Non-local Sigma Filter

Nikolay Ponomarenko[1], Vladimir Lukin[1], Jaakko Astola[2], and Karen Egiazarian[2(✉)]

[1] Department of Transmitters, Receivers and Signal Processing, National Aerospace University, 17 Chkalova St., Kharkiv 61070, Ukraine
nikolay@ponomarenko.info, lukin@ai.kharkov.com

[2] Department of Signal Processing, Tampere University of Technology, P.O. Box-553 FIN-33101, Tampere, Finland
{jaakko.astola,karen.egiazarian}@tut.fi

Abstract. This paper proposes a non-local modification of well-known sigma filter, Nonlocal Sigma filter (NSF), intended to suppress additive white Gaussian noise from images. Similarly to the Nonlocal Mean Filter (NLM), every output pixel value is computed as a nonlocal weighted average of pixels coming from similar patches to the patch around the current pixel. The main difference between the proposed NSF and NLM is in the following: there are pixels in NSF not used in a weighted averaging (if the difference between them and the central pixel value is above a predefined threshold value, and if the distance between patch neighborhood and the central patch neighborhood is greater than a second threshold value). The weights used to estimate the output pixel depend on the patch size as well as on a distance between considered and reference patches. The proposed filter is compared to its counter-parts, namely, the conventional sigma filter and the NLM filter. It is shown that NSF outperforms both of them in PSNR and visual quality metrics values, PSNR-HVS-M and MSSIM. In this paper, a novel filtering quality criterion that takes into account distortions introduced into processed images due to denoising is proposed. It is demonstrated that, according to this criterion, NSF has similar edge-detail preservation property as the conventional sigma filter but has better noise suppression ability.

Keywords: Image denoising · Non-local methods · Similarity-based methods · Sigma filter · Human perception · Visual quality metrics · Image self-similarity

1 Introduction

Noise suppression is one of the main problems in image processing during several decades [1]. Recently, non-local image denoising has become very popular image filtering technique. It is based not only on statistical and spatial information extracted from a given pixel neighborhood but uses similarity of image fragments [2-4]. In turn, this leads to a better separation of image from noise compared to local image denoising methods. Besides, recent studies in the field of visual image quality assessment [5] show that human perception uses a self-similarity in images. Then, one can expect that the use of this self-similarity in image denoising can result in improved visual quality of filtered images.

V. Murino and E. Puppo (Eds.): ICIAP 2015, Part II, LNCS 9280, pp. 483–493, 2015.
DOI: 10.1007/978-3-319-23234-8_45

Alongside with the design of rather complex non-local filters processing data in several iterations and/or use of orthogonal transforms in the search of similar patches and their joint processing [3], a special attention is paid to the design of non-local versions of simple (basic) filters as, e.g., the mean filter [4]. Such investigations are important [2] since they allow better understanding of "bricks" put into basis of more complex methods. To our surprise, well known sigma filter [6] still does not have a non-local version and our goal in this paper is to design and study such a filter.

Here it is worth mentioning the following. The requirement to preserve important information in images while filtering them was and continues to be very essential. Any image filter introduces certain distortions (e.g. smears edges and fine details, destroys textures) in less or greater extent [1]. Then, in practice, despite an increase of signal-to-noise ratio due to filtering, an efficiency of solving the final task (e.g., image interpreting) using filtered images can decrease [7]. Customers of digital cameras with embedded filtering are often unsatisfied by denoising outcome. Many of them switch off image filtering mode since processed images often occur to be smeared and important details appear visually worse than those before processing.

Thus, a traditional formulation "better noise suppression under a condition of acceptable information preservation" permanently changes to "less introduced distortion under a condition of sufficient noise suppression". Conditionally, this can be treated as a "distortion-free image denoising".

Among filters preserving an important information (edges, details, texture) conventional sigma filter is one of the best. As its drawbacks, insufficient noise suppression in homogeneous image regions is usually regarded. Taking this into account, several modifications of the conventional sigma filter have been designed to improve its noise suppression efficiency [8] and to provide robustness [9]. In this paper, a non-local version of the sigma filter is proposed. We check a hypothesis that the use of patch similarity is able to improve noise suppression without worsening information preservation ability. The proposed filter is compared to its closest counter-parts: conventional sigma-filter and non-local mean filter using PSNR values for output images and visual quality metrics PSNR-HVS-M [10] and MSSIM [11]. Besides, a new quality criterion is introduced where more attention is paid to distortions introduced to images.

2 Calculation of Non-local Sigma Filter

Recall that for the conventional sigma filter the output I^s_{kl} for a given distorted value I^n_{kl} which is the center of a sliding window in kl-th pixel is calculated as an average of sliding window pixel values that fall into interval $I^n_{kl}-2\sigma \dots I^n_{kl}+2\sigma$, where σ denotes a noise standard deviation, assumed a priori known or pre-estimated [12]. The value I^n_{kl} also takes part in averaging which is not weighted (all weights are equal).

