A motion and edge adaptive interlaced-to-progressive conversion using fuzzy logic-based systems

P. Brox
Instituto de Microelectrónica de Sevilla (CSIC) and University of Seville (Spain)
brox@imse.cnm.es

I. Baturone
Instituto de Microelectrónica de Sevilla (CSIC) and University of Seville (Spain)
lumi@imse.cnm.es

S. Sánchez-Solano
Instituto de Microelectrónica de Sevilla (CSIC)
Seville (Spain)
santiago@imse.cnm.es

Abstract

This paper presents an algorithm for video de-interlacing. The approach uses three fuzzy logic-based systems to adapt the interpolation strategy to the presence of motion and edges. Furthermore, the algorithm is able to deal with any kind of TV material independently of the source used to acquire the scene. Extensive simulations of standard and real sequences prove the efficiency of the proposed algorithm.

Keywords: Video de-interlacing, fuzzy logic-based system, motion adaptive, edge adaptive.

1 Introduction

Interlacing was introduced by the TV community since it provides an effective reduction of the video bandwidth. It reduces the bandwidth at half since only the even or odd lines that compose a frame are alternatively transmitted. Interlacing is currently used by all the analog TV standards (PAL, NTSC and SECAM) and also, by some of the more modern digital transmissions [1].

Recently, there is an increasing need of a progressive scanning format at the receiver side of TV signal. Many devices such as modern displays (LCDs, Plasma), DVDs, and projectors, work with progressive material and incorporate an embedded chip that implement a de-interlacing algorithm. It consists of converting interlaced video into a progressive form by interpolating the non-transmitted lines. Several features of the picture like the presence of motion and edges could complicate this task.

Many de-interlacing algorithms have been proposed in the literature during the last years [2]. Basically, they can be classified into two categories: motion (MC) and non-motion compensated (non-MC) algorithms. MC techniques look for a motion vector in each pixel or block of pixels of the image and achieve the best results in moving areas. However, the computational cost involved in the calculation of the most appropriate motion vector is quite high.

An alternative among non-MC algorithms are the motion adaptive de-interlacing techniques [3]-[6]. As its name indicates, this kind of algorithms estimates the level of motion in the image and adapt the interpolation strategy according to it. If there is no motion or the level of motion is barely appreciable then the temporal neighbors are suitable to perform the interpolation. On the contrary, when the level of motion is high a spatial interpolator is chosen to interpolate the new pixel.

The efficiency of motion adaptive algorithms relies on the quality of the motion detector. Primitive approaches use the difference between pixels with the same spatial coordinates from two consecutive frames to measure motion, and a crisp transition between the temporal and spatial interpolator [3]. However, they are far to achieve good results in tricky parts of the image, which contain high contrast detailed areas, high level of motion,

L. Magdalena, M. Ojeda-Aciego, J.L. Verdegay (eds): Proceedings of IPMU'08, pp. 1175–1182
Torremolinos (Málaga), June 22–27, 2008

noise and/or a high number of edges.

To improve the robustness of the motion detector several proposals have been presented in the literature during the last years [2], [4]. Some authors combine the output of several motion detectors [2], whereas others apply filtering techniques to field difference signal [4].

Other authors improve the performance of motion adaptive de-interlacing algorithms [4]-[6]. In [4], the crisp transition between the temporal and spatial interpolator is substituted by a soft transition. In this sense, several options are proven in [4] such as a linear or a step piecewise transition. Other alternative is to use fuzzy logic and to apply different heuristic rules with approximate levels or uncertainty, which implicitly perform a nonlinear filtering [5]-[6].

This paper describes a new motion adaptive de-interlacing as result of a work developed during the last years. The combination of the spatial and temporal interpolator is carried out by a fuzzy system (FS_1), whose input is a bi-dimensional convolution of field difference signal. Furthermore, the spatial and temporal interpolators are also calculated by two fuzzy systems: a second fuzzy system (FS_2) provides the temporal interpolator, which is capable of dealing with any kind of video material, and a third fuzzy system (FS_3) is able to adapt the spatial interpolation to edges.

