A MOTION ADAPTIVE DEINTERLACING METHOD WITH HIERARCHICAL MOTION DETECTION ALGORITHM

Elham Shahinfard, Maher A. Sid- Ahmed, Majid Ahmadi

University of Windsor, Windsor, Ontario, Canada

ABSTRACT

This paper presents a motion adaptive deinterlacing method for high quality conversion of interlace video format to progressive video format. A high performance and low complexity algorithm for motion detection is proposed. This algorithm uses five consecutive interlace video fields for motion detection, so it is able to capture a wide range of motions from slow moving objects to fast motions. The proposed motion detection algorithm benefits from a hierarchal structure where the algorithm starts with detecting motion in large partitions of a given field. Depending on the detected motion activity level, the motion detection algorithm might be recursively applied to sub-blocks of the original partition. Two low pass filters are used during the motion detection to increase the algorithm accuracy. The result of motion detection is then used in a motion adaptive interpolator for high quality deinterlacing. Excellent experimental results are obtained for motion detection and deinterlacing performance.

Index Terms: Deinterlacing, interlace, progressive, motion detection, interpolation, motion adaptive.

1. INTRODUCTION

Old analog television systems use interlaced video format in their standard benefiting from the fact that human visual system is less sensitive to flickering details than to large-area flicker [1-2]. However with recent advances in telecommunication and display technology, progressive video format has become the preferred format for new television systems. Transition to all-progressive video signals requires high performance deinterlacing methods to convert the signal from interlaced format to progressive format with minimal subsampling artifacts and computational complexity.

Over the last few decades many deinterlacing methods have been proposed. These algorithms could be categorized into three main categories: spatial methods, temporal methods and hybrid methods.

Spatial methods interpolate the missing lines of a field using the pixel values exclusively from the same field. They employ different types of spatial interpolations ranging from simple line repetition to edge dependent interpolations [2]. Spatial methods are independent of motion activity in the video sequence. However, since the high level of correlation that typically exists between successive video fields is ignored in these methods, their performance is not optimal.

Temporal methods such as field insertion [3] interpolate the missing lines of a field by using temporal data from previous and/or subsequent fields. Therefore, temporal methods ignore spatial correlation resulting in suboptimal performance.

The idea of hybrid methods is to take advantage of both of these categories. The performance of hybrid methods improves due to exploiting both spatial and temporal correlations. A wide range of hybrid methods are available in literature [4-10]. The most advanced hybrid methods are motion compensated and motion adaptive methods [2].

In most motion compensated methods the motion data is removed from the video signal prior to interpolation and will be added to the video at a later stage [4-6]. Motion compensated methods require complicated motion estimation algorithms since their performance is highly dependent on successful motion removal.

Motion adaptive methods are among the most popular deinterlacing methods. They adapt themselves to motion activity level of the video sequence. Depending on the detected level of motion activity they may use a simple temporal method in static parts of the video or a spatial method in highly dynamic parts [7-10].

This paper proposes a motion adaptive method with an accurate and simple motion detection algorithm. The hierarchical motion detection algorithm, which is explained in Section 2, measures motion activity level in video sequence and calculates adaptation factor for interpolation step. The interpolation step is explained in Section 2.2. Experimental results which are shown in Section 3 show that the algorithm has a high performance. Finally in Section 4 some concluding remarks are given.

2. DEINTERLACING METHOD

Proposed motion adaptive deinterlacing method measures the motion activity level of video data using a hierarchal motion detection algorithm. This measured activity level is subsequently used in interpolation algorithm as a weight factor for combining a temporal and a spatial deinterlacing method so that the performance benefits from both methods. Fig. 1 shows a block diagram representation of the proposed deinterlacing algorithm.

Each color component of video sequence is first fed to a motion detection block. The motion detection algorithm calculates MV (Motion Value) which is a measure of motion activity level of video data using a hierarchical structure explained in section 2.1.

