F.S.A.I., Vol. 8, Nos. 1–3, pp. 41–47, 2002

VLSI Implementation of a Fuzzy Image Filtering

Nicu Bizon*, Radian Raducu*

* Faculty of Electronics and Electromecanics, University of Pitesti

Abstract: This paper analyses the possibilities for VLSI implementation of the fuzzy image filtering. The proposed filtering method consists in two operations: a spatial meaning of the pixels from a window, determining the main pixel of this window and a fuzzy 3D interpolation, in which the representative median pixels determine the intensity value of the image pixel. Using a knowledge-based model the fuzzy 3D interpolation gives a natural solution for the coefficients function set of the linear interpolation function. The first operation can be easily implemented in a VLSI chip using resistive networks or, more economically, using junction or MOS diode. The second operation is more complex, because we have to obtain the filtered image by interpolation of the main samples for each node in each window. This can be done using a spatial smoothing circuit with MOS transistors, having space-dependant gate bias.

1. Introduction

Image restoration deals with the problem of reconstruction or estimation of the original image from a distorted or a noisy image. The interest in an image processing area is given by the application of restoration and segmentation to a wide variety of fields like: visual automation systems, vehicle guidance, surveillance, archaeology, intelligent diagnostic systems, intelligent robots and so [1,2].

The area addressed by the present paper is image spatial filtering and image recovery. These problems have been studied by a lot of researcher [3, 4, 5, 6, 7, 8]. The spatial operators used in most cases are of convolution matrix type that made a mediation and low pass spatial filtering. Practically, each pixel is substituted with a weighted medium of neighbour pixels:

$$z_{rs}^{f} = \sum_{(i,j)\in W} a_{ij} z_{ij}^{n} \tag{1}$$

where W is a given window, a_{ij} are the convolution matrix coefficients, $z_{i,j}^{n}$ is the pixels affected by noise and $z_{i,j}^{f}$ are filtered pixels.

The algorithms using fuzzy concepts and neural networks are frequently used in image processing [9-14]. These algorithms can be implemented using digital or analogic circuits. The implementation using digital technique (DSP's) has the disadvantage of big computing time. More attractive is the analogic implementation using VLSI, because of the parallel computing.

2. Fuzzy 3D Interpolation and Fuzzy Spatial Filtering

The proposed fuzzy 3D interpolation model [13, 15, 16] can be easily adapted for image reconstruction and filtering, if the 3-rd dimension is represented by pixels intensity.

Fuzzy 3D interpolation is based on the samples meaning concept and knowledge base model describes in [15]. The formula obtained for 3D interpolation is:

$$z_{c1}(x,y) = z_{c\phi}(x,y) + Q(x,y) \tag{2}$$

where $z_{c\phi}$ is basic estimation (zero level interpolation) and $Q(x, y)$ is the first level correction:

$$\begin{aligned} Q(x,y) = {} & \frac{1}{3}(1-\alpha)(1-\beta)\beta\left[-z_{i(j-1)} + z_{ij} + z_{i(j+1)} - z_{i(j+2)}\right] + \\ & + \frac{1}{3}\alpha(1-\alpha)\beta\left[-z_{(i+1)(j-1)} + z_{(i+1)j} + z_{(i+1)(j+1)} - z_{(i+1)(j+2)}\right] + \\ & + \frac{1}{3}\alpha\beta(1-\alpha)\left[-z_{(i-1)(j+1)} + z_{i(j+1)} + z_{(i+1)(j+1)} - z_{(i+2)(j+1)}\right] \end{aligned} \tag{3}$$

where $\alpha = \frac{x - x_i}{\Delta x}$, $\beta = \frac{y - y_i}{\Delta y}$, $\Delta x = x_{i+1} - x_i = ct$, $\Delta y = y_{j+1} - y_j = ct$.

Fuzzy spatial filtering represents, in fact, the fuzzy 3D interpolation algorithm applied for main pixels $(z^m_{i,j})$.