Let us modify this filter taking into account similarity of image patches.

For each image block (patch) A of size NxN pixels, let us estimate its similarity with respect to each block B in a given spatial neighborhood. A spatial neighborhood is determined by a present width M of a search area around the block A (see Fig. 1). A total number of blocks in the search area is equal to $(2M+N-1)^2-1$ whilst a total number of values potentially able to take part in averaging is equal to $(2M+N)^2-1$.

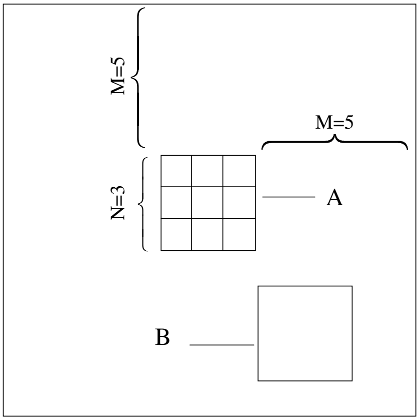

Fig. 1. Area of similar block search for the block *A* for *N=3, M=5*. Block *B* is one for which it's similarity to the block A is assessed.

Let us calculate the distance (similarity measure) for the blocks *A* and *B* as:

$$Diff(A,B)=\sum_{i=1}^{N}\sum_{j=1}^{N}(A_{ij}-B_{ij})^2 . \tag{1}$$

Then, for each pixel A_{ij}, it is possible to perform an estimation using the pixel B_{ij} with the weight W_{ij} as

$$W_{ij}=\begin{cases}0, & \left|A_{ij}-B_{ij}\right|>T\sigma \quad \vee \quad D_{ij}>\sigma^2\\ 1/max(D_{ij}+P,\sigma^2), & \left|A_{ij}-B_{ij}\right|\le T\sigma \quad \wedge \quad D_{ij}\le\sigma^2\end{cases}, \tag{2}$$

where T is a coefficient that determines the size of averaging σ-neighborhood (we have set $T=3$), D_{ij} is an estimate of MSE difference between the patches of pixels A_{ij} and B_{ij}, and P denotes a stabilizing term determined by the patch size N.

In the paper [4], an exponentially weighted aggregation of estimates is employed. Here, we use inverse proportionality calculation of weights depending on D_{ij} as it is done in [3]; such weighting seems to be more reasonable.

Let us explain the setting $T=3$. Recall that for the conventional sigma filter it is recommended to use $T=2$ which restricts its noise suppression ability. Meanwhile, the use of $T=3$ for the conventional sigma filters leads to worse preservation of an important information. In the modified version, we exploit image fragment similarity and the use of an additional restriction $D_{ij} \le \sigma^2$ which allows expecting acceptable edge/detail preservation even for $T=3$.

To calculate D_{ij}, we use *Diff(A,B)* subtracting contribution of the pixels A_{ij} and B_{ij}:

$$D_{ij}=\frac{Diff(A,B)-(A_{ij}-B_{ij})^2}{2(N^2-1)} \tag{3}$$

In [2, 4], D_{ij} is calculated without subtracting $(A_{ij} - B_{ij})^2$ as it is done in (3), i.e. a filtered pixel value is taken into account as well. To our opinion, this is not reasonable since in this case a larger weight W_{ij} is assigned to "similar" A_{ij} and B_{ij} which may result in less efficient noise suppression. There are also suggestions to use some kernels in (1) [4] to increase the weights for those pixels in the patch that are closer to the filtered one. However, we prefer not to use a multiplication by a kernel to avoid increased smearing of edges and details [2, 4].

The parameter P is introduced to compensate the influence of fluctuations of $Diff(A,B)$ that increase for smaller N. Suppose that a given image contains only noise and it has a Gaussian distribution with zero mean and variance σ^2. Then, it is easy to show that $Diff(A,B)/(N^2-1)$ is a random variable with the distribution $\chi^2(N^2-1)$. Taking into account the central limit theorem, for a large degree of freedom ($N^2>30$) the distribution $\chi^2(N^2-1)$ can be approximately considered Gaussian with the mean $2\sigma^2$ and the variance $8\sigma^4/(N^2-1)$.

