Information Sciences 179 (2009) 2194–2207

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Designing of a type-2 fuzzy logic filter for improving edge-preserving restoration of interlaced-to-progressive conversion

Gwanggil Jeon [a], Marco Anisetti [b], Valerio Bellandi [b], Ernesto Damiani [b,*], Jechang Jeong [a]

[a] Department of Electronics and Computer Engineering, Hanyang University, 17 Haengdang-dong, Seongdong-gu, Seoul, South Korea
[b] Department of Information Technology, University of Milan, via Bramante, 65, 26013 Crema (CR), Italy

ARTICLE INFO

Article history:
Received 11 December 2007
Received in revised form 28 October 2008
Accepted 16 January 2009

Keywords:
Fuzzy weight
Deinterlacing
Image restoration
Image enhancement
Type-2 fuzzy system

ABSTRACT

This paper focuses on recently advanced fuzzy models and the application of type-2 fuzzy sets in video deinterlacing. The final goal of the proposed deinterlacing algorithm is to exactly determine an unknown pixel value while preserving the edges and details of the image. To begin, we will discuss some artefacts of spatial, temporal, and spatio-temporal domain deinterlacing methods. In order to address the aforementioned issues, we adopted type-2 fuzzy sets concepts to design a weight evaluating approach. In the proposed method, the upper and lower fuzzy membership functions of the type-2 fuzzy logic filters are derived from the type-1 (or primary) fuzzy membership function. The weights from upper and lower membership functions are considered to be multiplied with the candidate deinterlaced pixels. Experimental results proved that the performance of the proposed method was superior, both objectively and subjectively to other different conventional deinterlacing methods. Moreover, the proposed method preserved the smoothness of the original image edges and produced a high-quality progressive image.

1. Introduction

Deinterlacing converts each field into a frame, so the number of pictures per second remains constant while the number of lines per picture is doubled [13]. The interlaced scanning format has been used widely in the TV industry and within consumer electronic video cameras including NTSC, PAL, and SECAM. The strength of the interlaced scanning format lies in that the refreshing rate is doubled without increasing the bandwidth. However, an interlaced TV signal in the vertical direction does not satisfy the demands of the Nyquist sampling theory. There is a trade-off between the video signal's bandwidth requirements and an optimal frame rate that would not be noticeably slow for human perception [20]. Because of the intrinsic nature of the interlaced scanning process, troublesome visual artefacts including like edge flickering, interline flickering, and line-crawling will appear when displaying interlaced content on a progressive device such as a PC monitor [34]. Therefore, in order to guarantee compatibility with existing TV broadcasting standards, deinterlacing is needed to convert interlaced video to progressive video.

In order to resolve these issues, deinterlacing methods have been investigated by many authors and various methods have been proposed with different degrees of complexity and qualities of reconstruction [1,5,6,8,11,33,36,41,45,47]. These methods can be roughly classified into two categories: spatial domain deinterlacing methods [1,10,38] and temporal domain deinterlacing methods [5,6,11,33,41,45,47]. In particular, temporal domain deinterlacing methods can be classified into methods with motion compensation [5,45] and methods without motion compensation [1,6,9,11,33,36,41,47]. Spatial domain deinterlacing methods are intra-field algorithms in which the pixel on the missing line is interpolated solely from

* Corresponding author. Tel.: +39 02 503 30064.
E-mail address: damiani@dti.unimi.it (E. Damiani).

0020-0255/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2009.01.044

the available lines in the current field. Motion adaptive deinterlacing algorithms analyze the characteristics of the motion in order to choose the appropriate interpolation scheme [6,11,33]. Motion compensated deinterlacing algorithms measure the inter-field motion and then align data between the two video fields, maximizing the vertical resolution of the image. Motion adaptive deinterlacing methods are chosen more often because of error propagation and computational complexity, especially in real-time applications. Usually, inter-field interpolation is used to prevent blurriness or dimness in static scenes, while intra-field interpolation is used for preventing artefacts when motion is detected. In this paper, we focus on a motion adaptive deinterlacing technique which does not employ a motion compensation process.

The fuzzy theory concept has been one of the most important areas in mathematics and has been successfully applied in process control in which binary decisions are not able to produce good results [46]. Some examples of the efforts of fuzzy controllers in low-level image processing are fuzzy rule-based motion detectors [19,37,38], fuzzy edge detectors [16], fuzzy relation equation for codec [23], fuzzy rate control for MPEG video [42], fuzzy operators for filtering and edge detection [14], rough sets-based deinterlacing [15], and fuzzy edge-dependent motion adaptive deinterlacing [3]. A method for detecting edge direction from an interlaced image has been constructed according to the gradient of the image [31,32].

A technique that is not employed in video deinterlacing is the type-2 fuzzy set theory [18,25], which was recently introduced by Zadeh [49] and has been researched by many authors [8,12,17,18,21,22,26–28,39,43,44] as an extension of the concept of an ordinary fuzzy set. Type-2 fuzzy sets have grades of membership that are themselves fuzzy. A number of algorithms applied to image processing exist. The control applications began to appear with an iterative method that allows for the fast execution of type-2 fuzzy systems [24,25]. Some contributions are used in medical applications [16,35]. Meanwhile, signal processing, like controls, has used only type-2 interval methods to date [17,21,30]. In the last decade, various type-2 neuro-fuzzy systems have been developed [40]. They are characterized by learning properties and natural language descriptions.

In this paper, a method to detect edge direction and measure the weight of each edge direction technique through type-2 fuzzy logic is presented. It will be shown that a fuzzy technique combined in a deinterlacing method is effective in reducing the method's sensitivity to edge direction detection errors and improves visual quality. We propose that the performance, such as edge-preserving and PSNR results through weight measuring, can be improved by applying type-2 fuzzy logic. The remainder of the paper is structured as follows. In Section 2, we explain some issues of the current contributions to this field. In Section 3, the proposed type-2 fuzzy deinterlacing (T2FD) algorithm is introduced. Simulation results demonstrating the usefulness of the approach are presented in Section 4. Finally, conclusions are presented in Section 5.

2. Background and existing contributions

Some issues of the spatial, temporal, and spatio-temporal domain deinterlacing methods are briefly discussed and investigated here. Fig. 1 shows a 3D localized window to interpolate the missing pixel value $x_m(i,j,k)$, where m represents the utilized method. The parameters i, j, and k refer to the column number, line number, and the field number, respectively. Fig. 2 shows the motion direction detection in the spatial or temporal domain with the displacement (Δ) and degree (θ) of edge direction. The abbreviations u, d, l, r, p, and n represent *up*, *down*, *left*, *right*, *previous*, and *next*, respectively. The parameter $LD_{\Phi,\theta,\Delta}$ is the luminance difference in a domain $\Phi\{=(SD,TD)\}$ with displacement from the centre pixel $\Delta\{=(+,z,-)\}$ and edge direction represented by $\theta\{=(45^\circ, 90^\circ, 135^\circ)\}$. The abbreviations *SD* and *TD* denote the *spatial domain* and *temporal domain*, respectively. The values $\{+,z,-\}$ of the displacement parameter Δ represent that the filter is a shifted $\{1,0,-1\}$ pixel in the horizontal direction. The equations to calculate the $LD_{\Phi,\theta,\Delta}$ values are shown in (1)

$$\begin{aligned}
&LD_{SD,45^\circ,-} = |u - dll|, \quad LD_{SD,45^\circ,z} = |ur - dl|, \quad LD_{SD,45^\circ,+} = |urr - d| LD_{TD,45^\circ,-} = |p - nll|,\\
&LD_{TD,45^\circ,z} = |pr - nl|, \quad LD_{TD,45^\circ,+} = |prr - n|\\
&LD_{SD,90^\circ,-} = |ul - dl|, \quad LD_{SD,90^\circ,z} = |u - d|, \quad LD_{SD,90^\circ,+} = |ur - dr| LD_{TD,90^\circ,-} = |pl - nl|,\\
&LD_{TD,90^\circ,z} = |p - n|, \quad LD_{TD,90^\circ,+} = |pr - nr|\\
&LD_{SD,135^\circ,-} = |ull - d|, \quad LD_{SD,135^\circ,z} = |ul - dr|, \quad LD_{SD,135^\circ,+} = |u - drr| LD_{TD,135^\circ,-} = |pll - n|,\\
&LD_{TD,135^\circ,z} = |pl - nr|, \quad LD_{TD,135^\circ,+} = |p - nrr|
\end{aligned} \tag{1}$$

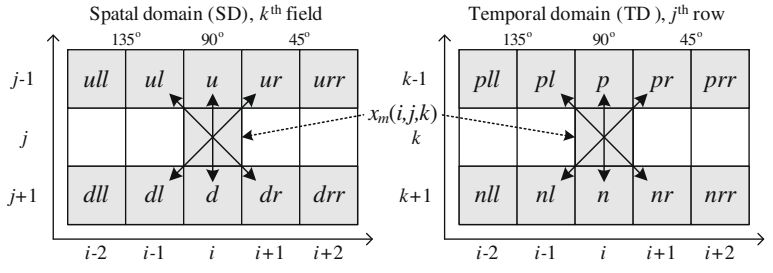

Fig. 1. Spatio-temporal window for the direction-based deinterlacing: (a) spatial direction filter and (b) temporal direction filter.