Median filtering assumes that pixels of a window are substituted with a median pixel. For the fuzzy 3D filtering, main sample is defined as the arithmetical mean of samples in a given window:

$$z^m_{ij} = \sum_{r,s=1,1}^{D,D} z^n_{i_r j_s}, \quad z^n_{i_r j_s} = z\left(x_{i_r j_s}, y_{i_r j_s}\right) + n_{i_r j_s} \tag{4}$$

where:

- the samples with noise are $z^n_{i_r j_s} = z^n\left(x_{i_r j_s}, y_{i_r j_s}\right)$;
- the original surface (without noise) is $- z_{i_r j_s} = z\left(x_{i_r j_s}, y_{i_r j_s}\right)$;
- the samples with noise are $n_{i_r j_s}$;

and grid points are $\left(x_{i_r j_s}, y_{i_r j_s}\right)$.

Notation (i, j) are used to address windows and (r, s) are used to address pixels in a given window (see Fig. 1). We assume that a window has $D \times D$ pixels.

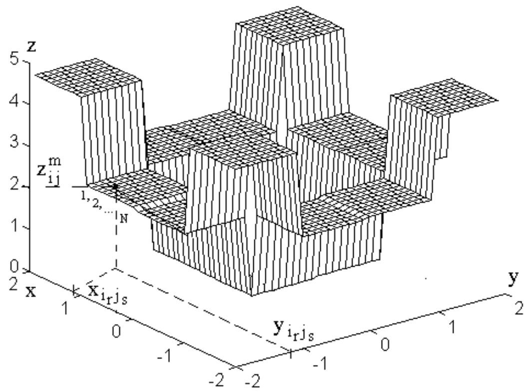

Fig. 1 – The model of interpolation windows

The proposed algorithm has been tested using both synthetic and real images, having the follow attributes: $N \times N$ pixels with 256 grey levels. Synthetic image was chosen with two constant grey level to create a vertical edge, with an edge strength of 40 grey level. Random noise with different amplitude was added to these images. The evaluation was made by visual inspection and using synthetic performance measure criterion, too:

$$e_r = \frac{\left\{ \sum_{i_r,\, j_s = 1,1}^{N,N} \left[z^{f\;2}_{i_r j_s} - z_{i_r j_s}^{\;2} \right] \right\}^{1/2}}{N^2 \left[\max_{i_r, j_s} z_{i_r j_s} - \min_{i_r, j_s} z_{i_r j_s} \right]} \tag{5}$$

In case of synthetic image chosen by us, because intensity variations occur only horizontal, this criterion can be used to measure preservation and strength [17]. The effectiveness of the algorithm has been proven through simulation [13].

3. Sensors for Image Acquisition

The acquisition can be done mainly using circuits based on the:

- charge coupled devices (CCD)
- CMOS based active pixel sensors (APS)
- APS only try to capture the image with a focus toward improving the image quality using standard processes.

APS have extensive applications for astronomical and space exploration applications, in addition to recent interest in multimedia applications, for video and still-image imaging. APS has the advantages that can achieve low noise, high speed, random access to pixels. APS are the main part of the vision chips (in which some high level image processing is performed at the image acquisition level).

These structures are based on a photodiode and the principle is represented in Fig. 2. Initially, the node X is reset and the charge stored in the photodiode area gives a potential of the node X dependant on the image brightness. After the integration cycle the charge is transferred to the output using Select pin.

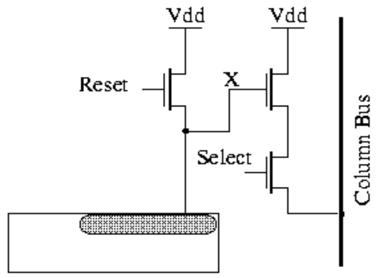

Fig. 2 – CMOS based active pixel sensors

4. Spatial meaning and fuzzy interpolation using VLSI

As it has been mentioned, the first operation consists in a spatial meaning of the samples in a $D \times D$ window. This can be easily implemented in a VLSI chip using resistive networks. A resistive grid receives the output current from the $D \times D$ nodes (pixels), the output current being proportional to the average input currents (for a constant window size). A resistive network wouldn't be economic in CMOS process because of the difficulties in implementing of high-value passive resistors. For this reason the resistive network have to be replaced with junction or MOS diode (see Fig. 3).

