2D Image Fuzzy Filters in Matlab

Dana Majerová, Jaromír Kukal, Aleš Procházka

ICT Prague, Department of Computing and Control Engineering

Abstract. *The paper is devoted to fuzzy image processing based on Łukasiewicz algebra with square root. A new method of image processing based on Modus Ponens Fuzzy Network with fuzzy logic function preprocessing in a hidden layer is presents. All the fuzzy algorithms are realized in the MATLAB system.*

Keywords: *Image enhancement, biomedical signals, Łukasiewicz algebra, Modus Ponens, fuzzy network, fuzzy filtering, Matlab.*

1 Introduction

The de–noising of 2D biomedical images is very actual problem. In case of MRI brain images this problem arises when the low–resolution apparatus is used. The noise reduction and signal structure saving are contradictory but useful aims. The Łukasiewicz algebra with square root seems to be an efficient tool for the 2D image de–noising because of sensitivity suppress, minimum number of basic operators and ability to construct weighted average filters.

2 Background of Fuzzy Image Processing

The Łukasiewicz algebra [4] enriched by square root was chose as a background of fuzzy image processing. The other basic logic algebras like Gougen or Gödel are not good models because of discontinuity of residuum or lack of square root function [7].

The *Łukasiewicz algebra with square root* (ŁA_{sqrt}) is defined as

$$\mathcal{L} = \{\mathbf{L}, 0, 1, \wedge, \vee, \otimes, \rightarrow, sqrt\}$$

where $\mathbf{L} =< 0;1 >$, $\mathbf{L} \subset \mathbf{R}$ and basic operators are defined for each $a, b \in \mathbf{L}$ as

$$a \wedge b = \min(a, b),$$

$$a \vee b = \max(a, b),$$

$$a \otimes b = \max(a + b - 1, 0),$$

$$a \rightarrow b = \min(1 - a + b, 1),$$

$$sqrt(a) = (1 + a)/2.$$

It is useful to define *derived operators* to simplify function notations. The derived operators in ŁA_{sqrt} are:

$$\begin{aligned} \neg a &= a \rightarrow 0 = 1 - a, \\ a \leftrightarrow b &= (a \rightarrow b) \wedge (b \rightarrow a) = 1 - |a - b|, \end{aligned}$$

$$\begin{aligned} a \circ b &= \neg(a \leftrightarrow b) = |a - b|, \\ a \oplus b &= \neg(\neg a \otimes \neg b) = \min(a + b, 1), \\ a \ominus b &= a \otimes \neg b = \max(a - b, 0), \\ na &= \oplus_{k=1}^{n} a = \min(n \cdot a, 1), \\ a^n &= \otimes_{k=1}^{n} a = \max(n \cdot a - n + 1, 0) \end{aligned}$$

for each $n \in \mathbf{N}$ and $a, b \in \mathbf{L}$.

Let $n \in \mathbf{N}$ and $f : \mathbf{L}^n \to \mathbf{L}$. Then f is called *fuzzy logic function* (FLF) in ŁA$_{sqrt}$ when f is composed from free variables and constants from $\mathbf{L}$, constants from $\mathbf{N}$ and finite number of ŁA$_{sqrt}$ operators and functions.

THEOREM: Any FLF in ŁA$_{sqrt}$ is Lipschitz continuous function with the *sensitivity*

$$\lambda = \max_{\overrightarrow{x} \neq \overrightarrow{y}} \frac{|f(\overrightarrow{x}) - f(\overrightarrow{y})|}{\sum_{k=1}^{n} |x_k - y_k|}.$$

The FIR filters [2, 6] are the standard tools for image filtering. It is possible to search an intersection between FIR filters and FLF's.

THEOREM: Let $N \in \mathbf{N}_0$, $n \in \mathbf{N}$ and $m_k \in \mathbf{N}$ for $k = 1, \ldots, n$. Let $\overrightarrow{x} \in \mathbf{L}^n$. Let $w_k = m_k / 2^N$ be dyadic weights for $k = 1, \ldots, n$ and $\sum_{k=1}^{n} w_k \leq 1$. Then any FIR filter

$$Y(\overrightarrow{x}) = \sum_{k=1}^{n} w_k \cdot x_k$$

is a FLF in the ŁA$_{sqrt}$ and it satisfies the Lipschitz condition with the sensitivity

$$\lambda = \max_{k=1,\ldots,n} w_k.$$

The FIR filters which are FLF's can be used as preprocessors in hidden fuzzy network layer. This fuzzy network is FLF and its sensitivity is not higher than hidden FLF sensitivities. This neural network is called Modus Ponens Fuzzy Network (MPFN) [3].