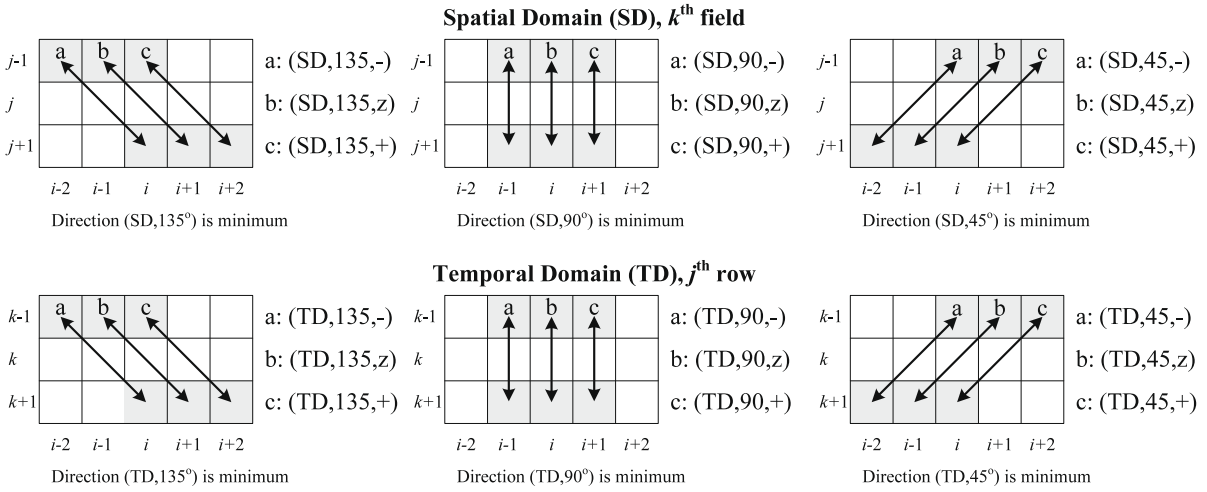

Fig. 2. Motion direction (Φ,θ,Δ) detection in spatial and temporal domains.

2.1. Spatial domain existing contributions

The spatial domain interpolation algorithm exploits the spatial correlation to interpolate the missing line. One well known spatial domain interpolator is the edge-based line average (ELA) method [9]. This method uses directional correlations between pixels to linearly interpolate a missing line between two adjacent lines in the interlaced signal before displaying the interpolated line. The ELA method is broadly used because it can be easily performed in a hardware system and requires a simple level of computation. Moreover, it performs well in regions with a dominant edge because the edge direction can be estimated accurately. However, in a high frequency region or horizontal edge region, poor visual quality may result because of unclear edges or inaccurate estimations of edge orientation.

A 2D 3-by-2 localized window is used to calculate directional correlations and interpolate the current pixel. Therefore, a parameter pair $\{SD,\theta,z\}$ is assigned to $\{\Phi,\theta,\Delta\}$. $LD_{SD,\theta,z}$ is then used to calculate the direction of the highest spatial correlation. The ELA operated pixel $x_{ELA}(i,j,k)$ is then calculated

$$x_{ELA}(i,j,k)=\begin{cases} x_{org}(i,j,k) & \text{if } j \bmod 2 = N \bmod 2,\\ \begin{cases}(ur+dl)/2 & \text{if } \min\ (LD_{SD,\theta,z})=LD_{SD,45^\circ,z},\\ (u+d)/2 & \text{if } \min\ (LD_{SD,\theta,z})=LD_{SD,90^\circ,z},\\ (ul+dr)/2, & \text{otherwise.}\end{cases}\end{cases} \tag{2}$$

Here (i,j,k) designates the position. The original image $x_{org}(i,j,k)$ is the input field defined for "$j \bmod 2 = N \bmod 2$" only. The parameter $\min(LD_{SD,\theta,z})$ represents the minimum value among the three values of $LD_{SD,45,z}$,$LD_{SD,90,z}$, and $LD_{SD,135,z}$. Fig. 3 shows the deinterlaced results from the ELA method.

2.2. Temporal domain existing contributions

The Weave method is the simplest method among inter-field deinterlacing methods [41] and its output pixel value $x_{Weave}(i,j,k)$ is defined in (3)

$$x_{Weave}(i,j,k)=\begin{cases} x(i,j,k) & \text{if } j \bmod 2 = N \bmod 2\\ p, & \text{otherwise}\end{cases} \tag{3}$$

It is known that the objective and subjective performances of the Weave method are better than that of the spatial domain deinterlacing methods in a static area. However, systems based on exclusively temporal domain interpolators cause a multiplicity of artefacts when there are moving objects in the image. Fig. 4b shows an example of artefacts that are caused by the Weave method. The edges exhibit significant serrations and the line-crawling effect occurs in motion areas, both of which are unacceptable artefacts in a broadcast or professional television environment.

2.3. Spatio-temporal domain existing contributions

The vertical-temporal median filter (VTMF), which adapts to motion or edges, is one of the most popular motion adaptive deinterlacing methods [11]. The deinterlaced pixels are found as the median luminance value of the upper and lower pixels in the current field and the temporal neighbour in the previous field

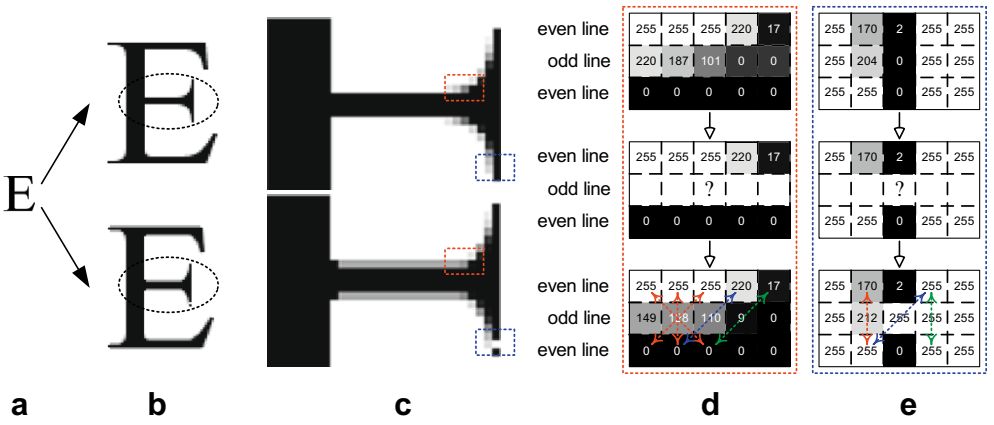

Fig. 3. Examples of errors with the ELA method: (a) original character image "E", (b) 350% zoomed original character (upper) and 350% zoomed ELA operated deinterlacing character (lower), (c) 1200% zoomed original character (upper) and 1200% zoomed ELA operated deinterlacing character (lower), (d) comparison between original and ELA operated results in red and (e) comparison between original and ELA operated results in blue. (For interpretation of the references in colour in this figure legend, the reader is referred to the web version of this article.)