Therefore, the obtained D_{ij} are random variables with variance $2\sigma^4/(N^2-1)$ and standard deviation $\sqrt{\frac{2\sigma^4}{N^2-1}}$. As the parameter P, we propose to use:

$$P = 2\sqrt{\frac{2\sigma^4}{N^2-1}}. \tag{4}$$

Each filtered pixel I^n_{kl} for different positions of a sliding window (block A) can occupy different positions in the block A. Then, after processing of the entire image, one obtains $U=(2M+N)^2-1$ estimates of B_{ij} with weights W_{ij} for this pixel. Let us group all these estimates and corresponding weights into arrays B^{all} and W^{all}, respectively. Then, the output of the non-local sigma-filter for the pixel I^n_{kl} is determined as:

$$I_{kl}^{NLS} = \frac{I_{kl}^n/\sigma^2 + \sum_{u=1}^{U} B_u^{all} W_u^{all}}{1/\sigma^2 + \sum_{u=1}^{U} W_u^{all}}. \tag{5}$$

Here $1/\sigma^2$ denotes the weight for the filtered pixel for which it is a priori known that it is corrupted by an additive white Gaussian noise with variance σ^2. In the paper [4], the authors propose to use the largest among W_u^{all} but, to our experience, the weighting in (5) works better.

3 Performance Comparison of Filtering Methods

In this section we compare a performance of the proposed filter to its counter-parts, conventional sigma-filter and non-local mean filter [4]. Three modifications of the non-local Sigma with $N=2$ (NLS2), $N=3$ (NLS3), $N=5$ (NLS5) are considered. For all three modifications we set $M=5$. In our comparison we use two variants of the non-local mean filter having patch sizes $3x3$ pixels (denoted as *NLM3*) and $5x5$ pixels (denoted as *NLM5*) with the same area of similar patch search as for *NLS*. Let us also consider conventional sigma filters with sliding window sizes *5x5* pixels (*Sig5*) and *7x7* pixels (*Sig7*) as well as a version of sigma filter with the neighborhood $\pm 3\sigma$ and the window size *7x7* pixels (*Sig73*). The filter *Sig73* is included into our analysis to demonstrate that for the conventional sigma filter the neighborhood increase leads to worse filtering.

Nine test images have been used in the experiments. Four of them are standard ones: ‘Cameraman’, ‘Barbara’, ‘Baboon’, and ‘Lena’. To this list we have added homogeneous (constant level) image (‘Homogeneous’), the gradient image (‘Gradient’) and three artificially created images, presented in Fig. 2. The image ‘Patterns’ contains different textures, gradients and small-size objects. The images ‘Cartoon’ and ‘Text’ have many sharp transitions (objects are inserted into background without antialiasing).

Fig. 2. Artificially created test images ‘Patterns’, ‘Cartoon’ and ‘Text’

The image ‘Cameraman’ is of size *256x256* pixels, all others are of size *512x512* pixels. All images are grayscale (they can be downloaded from http://ponomarenko.info/iciap2015set.zip).

Each test image has been corrupted by AWGN with variances σ^2 equal to *25, 100* and *400* and then processed by the considered filters. As it has been mentioned above, performance analysis is carried out using conventional criterion PSNR as well as two visual quality metrics PSNR-HVS-M and MSSIM, which are among the best for grayscale images.

Besides, to characterize distortions due to filtering, we have introduced the metric NAE (new aggregate distortions) and its derivatives, NMSE (new mean square error) and percentage of pixels that are more distorted after filtering (I_{kl}^{f}) than before filtering (denoted as NDP) with respect to the true value I_{kl}^{et}.

$$\mathrm{NDP} = \frac{100}{\mathrm{KL}} \sum_{k=1}^{K} \sum_{l=1}^{L} \delta_{kl}, \quad \delta_{kl} = \begin{cases} 1, & \left|I_{kl}^{et} - I_{kl}^{n}\right| < \left|I_{kl}^{et} - I_{kl}^{f}\right| \\ 0, & \left|I_{kl}^{et} - I_{kl}^{n}\right| \geq \left|I_{kl}^{et} - I_{kl}^{f}\right| \end{cases}, \tag{6}$$

$$\mathrm{NAE} = \frac{1}{\mathrm{KL}} \sum_{k=1}^{K} \sum_{l=1}^{L} \delta_{kl} \left(\left|I_{kl}^{et} - I_{kl}^{f}\right| - \left|I_{kl}^{et} - I_{kl}^{n}\right| \right)^2, \ \mathrm{NMSE} = 100\ \mathrm{NAE/NDP}.$$

Table 1 lists PSNR values for all filtered images. The best PSNR in each row is marked by **Bold**. As it is seen, the proposed filter *NLS2* outperforms *Sig5* and *Sig7* by *1...2* dB for standard test images and by *6...7* dB for artificial test images and the image 'Homogeneous'. The filter *Sig73* has PSNR smaller than for *NLS2* by *0.5...1* dB for all test images. The filter *NLM5* is sometimes the best, its PSNR can be larger than for *NLS2* by up to *1* dB (this happens for some standard test images in the cases of intensive noise). However, the difference in PSNR for *NLS2* and *NLM5* can be *10* dB and more for images with sharp transitions and small σ^2. In such cases, *NLM5* can even have smaller PSNR than before filtering. Interestingly, *NLS5* suppresses noise better than *NLM5* in homogeneous regions. This evidences in favor of weighted averaging methods determined by (2).

