image filtering,
weighted averaging,
adaptive selection of weights

Alina MOMOT[1], Janusz WROBEL[2], Krzysztof HOROBA[2],
Michal JEZEWSKI[3], Marek BERNYS[2]

FILTERING OF TWO-DIMENSIONAL DIGITAL IMAGES USING WEIGHTED AVERAGING FOR ADAPTIVE SELECTION OF WEIGHTS

Many digital images, especially in biomedical fields, contain some disturbances. The image analysis depends on quality of the images that is why reduction or elimination (if it is possible) the disturbances is the key issue. There are many methods of improvement in the quality of the images and thus improve the quality of the image analysis, among them one of the simplest method is low-pass filtering such as arithmetic mean or its generalization, weighted mean.

The basic problem of the weighted mean is the proper selection of the weights. This can be done using adaptive algorithms. This paper presents several such algorithms which are modifications of the existing weighted averaging methods created originally for noise reduction in electrocardiographic signal. The description of the new filtering methods and a few results of its application are also presented with comparison to existing arithmetic average filtering.

1. INTRODUCTION

In real biomedical signals there are almost always observed disturbances, the presence of which results from the specific acquisition of these signals. For example, for bioelectric signals, the disturbances may come from the bioelectric activity of body cells, the powerline or the hardware which retrieves these signals. Bioelectric signals, widely used in various fields of biomedicine, are generated by both muscle cells and nerve. Voltage is propagated through tissue and can be measured on the surface of the body, which provides a convenient, noninvasive method for measuring the electrical activity of internal organs. However, using surface electrodes, results in the presence of noise, often high amplitude, which must be reduced in order to extract the useful signal [1].

There are many methods for reducing interference in biomedical signals, such as low-pass filtering, exemplified by the moving average filter, as well as band-pass filtering. In the case of quasi-cyclic biomedical signals, the quality of these signals can be improved by synchronized averaging [7]. The averaging can be done using the existing arithmetic mean or its generalization, in the form of a weighted average where the weights are chosen adaptively.

Spatial domain methods of the image enhancement operate on the pixels composing the image and the processes could be denoted by

$$g(x, y) = T[f(x, y)], \tag{1}$$

where $f(x, y)$ is the input image, $g(x, y)$ is the output image and T is an operator on f, defined over some neighborhood of (x, y) [4]. The process of spatial filtering consists of moving the filter mask from point to point in an image. In the case of linear spatial filtering the response of the filter is given by a sum of products of the filter coefficients and the corresponding image pixels in the area spanned by the filter mask.

If all the coefficients are the same and sum up to one, this is called arithmetic mean filtering. The idea of mean filtering is to replace each pixel value in an image with the mean (‘average’) value of its neighbors, including itself. The median filter is very similar to the mean filter, but instead of simply

[1] Silesian University of Technology, Institute of Informatics, Akademicka St. 16, 44-100 Gliwice, alina.momot.@polsl.pl
[2] Institute of Medical Technology and Equipment, Biomedical Signal Processing Department,
Roosevelta St. 118, 41-800 Zabrze, Poland.
[3] Silesian University of Technology, Institute of Electronics, Akademicka St. 16, 44 100 Gliwice, Poland.

replacing the pixel value with the mean of neighboring pixel values, it replaces it with the median of those values. Due to such procedures there is obtained reduction the influence of pixel values which are unrepresentative of their surroundings [2].

Both mean filtering and median filtering can be treated as a convolution filter based around a kernel, which represents the shape and size of the neighborhood to be sampled when calculating the mean or the median. However in the case of median filtering the mask coefficients are not always constant, there is only one non-zero coefficient (equal one) and which coefficient is non-zero depends on result of the sorting operation.

In the case of the convolution filters with non-constant mask coefficients, the key issue is the choice of procedure which provides the values of the coefficients. In the adaptive fuzzy weighted average filter (AFWA) proposed in [8] the pixels in the filter window are regarded as a fuzzy set, and every pixel in the filter window can be depicted by membership function, which is actually the weight of this pixel. Whereas, the algorithm for computing values of the mask coefficient which has been described in [6], is based on the existing empirical Bayesian weighted averaging method created originally for noise reduction in electrocardiographic signal [5]. This paper presents several algorithms which are modifications of the existing weighted averaging methods created originally for noise reduction in electrocardiographic signal which are described in [7]. The modifications are based on the idea described in detail in [6]. The performance of the new algorithms are experimentally compared with the traditional average filtering and median filtering for both synthetic and real images.