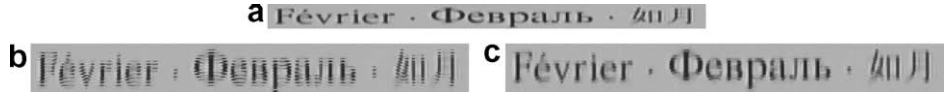

Fig. 4. Subjective view results for 120th Mobcal sequence [56]: (a) even field of the original sequence; (b) serration artefact due to the Weave method and (c) and alias effect due to the VTMF method.

$$x_{VTMF}(i,j,k) = \begin{cases} x_{org}(i,j,k) & \text{if } j \bmod 2 = N \bmod 2, \\ median(u,d,p), & \text{otherwise} \end{cases} \tag{4}$$

Unfortunately, however, with this filter it is also possible to shuffle the information on different lines causing disturbances to appear. To demonstrate the drawbacks of VTMF, Fig. 4c shows a VTMF operated image. At low resolution or details, there is little influence on the input image. However, with higher frequencies, the impact of the defect increases so that the fundamental details become coarser.

3. Type-2 fuzzy logic filter for deinterlacing

3.1. Fuzzy filters for image processing

The fuzzy set theory was first introduced by Zadeh in 1965 [48]. He used the word "fuzzy" to generalize the mathematical concept of the "set" to the "fuzzy set". Based on the fuzzy set theory, this technique provides a new and flexible mathematical framework to cope with qualitative properties, such as the edginess or contrast of a region as well as the ambiguity and vagueness often present in digital images [3,14,19,29,37,38]. The main difference from other methodologies in image processing is that input data of the gray level domain will occur in the so called membership plane of a fuzzy domain where one can use the great diversity of fuzzy measuring theories to aggregate the membership values to classify data or to make decisions using a fuzzy inference. The computed membership values are re-transformed in the gray level domain to generate new histograms, modified gray levels, image segments, or classes of objects.

For many practical problems, variables have different importances and make different contributions to the outputs. Therefore, it is necessary to find an optimal normalization and assign proper importance factors to the variables. From the previous section, we have seen that if only certain temporal information is considered in a motion region, some drawbacks such as serration or alias artefacts are introduced that degrade visual quality, especially on the contour of objects. Consequently, all edge direction information, not just the use of only one edge direction, should be considered together to calculate missing pixels. Motivated by these observations, we proposed the T2FD method.

3.2. Modelling of non-linear system using type-2 fuzzy logic systems

Fuzzy logic systems (FLSs) have been widely used in different industries. The membership functions and rules of FLS are formulated using numerical data or linguistic information. However, there is ambiguity and uncertainty associated with the data or information. A type-2 fuzzy set is able to handle uncertain information effectively. For example, in the study of noise

reduction, it is known that type-1 FLSs have the shortcoming of being sensitive to external noise [37,38]. This deficiency can be addressed by using type-2 FLSs.

In the following, type-2 FLSs are generally described and type-1 FLSs are compared with type-2 FLSs by presenting and analyzing experimental results. It will be shown that the efficiency of type-2 FLSs is superior to type-1 FLSs. Type-2 FLSs usually use type-2 fuzzy sets, where a type-reduction process is required for output processing. However, because of the manufacturing complexity and difficulty, we made the systems simple. Therefore, to moderate the computational burden, we skip the type-reduction process and calculate the output simply.

3.3. The T2FD architecture

In general, the fuzzy architectures constitute the fuzzifier, fuzzy inference engine, and defuzzifier in one structure. Fig. 5 shows the structure of the proposed T2FD filter. The interpolator is designed by integrating a candidate deinterlaced pixel calculator, a weighted average processor, six weighting factor multipliers (two domains with each domain having three degrees), and 18 type-2 fuzzy filters for weights evaluation where each type-2 fuzzy filter has three *LD* values as their inputs. The operator inputs the neighbour pixels, $x(i,j,k)$, in its filtering window and outputs the restored value of the missing pixel, $x_{T2FD}(i,j,k)$. In other words, each of the 18 fuzzy filters inputs the three *LD* values along with (Φ,θ) for different Λ values and then outputs a type-2 interval fuzzy set representing the uncertainty interval for the restored value of the deinterlaced pixels. Three weight values $(w_{\Lambda,\Phi,\theta,\Lambda})$ with the same domain and the same edge direction are combined into one weighting factor $(\omega_{\Phi,\theta})$, where the parameter Λ represents a fuzzy membership function such as PM (primary membership), UM (upper membership), and LM (lower membership). These fuzzy membership functions will be introduced in the following section. The scalar values obtained at the outputs of the six weighting factors and six candidate deinterlaced pixels become inputs of a defuzzifier stage and these six candidate deinterlaced pixel values are finally converted into a single scalar value by the defuzzifier (weighted average processor). Each defuzzifier transforms the input fuzzy set into a single scalar value by performing defuzzification. The output of the proposed T2FD filtering operator represents the restored value of the centre pixel of the filtering window.

3.4. Weight measurement using the S-type SMALL primary membership (PM) function

Since a type-2 fuzzy logic filter is derived from a type-1 (or primary, regular, skeleton) fuzzy logic filter, type-1 fuzzy logic filters need to be defined before designing the T2FD system. The type-1 fuzzy logic filter utilized is a direction-dependent interpolation technique, which is based on a sample correlation and fuzzy theory for weighting, where the detection of edges in all possible directions is followed by a weight-multiplied interpolation. The edge direction detector utilizes directional correlations among pixels in order to evaluate the weight in each direction to interpolate a missing line.

In order to design the desired weight measuring system, determining whether it is necessary to evaluate each edge direction is performed. The $LD_{\Phi,\theta,\Lambda}$ is computed between two adjacent pixels in the 18 directions. Then, the maximum observed

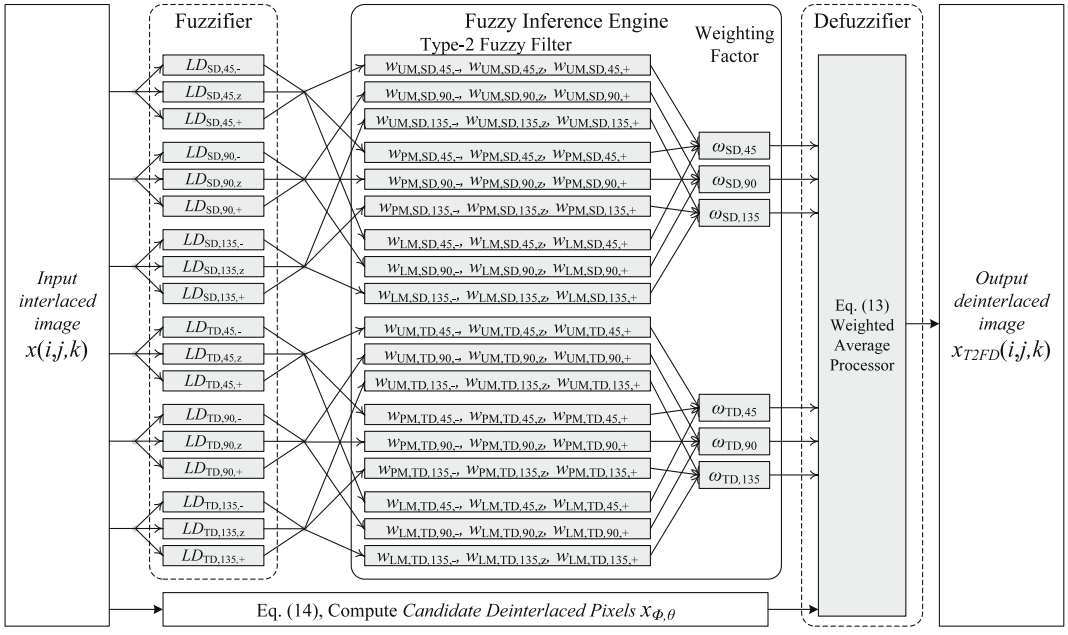

Fig. 5. The structure of the proposed type-2 fuzzy deinterlacing system. Each of the three type-2 fuzzy filters evaluates pixel neighborhoods in the 45°, 90°, and 135° directions. Boxes in gray represent the *membership plain*.

