FPGA IMPLEMENTATION OF A FUZZY BASED VIDEO DE-INTERLACING ALGORITHM

P. Brox, S. Sánchez-Solano, I. Baturone, A. Barriga

Instituto de Microelectrónica de Sevilla - Centro Nacional de Microelectrónica
Avda. Reina Mercedes s/n, (Edif. CICA),
E-41012, Sevilla, Spain.

Proc. SPIE Europe International Symposium on Microtechnologies
for the New Millennium, Sevilla, May 9-11, 2005.

This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author’s copyright. In most cases, these works may not be reposted without the explicit permission of the copyright holder.
FPGA implementation of a fuzzy based video de-interlacing algorithm

P. Brox, S. Sánchez-Solano, I. Baturone and A. Barriga
Instituto de Microelectrónica de Sevilla, Centro Nacional de Microelectrónica
Avda. Reina Mercedes s/n, Edificio CICA, 41012 Seville, Spain (brox@imse.cnm.es)

ABSTRACT

De-interlacing algorithms are used to convert interlaced video into progressive scan format. Among the different techniques reported in the literature, motion adaptive de-interlacing techniques that combine spatial and temporal interpolation according to the presence of motion achieve good results with a low computational cost. This paper presents the FPGA implementation of a motion adaptive algorithm which employs fuzzy logic in detecting motion and edges. Motion, which is evaluated at each pixel of the de-interlaced frame, determines the interpolation between an enhanced edge-dependent line average method and field insertion. Extensive simulations with video sequences show the advantages performance of the proposed method over other well-known de-interlacing techniques. The hardware implementation of the algorithm has been carried out on a FPGA obtaining a low-cost solution for real-time processing.

Keywords: De-interlacing motion adaptive, edge-dependent interpolation, fuzzy logic

1. INTRODUCTION

The ordinary TV formats (NTSC, PAL, SECAM) use an interlaced scan format. However, the advent of devices which require a progressive display (e.g. high quality monitors, projectors, HDTV systems) have provided the development of the interlaced to progressive conversion or de-interlacing algorithms [1]. They can be classified into three categories: spatial (or intra-field) techniques, temporal (or inter-field) techniques and hybrid methods. The first ones demand low hardware requirements since no additional field buffers are needed. Among them, line doubling and line average are the simplest. However, they introduce errors when de-interlacing diagonals or curves lines which result in the non-desired stair-case effect. This is avoided by the use of edge-dependent interpolation algorithms such as the ELA algorithm [2], which is a simple algorithm that achieves good results in reconstructing clear edges although introduces errors in areas where the edges are no clear. Therefore de-interlaced frames using solely intra-field techniques contain visual artefacts (blur and flicker) especially in static backgrounds. Intra-field techniques correctly reconstruct the original vertical resolution when no motion is presented although visual defects which are known as ghosting appear in moving parts. Hybrid techniques which combine inter and intra-field methods have been proposed to overcome these drawbacks. Vertical-temporal (VT) filtering interpolates between spatial and temporal samples independently of motion [3]. Motion adaptive de-interlacing techniques improve the results adapting the interpolation strategy to motion [4-6]. The idea is to select an inter-field method in absence of motion and an intra-field method in moving areas. Others researchers propose the weighted median filtering algorithms where the interpolated sample is obtained as the median luminance value of a group of spatial and temporal neighbors [7]. Weights of the median filter control the probabilities of intra or inter-field methods. In [8] a median filtering technique is implemented considering the minimum directional correlation in spatial and temporal domains.
The quality of a motion adaptive de-interlacing technique is related to the accuracy of the motion detector. Since the motion degree is something fuzzy, fuzzy logic-based motion detectors have been used to improve the performance of motion adaptive de-interlacing techniques [4-5].

The motion adaptive de-interlacing algorithm described in this paper exploits fuzzy logic to improve the performance not only of the motion detector but also of the intra-field technique employed. It implements an interpolation between an enhanced edge-dependent interpolation technique and the field insertion technique, which achieves a good quality of de-interlaced video with a low computational complexity. A cost-efficient hardware implementation of the algorithm is also presented herein.

This paper is organized as follows. Section 2 describes the algorithm while simulations results of de-interlacing video sequences are given in Section 3. Hardware implementation is presented in Section 4. Finally, conclusions are given in Section 5.