Fig. 3 presents PSNR averaged for all nine test images. As it is seen, the best results are provided by the proposed filter *NLS5*, other considered filters can be ranked as *Sig73, NLM5*, and *Sig7*. The reason why we present data for *NLS5* (but not for *NLS2* which usually has the better PSNR) will follow from the analysis of data presented in Tables 2 and 3 for the metrics PSNR-HVS-M and MSSIM, respectively.

Table 1. Comparison of analyzed methods, PSNR, dB (average of 100 experiments)

σ^2	Test image	Denoising method								
		none	Sigma filter			Non-local Mean		Non-local Sigma		
			Sig5	*Sig7*	*Sig73*	*NLM3*	*NLM5*	*NLS2*	*NLS3*	*NLS5*
25	Cameraman	34.1	36.8	36.9	36.8	34.8	33.9	37.5	**37.6**	37.4
	Barbara	34.1	35.7	35.8	35.7	36.6	36.7	36.8	**36.9**	36.5
	Lena	34.2	36.6	36.6	36.7	37.6	37.5	37.6	**37.7**	37.2
	Baboon	34.1	34.3	34.4	34.0	31.0	30.0	34.5	**34.6**	34.4
	Cartoon	36.1	41.3	41.7	42.5	32.3	30.5	**42.7**	42.5	41.7
	Patterns	35.0	38.6	38.9	39.4	31.5	30.2	41.6	**41.7**	41.2
	Text	34.1	41.4	42.3	47.2	38.6	36.5	**48.4**	47.8	45.5
	Gradient	34.2	41.6	42.4	47.9	46.4	48.3	**49.0**	**49.0**	48.9
	Homogeneous	34.1	41.7	42.5	48.2	46.7	48.7	49.5	**49.6**	**49.6**
100	Cameraman	28.3	32.0	32.1	32.4	32.4	32.0	33.2	**33.3**	32.9
	Barbara	28.1	30.8	31.0	31.3	33.4	**33.9**	32.8	33.2	32.7
	Lena	28.1	32.3	32.2	33.2	34.6	**34.9**	34.2	34.5	34.2
	Baboon	28.1	29.4	29.5	29.2	29.1	28.6	**29.7**	**29.7**	29.3
	Cartoon	30.0	35.1	35.4	36.0	31.3	29.6	**36.4**	36.3	35.7
	Patterns	29.0	32.7	33.0	33.4	30.4	29.4	35.9	**36.1**	35.7
	Text	28.1	35.2	36.0	41.1	36.0	34.7	**42.1**	41.6	39.5
	Gradient	28.1	35.4	36.2	42.2	40.8	42.9	43.4	**43.5**	**43.5**
	Homogeneous	28.1	35.4	36.3	42.3	40.9	43.0	43.6	**43.7**	**43.7**
400	Cameraman	22.5	27.2	27.5	28.4	29.3	**29.4**	29.3	29.5	29.4
	Barbara	22.1	26.2	26.4	27.2	29.3	**30.0**	28.9	29.6	29.7
	Lena	22.1	27.6	27.8	29.9	30.8	**31.5**	30.8	31.1	31.1
	Baboon	22.1	24.9	25.0	24.9	**26.0**	25.8	25.7	25.7	25.1
	Cartoon	24.1	28.9	29.1	29.3	28.1	27.2	**30.0**	30.0	29.6
	Patterns	23.1	27.2	27.3	27.9	27.7	27.3	29.9	**30.4**	30.1
	Text	22.1	29.0	29.7	34.0	31.6	31.6	**35.4**	35.1	33.4
	Gradient	22.1	29.3	30.1	36.2	34.9	37.0	37.5	**37.6**	**37.6**
	Homogeneous	22.1	29.3	30.1	36.2	34.9	37.0	37.5	**37.6**	**37.6**