$LD_{\Phi,\theta,\Delta}$ value among the 18 $LD_{\Phi,\theta,\Delta}$ values in the 3D localized window is determined and is denoted as $LD_{\max}$. For each image pixel $x(i,j,k)$ the fuzzy index, $FI_{\Phi,\theta,\Delta}$, is computed as,

$$FI_{\Phi,\theta,\Delta} = \frac{LD_{\Phi,\theta,\Delta}}{LD_{\max}}. \tag{5}$$

Fig. 6 shows the S-type SMALL primary membership function, $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$, where the vertical axis represents a membership degree ranging from 0.0 to 1.0 and the horizontal axis of this function represents all the possible $FI_{\Phi,\theta,\Delta}$ values with a range of 0.0 to 1.0. The $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$ is calculated in (6). The membership degree of $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$ is assumed to be the weight $w_{\mathrm{PM},\Phi,\theta,\Delta}$

$$\mu_{\mathrm{PM}} = w_{\mathrm{PM},\Phi,\theta,\Delta} = S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta}) = \begin{cases} 1 & \text{if } FI_{\Phi,\theta,\Delta} \leqslant \kappa_{LB}, \\ \frac{1}{2}\left(\cos\left(\frac{\pi\left(FI_{\Phi,\theta,\Delta}-\kappa_{LB}\right)}{\kappa_{UB}-\kappa_{LB}}\right)+1\right) & \text{if } \kappa_{LB} < FI_{\Phi,\theta,\Delta} \leqslant \kappa_{UB}, \\ 0 & \text{if } \kappa_{UB} < FI_{\Phi,\theta,\Delta}. \end{cases} \tag{6}$$

Here κ_{UB} and κ_{LB} represent the upper and lower bounds of the membership function, $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$, respectively. It is clearly seen that the value of the fuzzy index $FI_{\Phi,\theta,\Delta}$ is higher when $LD_{\Phi,\theta,\Delta}$ is higher, which means that the missing pixel is on the edge region (in the case of Φ = SD) or motion region (in the case of Φ = TD). Then, the membership degree weight values $w_{\mathrm{PM},\Phi,\theta,\Delta}$, which represent the certainty of the direction (Φ,θ,Δ) to be in an edge or motion, is calculated using the $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$ and is close to zero. If the$LD_{\Phi,\theta,\Delta}$ between two pixels on a (Φ,θ,Δ) direction is lower, then the membership degree $(w_{\mathrm{PM},\Phi,\theta,\Delta})$ becomes higher. We observed both salient and non-salient intensity changes of $LD_{\Phi,\theta,\Delta}$ using $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$. The output of the type-1 fuzzy filter, $x_{T1FD}(i,j,k)$, is obtained as follows. Since the type-1 fuzzy logic filter employs only a primary membership function, $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$, the weighting factor, $\omega_{\Phi,}\theta$, can be computed as seen in (7)

$$\omega_{\Phi,\theta} = \prod_{\Delta\in\{-,z,+\}} w_{\mathrm{PM},\Phi,\theta,\Delta}. \tag{7}$$

In the proposed algorithm we only consider fuzzy set SMALL. The idea behind our decision was to assign large weights to the directions that had a small luminance distance as the centre. As can be seen in [9,33], the most famous deinterlacing algorithms are based on linear interpolation algorithm which uses directional correlations between pixels to linearly interpolate a missing line between two adjacent lines in the interlaced signal before displaying. However, since the above methods choose edge direction crisply, they can easily cause artefacts when there is not an edge or it is not clear [17]. In order to build the desired weight evaluation system, we must first distinguish each edge direction and whether it has worth or not. To this end, certain weights are assigned to each of the 18 directions (two domains × three edge degrees × three displacements) in the 3D localized window, namely with the parameters $w_{\mathrm{PM},\Phi,\theta,\Delta}$, for the direction (Φ,θ,Δ) when considering the primary membership function. The adequate values of the two parameters κ_{UB} and κ_{LB} will be discussed in Section 4.

3.5. Weight measurement by type-2 fuzzy filter

Until now, in literature, a large number of fuzzy techniques for deinterlacing have been introduced. However, studies involving deinterlacing systems that are based on type-2 fuzzy techniques have not been proposed yet. The motivation of our paper is to remove the uncertainty of membership values by using type-2 fuzzy sets. Type-2 fuzzy logic is an extension of type-1 fuzzy logic where the membership grade in a fuzzy set is itself measured as a fuzzy number. In many works

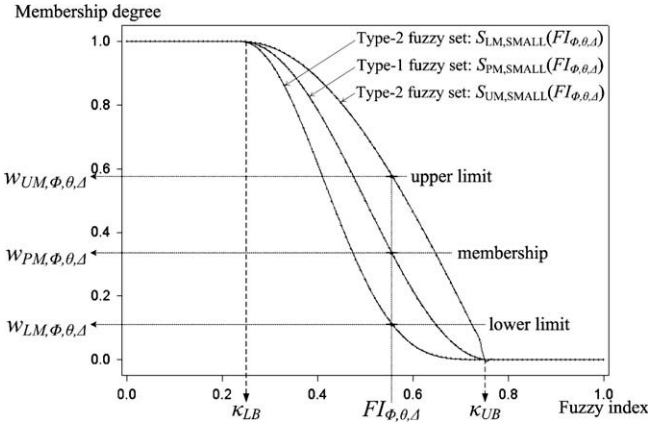

Fig. 6. The construction and behavior of the S-type SMALL membership function: $S_{\mathrm{LM,SMALL}}(FI_{\Phi,\theta,\Delta})$, $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$, and $S_{\mathrm{UM,SMALL}}(FI_{\Phi,\theta,\Delta})$ for $\alpha = 2$.

[4,7,10,16], several researchers investigated type-2 fuzzy sets and showed that those membership functions are well adapted in image processing. Also, Mendel introduced the footprint of uncertainty (FOU) to describe the erratic shape of a type-2 fuzzy set. Therefore, we adopted the same membership functions and ultrafuzzifier value ($\alpha = 2$) in our proposed T2FD method. The value α for each spatio-temporal position (i,j,k) can be computed with regard to some characteristics of the corresponding neighbourhood, and the value range is from 0 to ∞. We now present the application of type-2 fuzzy sets to deinterlacing.

In order to define a type-2 fuzzy set, we defined a type-1 fuzzy set, and assigned upper and lower membership functions to each element to construct the FOU, as shown in Fig. 6. We assumed $\mu_{\mathrm{PM}}(x)$ as the primary membership function. A more practical definition for a type-2 fuzzy set (upper and lower membership functions) can be given as follows:

$$\mathrm{PM} = \{(x, \mu_{\mathrm{UM}}(x), \mu_{\mathrm{LM}}(x)) | \forall x \in X\} \quad \text{where } \mu_{\mathrm{LM}}(x) \leqslant \mu_{\mathrm{PM}}(x) \leqslant \mu_{\mathrm{UM}}(x), \ \ \mu \in [0,1]. \tag{8}$$

In order to establish the upper and lower membership functions, μ_{UM} and μ_{LM}, linguistic hedges can be employed to modify the primary function. Hedges are generally available as pairs, which represent diagonally different modifications of a basic term [25]. Since the membership functions, $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$ are interval membership functions, the boundaries of their FOU are characterized by their upper and lower membership functions, which are defined as

$$\begin{aligned}\mu_{\mathrm{UM}} &= w_{\mathrm{UM},\Phi,\theta,\Delta} = S_{\mathrm{UM,SMALL}}\ (FI_{\Phi,\theta,\Delta}) \\ &= \left[\mu_{\mathrm{PM}}\right]^{\frac{1}{\alpha}} = \left[w_{\mathrm{PM},\Phi,\theta,\Delta}\right]^{\frac{1}{\alpha}} = \left[S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})\right]^{\frac{1}{\alpha}}, \end{aligned} \tag{9}$$

$$\begin{aligned}\mu_{\mathrm{LM}} &= w_{\mathrm{LM},\Phi,\theta,\Delta} = S_{\mathrm{LM,SMALL}}(FI_{\Phi,\theta,\Delta}) \\ &= \left[\mu_{\mathrm{PM}}\right]^{\alpha} = \left[w_{\mathrm{PM},\Phi,\theta,\Delta}\right]^{\alpha} = \left[S_{\mathrm{PM},SMALL}(FI_{\Phi,\theta,\Delta})\right]^{\alpha}. \end{aligned} \tag{10}$$