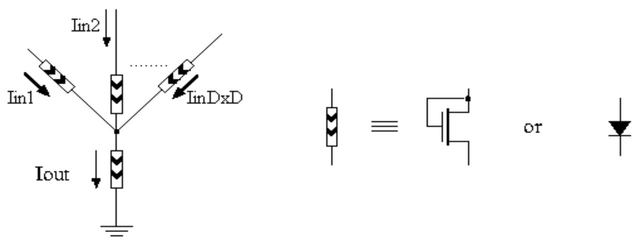

Fig. 3 – Spatial meaning

The next operation is fuzzy interpolation. Re-writing equation (5) for the main sample $\left(z_{i,j}^{m}\right)$ obtained in a $D \times D$ window and replacing the zero level interpolation value, we can obtain the filtered samples as:

$$
\begin{aligned}
z_f(i_r, j_s) = & \frac{1}{3}(1-\alpha_r)(1-\beta_s)(3+\alpha_r+\beta_s)z^m_{i,j} + \frac{1}{3}\alpha_r(1-\beta_s)(4-\alpha_r+\beta_s)z^m_{i+1,j} + \\
& + \frac{1}{3}(1-\alpha_r)\beta_s(4-\alpha_r+\beta_s)z^m_{i,j+1} + \frac{1}{3}\alpha_r\beta_s(5-\alpha_r-\beta_s)z^m_{i+1,j+1} - \\
& - \frac{1}{3}(1-\alpha_r)\beta_s(1-\beta_s)(z^m_{i,j-1} + z^m_{i,j+2}) - \frac{1}{3}\alpha_r\beta_s(1-\beta_s)(z^m_{i+1,j-1} + z^m_{i+1,j+2}) - \\
& - \frac{1}{3}\alpha_r(1-\alpha_r)(1-\beta_s)(z^m_{i-1,j} + z^m_{i+2,j}) - \frac{1}{3}\alpha_r(1-\alpha_r)\beta_s(z^m_{i-1,j+1} + z^m_{i+2,j+1})
\end{aligned}
\qquad (6)
$$

This is the equation that is to be implemented using a VLSI chip. The intensity of the current pixel will result taking into account the twelve main samples and its position in $D \times D$ window.

The main sample $(z^m_{i,j})$ is represented in Fig. 4 by the current sources $Iout_{i,j}$, obtained after smoothing in $D \times D$ window. Because in the equation (6) appear main samples having both plus and minus sign, each of these main samples is represented as normal and complemented current.