2. ADAPTIVE LINEAR SPATIAL FILTERING

2.1. THE BASIC MODIFICATED METHOD OF AVERAGING

In this section there is briefly presented the algorithm proposed in [6] which provides the idea of constituting a base for further modifications other algorithms described in [7]. Let us assume that *r* is the radius of the square mask, i.e. $r = (m-1)/2$ where *m* is the size of mask, *X* and *Y* are dimensions of the input $X \times Y$ image f. The output image g size is $(X-2r)\times(Y-2r)$. For each pixel of the input image $f(x,y)$, i.e. $x \in \{r+1, r+2, \ldots, X-r\}$ and $y \in \{r+1, r+2, \ldots, Y-r\}$, there is calculated the pixel of the output image $g(x,y)$ as the sum

$$g(x,y) = \sum_{i=x-r}^{x+r} \sum_{j=y-r}^{y+r} w_{ij} f(i,j) . \tag{2}$$

Although as it is described in [6] the weights w_{ij} need not be explicitly computed and the output value $g(x,y)$ is calculated by following iterative algorithm:

1. Initialize $g(x,y)^{(0)}$ as in the case of the arithmetic filtering, as the mean value of its neighbors, including itself. If the sample variance of the neighborhood of the pixel is greater than zero set the iteration index k = 1 else stop.
2. Calculate the parameters $\beta^{(k)}$ and $\alpha_{ij}^{(k)}$ for $i, j = 1,2,\ldots,2r+1$:

$$\beta^{(k)} = \left(g(x,y)^{(k-1)}\right)^{-2} \tag{3}$$

$$\alpha_{ij}^{(k)} = \left(f(i,j) - g(x,y)^{(k-1)}\right)^{-2} \tag{4}$$

3. Update the average $g(x,y)^{(k)}$ for *k*th iteration

$$g(x,y)^{(k)} = \frac{\sum_{i=x-r}^{x+r}\sum_{j=y-r}^{y+r}\alpha_{ij}^{(k)} f(i,j)}{\beta^{(k)} + \sum_{i=x-r}^{x+r}\sum_{j=y-r}^{y+r}\alpha_{ij}^{(k)}} \tag{5}$$

4. If $\left(g(x,y)^{(k)} - g(x,y)^{(k-1)}\right)^2 > \varepsilon$, then $k \leftarrow k+1$ and go to 2, else stop.

2.2. THE MODIFICATION OF EBWA METHOD

The method described in previous section is the modification of the Simplified Empirical Bayesian Weighted Averaging algorithm (SEBWA) which is presented in [7]. The original Empirical Bayesian Weighted Averaging algorithm (EBWA, also presented in [7]) assumes the gamma prior for parameter β with scale parameter λ and shape parameter p and exploits the iterative expectation-maximization technique. Thus the modified algorithm differs from previous only in step 2 where parameter $\beta^{(k)}$ can be calculated as conditional expected value

$$E(\beta^{(k)} \mid g(x,y)) = \frac{2p+1}{\left(g(x,y)^{(k-1)}\right)^2 + 2\lambda} \tag{6}$$

and assuming that p is a positive integer, the estimate of hyperparameter λ can be calculated by means of the first absolute sample moment:

$$\hat{\lambda} = \left(\frac{\Gamma(p)(2p-1)}{(2p-1)!!} 2^{p-\frac{3}{2}} g(x,y)\right)^2, \tag{7}$$

where $(2p-1)!! = 1\cdot 3 \cdot \ldots \cdot (2p-1)$.

2.3. THE MODIFICATION OF WACFM METHOD

In this section there is presented modification of the Weighted Averaging method based on Criterion Function Minimization (WACFM) [7]. The output value $g(x,y)$ can be calculated by iterative algorithm, alternately using the two formulas:

$$g(x,y) = \frac{\sum_{i=x-r}^{x+r}\sum_{j=y-r}^{y+r} w_{ij}^m f(i,j)}{\sum_{i=x-r}^{x+r}\sum_{j=y-r}^{y+r} w_{ij}^m}, \tag{8}$$

$$w_{ij} = \frac{\left(f(i,j) - g(x,y)\right)^{\frac{2}{1-m}}}{\sum_{i=x-r}^{x+r}\sum_{j=y-r}^{y+r}\left(f(i,j) - g(x,y)\right)^{\frac{2}{1-m}}}, \tag{9}$$

where $m \in (1,+\infty)$ is the parameter of the method and $g(x,y)^{(0)}$ is initialized as in the case of the arithmetic filtering, as the mean value of its neighbors.