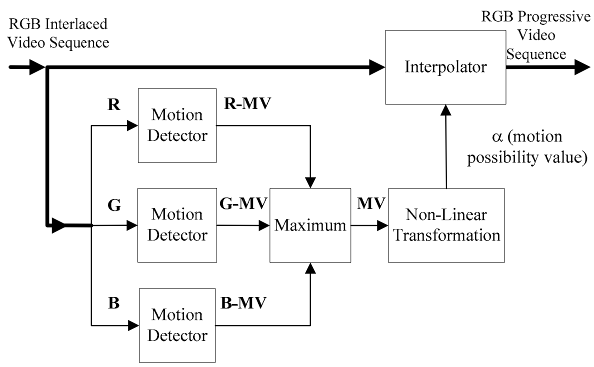


Fig. 1 Block diagram of the proposed deinterlacing method.

The result of motion detection on R, G, and B color components are then compared and the maximum of three motion values is used as the motion value for all three components. Using a single value for all three components avoids distortion artifacts in the output. The MV is then delivered to a non-linear transformation block which transforms the motion value to a number between zero and one. This number (the coefficient α) is motion possibility value. The non-linear transformation is explained in section 2.1. The interpolation algorithm uses motion possibility value to make a linear combination of a temporal and a spatial deinterlacing method.

2.1 Motion Detection Algorithm

In motion adaptive deinterlacing a motion detection algorithm detects motion in the interlaced video signal. The result of this algorithm is subsequently used in interpolation algorithm to estimate the missing lines of interlaced video. Motion detection algorithm should be as accurate as possible to avoid introducing extra error to interpolation algorithm.

Proposed motion detection algorithm uses five consecutive fields of video data to increase the ability of detecting fast motions. Fig. 2 shows a sequence of five video fields which will be transferred to motion detector at the same time, F represents the current field, $F+1$ and $F+2$ represent the two subsequent fields, and $F-1$ and $F-2$ represent the two fields prior to F.

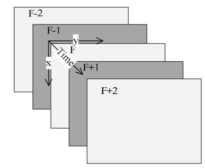


Fig. 2 A sequence of five consecutive fields.

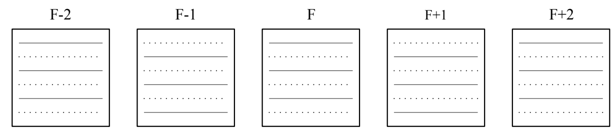


Fig.3 Sample position of missing lines in a sequence of 5 fields.

Fig. 3 shows the position of missing lines in these fields. Note that the pairs (*F-2, F*), (*F-1, F+1*) and (*F, F+2*) have same missing lines and the same time difference. As shown in Fig 4, these three pairs are used for motion detection. The difference between the above mentioned pairs are low-pass filtered for noise removal. The use of low-pass filter is justified by the fact that signal is large and noise is small and the low frequency energy in signal is greater than low frequency energy in noise. Then the absolute value of these filtered signals is obtained. The rectified signals are then low-pass filtered to improve the consistency of the outputs based on the assumption that moving objects are large compared to pixels.

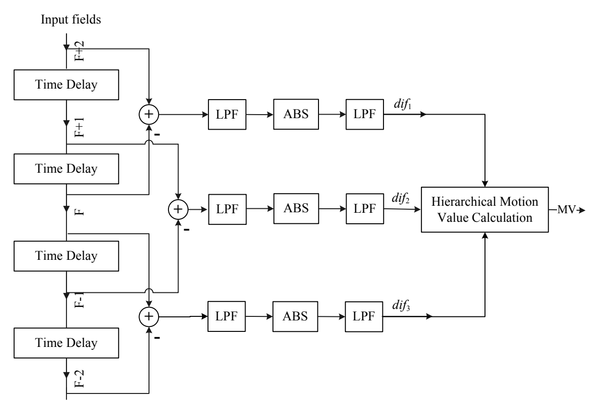


Fig. 4 Block diagram of proposed motion detection algorithm.

In the next stage, the rectified difference signals dif_1, dif_2 and dif_3 are delivered to the hierarchical motion calculation module. This module partitions the rectified difference signals into data blocks with the same spatial positions. The average of each three corresponding rectified difference block is calculated and the maximum of the three averages is compared with a predefined threshold value.