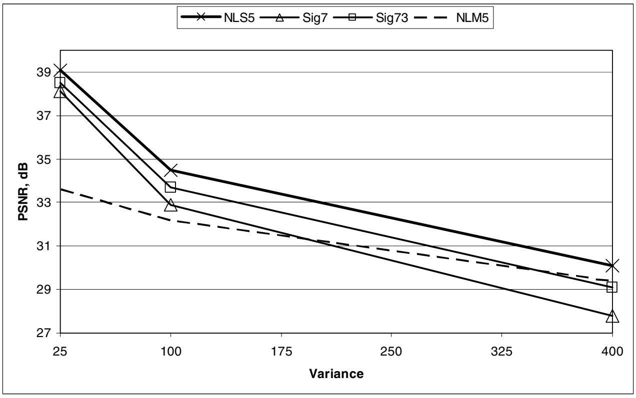

Fig. 3. PSNR averaged for all test images for three noise variances

Table 2. Comparison of analyzed methods, PSNR-HVS-M, dB

σ^2	Test image	none	Denoising method							
			Sigma filter			Non-local Mean		Non-local Sigma		
			Sig5	*Sig7*	*Sig73*	NLM3	NLM5	*NLS2*	*NLS3*	*NLS5*
25	Cameraman	38.2	40.6	40.9	39.8	39.3	38.2	40.8	41.3	**41.4**
	Barbara	39.0	40.5	40.5	39.2	39.9	39.6	40.1	40.5	**40.7**
	Lena	38.3	40.2	40.2	38.7	39.9	39.7	39.5	40.1	**40.4**
	Baboon	41.3	41.1	40.9	38.7	36.5	35.1	40.1	40.9	**41.4**
	Cartoon	38.0	39.8	40.0	40.0	37.2	36.3	**40.2**	**40.2**	**40.2**
	Patterns	40.2	42.9	43.5	42.8	36.6	35.0	44.5	45.1	**45.5**
	Text	37.5	42.1	43.8	45.9	45.3	44.1	48.9	**48.9**	48.6
	Gradient	36.2	40.9	42.7	45.3	46.5	47.6	**48.0**	47.9	47.8
	Homogeneous	36.1	40.8	42.6	45.3	46.8	47.9	**48.3**	**48.3**	**48.3**
100	Cameraman	31.5	34.3	34.8	33.7	34.7	34.1	34.9	35.5	**35.8**
	Barbara	31.7	33.8	34.0	33.0	34.7	34.9	34.1	34.7	**35.0**
	Lena	31.3	34.1	34.3	33.3	34.7	34.8	34.0	34.6	**35.2**
	Baboon	32.9	**33.6**	33.4	31.3	32.1	31.4	32.4	33.1	33.5
	Cartoon	31.7	33.5	33.8	33.7	33.5	32.7	**34.0**	**34.0**	**34.0**
	Patterns	33.6	36.2	36.7	35.8	34.4	33.1	37.8	38.5	**38.9**
	Text	31.3	35.8	37.4	39.5	40.2	39.7	42.2	**42.4**	42.1
	Gradient	30.1	34.7	36.5	39.3	40.8	42.0	**42.3**	**42.3**	**42.3**
	Homogeneous	30.1	34.7	36.5	39.4	41.0	42.2	**42.5**	**42.5**	**42.5**
400	Cameraman	25.2	28.3	28.9	28.1	29.8	29.6	29.4	29.8	**30.1**
	Barbara	24.9	27.6	28.1	27.5	29.4	29.5	28.8	29.3	**29.9**
	Lena	24.7	28.1	28.6	28.2	29.6	29.7	29.0	29.5	**30.0**
	Baboon	25.4	**27.0**	26.9	24.9	26.6	26.1	26.0	26.5	26.9
	Cartoon	25.3	27.1	27.3	26.9	**27.7**	27.3	27.5	27.6	27.5
	Patterns	26.7	29.2	29.6	28.4	29.8	29.3	30.6	31.6	**32.1**
	Text	24.9	29.4	31.0	32.7	34.0	34.2	35.5	**35.7**	35.2
	Gradient	24.1	28.7	30.4	33.2	34.9	36.0	**36.3**	**36.3**	**36.3**
	Homogeneous	24.0	28.6	30.3	33.2	34.8	36.0	36.2	**36.3**	**36.3**

According to data presented in Table 2, the filter *NLS5* is obviously the best (larger PSNR-HVS-M corresponds to better visual quality). One more interesting observation is that *Sig73* is often worse than *Sig7*, this is also confirmed by data in Fig 4.