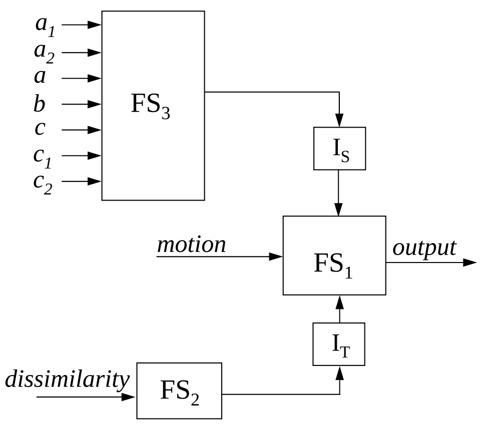


Figure 1: Block diagram of the proposed algorithm.

2 Description of the algorithm

Figure 1 shows a descriptive diagram of the implemented algorithm. The following subsections describe the three fuzzy systems used in the proposed algorithm.

2.1 Fuzzy system to combine the interpolators with the presence of motion (FS_1)

Since current TV standards work with video coding algorithms where luminance component contains more information than chrominance components [1], our study is only developed for this unique component. However, its extension to color images is simple and direct by applying the final interpolation expression to the color components.

Our proposal uses as input value the bi-dimensional convolution of field difference signals that can be mathematically expressed as follows:

$$motion = \frac{\Sigma C_{j,i} H_{i,j}}{\Sigma C_{j,i}} = \\ = \frac{(1\ 2\ 1)\,(H_{1,1}\ H_{1,2}\ H_{1,3})^T}{4} \quad (1)$$

where $C_{j,i}$ are the values of the weights and $H_{i,j}$ are described by the following differences of luminance values (see Figure 2(a)):

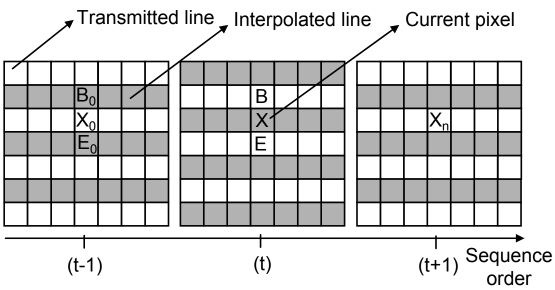
$$H_{1,1} = |B_0 - B| \quad (2)$$

$$H_{1,2} = |X_0 - X| \quad (3)$$

$$H_{1,3} = |E_0 - E| \quad (4)$$

Different weights and sizes of matrix H have been studied to achieve a good trade-off between the computing resources and the quality of motion measurement [7]-[8]. As can be seen in expression (1), the selected convolution only includes neighbors in vertical direction since a wide number of video sequence simulations shown a non-decisive influence of horizontal neighbors to measure the level of motion.

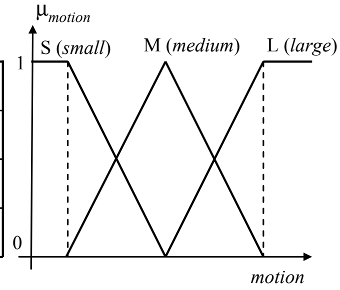
Unlike the proposals in [5]-[6], which use four fields, our algorithm reduces the temporal aperture up to three fields as shown Figure



(a)

If	*antecedent*	*consequent*
1st	*motion* is S	$X=I_T$
2nd	*motion* is L	$X=I_S$
3rd	*motion* is M	$X=\lambda I_T+\delta I_S$

(b)



(c)

Figure 2: (a) Pixels involved in the calculation of the bi-dimensional convolution. (b) Rulebase of the FS_1. (c) Membership functions used in FS_1.

2(a). The interpolated values calculated in the previous field (B_0, E_0) are necessary to evaluate the *motion* value in expression (1). The spatial interpolator (I_S) is employed to calculate the first progressive frame.

The influence of motion in selecting the kind of interpolation is evaluated by considering three rules that are linguistically expressed as follows:

1. If *motion* in the current pixel is small (S), the most adequate interpolated value is obtained by applying a temporal interpolation (I_T).
2. If *motion* in the current pixel is large (L), the best result is obtained by performing a spatial interpolation (I_S).
3. If *motion* in the current pixel is medium (M), then the value is better calculated by applying a linear combination of the temporal and spatial interpolators $(\lambda I_T + \delta I_S)$.