$S_{\mathrm{UM,SMALL}}(FI_{\Phi,\theta,\Delta})$ and $S_{\mathrm{LM,SMALL}}(FI_{\Phi,\theta,\Delta})$ are the upper and lower membership functions, respectively, of the type-2 interval membership function $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$. The weighting factor, $\omega_{\Phi,\theta}$, represents the output weight of each direction (θ) in each domain (Φ) and is calculated by evaluating the membership expressions. This is accomplished by first converting the input values to fuzzy membership values by utilizing the two membership functions, $S_{\mathrm{UM,SMALL}}(FI_{\Phi,\theta,\Delta})$ *and* $S_{\mathrm{LM,SMALL}}(FI_{\Phi,\theta,\Delta})$, and then applying the minimum *t*-norm operator to these membership values [19]. The weighting factor, $\omega_{\Phi,\theta}$, can be computed as shown in (12)

$$\widehat{\Delta} = \arg \min_{\Delta=\{-,Z,+\}} (w_{\Lambda,\Phi,\theta,\Delta}), \tag{11}$$

$$\omega_{\Phi,\theta} = \prod_{\Lambda\in\{\mathrm{LM,UM}\}} w_{\Lambda,\Phi,\theta,\widehat{\Delta}}. \tag{12}$$

A weighted average would be the ideal way to determine a single value for each pixel. After obtaining the weighting factor, the output of the type-2 fuzzy filter, $x_{T2FD}(i,j,k)$, can be found by calculating the weighted average of the individual rule outputs by using a weighted sum

$$x_{T2FD}(i,j,k) = \begin{cases} x_{org}(i,j,k) & \text{if } j \bmod 2 = N \bmod 2, \\ \dfrac{\sum_{\Phi\in\{SD,TD\}}\sum_{\theta\in\{45^\circ,90^\circ,135^\circ\}} \omega_{\Phi,\theta}(i,j,k)\cdot x_{\Phi,\theta}(i,j,k)}{\sum_{\Phi\in\{SD,TD\}}\sum_{\theta\in\{45^\circ,90^\circ,135^\circ\}} \omega_{\Phi,\theta}(i,j,k)}, & \text{otherwise}. \end{cases} \tag{13}$$

The candidate deinterlaced pixels $x_{\Phi,\theta}(i,j,k)$ are the average values between two pixels on each edge direction, which are calculated as follows:

$$\begin{aligned} &x_{SD,45^\circ} = (ur+dl)/2, \quad x_{SD,90^\circ} = (u+d)/2, \quad x_{SD,135^\circ} = (ul+dr)/2, \\ &x_{TD,45^\circ} = (pr+nl)/2, \quad x_{TD,90^\circ} = (p+n)/2, \quad x_{TD,135^\circ} = (pl+nr)/2. \end{aligned} \tag{14}$$

4. Simulation results and performance analysis

The simulations were performed for the deinterlacing problem. The tests were performed on the following seven CIF images, "Akiyo," "Flower Garden," "Foreman," "Mobile," "News," "Stefan," and "Table Tennis," and were also compared with other representative methods from the literature [50]. These video sequences were chosen because they represent different motion types such as translation, rotation, and zooming details. As a result, they give a more complete evaluation of the proposed method. We conducted an extensive simulation to test the performance of our proposed method using a Pentium IV processor (3.2 GHz). The experiments were implemented in C++.

The simulations were selected to demonstrate the abilities of the proposed T2FD method for deinterlacing with noise-free situations. The peak signal to noise ratio (PSNR) in decibels (dB) was used as an indicator for image comparison [2]. The ratio is described by (16)

$$\mathrm{MSE}(x_{org}, x_{rec}) = \frac{\sum_{i=1}^{cols}\sum_{j=1}^{rows}\left(x_{org}(i,j) - x_{rec}(i,j)\right)^2}{cols \times rows}, \tag{15}$$

$$\mathrm{PSNR}(x_{org}, x_{rec}) = 10\log_{10}\frac{255^2}{\mathrm{MSE}(x_{org}, x_{rec})}. \tag{16}$$

Here, "*cols*" is the column number (=352) of the image, "*rows*" is the row number (=288), and 255 is the maximum value that a pixel can have. The images x_{org} and x_{rec} are the *original image* and the *reconstructed image* with a size of *cols* × *rows*, respectively.

Each CIF standard video sequence has been converted from a progressive format to an interlaced format according to the algorithm shown in Fig. 7. The original 30 progressive frames/s video sequence was decimated into the 30 interlaced fields/s sequences. After deinterlacing, the 30 interlaced fields/s video sequences were reconstructed into the 30 progressive frames/s sequences. Then, we compared the original progressive sequences and output deinterlaced sequences.

4.1. Training of the T2FD filter

The appropriate parameter values for the membership functions, such as the upper bound and lower bound values (κ_{LB} and κ_{UB} in Fig. 6), are difficult to acquire theoretically and therefore, should be found empirically. However, heuristic knowledge does not always provide reliable information on this issue [31,32], except that κ_{UB} should be bigger than κ_{LB} to create the "upper bound" and "lower bound" parameters. We assumed that some κ_{LB} and κ_{UB} values which satisfy the constraints would surely produce better results than other values.

We found the difference between some parameters and others using PSNR. In order to determine the parameters of $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$, we conducted experiments using variable values of κ_{LB} and κ_{UB} in which we fixed the value $\kappa_{LB} = (1 - \kappa_{UB})$. Fig. 8 shows the average PSNR of deinterlaced images for differentκ_{LB} values ranging from 0.01 to 0.49 at intervals of 0.01. Tuning the membership function (determining κ_{LB} and κ_{UB}) resulted in the parameter κ_{LB} (orκ_{UB}) of the $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$ of about 0.24 (or 0.76). The value α is obtained from previous work [10]. We used the same ultrafuzzifier value in our proposed deinterlacing method. Finally, we used the following parameters for the membership functions: $\kappa_{LB} = 0.24$, $\kappa_{UB} = 0.76$, and $\alpha = 2$.

4.2. PSNR analysis

The average PSNR values for test sequences using spatial domain methods (ELA [9], NEDI [36], and FDED [29]), a temporal domain method (Weave [41]), and spatio-temporal domain methods (VTMF [11], STELA [33], EDT [6], and FDOI [14]) are shown in Table 1. The rough sets theory-based deinterlacing method was excluded from the comparison of PSNR analysis, because it described a selection rule for the best method among several conventional methods [15].

In order to demonstrate the merits of the T2FD method, we considered the T1FD method, which uses only the primary membership function. Fig. 9 shows the PSNR values in decibel (dB) curves for the 250 frames of the CIF video sequences. It shows the PSNR using the FDED, Weave, FDOI, T1FD, and the proposed T2FD method. We assumed that the three best techniques in each spatial (FDED), temporal (Weave), and spatio-temporal domain (FDOI) were sufficient to prove the superiority of the proposed method. From Table 1 and Fig. 9, it can be noticed that the proposed T2FD method gave higher PSNR values for the test sequences except for the Akiyo sequence. In particular, it is seen in Fig. 9g that the methods using temporal information behaved worse in frames 131 and 132. These frames in the "Table Tennis" sequences have scene changes; therefore, the methods with spatial information provided better PSNR results. However, the proposed method still maintained the best PSNR level over all frames. The performance of the T2FD method was almost the same or better than the FDOI method in all the tested sequences (except for Akiyo, Foreman, and News sequences) while maintaining a reasonable computational level. The proposed T2FD method required only 55.35% of the computational CPU time of FDOI, with a 0.061 dB PSNR loss. Mean-

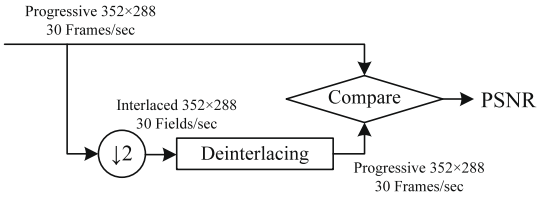

Fig. 7. Performance measurement method of PSNR.