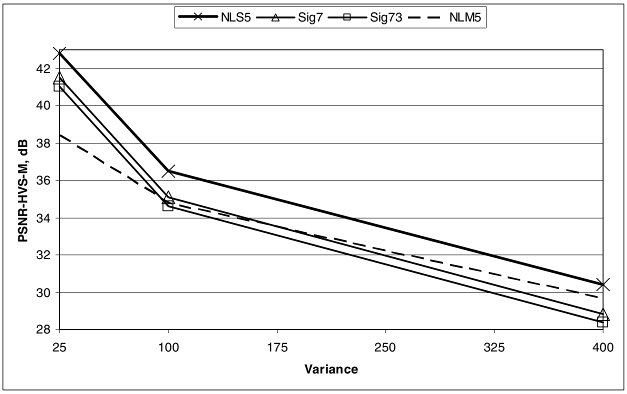

Fig. 4. PSNR-HVS-M averaged for all test images for three noise variances

Table 3. Comparison of analyzed methods, MSSIM

σ^2	Test image	Denoising method								
		none	Sigma filter			Non-local Mean		Non-local Sigma		
			Sig5	*Sig7*	*Sig73*	*NLM3*	*NLM5*	*NLS2*	*NLS3*	*NLS5*
25	Cameraman	0.977	0.990	0.991	0.991	0.991	0.990	**0.992**	**0.992**	**0.992**
	Barbara	0.984	0.991	0.991	0.990	**0.992**	**0.992**	**0.992**	**0.992**	**0.992**
	Lena	0.979	0.989	0.989	0.988	**0.990**	**0.990**	0.989	**0.990**	**0.990**
	Baboon	0.992	0.992	0.992	0.989	0.988	0.985	0.991	0.992	0.992
	Cartoon	0.991	0.998	0.999	0.999	0.998	0.997	**1.000**	0.999	0.999
	Patterns	0.989	0.996	0.997	0.998	0.996	0.994	0.998	**0.999**	**0.999**
	Text	0.974	0.993	0.995	0.998	0.998	0.998	**0.999**	**0.999**	**0.999**
	Gradient	0.948	0.985	0.989	0.995	0.995	0.996	**0.997**	**0.997**	**0.997**
	Homogeneous	0.947	0.984	0.988	0.995	0.995	**0.997**	**0.997**	**0.997**	**0.997**
100	Cameraman	0.928	0.969	0.973	0.976	0.979	0.979	0.980	0.982	**0.983**
	Barbara	0.947	0.973	0.975	0.976	0.981	**0.983**	0.980	0.982	**0.983**
	Lena	0.930	0.969	0.971	0.974	0.978	**0.979**	0.976	0.978	**0.979**
	Baboon	0.970	**0.977**	**0.977**	0.968	0.974	0.971	0.973	0.975	**0.977**
	Cartoon	0.972	0.993	0.995	0.997	0.996	0.995	**0.998**	**0.998**	**0.998**
	Patterns	0.965	0.987	0.989	0.993	0.991	0.991	0.995	0.995	**0.996**
	Text	0.923	0.974	0.980	0.991	0.992	0.993	**0.995**	**0.995**	**0.995**
	Gradient	0.838	0.943	0.957	0.980	0.983	0.988	**0.989**	**0.989**	**0.989**
	Homogeneous	0.834	0.940	0.955	0.979	0.983	0.987	0.988	**0.989**	**0.989**
400	Cameraman	0.825	0.916	0.928	0.944	0.953	0.955	0.955	0.957	**0.958**
	Barbara	0.856	0.922	0.931	0.939	0.953	0.958	0.952	0.958	**0.961**
	Lena	0.818	0.912	0.923	0.944	0.949	0.953	0.950	0.953	**0.955**
	Baboon	0.907	0.936	0.936	0.915	0.935	0.927	0.925	0.931	**0.937**
	Cartoon	0.925	0.976	0.981	0.984	0.987	0.986	**0.989**	**0.989**	**0.989**
	Patterns	0.911	0.959	0.965	0.976	0.978	0.979	0.983	**0.985**	**0.985**
	Text	0.828	0.919	0.935	0.968	0.970	0.977	**0.980**	**0.980**	**0.980**
	Gradient	0.616	0.814	0.855	0.927	0.939	0.955	0.958	**0.959**	**0.959**
	Homogeneous	0.605	0.803	0.846	0.920	0.934	0.951	0.954	**0.955**	**0.955**

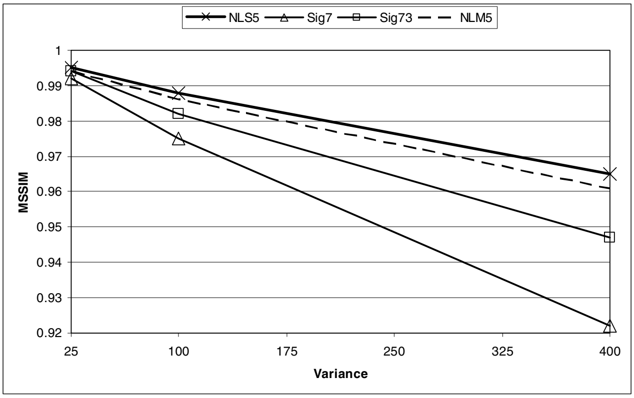

Fig. 5. MSSIM averaged for all test images from three noise variances

According to MSSIM (see Table 3 and Fig. 5, MSSIM=1 relates to perfect quality), the filter *NLS5* is again the best although the second place is occupied by *NLM5*.