This rulebase is summarized in the Table of Figure 2(b). The fuzzy concepts small, large and medium used in the rules are modeled according to the membership functions shown in Figure 2(c). Using the Fuzzy Mean as defuzzification method the new pixel value is calculated as follows:

$$X = \alpha_1 I_T + \alpha_2 I_S + \alpha_3(\lambda I_T + \delta I_S) \qquad (5)$$

where α_i is the corresponding activation degree of each rule in the Table of Figure 2(b).

Three degrees of *motion* (small, medium and large) are considered in this fuzzy system. After analyzing up to five degrees of *motion* [9], the rulebase with three rules has been selected since it provides the most attractive solution in terms of hardware resources and quality of the interpolated image.

2.2 Fuzzy logic-based system for the temporal interpolation (FS_2)

In order to understand the strategy implemented in FS_2 to obtain the temporal interpolator, it is necessary to review the origin of material. If the sequence was recorded by a video camera at a picture rate of 50 Hz (PAL) or 60 Hz (NTSC), the three fields of the aperture are different in moving areas of the image (different numbers in Figure 3(a)). However, if the material was registered with a cine-camera the picture rate is 24 Hz and a conversion of film material is necessary to display it on TV. The conversion to adapt both picture rates basically consists of repeating the fields twice (to achieve 50 Hz), or twice and three times alternatively (to achieve 60 Hz) as it is shown in Figure 3(b). This process is known as pull-down 2:2 and pull-down 3:2, respectively.

Since the temporal aperture of this approach would be composed from film material (for instance 2-even, 2-odd, 3-odd), two of the fields in the aperture has to come from the same original frame. The detection of these cases is very interesting due to two reasons. Firstly, the risen presence of film and hybrid material

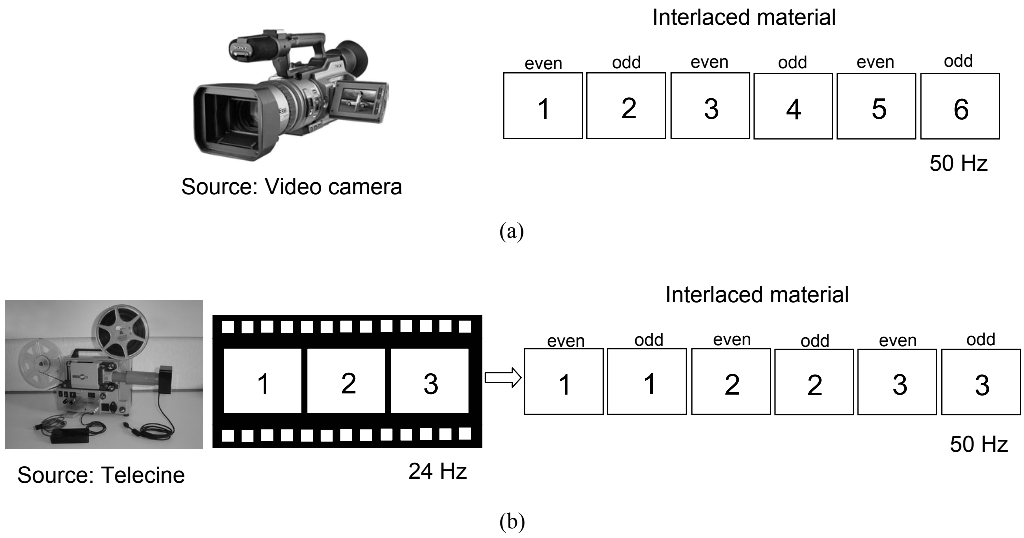
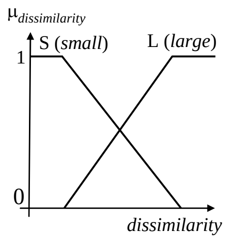


Figure 3: (a) Video sequence. (b) Sequence of film material.

If	*antecedent*	*then*	*consequent*
1st	*dissimilarity* is S		$I_T = X_0$
2nd	*dissimilarity* is L		$I_T = X_n$

(a)



(b)

Figure 4: (a) Rulebase of the FS_2. (b) Membership functions used in FS_2.

on TV and secondly, a perfect de-interlacing can be achieved by copying this information from the repeated field in the aperture at a expense of a minimal cost (if the repeated field is correctly detected in the aperture).