2. A FUZZY-BASED MOTION ADAPTIVE DE-INTERLACING TECHNIQUE

Let us represent by $I(x,y,t)$ the luminance component of a de-interlaced pixel, where $x$ and $y$ are spatial coordinates, and variable $t$ determines the order in the frame sequence. Motion adaptive de-interlacing techniques calculate $I(x,y,t)$ by applying:

$$I(x,y,t) = (1 - \gamma(x,y,t))I_{\text{INTER}}(x,y,t) + \gamma(x,y,t)I_{\text{INTRA}}(x,y,t)$$ (1)

where $I_{\text{INTER}}(x,y,t)$ and $I_{\text{INTRA}}(x,y,t)$ correspond to the output of inter-field and intra-field methods respectively, and the motion detector output is denoted by $\gamma(x,y,t)$. They combine the advantages of intra and inter-field methods according to the presence of motion. Intra-field technique is selected in moving areas ($\gamma=1$) whereas inter-field technique is adopted in static areas ($\gamma=0$). The fuzzy-based motion adaptive technique proposed in this paper follows the block diagram showed in Figure 1. The fuzzy motion detector and intra-field techniques employed are described in detail in the next sections.

2.1. FUZZY MOTION DETECTOR

The fuzzy motion detector is that proposed in [5], which in turns inspired on the method developed in [4] but improving its results and reducing the computational cost considerably. It uses inference techniques based on fuzzy logic to evaluate the level of motion. The input to the fuzzy motion detector is a ‘difference matrix’ $H(x,y,t)$ described as follows:

$$H(x,y,t) = \frac{I(x,y,t-1) - I(x,y,t+1)}{2}$$ (2)

The presence of motion is detected for each sample of the de-interlaced frame evaluating a set of fuzzy rules. The rule base of the fuzzy motion detector is showed in Table 1. The first rule establishes that motion is present ($M(x,y,t) = \text{YES}$) when the values of $H(x,y,t)$ in the current pixel or in the pixels on both sides of the current pixel are large. The others rules are similar taking into account samples from the previous and subsequent fields. This heuristic
knowledge is fuzzy since the concept of LARGE is not understood as a threshold value but as fuzzy one. Hence, it is represented by a fuzzy set whose membership function changes continuously instead of abruptly between 0 and 1 membership values, as shown in Figure 2a. The pixels evaluated by the fuzzy system are shown in Figure 2b.

Table 1
RULE BASE OF FUZZY MOTION DETECTOR

<table>
<thead>
<tr>
<th>if</th>
<th>antecedents</th>
<th>then</th>
<th>consequent</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$(H(x+2,y,t)$ is LARGE) and $(H(x+1,y,t)$ is LARGE) and $(H(x,y,t)$ is LARGE) and $(H(x-1,y,t)$ is LARGE)</td>
<td></td>
<td>$M(x,y,t) = \text{YES}$</td>
</tr>
<tr>
<td></td>
<td>and $(H(x-2,y,t)$ is LARGE)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$(H(x+2,y-1,t-1)$ is LARGE) and $(H(x+1,y-1,t-1)$ is LARGE) and $(H(x,y-1,t-1)$ is LARGE)</td>
<td></td>
<td>$M(x,y-1,t-1) = \text{YES}$</td>
</tr>
<tr>
<td></td>
<td>and $(H(x-1,y-1,t-1)$ is LARGE) and $(H(x-2,y-1,t-1)$ is LARGE)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$(H(x+2,y+1-1,t-1)$ is LARGE) and $(H(x+1,y+1,t-1)$ is LARGE) and $(H(x,y+1,t-1)$ is LARGE)</td>
<td></td>
<td>$M(x,y+1,t-1) = \text{YES}$</td>
</tr>
<tr>
<td></td>
<td>and $(H(x-1,y+1,t-1)$ is LARGE) and $(H(x-2,y+1,t-1)$ is LARGE)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The inference process of the fuzzy logic-based system can be approximated using bi-dimensional convolution techniques according to the expression (3) and is explained in detail in [5]. This reduces considerably the computational process and provides high flexibility because it allows using different weights in the coefficients of the matrix C showed in (5).
The result of convolution, $A$, is the input to the fuzzification process in which the membership degree to the fuzzy set shown in Figure 2 is evaluated. After convolution and fuzzification the result $\gamma(x,y,t)$ is obtained. Its value is in the interval $[0,1]$ and indicates the level of motion.