Table 4. Comparison of analyzed methods, NMSE (NDP)

σ^2	Test image	Denoising method							
		Sigma filter			Non-local Mean		Non-local Sigma		
		Sig5	*Sig7*	*Sig73*	*NLM3*	*NLM5*	*NLS2*	*NLS3*	*NLS5*
25	Cameraman	5.1 (24)	4.8 (23)	9.9 (28)	36 (27)	47 (28)	7.1 (24)	6.0 (18)	**5.2 (14)**
	Barbara	5.2 (30)	4.3 (29)	8.4 (34)	12 (28)	13 (29)	6.7 (29)	6.2 (24)	**5.8 (17)**
	Lena	4.6 (25)	**4.2 (25)**	8.2 (30)	9.6 (26)	11 (27)	7.3 (28)	6.9 (25)	6.3 (21)
	Baboon	6.2 (39)	5.1 (38)	11 (44)	47 (51)	63 (53)	8.0 (34)	8.0 (17)	**6.7 (7)**
	Cartoon	3.1 (45)	**3.1 (44)**	3.8 (45)	75 (44)	118 (44)	3.7 (45)	3.7 (45)	3.7 (44)
	Patterns	3.7 (31)	3.4 (30)	4.6 (33)	130 (28)	189 (28)	3.7 (28)	3.8 (27)	**3.9 (25)**
	Text	1.6 (7)	**1.2 (4)**	1.6 (5)	110 (5)	146 (7)	1.4 (3)	1.6 (4)	1.9 (4)
	Gradient	1.5 (7)	1.2 (4)	1.3 (5)	**1.0 (3)**	**1.0 (3)**	1.1 (3)	1.1 (3)	1.1 (4)
	Homogeneous	1.5 (6)	1.1 (4)	1.2 (4)	**1.0 (2)**	**1.0 (2)**	1.0 (3)	1.1 (3)	1.1 (3)
100	Cameraman	15 (21)	15 (20)	34 (24)	48 (21)	64 (22)	28 (22)	24 (20)	**17 (16)**
	Barbara	14 (26)	**12 (25)**	27 (30)	19 (22)	20 (21)	21 (24)	19 (23)	16 (20)
	Lena	11 (20)	**10 (20)**	22 (23)	16 (18)	18 (19)	18 (21)	17 (21)	15 (19)
	Baboon	17 (36)	15 (35)	36 (41)	55 (39)	70 (41)	29 (38)	25 (29)	**23 (15)**
	Cartoon	12 (47)	**12 (46)**	15 (48)	71 (46)	115 (46)	14 (47)	15 (47)	14 (46)
	Patterns	12 (34)	**11 (33)**	17 (35)	120 (30)	173 (30)	13 (30)	13 (29)	14 (28)
	Text	3.8 (9)	**2.7 (6)**	3.9 (7)	86 (7)	115 (8)	3.8 (5)	4.1 (5)	4.9 (6)
	Gradient	3.4 (9)	2.2 (6)	2.5 (6)	**1.5 (3)**	**1.5 (3)**	1.7 (4)	1.8 (4)	1.9 (4)
	Homogeneous	3.3 (8)	2.1 (5)	2.4 (6)	**1.5 (3)**	**1.5 (3)**	1.6 (4)	1.7 (4)	1.8 (4)
400	Cameraman	41 (18)	**41 (17)**	84 (21)	76 (16)	93 (16)	80 (18)	75 (18)	70 (17)
	Barbara	40 (22)	**35 (21)**	92 (25)	45 (17)	49 (17)	63 (20)	55 (19)	50 (18)
	Lena	25 (16)	**25 (15)**	53 (17)	32 (14)	37 (14)	46 (16)	43 (16)	40 (15)
	Baboon	52 (29)	**50 (29)**	131 (35)	97 (30)	122 (31)	106 (32)	98 (31)	74 (24)
	Cartoon	**44 (49)**	45 (48)	62 (51)	100 (48)	136 (48)	59 (49)	59 (49)	59 (48)
	Patterns	42 (35)	**41 (34)**	67 (36)	141 (31)	188 (31)	54 (33)	52 (31)	52 (29)
	Text	12 (11)	**8.6 (8)**	18 (9)	81 (8)	118 (8)	17 (6)	18 (7)	21 (7)
	Gradient	11 (10)	6.1 (7)	7.0 (7)	**3.5 (4)**	**3.4 (4)**	4.1 (5)	4.3 (5)	4.6 (5)
	Homogeneous	11 (10)	6.0 (7)	7.0 (7)	**3.5 (4)**	**3.4 (4)**	4.1 (5)	4.3 (5)	4.7 (5)

Finally, Table 4 and Fig. 6 contain data for new criteria NMSE, NDP and NAE.