2.4. THE MODIFICATION OF WAPM METHOD

The WAPM method (Weighted Averaging method based on Partition of input data set in time domain and using Criterion Minimization function) [7] can minimize the distance between the two averaged:

$$\left\| f^{(1)} w^{(1)} - f^{(2)} w^{(2)} \right\|, \tag{10}$$

where the set of input (the neighbors of the pixel) is divided into two subsets $f^{(1)}$ and $f^{(2)}$, for which they are determined appropriate weights in an iterative manner:

$$w^{(1)} = \left(\left(f^{(1)}\right)^T f^{(1)}\right)^{-1} \left(f^{(1)}\right)^T f^{(2)} w^{(2)} + \frac{1 - 1^T \left(\left(f^{(1)}\right)^T f^{(1)}\right)^{-1} \left(f^{(1)}\right)^T f^{(2)} w^{(2)}}{1^T \left(\left(f^{(1)}\right)^T f^{(1)}\right)^{-1} 1} \left(\left(f^{(1)}\right)^T f^{(1)}\right)^{-1} 1 \tag{11}$$

$$w^{(2)} = \left(\left(f^{(2)}\right)^T f^{(2)}\right)^{-1} \left(f^{(2)}\right)^T f^{(1)} w^{(1)} + \frac{1 - 1^T \left(\left(f^{(2)}\right)^T f^{(2)}\right)^{-1} \left(f^{(2)}\right)^T f^{(1)} w^{(1)}}{1^T \left(\left(f^{(2)}\right)^T f^{(2)}\right)^{-1} 1} \left(\left(f^{(2)}\right)^T f^{(2)}\right)^{-1} 1 \tag{12}$$

The output value $g(x, y)$ can be calculated as

$$g(x, y) = \frac{N_1 f^{(1)} w^{(1)} + N_2 f^{(2)} w^{(2)}}{N}, \tag{13}$$

where N_1 and N_2 are the cardinalities of the two subsets $f^{(1)}$ and $f^{(2)}$, and $N_1 + N_2 = N$.

3. NUMERICAL EXPERIMENTS

In this section there is presented performance of the described method for two synthetic images and a real one in presence of salt-and-pepper (appearing as white and black dots superimposed on an image) and Gaussian noise. In the case of salt-and-pepper noise the probability of the black pixel or the white pixel was equal 0.05 and in the Gaussian noise the mean was zero and the standard deviation was 0.15 (the values of pixels were in the range from zero to one). For computed output images the performance of tested methods is evaluated by the root mean-square error (RMSE) between the original image (without noise) and the output image. All experiments were run in the R environment (www.r-project.org).

First there was performed experiment with synthetic image of size 100x100 with only three gray levels. The image is presented in figure 1 (on the left). Figure 1 also presents the image disturbed by salt-and-pepper noise (in the middle) and the image disturbed by Gaussian noise (on the right).

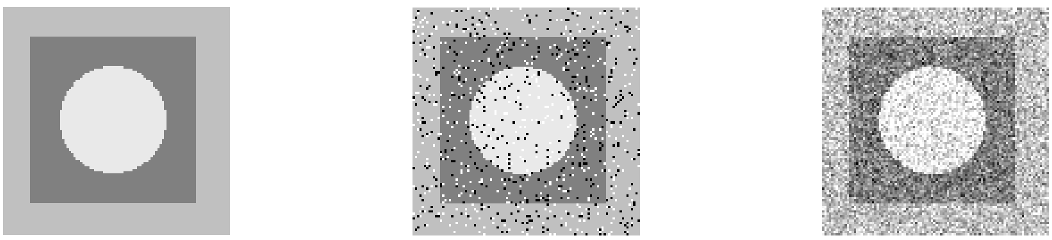

Fig. 1. Synthetic image of size 100×100, the image with salt-and-pepper noise and with Gaussian noise.

In this experiment two sizes of mask was used, namely 3x3 and 5x5, for both types of noise. The results, in form of RMSE, are presented in table 1, together with measured execution times of presented algorithms. Unfortunately in the case of the modification of WAPM method the implementation of the algorithm was often impossible because of the numerical problems arising, namely matrices appearing in the model proved to be singular. That is why the results presented in this section do not include the modified WAPM method.

As it is expected in the case of salt-and-pepper noise the best results are obtained using median filter. Bayesian averaging SEBWA and EBWA give similar results, and interesting fact relates to the same RMSE values obtained for WACFM and mean filter. Apart from these two methods it is visible deterioration of the results.

Table 1. The results for the synthetic image of size 100×100.