A simple fuzzy system is proposed that is able to deal with film material. It selects the most adequate temporal interpolation depending on *dissimilarity* signal between two consecutive fields, given by the following expression:

$$dissimilarity = \frac{|B - B_0| + |E - E_0|}{2} \quad (6)$$

The heuristic knowledge of this fuzzy system is expressed by means of the following linguistic rules:

1. If *dissimilarity* between the fields *(t-1)* and *(t)* is small (S), the most adequate interpolated value is obtained by selecting the pixel value in the previous field at the same spatial position (X_0) (see Figure 4(a)).
2. On the contrary, if *dissimilarity* is large (L), the pixel value in the previous field is not a good choice and is better to bet on the pixel in the next field (X_n) (see Figure 4(a)).

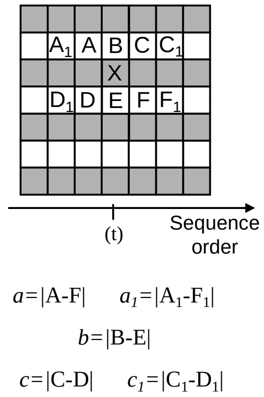
Table in Figure 4(a) summarizes the rulebase of this second fuzzy system. The shape of membership functions to model the fuzzy concepts small and large are shown in Figure 4(b). The output of this fuzzy system is given by the following expression:

$$I_T = \beta_1 X_0 + \beta_2 X_n \quad (7)$$

where β_i is the activation of each rule in the Table of Figure 4(a).

2.3 Fuzzy logic-based system for the spatial interpolation (FS_3)

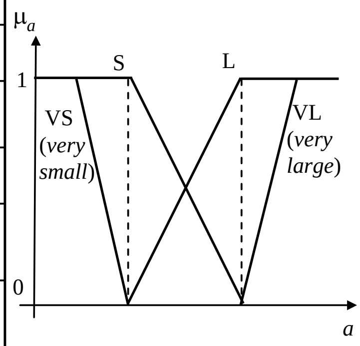
FS_3 performs a smart interpolation among pixels in the spatial neighborhood. The heuristic knowledge developed in the fuzzy rulebase adapts the interpolation strategy according to the presence of edges in the picture. To detect edges the following differences



(a)

If	*antecedent*	*then*	*consequent*
1st	a is S *and* b is L *and* c is L		I_S=(A+F)/2
2nd	a is L *and* b is L *and* c is S		I_S=(C+D)/2
3rd	a is VS *and* b is L *and* c is VS		I_S=(A+F+C+D)/4
4th	a_1 is S *and* a is L *and* b is L *and c is* VL *and* c_1 is VL		I_S=(A_1+F_1)/2
5th	a_1 is VL *and* a is VL *and* b is L *and c is* L *and* c_1 is S		I_S=(C_1+D_1)/2
6th	*otherwise*		I_S=(B+E)/2

(b)



(c)

Figure 5: (a) Pixels involved in the spatial interpolator. (b) Rulebase of the FS_3. (c) Membership functions used in FS_3.

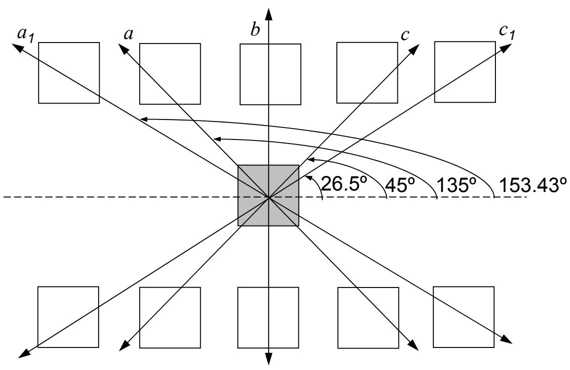


Figure 6: Directions evaluated by the FS_3.

of pixel values along five directions are calculated (see Figure 5(a) and Figure 6):

$$a_1 = |A_1 - F_1| \tag{8}$$

$$a = |A - F| \tag{9}$$

$$b = |B - E| \tag{10}$$

$$c = |C - D| \tag{11}$$

$$c_1 = |C_1 - D_1| \tag{12}$$

The following knowledge is employed to estimate the edge adaptive interpolation:

1. If there is a small (S) difference in direction a, and if b and c are large (L), then an edge could be in direction a and the best solution is to apply the average between the two pixels that defines a direction.
2. If there is a small (S) difference in direction c, and if b and a are large (L), then an edge could be in direction c and the best solution is to apply the average between the two pixels that defines c direction.
3. If there is a very small (VS) difference in directions a and c, and a large (L) difference in direction b, neither there is an edge nor vertical linear interpolation performs well; the best option is a linear interpolation between the neighbors with small differences: A, C, D, F.
4. An edge is clear in direction a_1 not only if a_1 is small (S), but also if a and b are large (L) and c and c_1 are very large (VL). Then the spatial interpolation is calculated by applying the average between the two pixels that defines a_1 direction.
5. An edge is clear in direction c_1 not only if c_1 is small (S), but also if b and c are large (L) and a and a_1 are very large (VL). Then the spatial interpolation is calculated by applying the average between the two pixels that defines c_1 direction.
6. Otherwise, a vertical linear interpolation would be the most adequate.

Table of the Figure 5(b) summarizes the rule-

Table 1: Average PSNR values (in dBs) using video sequences.

SEQUENCE FORMAT	**Missa CIF**	**Paris CIF**	**Trevor CIF**	**Salesman CIF**	**News QCIF**	**Mother QCIF**	**Carphone QCIF**
Line Doubling	36.44	23.61	31.05	29.75	25.18	31.81	28.25
Line Average	40.47	26.67	35.04	33.53	29.25	35.94	32.61
ELA 3+3	39.49	25.53	34.11	32.11	26.63	35.39	32.65
ELA 5+5	38.56	24.64	33.31	30.17	25.92	34.2	31.51
Field Insertion	38.36	29.86	34.36	36.17	33.13	36.14	30.34
VT 2fields	40.25	30.73	36.61	36.54	35.46	39.61	34.08
VT 3fields	40.52	31.37	37.16	36.95	35.67	40.89	34.54
Technique in [5]	40.01	33.12	35.38	37.62	34.73	39.49	32.27
Technique in [6]	40.18	35.28	36.69	38.29	37.51	41.87	34.78
Proposal	40.81	35.87	37.63	38.35	38.78	42.11	35.09

base of this second fuzzy system. From the analysis of these rules, we can see that a higher number of antecedents are used in the rules that evaluate a_1 and c_1 directions, since a reinforcement is necessary to avoid the detection of false edges when the system works with 5+5 pixels in the neighborhood.

The shape of membership functions to model the fuzzy concepts small, large, very small and very large are shown in Figure 5(c). The output of this fuzzy system is obtained by applying the Fuzzy Mean as follows:

$$
\begin{aligned}
I_S = \chi_1(\frac{A+F}{2}) + \chi_2(\frac{C+D}{2}) + \\
+\chi_3(\frac{A+C+D+F}{4}) + \chi_4(\frac{A_1+F_1}{2}) + \\
+\chi_5(\frac{C_1+D_1}{2}) + \chi_6(\frac{B+E}{2}) \qquad (13)
\end{aligned}
$$

where χ_i is the activation of each rule in the Table of Figure 5(b).

3 Simulation results

The performance of the proposed algorithm has been analyzed by de-interlacing several video sequences. They can be divided into two categories: a first group of standard video sequences and a second one of real film sequences.

The video sequences considered have widely been used as benchmarks in video processing applications. After obtaining the interlaced video data from these progressive sequences by eliminating lines, several de-interlacing algorithms have been applied. The Peak Signal to Noise Ratio (PSNR) is used as figure of merit, to evaluate the quality between the obtained interpolated frames and the original ones.

The proposed algorithm has been also compared with other de-interlacing algorithms with less or similar computational cost: four spatial method such as line doubling, line average, and conventional ELA (edge-adaptive interpolation algorithm [2]) using 3+3 and 5+5 taps; the simplest temporal de-interlacing algorithm called field insertion, and two vertico-temporal filtering with two and three fields [2]; and, finally the fuzzy motion adaptive algorithms reported in [5] and [6].