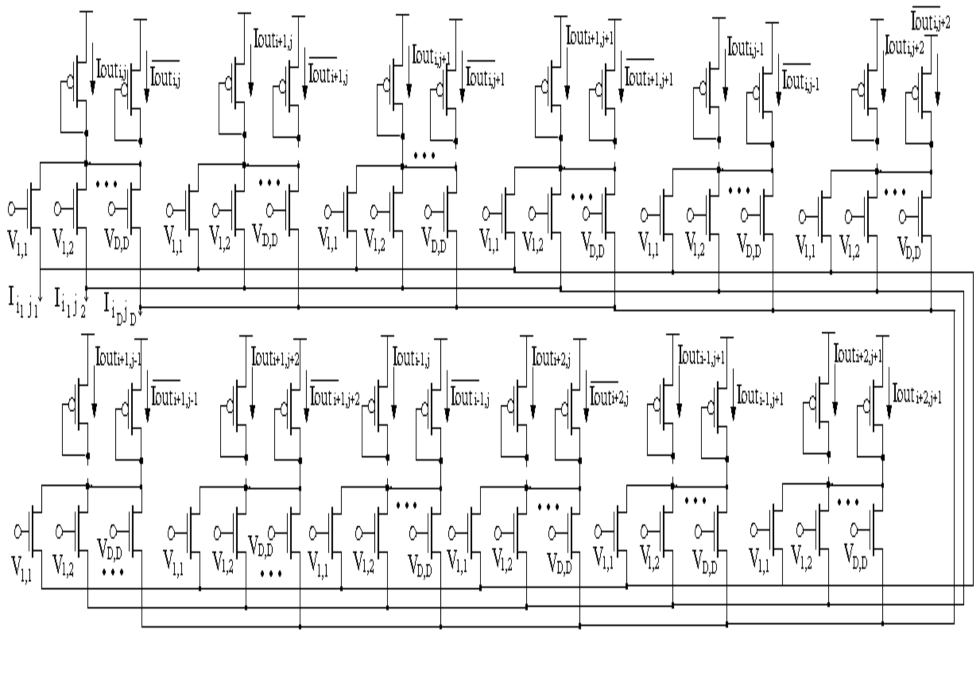

Fig. 4 – Part of VLSI chip

The potentials Vrs, r,s=1÷D are obtained using resistive divisors grids, in order to obtain the appropriate current ratio for each pixel in $D \times D$ window.

Using MATLAB program it is quite easy to compute the current ratios for each pixel in $D \times D$, current flowing from each of the twelve main pixels (see Fig. 5).

a)

0.8640	0.7920	0.7140	0.6300	0.5400	0.4440	0.3420	0.2340	0.1200	0
0.7920	0.7253	0.6533	0.5760	0.4933	0.4053	0.3120	0.2133	0.1093	0
0.7140	0.6533	0.5880	0.5180	0.4433	0.3640	0.2800	0.1913	0.0980	0
0.6300	0.5760	0.5180	0.4560	0.3900	0.3200	0.2460	0.1680	0.0860	0
0.5400	0.4933	0.4433	0.3900	0.3333	0.2733	0.2100	0.1433	0.0733	0
0.4440	0.4053	0.3640	0.3200	0.2733	0.2240	0.1720	0.1173	0.0600	0
0.3420	0.3120	0.2800	0.2460	0.2100	0.1720	0.1320	0.0900	0.0460	0
0.2340	0.2133	0.1913	0.1680	0.1433	0.1173	0.0900	0.0613	0.0313	0
0.1200	0.1093	0.0980	0.0860	0.0733	0.0600	0.0460	0.0313	0.0160	0
0	0	0	0	0	0	0	0	0	0

0.1200	0.1093	0.0980	0.0860	0.0733	0.0600	0.0460	0.0313	0.0160	0
0.2340	0.2133	0.1913	0.1680	0.1433	0.1173	0.0900	0.0613	0.0313	0
0.3420	0.3120	0.2800	0.2460	0.2100	0.1720	0.1320	0.0900	0.0460	0
0.4440	0.4053	0.3640	0.3200	0.2733	0.2240	0.1720	0.1173	0.0600	0
0.5400	0.4933	0.4433	0.3900	0.3333	0.2733	0.2100	0.1433	0.0733	0
0.6300	0.5760	0.5180	0.4560	0.3900	0.3200	0.2460	0.1680	0.0860	0
0.7140	0.6533	0.5880	0.5180	0.4433	0.3640	0.2800	0.1913	0.0980	0
0.7920	0.7253	0.6533	0.5760	0.4933	0.4053	0.3120	0.2133	0.1093	0
0.8640	0.7920	0.7140	0.6300	0.5400	0.4440	0.3420	0.2340	0.1200	0
0.9300	0.8533	0.7700	0.6800	0.5833	0.4800	0.3700	0.2533	0.1300	0

b)

0.1200	0.2460	0.3780	0.5160	0.6600	0.8100	0.9660	1.1280	1.2960	1.4700
0.1040	0.2133	0.3280	0.4480	0.5733	0.7040	0.8400	0.9813	1.1280	1.2800
0.0887	0.1820	0.2800	0.3827	0.4900	0.6020	0.7187	0.8400	0.9660	1.0967
0.0740	0.1520	0.2340	0.3200	0.4100	0.5040	0.6020	0.7040	0.8100	0.9200
0.0600	0.1233	0.1900	0.2600	0.3333	0.4100	0.4900	0.5733	0.6600	0.7500
0.0467	0.0960	0.1480	0.2027	0.2600	0.3200	0.3827	0.4480	0.5160	0.5867
0.0340	0.0700	0.1080	0.1480	0.1900	0.2340	0.2800	0.3280	0.3780	0.4300
0.0220	0.0453	0.0700	0.0960	0.1233	0.1520	0.1820	0.2133	0.2460	0.2800
0.0107	0.0220	0.0340	0.0467	0.0600	0.0740	0.0887	0.1040	0.1200	0.1367
0	0	0	0	0	0	0	0	0	0

c)