2.2. FUZZY EDGE-BASED LINE AVERAGE METHOD

The ELA method is as well-known intra-field technique which offers advantages in reconstructing the edges of a de-interlaced image. However, some errors appear when the edges are not clear or there is noise. These drawbacks are eliminated by using an enhanced algorithm named fuzzy ELA, which is based on modelling heuristic knowledge with a simple fuzzy system.

Figure 3a shows the samples used by the ELA algorithm to interpolate the pixel value $I(x,y,t)$. The pseudo-code of this algorithm is as follows:

$$a = A - F \quad b = B - E \quad c = C - D$$

if $\min[a,b,c] = a \Rightarrow I(x,y,t) = (A + F)/2$

if $\min[a,b,c] = c \Rightarrow I(x,y,t) = (C + D)/2$

else $\Rightarrow I(x,y,t) = (B + E)/2$
The problem with ELA is that minimum correlation does not always correspond to the direction of an edge. To ensure the presence of an edge new restrictions are imposed: an edge is clear in direction a not only if a is small but if b and c are large (and this rule is also valid for direction c). In case of small correlations in directions a and c, neither there is an edge nor vertical linear interpolation performs well, so the best option is a linear interpolation between the samples with small correlations. In other cases, a vertical linear interpolation would be the most adequate. This heuristic knowledge is modelled by a fuzzy system where the concepts of SMALL, LARGE and strongly SMALL are not understood as threshold values but as fuzzy ones (Figure 3b).

The rule base of the fuzzy system is described in Table 2. The minimum operator is used as connective and of antecedents and the activation degrees of the rules, $\alpha_i$, are calculated as follows:

$$
\begin{align*}
\alpha_1 &= \min\left(\mu_{\text{SMALL}}(h), \mu_{\text{LARGE}}(h), \mu_{\text{LARGE}}(h)\right) \\
\alpha_2 &= \min\left(\mu_{\text{LARGE}}(h), \mu_{\text{LARGE}}(h), \mu_{\text{SMALL}}(h)\right) \\
\alpha_3 &= \min\left(\mu_{\text{strongly SMALL}}(h), \mu_{\text{LARGE}}(h), \mu_{\text{strongly SMALL}}(h)\right) \\
\alpha_4 &= 1 - \alpha_1 - \alpha_2 - \alpha_3
\end{align*}
$$

(7)

The output of the fuzzy system is calculated by applying the Fuzzy Mean defuzzification method. Substituting the consequents, $c_i$, by their values (showed in Table 2) and applying that $\sum_{i=1}^{4} \alpha_i = 1$ equal to 1 we obtain:

$$
I_{\text{INTRA}}(x,y,t) = \frac{\sum_{i=1}^{4} \alpha_i c_i}{\sum_{i=1}^{4} \alpha_i} = \alpha_1 \left( \frac{A + F}{2} \right) + \alpha_2 \left( \frac{C + D}{2} \right) + \alpha_3 \left( \frac{A + F + C + D}{4} \right) + \alpha_4 \left( \frac{B + E}{2} \right)
$$

(8)

This is the fuzzy intra-field technique used in our motion adaptive de-interlacing algorithm.

![3x3 window for the ELA algorithm in frame t. (b) Memberships functions of the fuzzy sets used.](image)

Figure 3: (a) 3x3 window for the ELA algorithm in frame t. (b) Memberships functions of the fuzzy sets used.
3. SIMULATION RESULTS

The proposed algorithm has been tested with several progressive video sequences which were artificially interlaced. Our algorithm has been analyzed and compared with others well-known algorithms of less or similar computational cost. In particular, three intra-fields methods (line doubling, line average and ELA) and the simplest intra-field method (insertion of previous field). The rest of algorithms considered have been hybrid methods which realize a spatial-temporal interpolation: a VT filtering reported in [3], a weighted median filtering described in [7], the technique developed in [8] and the motion adaptive algorithms in [4-5]. Fifty frames of each sequence have been analyzed and the mean of PSNR values have been considered as figure of merit as shown Table 3.

Figure 4 allows evaluating subjectively the quality of the methods mentioned above for one frame of the de-interlaced ‘Mother & Daughter’ video sequence. The moving area in the frame selected corresponding to the mother’s hand. The analysis of frames corroborates what is known about inter-field techniques that introduce a ghosting effect in areas with a major presence of motion (Figure 4c). On the contrary, intra-field methods reconstruct better this area but introduce several visual defects in the background static area (Figure 4d-f). The proposed algorithm gets more effective visual quality avoiding these drawbacks as shown Figure 4l.