The MPFN is a four layer neural network. The first layer contains the n input nodes for the input signal. The second layer of size H is composed from FLF nodes for the FLF preprocessing with constrained sensitivity. The third layer realizes the Modus Ponens law with learnable rule weights. The fourth layer with m output nodes realizes the compromise solution of given task. Any MPFN output is FLF of input vector. The MPFN is described on the Fig. 1. The MPFN was realized in MATLAB system [5].

MPFN PROCESSING RULE ($m, n, H \in \mathbf{N}$ and $\overrightarrow{x} \in \mathbf{L}^n$):

$$\begin{aligned} h_j(\overrightarrow{x}) &= FLF_j(\overrightarrow{x}) \quad for\ j = 1, \ldots, H, \\ \mathcal{E}_i(\overrightarrow{x}) &= \bigvee_{j=1}^{H} w_{ij} \otimes h_j(\overrightarrow{x}) \quad for\ i = 1, \ldots, m, \\ \mathcal{E}_i^*(\overrightarrow{x}) &= \bigvee_{j=1}^{H} w_{ij}^* \otimes h_j(\overrightarrow{x}) \quad for\ i = 1, \ldots, m, \\ Y_i(\overrightarrow{x}) &= sqrt(\mathcal{E}_i(\overrightarrow{x})) \otimes sqrt(\neg\mathcal{E}_i^*(\overrightarrow{x})) \quad for\ i = 1, \ldots, m. \end{aligned}$$

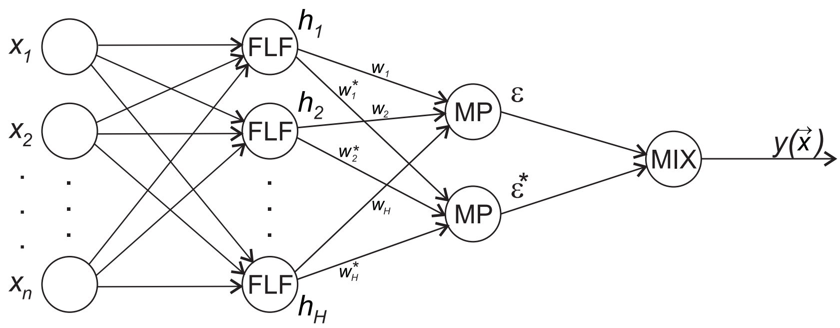

Figure 1: Modus Ponens Fuzzy Network for $m = 1$

The MPFN could be used as a 2D image filtering tool. Its weights must be optimized to increase de–noising quality.

3 Fuzzy 2D Image Processing

The Fig. 2 represents the MRI T2 phantom data [1] corrupted by the real MRI T2 noise. Then the results of all the FLF FIR 3x3 filters were compared by a SNR criterion with the Wiener 3x3 adaptive filter in the first experiment.

$$SNR = 10 \cdot \log \frac{Var(X)}{Var(Y - X)}$$

where X is original image matrix and Y is noised one. $Var(A) = \mathbf{E}\{(A - \mathbf{E}A)^2\}$, where $\mathbf{E}$ is symbol of average value. The result of Wiener filter is presented on the Fig. 3a. Five of tested FIR filters were better than the Wiener 3x3 filter.

Tab. 1 and Figs. 3b–5a represent the results of experiment. The best FLF filters were used as FLF's in the hidden layer of MPFN in the second phase. The MPFN weights were optimized. The result of MPFN filtering is presented in the Tab. 1 and on the Fig. 5b.

Filter	λ	**SNR [dB]**	Δ **SNR [dB]**
NO	-	12.78	0
WIENER	-	23.61	10.83
FIR-12-7-6	3/16	23.68	10.90
FIR-12-8-5	3/16	23.81	11.03
FIR-8-3-3	1/4	23.70	10.92
FIR-4-2-1	1/4	23.77	10.99
FIR-16-7-5	1/4	23.78	11.00
MPFN	1/4	23.91	11.13

Table 1: FLF filter efficiency

The FIR filters are labeled by three numbers. The first of them is central weight, the second one is the weight of neighbour pixels and the third one is the weight of corner pixels. The denominator is not included.

