolution of 720x 480 pixels .i.e. in a frame each row is having 720 pixels and each column having 480 pixels. The input video sequences are having all types of conditions .i.e. they are in motion and stationary and the combination of motion and stationary as shown in fig 4.1, 4.2, 4.3..The results discussed for three methods those are 3-field motion adaptive deinterlacing method and 4- field motion compensation deinterlacing method and the adaptive 4-field motion compensation method. The corresponding peak signal to noise ratio calculated for three methods outputs.



Fig 4.1 Interlaced input, deinterlaced output of car race video of proposed method.

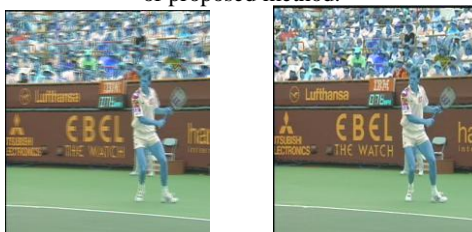


Fig 4.2 Interlaced input, deinterlaced output of tennis video of proposed method



Fig 4.3 Interlaced input, deinterlaced output of Ice hockey video of proposed method

4.2 Comparison of results with PSNR:

The actual interlaced input is given to the system. This input is down sampled by two means it separates the even and odd fields. After that the deinterlacing block gives the deinterlaced output. Then it compares the deinterlaced output and actual input by using compare block finally we get the PSNR output.

Method Type Of video	3- field Motion adaptive deinterlacing(PSNR in db)	4- field motion compensation deinterlacing (PSNR in db)	Adaptive 4- field motion compensation deinterlacing (PSNR in db)
Car race	34.23	38.34	40.40
tennis	33.76	36.53	38.83
Ice hockey	37.34	39.02	41.23

5. CONCLUSION

The five deinterlacing methods intra field deinterlacing and inter field deinterlacing and motion adaptive deinterlacing and motion compensation deinterlacing and adaptive 4-field motion compensation deinterlacing are discussed in the paper. From the discussion conclude that intra field interpolation methods are given the better results when motion present in the video sequence but jitter effect and edge flicker artifacts are present.

The inter field methods are given better results when the video sequence contain stationary objects but feathering effect present. The motion adaptive deinterlacing takes the advantage of both intra and inter deinterlacing methods. The 3-field and 4-field motion adaptive deinterlacing there are some missed detections when there are some fast moving objects. The 3-field and 4-field motion compensation deinterlacing also having same problem when there are fast moving objects.

The adaptive 4-field motion compensation method given the better results as compared to 3-field motion adaptive deinterlacing, 4-field motion compensation deinterlacing. The output of the proposed deinterlaced method remove the interlaced artifacts and given the better visual output on the progressive display. This deinterlacing method given the better results for all types of video sequences contain stationary and motion objects and the combination of stationary and motion objects.

The adaptive 4-field motion compensation method remove all types of interlaced artifacts such as feathering effect occurred in the inter frame method ,and the jitter effect occurred in the intra frame method , edge flicker in the intra frame methods , and the blurring. The adaptive 4-field motion compensation method gives the better PSNR values (2 to 3 db) than the 4-field motion compensation and (4 to 6 db) than the 3-field motion adaptive deinterlacing method.

REFERENCES:

- [1]. Iain E.G. Richardson, "H.264 and MPEG-4 Video Compression, Video Coding for Next-generation Multimedia",2003 The Robert Gordon University, Aberdeen,UK.
- [2]. Motion detection in interlaced sequences available at www.bdti.com.
- [3]. A. Shoham and J. Bier "Introduction to Video Compression" Berkeley Design Technology, March 2004.
- [4]. Berna Erol, Michael Gallant, Student Member, IEEE, and Faouzi Kossentini, Member, IEEE , Guy Cote, Student Member, IEEE " H.263+ Video Coding at Low Bit Rates",vol .8,no.7 ,November 1998.
- [5]. Brian A.Heng, "Application of Deinterlacing for the Enhancement of Surveillance Video".B.S. Electrical Engineering University of Minnesota, 1999.
- [6]. Xin Li, Member, IEEE, and Michael T. Orchard, Fellow, IEEE," New Edge- Directed Interpolation" VOL 10, No. 10, October 2001.
- [7]. Yoon Kim¹, Kang-Sun Choi¹, Jae-Young Pyun¹, Byung-Tae Choi², and Sung-Jea Ko¹ "A Novel De-interlacing Technique Using Bi-directional Motion Estimation "Department of Electronics Engineering,Korea University, Anam-Dong, Sungbuk-Ku, Seoul 136-701, Korea.
- [8]. De Haan, G.; Bellers, E.B " De-interlacing of video data, Consumer Electronics"IEEE Transactions on, vol. 43, pp. 819–825, Aug. 1997.
- [9]. E. B. Bellers and G. de Haan, Advanced de-interlacing techniques," Proc. ProRISC / IEEE Workshop Circ., Syst. and Sig. Proc. Mierlo,The Netherlands, Nov.27–28, 1996,pp. 7–17.
- [10]. Tero Koivunen "Motion detection of an interlaced video signal", IEEE Trans. Consumer Electronics, pp.753-759, 1994.
- [11]. 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.
- [12]. Yu-Lin Chang,Shyh-Feng Lin,Ching-Yeh Chen, and Liang-Gee Chen "Video De-Interlacing by Adaptive 4-Field Global/Local Motion Compensated Approach", Fellow, IEEE ,vol 15, no.12, December 2005.
- [13]. E.B.Bellers and G.de Haan "New Algorithm for Motion Estimation on Interlaced Video" in Philips Research Laboratories, Television Systems Group, Prof. Holstlaan 4, 5656 AA Eindhoven, The Netherlands.
- [14].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.
- [15]. 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.