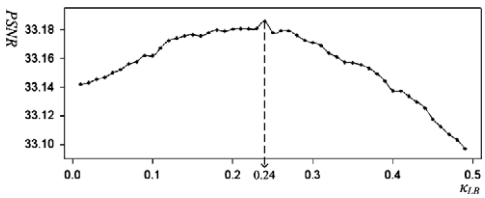

Fig. 8. PSNR results for various κ_{LB} in $S_{\mathrm{PM,SMALL}}(FI_{\Phi,\theta,\Delta})$ using seven CIF test sequences.

Table 1
Average PSNR and computational CPU time of each algorithm over the corresponding seven CIF test sequences.

Sequence	Akiyo	Flower	Foreman	Mobile	News	Stefan	Table Tennis	Average
Spatial domain methods								
Bob (dB)	39.858	22.191	30.172	25.511	33.615	27.724	28.566	29.662
(s)	8.159	7.65	8.963	8.504	8.65	7.598	9.248	8.396
ELA (dB)	37.931	21.681	30.323	23.532	31.474	26.391	27.408	28.391
(s)	18.467	18.341	19.037	18.447	17.837	17.585	22.073	18.827
NEDI (dB)	39.068	22.063	29.394	24.697	32.819	26.554	27.945	28.934
(s)	38.695	35.695	37.297	37.248	35.955	35.13	42.984	37.572
FDED (dB)	40.039	22.068	30.383	25.225	33.548	27.498	28.405	29.595
(s)	121.382	97.472	121.146	147.809	123.211	149.61	95.447	122.297
Temporal domain method								
Weave (dB)	43.786	20.294	26.307	23.537	36.471	21.549	27.996	28.563
(s)	11.301	7.289	7.984	6.537	6.947	7.191	9.093	8.049
Spatio-temporal domain method								
VTMF (dB)	42.827	22.948	32.266	25.459	38.068	25.969	30.119	31.094
(s)	17.528	17.187	18.285	17.663	17.065	17.244	20.984	17.994
STELA (dB)	44.655	22.99	30.449	27.26	39.284	26.996	31.588	31.889
(s)	27.301	27.358	28.919	29.004	28.057	27.179	34.065	28.84
EDT (dB)	44.938	22.271	29.932	25.107	38.105	23.381	30.12	30.551
($T = 20$) (s)	13.691	33.122	23.447	30.78	15.191	31.488	22.085	24.258
FDOI (dB)	47.259	23.173	30.387	27.164	40.167	24.875	31.07	32. 014
(s)	187.602	147.547	177.344	201.793	179.996	203.299	137.093	176.382
The method based on type-1 fuzzy logic (use only primary membership function)								
T1FD (dB)	44.964	23.629	30.366	27.554	38.253	26.522	31.32	31.801
(s)	115.095	110.564	78.04	105.74	87.922	73.342	89.016	94.246
The proposed method based on type-2 fuzzy logic								
T2FD (dB)	45.1	23.681	30.368	27.592	38.806	26.544	31.58	31.953
(s)	119.259	113.098	85.147	105.942	92.415	78.956	92.375	97.624

while, if the Akiyo, Foreman and News sequences were not included in the experiment dataset, the T2FD method provided 0.779 dB better PSNR performance than the FDOI method. We can easily remark that the proposed T2FD method may eventually receive slightly better PSNR values than their counterparts using type-1 fuzzy sets (T1FD). The subjective comparison will be discussed in the following section.

4.3. Image quality analysis

It was noticed that sometimes objective image quality measures, such as MSE or PSNR, are not always suitable criteria for image quality. The reason behind this idea is that there were no direct and logical relationships between such objective measures and the subjective impression of the human observer. Another reason to avoid such objective quality measures was that they compute the luminance difference between the original image and the reconstructed one. However, in practical cases, muted noise and good original images are not observed. Therefore, subjective tests could be used to ensure the quality of the results.

For the subjective image quality evaluation, we chose the 168^{th} (352×288) "Foreman" and "Table Tennis" original images as shown in Fig. 10. The deinterlacing performance of the T2FD method is further illustrated in Figs. 11 and 12, where original images and the corresponding deinterlaced images obtained using the recently proposed methods are shown. The Bob and ELA methods were excluded from the subjective image quality comparison because they are common methods. As can be seen in Figs. 11 and 12, the images deinterlaced by conventional methods showed some shortcomings, which can be observed for example in the helmet's outer boundaries, an oblique line on the wall, or a feature of the player.

Spatial domain methods did not use temporal information and show no motion artefacts in the motion region as shown in Figs. 11a, b and 12a, b. However, they did not work properly with complex structures and the edges are severely degraded. Because the input vertical resolution was halved before the image was interpolated, the detail in the progressive image was reduced.

The results of the Weave method are shown in Figs. 11c and 12c. The edges exhibit significant serrations and the objects from the previous field are still shown and become feathering defects on the helmet, ear, moving hand, and paddle of the player. In the VTMF method, the vertical detail was steadily reduced as the temporal frequencies increased as shown in Figs. 11d and 12d. However, this method still caused staircase artefacts. In Figs. 11e ,f and 12e, f, an outline of the hair or ear from

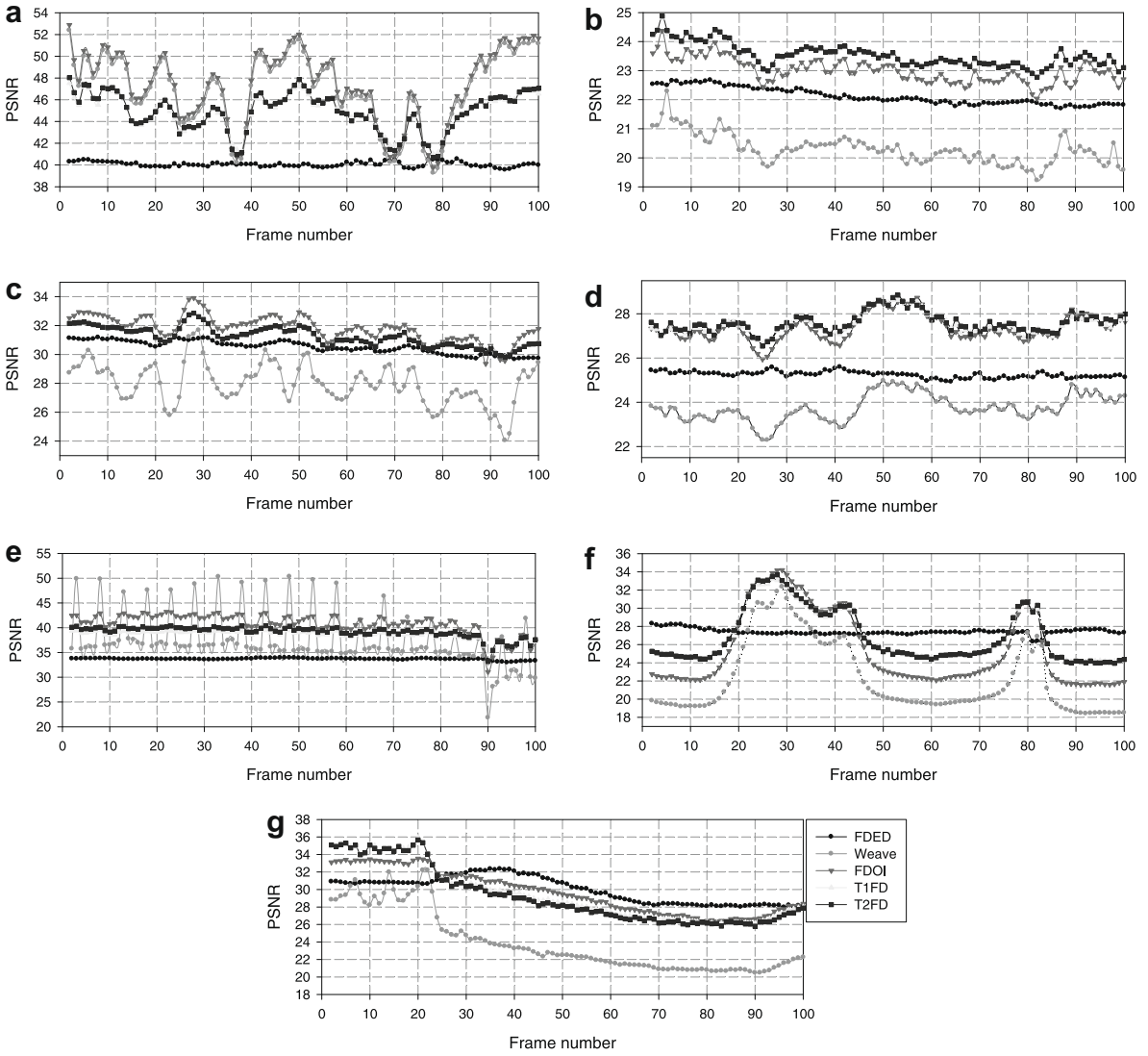

Fig. 9. Performance comparison of different schemes for different sequences: (a) Akiyo, (b) Flower, (c) Foreman, (d) Mobile, (e) News, (f) Stefan, and (g) Table Tennis.