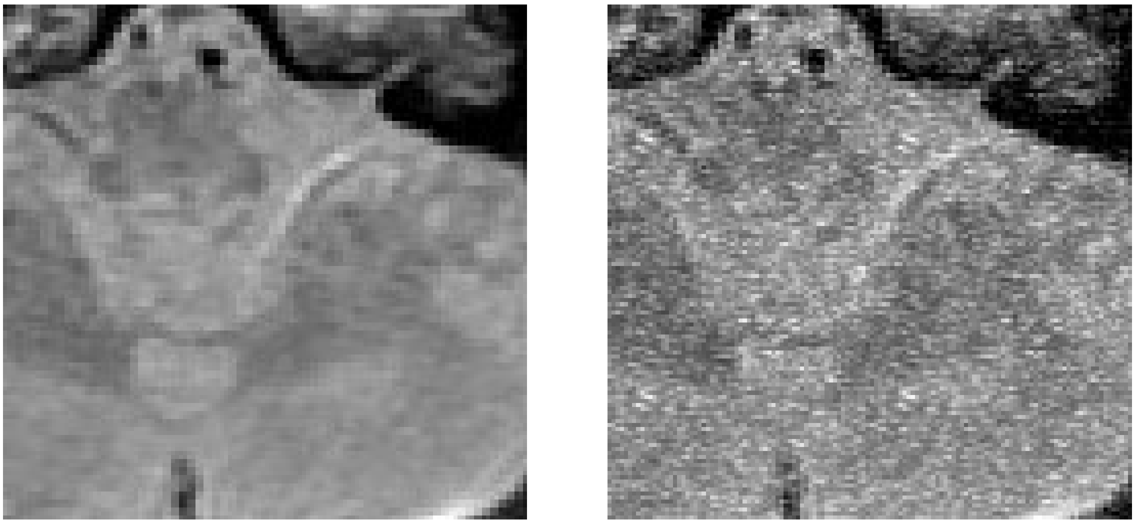

Figure 2: **(a)** original image, **(b)** noised image

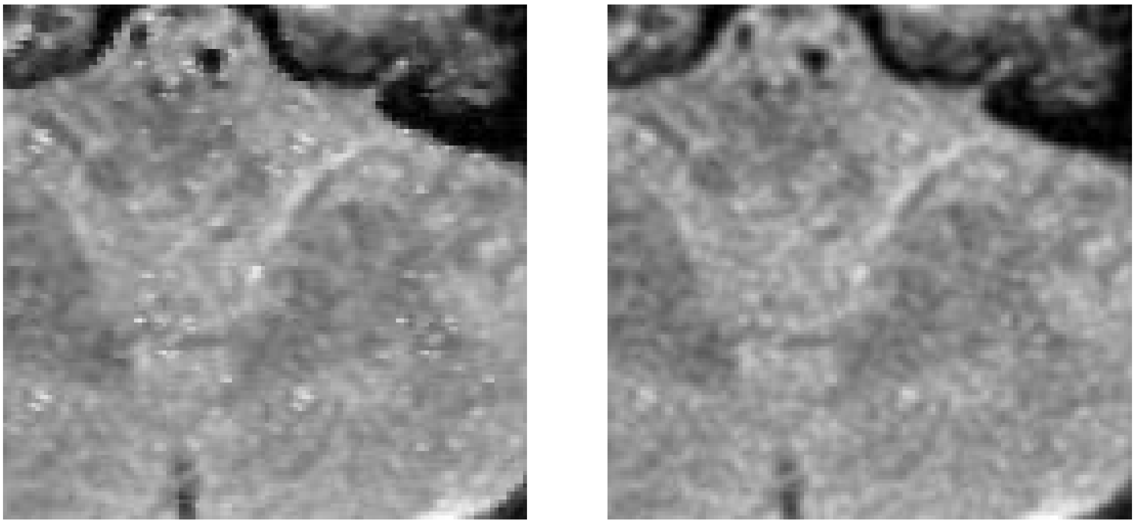

Figure 3: **(a)** Wiener filter, **(b)** FLF filter FIR-12-8-5

4 Conclusions

The LA_{sqrt} is a applicable tool for fuzzy de–noising of 2D MRI images. The application of FLF 3x3 filters permits to increase the SNR of image. An optimum FLF combination in the hidden layer of MPFN produces the better FLF filter. The hidden layer optimization is a subject of our future research.

References

[1] http://www.med.harvard.edu/aanlib/cases/casem/mr2/011.html.

[2] Klette, R., P. Zamperoni. *Handbook of Image Processing Operators.* John Wiley and Sons. Chichester. 1996.

[3] Kukal J. *Modus Ponens Fuzzy Network*, Mendel 2000, Brno. 2000.

[4] Łukasiewicz J. *Selected Works*, Amsterdam. 1970.