Method		**Salt-and-pepper noise**		**Gaussian noise**	
		3x3 mask	**5x5 mask**	**3x3 mask**	**5x5 mask**
SEBWA	RMSE Time [s]	0.02757982 2.53	0.03421315 2.79	0.06272246 2.93	0.05269513 2.68
EBWA	RMSE Time [s]	0.0296515 2.58	0.03426524 2.82	0.06272739 3	0.05268687 2.73
WACFM	RMSE Time [s]	0.06682974 2.19	0.05769279 2.31	0.05798072 2.43	0.05266158 2.34
Mean filter	RMSE Time [s]	0.06682974 0.01	0.05769279 0.01	0.05798072 0.01	0.05266158 0.01
Median filter	RMSE Time [s]	0.02038201 0.78	0.02621715 0.78	0.06638568 0.82	0.05208484 0.78

In the case of Gaussian noise as it is expected the best results are obtained using mean filter, although unexpected is the smallest RMSE value for median filter with 5x5 mask. Unlike the case of salt-and-pepper noise the larger mask leads to improvement of the results for all methods applied.

Times of executions do not seem to depend of the size of the mask. As it is expected the simplest algorithm, namely mean filter is the fastest one. Median filter is slower and uses the built-in *quicksort* algorithm. The proposed iterative methods are the slowest.

Figure 2 presents next synthetic image using in experiments (on the left) with the same image disturbed by salt-and-pepper noise (in the middle) and the image disturbed by Gaussian noise (on the right). The original image size is 256x256 and it contains 4 gray levels.

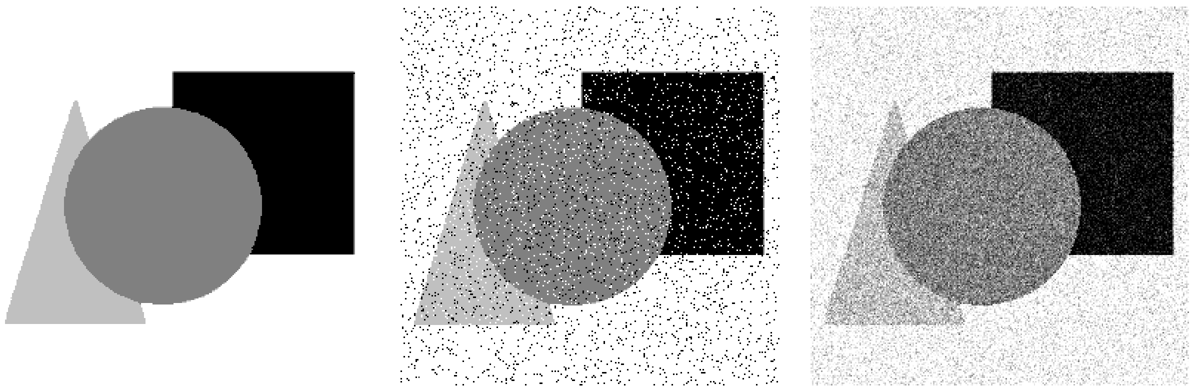

Fig. 2. Synthetic image of size 256×256, the image with salt-and-pepper noise and with Gaussian noise.

Results of the experiments for the image with two sizes of mask (3x3 and 5x5) and with both types of noise are presented in table 2. They are similar to the previous ones obtained for image visible in figure 1, but it can be seen longer computation time caused by the greater size of the image. Again unexpected is the smallest RMSE value for median filter in the case of Gaussian noise, probably caused by simplicity of the synthetic image.

Table 2. The results for the synthetic image of size 256×256.

Method		Salt-and-pepper noise		Gaussian noise	
		3x3 mask	**5x5 mask**	**3x3 mask**	**5x5 mask**
SEBWA	RMSE Time [s]	0.02527094 15.58	0.03017638 17.83	0.05913966 19.69	0.0605278 18.93
EBWA	RMSE Time [s]	0.02525737 16.26	0.03147555 18.24	0.05892399 20.34	0.0604633 19.55
WACFM	RMSE Time [s]	0.08116536 13.85	0.06973163 15.33	0.06512383 16.71	0.06779615 16.27
Mean filter	RMSE Time [s]	0.08116536 0.06	0.06973163 0.01	0.06512383 0.03	0.06779615 0.03
Median filter	RMSE Time [s]	0.02265219 5.29	0.02094514 5.31	0.0520038 5.35	0.03970118 5.49

Figure 3 presents real biomedical image using in experiments (on the left) with the same image disturbed by salt-and-pepper noise (in the middle) and the image disturbed by Gaussian noise (on the right). The original image size is 256x256 and it contains 256 gray levels.