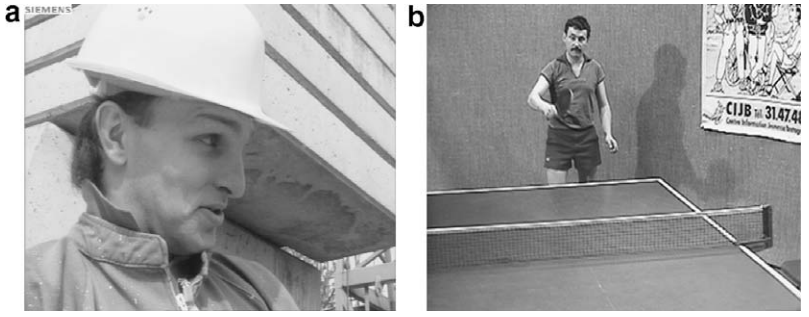

Fig. 10. The test sequences that were used for subjective performance comparisons: (a) the 168th (352 × 288) "Foreman" sequence and (b) the 168th (352 × 288) "Table Tennis" sequence.

a previous field was weakly displayed due to the STELA and EDT methods. After the FDOI method, there were still many feathering defects in the picture as shown in Figs. 11g and 12g. The proposed T2FD method operated images are shown

Fig. 11. Experimental results for visual comparison, frame number 168 from the "Foreman" video sequence: (a) detail of the Foreman sequence after the deinterlacing processes of the NEDI method, (b) FDED method, (c) Weave method, (d) VTMF method, (e) STELA method, (f) EDT method, (g) FDOI method, (h) T1FD method and (i) T2FD method.

in Figs. 11i and 12i. Given that the interlaced images were a fairly loose feature, the T2FD results showed reasonable detail preservation. In particular, the T2FD method provided good visual quality on several patterns and the outer boundaries of the helmet, the oblique diagonal lines on the wall, and the place with the moving hand were faithfully reconstructed. Regarding the subjective visual quality, various image details were well preserved and, at the same time, very few artefacts were introduced. The state-of-the-art subjective visual quality of our algorithm was confirmed by criteria such as perceived blurriness and overall visual quality. Furthermore, we considered the subjective visual quality of the T2FD method to be significantly better (in terms of detail preservation) than that of the other methods. For the T2FD method, feathering defects were no longer seen in the picture. Also, the processed video sequences had a high-quality subjective view.

The T2FD method developed the weight measuring systems that delivered the appropriate weight to the methods, in the correct amount, at the right time. The main advantage of the proposed method was that the edge region was not suppressed well, while fine details and edges did not lose much sharpness. Additionally, the visual results illustrated that the proposed method interpolated an interlaced image much better than the other methods. In real-time, the benefit of this method becomes noticeable since the stability of the moving horizontal lines was assured and the traditional flicker effect of the conventional methods was removed.

4.4. The limitations and the future work

In this paper, a straightforward technique was introduced to prove how type-2 fuzzy logic can be applied with respect to practical issues in deinterlacing. However, one problem encountered was that the computational cost of T2FD was not lower

Fig. 12. Experimental results for visual comparison, frame number 168 from the "Table Tennis" video sequence: (a) detail of the Table Tennis sequence after the deinterlacing processes of the NEDI method, (b) FDED method, (c) Weave method, (d) VTMF method, (e) STELA method, (f) EDT method, (h) FDOI method, (h) T1FD method and (i) T2FD method.

if applied in video deinterlacing. Also, there were other deinterlacing techniques that had lower computational requirements than the proposed T2FD method. The benefit of uncertainty control via type-2 fuzziness decreased because of the high computational expense.

Another limitation of this study was that the parameters for the fuzzy membership functions were determined experimentally. The dataset used in this research was too small to design an overall membership function for the proposed T2FD method as a large dataset would give better results.

It was considered that type-2 fuzzy sets removed the membership uncertainty. This could lead to a new and robust class of deinterlacing algorithms capable of achieving a higher level of accuracy. How type-2 fuzzy sets should be embedded within existing fuzzy techniques and applied to a practical deinterlacing issue, however, still need to be investigated.

Since deinterlacing techniques should be performed in real-time and consume less time, it is worth adopting one of the simple conventional techniques that may be a better design alternative. Because the deinterlacing process is not an easy task in image processing, a unique algorithm will never be established for application to all types of images. A more feasible method of type-2 fuzzy logic filter based deinterlacing should be explored, especially when real-time processing and low computational levels are given constraints.

In the future, we wish to extend the fuzzy technique-assisted edge-preserving system for deinterlacing. Also, we want to develop a fuzzy weight evaluating system beyond our fuzzy weight measurement system. Widening existing fuzzy techniques and covering deeper and broader details of the advantages and challenges of type-2 fuzzy sets remain as topics for future work.

The potentials of the type-2 fuzzy set theory with respect to image enhancement are yet to be investigated as fully as other established methodologies. Because of the intrinsic nature of the type-2 FLS such as difficulties of complexity control

under real-time constraints or industrialization, we tried to make the systems simpler. Therefore, the proposed T2FD method only uses two points of FOU and does not have a type-reduction process. This is quite different from what is usually done in the interval type-2 fuzzy sets community, where the entire interval between the upper and lower membership functions is used. Therefore, our future research will concentrate on the construction of a type-reduction supported T2FD which fully uses the entire interval between the upper and lower membership functions.

5. Conclusion

In this paper, we proposed an efficient deinterlacing method based on a type-2 fuzzy logic filter. The central idea of this paper was to introduce the application of type-2 fuzzy sets to the interpolation of interlaced fields. The upper and lower fuzzy membership functions were derived from the type-1 primary fuzzy filter. The method presented in this paper employed edge direction information and fuzzy weighted filtering schemes. Through the edge direction detection process, weights of each direction in each domain were derived so that a specific interpolation technique could be utilized to better preserve edges during the deinterlacing process. Meanwhile, interpolation based on the fuzzy weighted filtering approach can take a relatively large number of neighbouring pixels into consideration, which reduces edge direction-based interpolation errors. We compared this algorithm with conventional algorithms and intend to conduct additional experiments with different test images to confirm the obtained results and to reinforce the potential of this new method in the future.

Acknowledgements

The authors would like to thank the associate editor and all of the anonymous reviewers for their valuable comments and questions that helped improve both the technical content and the presentation quality of this paper. This work was sponsored by ETRI SoC Industry Promotion Centre, Human Resource Development Project for IT SoC Architect.