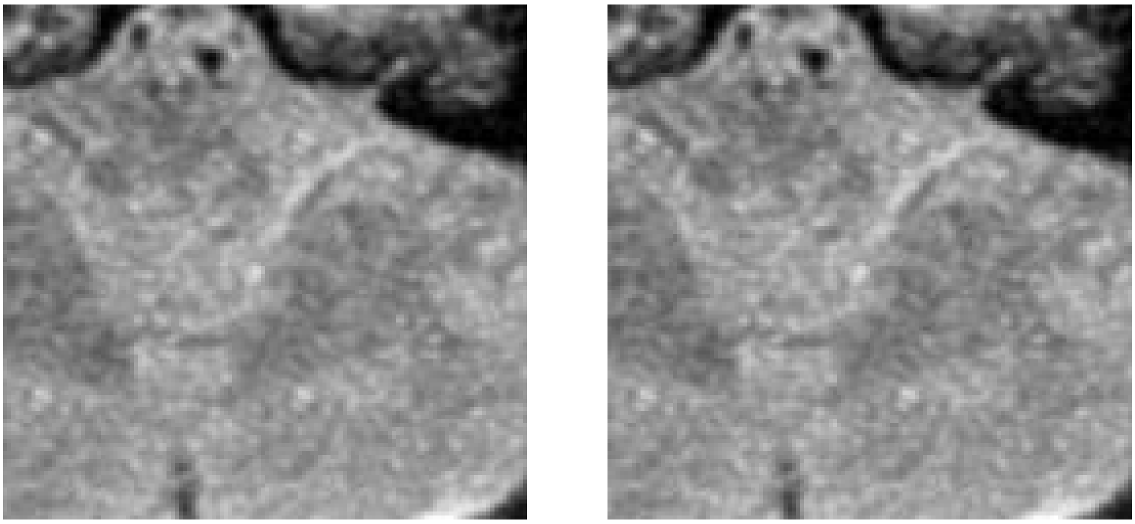

Figure 4: FLF filters: **(a)** FIR-16-7-5, **(b)** FIR-4-2-1

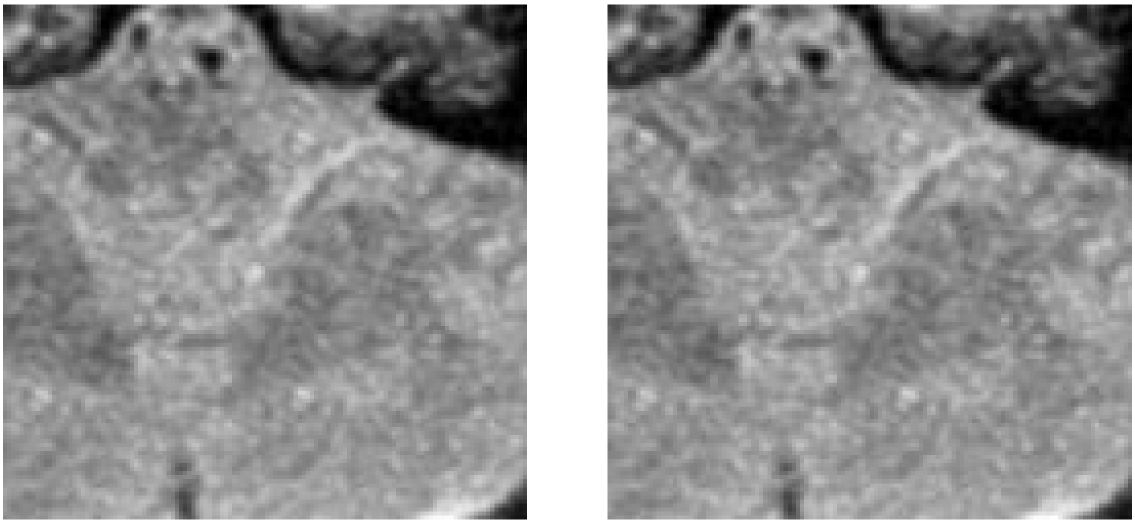

Figure 5: FLF filters: **(a)** FIR-12-7-6, **(b)** MPFN

[5] Majerová D., Kukal J. *Modus Ponens Fuzzy Network in MATLAB.* In: MATLAB 2000. Prague. 2000.

[6] Mitra S. K., Kaiser J. F. *Handbook for Digital Signal Processing.* John Wiley & Sons. New York. 1993.

[7] Novák V., Perfilieva I., Močkoř J. *Mathematical Principles of Fuzzy Logic*, Kluwer Academic Publishers, Boston. 1999.

Contact

Dana Majerová, Jaromír Kukal, Aleš Procházka

INSTITUTE OF CHEMICAL TECHNOLOGY, Prague
Department of Computing and Control Engineering
Technická 5, 166 28 Prague 6 Dejvice
Phone: 420-2-2435 4170, fax: 420-2-2435 5053
E-mails: {Dana.Majerova, Jaromir.Kukal, Ales.Prochazka}@vscht.cz
WWW address: http://uprt.vscht.cz/