If the maximum value is smaller than the threshold, the block is considered as a static part of video and by setting its MV equal to zero the interpolation algorithm is forced to use a purely temporal interpolation. If the maximum value is greater than threshold, the block of data is considered as a dynamic block and will be recursively partitioned to smaller blocks and the procedure continues. The threshold value is predefined through recursive error measurement on standard videos and will be changed on each stage as a function of the block size. In our implementation we chose the threshold value to be in proportion to the logarithm of the size of the block. The optimal value of the threshold also depends on the type of the video signal. The final output of hierarchical structure is an MV matrix with the same size as the original video fields. Experimental results in section 3 show that the algorithm is powerful in detecting all dynamic parts of video sequence.

A non-linear transformation illustrated in Fig. 5 will be applied to the motion value matrix to find the motion possibility value. This value is used as the weight factor for combining the results of temporal and spatial interpolations.

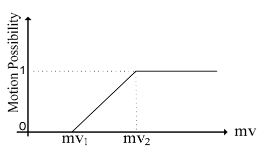


Fig. 5 Non-linear transfer function.

2.1 Interpolation Algorithm

In the proposed motion adaptive method simple line averaging is chosen as the spatial deinterlacing and median filtering is chosen as the temporal deinterlacing method. Motion possibility values, calculated after motion detection step, are used as the weighting factor for combining the results of spatial and temporal interpolations to improve the performance. The interpolation step could be formulized by equation (1) where α is the motion possibility value measured in motion detection algorithm.

$$F_o(X,n) = \begin{cases} F(X,n) & where \quad y \bmod 2 = n \bmod 2 \\ \alpha F_{spatial}(X,n) + (1-\alpha)F_{temporal}(X,n) & otherwise \end{cases} \quad (1)$$

In the above equation $X=(x,y)$ represents the pixel position, n represents field index and α represents motion possibility value. Note that α is also a function of X and n. For highly dynamic parts of the video the value of α would be equal to unity, resulting in pure spatial interpolation and for highly static parts of the video signal it would be equal to zero resulting in pure temporal interpolation.

3. EXPERIMENTAL RESULTS

The algorithm has been tested on several standard video sequences. These standard videos are originally in progressive format. Interlaced video sequences have been generated by removing every other line from the original progressive video as shown in Fig.6. Optimal threshold values have been found through recursive error minimization on these sequences.

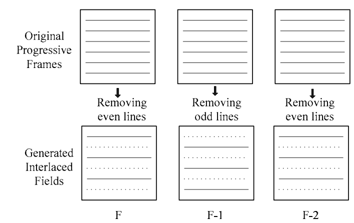


Fig. 6 Generating interlaced video from originally progressive video sequence.

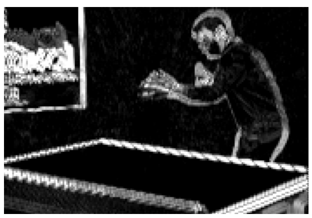
Fig 7 shows a grayscale display of implementation results for a single frame of TABLETENNIS sequence when mv_1 and mv_2 (thresholds shown on Fig. 5) are set to 0.1 and 0.4 of the maximum possible value for mv. A simple 3 by 3 averaging filter has been used as the low-pass filter. Fig. 7-a shows the original progressive video frame. Fig 7-b is a visualization of the motion possibility matrix generated at the output of the motion detector. This image shows that the motion detector has been successful in detecting all moving parts. Fig 7-c is the deinterlaced output which confirms the high performance of the proposed deinterlacing algorithm.

For quality comparison the proposed algorithm results on three different video sequences has been compared with the results of several well-known deinterlacing algorithms.

Tested video sequences are MOM video sequence which has minimum motion, TABLETENNIS video sequence which has fast motion on hands, head, ball and table edges, and FLOWER-GARDEN video sequence with moving camera. For all three cases 60 successive frames of an RGB sequence of size 240×352 has been used.

We have implemented several interlacing algorithms in literature to compare with the performance of our algorithm. Specifically, methods surveyed in [2], including line repetition and line averaging from spatial deinterlacing category; field insertion and bilinear field averaging from temporal deinterlacing category; and vertical-temporal median filtering, motion adaptive with field insertion and line repetition and motion adaptive with field insertion and line averaging. We have also compared our algorithm with the deinterlacing method proposed in [6] from hybrid deinterlacing category. The average Mean Square Error (MSE) of any of these deinterlacing algorithms is calculated for MOM, TABLETENNIS, and FLOWER-GARDEN sequences. The proposed algorithm has shown the smallest MSE between all of them. Fig. 8 shows the average MSE of all the implemented methods in dB. The last group of bars in this graph which has the minimum value of all is the average MSE of the proposed method which proves the proposed method superiority in terms of deinterlaced image quality.