References

[1] E.B. Bellers, G. de Haan, Advanced de-interlacing techniques, in: Proceedings of the ProRisc/IEEE Workshop on Circuits, Systems and Signal Processing, 1996, pp. 7–17.
[2] E.B. Bellars, G. De Haan, De-interlacing: A Key Technology for Scan Rate Conversion, Elsevier, Amsterdam, 2000.
[3] P. Brox, I. Baturone, S. Sánchez-Solano, J. Gutiérrez-Ríos, F. Fernández-Hernández, A fuzzy edge-dependent motion adaptive algorithm for de-interlacing, Fuzzy Sets and Systems 158 (3) (2007) 337–347.
[4] H. Bustince, E. Barrenechea, M. Pagola, R. Orduna, Construction of interval type 2 fuzzy images to represent images in grayscale, false edges, in: Proceedings of FUZZ-IEEE, 2007, pp. 1–6.
[5] Y.-L. Chang, S.-F. Lin, C.-Y. Chen, L.-G. Chen, Video de-interlacing by adaptive 4-field global/local motion compensated approach, IEEE Transactions on Circuits and Systems for Video Technology 15 (12) (2005) 1569–1582.
[6] M.-J. Chen, C.-H. Huang, C.-T. Hsu, Efficient de-interlacing technique by inter-field information, IEEE Transactions on Consumer Electronics 50 (4) (2004) 1202–1208.
[7] J. Clairet, A. Bigand, O. Colot, Color image segmentation using type-2 fuzzy sets, in: Proceedings of the IEEE ICELIE, 2006, pp. 52–57.
[8] F. Doctor, H. Hagras, V. Callaghan, A type-2 fuzzy embedded agent to realise ambient intelligence in ubiquitous computing environments, Information Sciences 171 (4) (2005) 309–334.
[9] T. Doyle, Interlaced to sequential conversion for EDTV applications, in: Proceedings of the 2nd International Workshop Signal Processing of HDTV, 1990, pp. 412–430.
[10] P. Ensafi, H.R. Tizhoosh, Type-2 fuzzy image enhancement, Lecture Notes in Computer Science 3656 (2005) 159–166.
[11] G. de Haan, E.B. Bellers, Deinterlacing – an overview, Proceedings of the IEEE 86 (9) (1998) 1839–1857.
[12] M.-Y. Hsiao, T.-H.S. Li, J.-Z. Lee, C.-H. Chao, S.-H. Tsai, Design of interval type-2 fuzzy sliding-mode controller, Information Sciences 178 (6) (2008) 1696–1716.
[13] K. Jack, Video Demystified a Handbook for the Digital Engineer, Elsevier, 2005.
[14] G. Jeon, D. Kim, J. Jeong, Rough sets attributes reduction based expert system in interlaced video sequences, IEEE Transactions on Consumer Electronics 52 (4) (2006) 1348–1355.
[15] G. Jeon, J. Jeong, Designing Takagi–Sugeno fuzzy model-based motion adaptive deinterlacing system, IEEE Transactions on Consumer Electronics 52 (3) (2006) 1013–1020.
[16] R.I. John, P.R. Innocent, M.R. Barnes, Neuro-fuzzy clustering of radiographic tibia image data using type 2 fuzzy sets, Information Sciences 125 (1–4) (2000) 65–82.
[17] N.N. Karnik, J.M. Mendel, Applications of type-2 fuzzy logic systems to forecasting of time-series, Information Sciences 120 (1–4) (1999) 89–111.
[18] N.N. Karnik, J.M. Mendel, Centroid of a type-2 fuzzy set, Information Sciences 132 (1–4) (2001) 195–220.
[19] E.E. Kerre, M. Nachtegael, Fuzzy techniques in image processing, Studies in Fuzziness and Soft Computing 52 (2000) 336–369.
[20] R. Li, B. Zeng, L. Liou, Reliable motion detection/compensation for interlaced sequences and its applications to deinterlacing, IEEE Transactions on Circuits and Systems for Video Technology 10 (1) (2000) 23–29.
[21] Q. Liang, J.M. Mendel, Equalization of nonlinear time-varying channels using type-2, fuzzy adaptive filters, IEEE Transactions on Fuzzy Systems 8 (5) (2000) 551–563.
[22] F. Liu, An efficient centroid type-reduction strategy for general type-2 fuzzy logic system, Information Sciences 178 (9) (2008) 2224–2236.
[23] V. Loia, S. Sessa, Fuzzy relation equations for coding/decoding processes of images and videos, Information Sciences 171 (1–3) (2005) 145–172.
[24] P. Melin, O. Castillo, An intelligent hybrid approach for industrial quality control combining neural networks, fuzzy logic and fractal theory, Information Sciences 177 (7) (2007) 1543–1557.
[25] J.M. Mendel, Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions, Prentice Hall, New Jersey, 2001.
[26] J.M. Mendel, On a 50% savings in the computation of the centroid of a symmetrical interval type-2 fuzzy set, Information Sciences 172 (3–4) (2005) 417–430.
[27] J.M. Mendel, Advances in type-2 fuzzy sets and systems, Information Sciences 177 (1) (2007) 84–110.
[28] J.M. Mendel, H. Wu, New results about the centroid of an interval type-2 fuzzy set, including the centroid of a fuzzy granule, Information Sciences 177 (2) (2007) 360–377.

[29] F. Michaud, C.T. Le Dinh, G. Lachiver, Fuzzy detection of edge-direction for video line doubling, IEEE Transactions on Circuits and Systems for Video Technology 7 (3) (1997) 539–542.
[30] H.B. Mitchell, pattern recognition using type-II fuzzy sets, Information Sciences 170 (2–4) (2005) 409–418.
[31] M. Nachtegael, D. Van der Weken, D. Van de Ville, E.E. Kerre, Fuzzy filters for image processing, Studies in Fuzziness and Soft Computing 122 (2002) 3–24.
[32] M. Nachtegael, D. Van der Weken, E.E. Kerre, W. Philips, Soft computing in image processing, Studies in Fuzziness and Soft Computing 210 (2007) 155–185.
[33] H.-S. Oh, Y.Kim, Y.-Y. Jung, A.W. Morales, S.-J. Ko, Spatio-temporal edge-based median filtering for deinterlacing, in: Proceedings of the IEEE ICCE, 2000, pp. 52–53.
[34] K. Ouyang, G. Shen, Shipeng Li, M. Gu, Advanced motion search and adaptation techniques for deinterlacing, in: Proceedings of the IEEE ICME, 2005, pp. 374–377.
[35] T. Ozen, J.M. Garibaldi, Investigating adaptation in type-2 fuzzy logic systems applied to umbilical acid–base assessment, in: Proceedings of the European Symposium on Intelligent Technologies, 2003, pp. 289–294.
[36] M.K. Park, M.G. Kang, K. Nam, S.G. Oh, New edge dependent deinterlacing algorithm based on horizontal edge pattern, IEEE Transactions on Consumer Electronics 49 (4) (2003) 1508–1512.
[37] S. Schulte, V. De Witte, M. Nachtegael, D. Van der Weken, Etienne E. Kerre, Fuzzy random impulse noise reduction method, Fuzzy Sets and Systems 158 (3) (2007) 270–283.
[38] S. Schulte, V. De Vitte, E.E. Kerre, A fuzzy noise reduction method for color images, IEEE Transactions on Image Processing 16 (5) (2007) 1425–1436.
[39] R. Sepúlveda, O. Castillo, P. Melin, A. Rodríguez-Díaz, O. Montiel, Experimental study of intelligent controllers under uncertainty using type-1 and type-2 fuzzy logic, Information Sciences 177 (10) (2007) 2023–2048.
[40] J. Starczewski, L. Rutkowski, Connectionist structures of type 2 fuzzy inference systems, Lecture Notes in Computer Science 2328 (2001) 634–642.
[41] P.L. Swan, Method and apparatus for providing interlaced video on a progressive display, US Patent 5 864 369, January 26, 1999.
[42] D. Tsang, B. Bensaou, S. Lam, Fuzzy-based rate control for real-time MPEG video, IEEE Transactions on Fuzzy Systems 6 (4) (1998) 504–516.
[43] D. Wu, J.M. Mendel, Uncertainty measures for interval type-2 fuzzy sets, Information Sciences 177 (23) (2007) 5378–5393.
[44] D. Wu, J.M. Mendel, A vector similarity measure for linguistic approximation: Interval type-2 and type-1 fuzzy sets, Information Sciences 178 (2) (2008) 381–402.
[45] S. Yang, Y.-Y. Jung, Y.H. Lee, R.-H. Park, Motion compensation assisted motion adaptive interlaced-to-progressive conversion, IEEE Transactions on Circuits and Systems for Video Technology 14 (9) (2004) 1138–1148.
[46] E.C.C. Tsang, D.M. Huang, D.S. Yeung, J.W.T. Lee, X.Z. Wang, A weighted fuzzy reasoning and its corresponding neural network, in: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, 2002.
[47] A. Zaccarin, B. Liu, Block motion compensated coding of interlaced sequences using adaptively deinterlaced fields, Signal Processing: Image Communication 5 (5–6) (1993) 473–485.
[48] L.A. Zadeh, Fuzzy sets, Information and Control 8 (1965) 338–353.
[49] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning, Information Sciences 8 (3) (1975) 199–249.
[50] Available: <http://media.xiph.org/video/derf/>.