Fig. 3. Real image of size 256×256, the image with salt-and-pepper noise and with Gaussian noise.

Results of the experiments for the image with two sizes of mask (3x3 and 5x5) and with both types of noise are presented in table 3. As it is expected in the case of salt-and-pepper noise the best results are obtained using median filter and in the case of Gaussian noise, computation time depends on the size of the image. In the case of salt-and-pepper noise it is visible deterioration of results for larger mask and in the case of Gaussian noise using larger mask results in improvement of the RMSE values.

Summarizing, the results of performed experiments show that proposed adaptive weighted averaging filtering methods, especially the Bayesian ones, are close to optimal for both salt-and-pepper and Gaussian noise. Since in reality the noise is usually characterized by mixture of these two types, the authors expect that the new methods could be useful in application to real images.

Unfortunately in the case of the modification of WAPM method in the direct implementation of the algorithm appears numerical instability during inverse matrices calculating which makes difficult to compare the obtained results with other algorithms. Probably it is caused by too little variability of gray levels occurring in the analyzed images. Described above algorithms could be also used for color images where each pixel is described by three dimensions and in such cases the problem of little variability of vector values of neighboring pixels will be less troublesome.

Table 3. The results for the real image of size 256×256.

Method		Salt-and-pepper noise		Gaussian noise	
		3x3 mask	**5x5 mask**	**3x3 mask**	**5x5 mask**
SEBWA	RMSE Time [s]	0.03365077 18.72	0.04636494 17.52	0.06021915 19.80	0.04925835 18.91
EBWA	RMSE Time [s]	0.03361484 19.09	0.04635727 18.36	0.06024201 20.69	0.04924985 18.47
WACFM	RMSE Time [s]	0.06429353 16.79	0.05428829 16.10	0.05414105 16.72	0.04857111 16.17
Mean filter	RMSE Time [s]	0.06429353 0.03	0.05428829 0.03	0.05414105 0.03	0.04857111 0.04
Median filter	RMSE Time [s]	0.02155204 5.65	0.03161807 5.33	0.06483689 5.34	0.05176399 5.41

It is worth noting that the iterative procedures to obtain weights for each pixel in image can be performed in parallel. Thus the main disadvantage of the proposed methods –computational complexity is to be reduced when the algorithms will be implemented in the NVIDIA programming model – CUDA. It allows programmers to write scalable parallel programs using a straightforward extension of the C language [3].

4. ACKNOWLEDGMENTS

This work was financed in part by the Polish National Science Centre.

BIBLIOGRAPHY

[1] BRUCE E.N., Biomedical signal processing and signal modelling, Wiley, New York, 2001.
[2] DAVIES E.R., Machine Vision: Theory, Algorithms and Practicalities, Academic Press, San Diego, 1990.
[3] GARLAND M., GRAND LE.S., NICKOLLS J., ANDERSON J., HARDWICK J., MORTON S., PHILLIPS E., ZHANG Y., VOLKOV V., Parallel Computing Experiences with CUDA, Journal IEEE Micro, 2008, Vol. 28, No. 4, pp. 13-27.
[4] GONZALEZ R.C., WOODS R.E., Digital Image Processing. Prentice Hall, Upper Saddle River, New Jersey, 2002.
[5] MOMOT A., Fuzzy Weighted Averaging of Biomedical Signal Using Bayesian Inference., In: CYRAN K.A. et al. (Eds.) Man-Machine Interactions, Advances in Intelligent and Soft Computing, Springer-Verlag Berlin Heidelberg, 2009, Vol. 59, pp. 133-140.
[6] MOMOT A., Application of Adaptive Weighed Averaging to Digital Filtering of 2D Images, In: PIETKA E., KAWA J. (Eds.) Information Technologies in Biomedicine (Vol. 2), Advances in Soft Computing, Springer-Verlag Berlin Heidelberg, 2010, Vol. 69, pp. 33–44.
[7] MOMOT A., Methods of Weighted Averaging with Application to Biomedical Signals, In: GARGIULO G., MCEWAN A. (Eds.) Applied Biomedical Engineering, InTech, Rijeka, Croatia, 2011, pp. 361-386.
[8] XU Q., MA L., NIE W., PENG L., JIAWAN Z., JIZHOU S., Adaptive Fuzzy Weighted Average Filter for Synthesized Image, In: GERWASI O. et al.(eds.) Proc. Int. Conf. on Computational Science and Its Applications ICCSA 2005, Lecture Notes in Computer Science, Springer, Heidelberg, 2005, Vol. 3482, pp. 292-298.