Fig. 5 – The current ratios in a $D \times D = 10 \times 10$ window for $I_{out}i,j$ (*a*), $I_{out}i+1,j$ (*b*), $I_{out}i,j+1$ (c)

5. Conclusions

An algorithm for spatial filtering has been presented. The algorithm turned out to give better performances in comparison to other filtering methods (see the results obtained in [13, 15, 16]).

In [13] the proposed algorithm and the spatial filtering method was tested for synthetic or real images in different conditions of noise level and window size. The obtained results confirms that proposed algorithm has good performances in all situation mentioned above.

A principle for VLSI implementation has been also discussed. In the paper are presented principles for both spatial meaning and fuzzy interpolation.

References

[1] Y. W. LIN, R. SRINIVASAN: *Digital Image Processing Applications,* SPIE-ISOE, 1989.

[2] W. L. YING, R. SRINIVASAN: *Digital Image Processing Applications,* Proceedings SPIE, vol. **1075**, Washington, 1989.

[3] K. R. CASTLEMAN: *Digital Image processing*, ed. Prentice Hall, 1996.
[4] R. C. GONZALES, R.E. WOODS: *Digital Image Processing*, ed. Addison Wesley, 1993.
[5] A. K. JAIN: *Fundamentals of Digital Image Processing*, ed. Prentice Hall, New Jersey, 1989.
[6] J. S. LIM: *Two Dimensional Signal and Image Processing*, ed. Prentice Hall, Englewood Cliffs, New Jersey, 1990.
[7] J. S. LIM: *Image Enhancement in Digital Image Processing Techniques*, ed. Academic Press, New York, 1984.
[8] W. K. PRATT: *Digital Image Processing*, Wiley, New York, 1991.
[9] H. CHEN: *A Fast Filtering Algorithm for Image Enhancement*, IEEE Transaction on Image Processing, vol. **4**, no. 5, 1995.
[10] C. Y. TYANG, P. P. WANG: *Image processing – Enhancement, filtering and Edge detection using the Fuzzy logic Approach*, second IEEE international conference on Fuzzy Systems, vol. **1**, pp. 600 – 605, 1993.
[11] E. LITTMANN, H. RITTER: *Adaptive Color Segmentation – A Comparison of Neural and Statistical Methods*, IEEE Tansaction on Neural Networks, vol. **8**, no. 1, 1997.
[12] F. RUSSO: *A New Class of Fuzzy Operators for Image processing: Design and Implementation*, second IEEE international conference on Fuzzy Systems, vol. **1**, pp. 821 – 826, 1993.
[13] N. BIZON, E. SOFRON, R. RADUCU: *Fuzzy Method for Image Filternig*, International AMSE Conference on Modelling and Simulation, vol.**1**, pp. 43-48, Santiago de Compostela, Spain, 1999.
[14] O. K. ALSHAYKH, S. RAMASAWAMY, H. S. HUNG: *Fuzzy Techniques for Image Enhancemet and Reconstruction*, Second IEEE international conference on Fuzzy Systems, vol. **1**, pp. 582 – 587, 1993.
[15] N. BIZON, E. SOFRON, R. RADUCU: *Fuzzy 3D Interpolation -Zero Level*, 4th International IFAC Conference on System Structure and Control, pp. 367-372, Bucharest, Romania, 1997.
[16] N. BIZON, R. RADUCU: *Model for 3D Signal Processing*, Int. AMSE Conf. on Contribution of Cognition to Modelling, pp. 15.1-15.4, Lyon, 1998.
[17] V. V. PHOHA, W. J. B. OLDHAM: *Image Recovery and Segmentation Using Competitive Learning in a Layered Network*, IEEE Tansaction on Neural Networks, vol. **7**, no. 4, 1996.

Paper presented in a preliminary form in the **Second European Conference on Intelligent Systems and Technologies**, **ECIT'2002**, *Iasi, July 17-20, 2002*.