The advantageous of our algorithm over the hybrid methods considered are more evident in the zooms of the de-interlaced frame from ‘Salesman’ sequence (Figure 5). It reconstructs considerably better the edges in moving area (salesman’s hand) as illustrated Figure 5j.

4. HARDWARE IMPLEMENTATION

The proposed algorithm has been implemented following three architectures which differ in the parallelism employed to realize the bidimensional convolution (expression (3)) and to evaluate the rule’ activation degrees involved in intra-field method. The parallel design provides a new interpolated pixel value every clock cycle and requires two multipliers to implement the interpolator (expression (1)) and three multipliers to implement the fuzzy ELA intra-field method (taking into account the expression of \( \alpha_4 \) in (7)). The mixed design realizes the operations in sequential form for.
Table 4

POST-SYNTHESIS RESULTS OF THE FPGA IMPLEMENTATIONS

<table>
<thead>
<tr>
<th>FPGA Virtex 2 xc2v500_6fg256</th>
<th>Parallel</th>
<th>Mixed</th>
<th>Sequential</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. Slices</td>
<td>1017 (33%)</td>
<td>569 (18.5%)</td>
<td>421 (13.7%)</td>
</tr>
<tr>
<td>Processing Period (ns)</td>
<td>11.7</td>
<td>28.7</td>
<td>98.6</td>
</tr>
<tr>
<td>Processing Frequency (MHz)</td>
<td>85.4</td>
<td>38.4</td>
<td>10.14</td>
</tr>
<tr>
<td>Resolution pixels</td>
<td>2,849,002</td>
<td>1,161,440</td>
<td>338,066</td>
</tr>
</tbody>
</table>

the three values of each row of the matrix, whereas operations from different rows of the matrix are realized in parallel. Activation degrees of intra-field method are processed in sequential form and the result is obtained every three clock cycles. To produce the most effective realization of the algorithm in terms of the hardware resources employed a global sequential architecture of the motion adaptive algorithm is proposed. A new interpolated pixel value is obtained every fifteen clock cycles. The mixed and sequential architectures, whose block diagrams are shown in Figures 6 and 7 respectively, require two multipliers to realize the interpolation and one for the fuzzy ELA algorithm.

The designs have been developed with SysGen (System Generator), which is a tool from Xilinx integrated in the Matlab environment. SysGen accelerates process since allows simulating in the Simulink environment and, especially, generating the VHDL descriptions. Behaviour of these descriptions has been also tested by using the ModelSim simulator with a testbench file. The generated digital systems compute data using two’s complement binary signal with a resolution of 16 bits. The results obtained with the algorithms programmed in Matlab are very similar to the simulation ones of the hardware implementations.

The implementations have been developed on an FPGA Virtex2 from Xilinx which provides a wide variety of flexible features (block RAMS, multipliers), 500k system gates and an internal clock speed of 420 MHz. Table 4 shows the post-synthesis results in terms of area and speed of the implementations. The last row of Table 4 shows the maximum...
resolution per frame that could be provided for frame sequences displayed at a rate of 30 frames per second. It can be seen how the sequential design achieves a resolution over VGA (640x480), the mixed over XGA (1024x768), and the parallel over HDTV (1080x1920).

5. CONCLUSIONS

The proposed motion adaptive de-interlacing algorithm employs fuzzy logic to improve the performance of the motion detector and also the intra-field technique. The improvements are especially significant in the reconstruction of moving areas of de-interlaced frames due to the enhanced ELA technique employed. This has been proven by extensive simulation results of video sequences. These advantages are obtained with a low increase in computational cost. The hardware implementation of the algorithm is very simple and has been proven with FPGA implementations.
Figure 4: (a) Original progressive frame from Mother and Daughter sequence. (b) Artificial interlaced frame used to test the de-interlacing algorithms. De-interlaced frame applying line insertion (c), line doubling (d), line average (e), ELA method (f), VT filtering (g), method reported in [8] (h), weighted median filtering [7] (i), motion adaptive method described in [4] (j), motion adaptive [5] (k) and the proposed technique (l).
Figure 5: (a) Original progressive frame from Mother and Daughter sequence. De-interlaced frame applying line insertion technique (b), line doubling (c), line average (d), ELA method (e), VT filtering (f), method reported in [8] (g), weighted median filtering [7] (h), motion adaptive method described in [4] (i), motion adaptive [5] (j) and the proposed technique (k).
REFERENCES