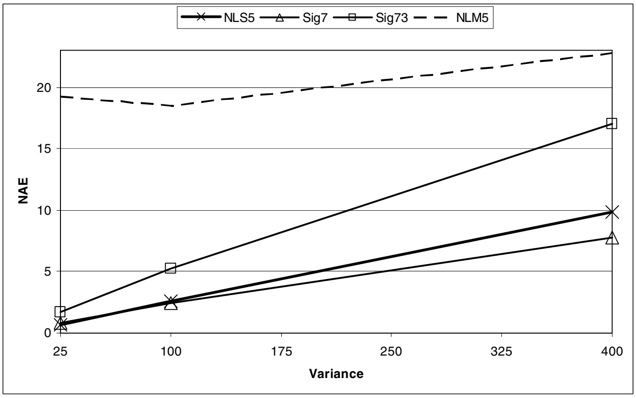

Fig. 6. Averaged values of NAE

According to these criteria characterizing information preservation (both tend to zero for ideal preservation), the best filter is *Sig7*. However, the proposed filter *NLS5* is only slightly worse and only for $\sigma^2=400$ although there are cases when *NLS5* is better (for example, for the test image 'Baboon' for $\sigma^2=25$).

Therefore, the obtained data allow concluding that the designed filter demonstrates better performance than the analyzed counter-parts. It suppresses noise better (especially, in images with sharp edges) whilst edge/detail preservation is practically the same as for the conventional sigma-filter.

4 Conclusions and Future Work

This paper presents a novel non-local Sigma filter. It is shown that this filter provides noise suppression efficiency at the same level as the non-local means filter whilst edge/detail preservation is practically similar to the conventional sigma filter.

Further studies might deal with a use of iterative filtering, orthogonal transforms as in [3], etc. Besides, it seems possible to use soft and adaptive thresholds [13], different sizes of patches and search area, combine different size patches, etc.

In this paper also new criteria of filtering efficiency are proposed. They are based on estimating level of distortions introduced by denoising.

References

1. Astola, J., Kuosmanen, P.: Fundamentals of nonlinear digital filtering, vol. 8. CRC Press (1997)
2. Deledalle, C.A., Duval, V., Salmon, J.: Non-Local Methods with Shape-Adaptive Patches (NLM-SAP). Journal of Mathematical Imaging and Vision **43**(2), 103–120 (2012).

3. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.O.: Image denoising by sparse 3-D transform-domain collaborative filtering. IEEE Trans. Image Process. **16**(8), 2080–2095 (2007)
4. Buades, A., Coll, B., Morel, J.M.: A non-local algorithm for image denoising. IEEE Computer Society Conference on Computer Vision and Pattern Recognition **2**, 60–65 (2005)
5. Ponomarenko, N., Jin, L., Lukin, V., Egiazarian, K.: Self-similarity measure for assessment of image visual quality. In: Blanc-Talon, J., Kleihorst, R., Philips, W., Popescu, D., Scheunders, P. (eds.) ACIVS 2011. LNCS, vol. 6915, pp. 459–470. Springer, Heidelberg (2011)
6. Lee, J.S.: Digital Image Smoothing and the Sigma Filter. Computer Vision, Graphics, and Image Processing, 255–269 (1983)
7. Lukin, V., Ponomarenko, N., Zelensky, A., Astola, J., Egiazarian, K.: Automatic design of locally adaptive filters for pre-processing of images subject to further interpretation. In: Proceedings of 2006 IEEE Southwest Symp. on Image Analysis and Interpretation, Denver, USA, pp. 41–45 (2006)
8. Lukin, V.V., Zelensky, A.A., Ponomarenko, N.N., Kurekin, A.A., Astola, J.T., Koivisto, P.T.: Modified sigma filter with improved noise suppression efficiency and spike removal ability. In: Proceedings of the 6-th Intern. Workshop on Intelligent Signal Processing and Communication Systems, Melbourne, Australia, pp. 849–853 (1998)
9. Alparone, L., Baronti, S., Garzelli, A.: A hybrid sigma filter for unbiased and edge-preserving speckle reduction. In: Proceedings of International Geoscience and Remote Sensing Symposium, pp.1409–1411 (1995)
10. Ponomarenko, N., Silvestri, F., Egiazarian, K., Carli, M., Astola, J., Lukin, V.: On between-coefficient contrast masking of DCT basis functions. In: Proc. of the Third International Workshop on Video Processing and Quality Metrics, USA, p. 4 (2007)
11. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multi-scale structural similarity for image quality assessment. In: IEEE Asilomar Conference on Signals, Systems and Computers, pp. 1398–1402 (2003)
12. Lukin, V., Abramov, S., Ponomarenko, N., Uss, M., Zriakhov, M., Vozel, B., Chehdi, K., Astola, J.: Methods and automatic procedures for processing images based on blind evaluation of noise type and characteristics. SPIE Journal on Advances in Remote Sensing (2011). doi:10.1117/1.3539768
13. Van De Ville, D., Kocher, M.: SURE-based non-local means. IEEE Signal Processing Letters **16**(11), 973–976 (2009)