(a) Original progressive frame



(b) Motion possibility; Brigh points represent daynamic parts & dark points represent

(c) Deinterlaced frame

Fig. 7 Proposed algorithm deinterlacing results.

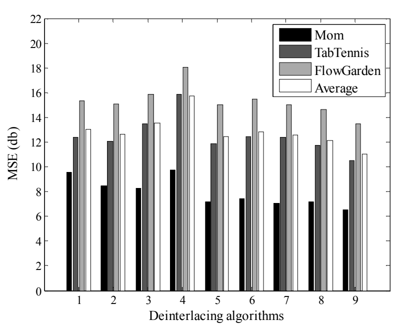


Fig. 8 The average MSE for different deinterlacing methods, 1 to 9 on x-axis shows 1-line repetition, 2-line averaging, 3- field insertion, 4- bilinear field averaging, 5- VT median filtering, 6- Motion adaptive with field insertion and line repetition, 7- Motion adaptive with field insertion and line averaging, 8- method of [6] 9-Proposed method.

4. CONCLUSION

A motion adaptive deinterlacing method with high deinterlacing quality is proposed. The method has a powerful motion detection algorithm which determines the possibility of motion in the video by a hierarchical structure. In order to save the consistency of motion detection algorithm and to avoid introducing extra artifacts, the motion detection algorithm is separately applied to all color components and the results are compared to each other to reach to a unique motion possibility result. The interpolation step uses motion detection results to obtain best weighting factor for combining temporal and spatial interpolation values. Experimental results prove that proposed deinterlacing algorithm has outperformed other available deinterlacing methods.

5. REFERENCES

[1] W. Engstrom, “A study of television image characteristics, Part II: Determination of frame frequency for television in terms of flicker characteristics,” *Proc. IRE*, vol. 23, no. 4, pp.295–310, 1935.

[2] De Haan, G.; Bellers, E.B., "Deinterlacing-an overview," *Proceedings of the IEEE* , vol.86, no.9, pp.1839-1857, Sep 1998

[3] US Patent #5621470, Maher A. Sid-Ahmed “Interpixel and interframe interpolation of television pictures with conversion from interlaced,” Apr 15, 1997.

[4] De Haan, G.; Bellers, E.B., “De-interlacing of video data,” *Consumer Electronics, IEEE Transactions on,* vol. 43, pp. 819–825, Aug. 1997.

[5] 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.

[6] Renxiang Li; Bing Zheng; Liou, M.L., "Reliable motion detection/compensation for interlaced sequences and its applications to deinterlacing," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol.10, no.1, pp.23-29, Feb 2000.

[7] Dongil Han; Chang-Yong Shin; Seung-Jong Choi; Jong-Seok Park, "A motion adaptive 3-D de-interlacing algorithm based on the brightness profile pattern difference," *Consumer Electronics, IEEE Transactions on* , vol.45, no.3, pp.690-697, Aug 1999.

[8] Bhatt, B.; Templin, F.; Cavallerano, A.; Gornstein, V.; Hogstrom, B.; Derovanessian, H.; Lamadrid, S.; Mailhot, J., "Grand Alliance HDTV multi-format scan converter," *Consumer Electronics, IEEE Transactions on* , vol.41, no.4, pp.1020-1031, Nov 1995.

[9] Qian Huang; Wen Gao; Debin Zhao; Huifang Sun, "An efficient and robust adaptive deinterlacing technique," *Consumer Electronics, IEEE Transactions on*, vol.52, no.3, pp. 888-895, Aug. 2006

[10] Yanfei Shen; Dongming Zhang; Yongdong Zhang; Jintao Li, "Motion adaptive deinterlacing of video data with texture detection," *Consumer Electronics, IEEE Transactions on*, vol.52, no.4, pp.1403-1408, Nov. 2006