Table 1 shows the average PSNR values obtained when de-interlacing fifty fields of seven video sequences. The PSNR results show that the proposed algorithm performs better than the other algorithms since it achieves the highest values. Moreover, its computational complexity is quite low since the three fuzzy systems are very simple.

The algorithm has also been tested to de-interlace the real film sequences shown in Table 2. These results prove the advantages of the inclusion of the second fuzzy system (FS_2).

Finally, the superior performance of our approach can be corroborated by the visual in-

Table 2: Average PSNR values (in dBs) using film sequences.

SEQUENCE FORMAT	Fire Rose PAL TV	Chop Hunt PAL TV	Fargo Repair PAL TV	Fargo PAL TV	Tokyo PAL TV
Line Doubling	34.51	39.97	30.48	28.79	27.22
Line Average	38.76	44.61	35.92	34.31	31.46
ELA 3+3	35.55	44.07	35.28	33.66	30.02
ELA 5+5	33.61	43.16	34.33	32.16	28.53
Field Insertion	36.41	24.06	31.23	33.07	36.49
VT 2fields	40.32	44.18	35.87	40.99	36.84
VT 3fields	41.16	46.08	38.43	38.91	35.13
Technique in [5]	39.36	43.71	36.64	40.11	34.88
Technique in [6]	41.14	42.64	37.33	42.54	37.71
Proposal	42.11	48.81	41.63	42.81	37.75

spection of the de-interlaced frames from the Carphone sequence shown in Figure 7.

4 Conclusions

The algorithm presented herein is the result of the application of fuzzy logic-based systems to video processing. Especially this approach tackles the problem of de-interlacing, which is currently more demanded in consumer devices. The algorithm overcomes the performance of other well-known de-interlacing algorithms by adapting the interpolation strategy to the presence of motion and edges. To achieve it, the approach includes three fuzzy systems: one is used to combine a spatial and a temporal interpolator according to the level of motion, and the other two provide a smart temporal and spatial interpolator.

Acknowledgements

This work has been supported in part by the Spanish MEC Projects TEC2005-04359 and DPI2005-02293, and by the Projects TIC2006-635 and TEP2006-375 from the Andalusian regional Government.

References

[1] J. Whitaker. *Television transmissions systems*, chapter book of 'Standard handbook of video and television engineering'. McGraw-Hill Editorial, Blacklick OH (USA), 2002.

[2] G. de Haan. *De-interlacing*, chapter book of 'Digital Video. Post Processing', pages 185-201, University Press Eindhoven, Sep. 2006.

[3] A. M. Bock. *Motion-adaptive standards conversion between formats of similar field rates.* Signal Processing: Image Communication, vol.6, no.3, pages 275-280, 1994.

[4] H. Jiang, D. Huu and E. Tinyork. *Motion adaptive deinterlacing.* United States Patent (US 6,459,455), Oct. 2002.

[5] D. Van de Ville, R. Van de Wall, W. Philips and I. Lemahieu. *Motion adaptive de-interlacing using fuzzy logic.* Proc. International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU), pages 1989-1996, Jul. 2002.

[6] J. Gutiérrez-Ríos, F. Fernández-Hernández, J. C. Crespo and G. Treviño. Motion adaptive fuzzy video de-interlacing method based on convolution techniques. In *Proceedings of the conference IPMU'2004*, pages 1635-1642, Perugia, Italy, Jul. 2004.

[7] P. Brox I. Baturone S. Sánchez-Solano J. Gutiérrez-Ríos and F. Fernández-Hernández. *A fuzzy edge-dependent motion adaptive algorithm for de-interlacing.* Fuzzy Sets and Systems, vol.158, no.3, pages 337-347, Feb.2007.

Figure 7: De-interlaced frames of Carphone sequence.

[8] P. Brox I. Baturone and S. Sánchez-Solano. A fuzzy motion adaptive algorithm for interlaced-to-progressive conversion. In *Proceedings of the Conference IPMU'2006*, París, France, Jul. 2006.

[9] P. Brox I. Baturone and S. Sánchez-Solano. Fuzzy motion adaptive algorithm for video de-interlacing. In *Proceedings of the Conference KES'2006*, Bournemouth, United Kingdom, Oct. 2006.