Appeared in *Signal, Image and Video Processing*, Vol. 12, Issue 2, pp. 315-322, February 2018

Studying the Influence of Search Rule and Context Shape in Filtering Impulse Noise Images with Markov Chains

Arpad Gellert *, Remus Brad

Computer Science and Electrical Engineering Department, Lucian Blaga University of Sibiu, Emil Cioran Street, No. 4, 550025 Sibiu, Romania
* E-mail: arpad.gellert@ulbsibiu.ro

Abstract: The paper presents an improved context-based denoising method for grayscale images affected by impulse noise. The proposed algorithm is using Markov chains to replace the detected noise with the intensity having the highest number of occurrences in similar contexts. The context of a noisy pixel consists in its neighbor pixels and is searched in a larger but limited surrounding area. We have analyzed different search methods and different context shapes. The experimental results obtained on the test images have shown that the most efficient model applies the search in form of "*" of contexts in form of "+". Beside the better denoising performance obtained on all the noise levels, the computational time has been also significantly improved with respect to our previous context-based filter which applied full search of full context. We have also compared this improved Markov filter with other denoising techniques existing in the literature, most of them being significantly outperformed.

Keywords: Context-based filtering, Markov chain, denoising, impulse noise, salt-and-pepper noise

1. Introduction

As digital images are affected by noise during their acquisition or transfer, we are proposing a context-based method to eliminate salt-and-pepper noise from grayscale images, with an improved prediction scheme based on Markov chains. Salt-and-pepper is an impulse noise, consisting in white and black pixels altering the image. Our main goal is to restore the missing information and to preserve the unaffected pixels. In this respect, we are replacing the noisy pixel with the intensity having the highest number of occurrences in similar contexts within a limited surrounding area, like in a Markov chain. By using context information, our proposed filter can rebuild details in images altered by salt-and-pepper noise. In our previous work [1], we have applied a complete search in the limited surrounding area of the context, consisting in all the neighbor pixels. In this paper, we have continued our research by studying different search methods and different context shapes.

For validation, we have compared our technique with several denoising methods from the current literature, by measuring the mean square error (MSE) on some well-known test images like "Cameraman", "Boat" and "Airplane". The experiments have shown that our Markov filter significantly outperforms many existing impulse noise filters.

Further, we present some existing related denoising techniques. The median filter is one of the most employed methods to reduce impulse noise, with the drawback of being suitable only for low noise levels. Therefore, different improved median filter variants have been proposed over the years, which worked better on high noise densities. In [2], Srinivasan and Ebenezer proposed a decision-based method, which applies a 3×3 denoising window only on black and white pixels. In [3], the authors have introduced another median filter based method, which relaxes the order statistic for intensity substitution. The authors of [4] have presented the progressive switching median filter, which applies through several iterations an impulse noise detection algorithm and filtering.

In [5], the authors have introduced a two-level noise-adaptive fuzzy switching median filter. It identifies in the first stage the noisy pixels based on a histogram and replaces in the second stage the noisy pixels with the median of uncorrupted pixel values, applying also fuzzy reasoning. In [6] and [7], Chan et al. presented a two-stage scheme, using a median filter to identify pixels contaminated by noise and a specialized regularization method to restore the noisy pixels, by minimizing an objective function. In [8], the authors introduced an adaptive progressive filtering technique, which detects corrupted pixels based on two-dimensional geometric and size features of the noise. Based on the result of the first stage, an adaptively sized and shaped filtering window

(which in our work is fix sized and shaped) is employed in the second stage. Another two-stage scheme has been presented in [9] by Nasri et al., with noise detection in the first stage and Adaptive Gaussian Filtering in the second stage. The pixels detected as being noisy are stored in a binary noise matrix. The uncorrupted pixels from a fixed-size window are weighted by a Gaussian function. Then the denoised pixel intensity is computed as the normalized sum of these weighted values.

The main difference between the above described methods and our denoising scheme is that we use context information and therefore we can reconstruct better the details in the corrupted images. Our Markov filter could be also applied for defect detection, as in [10]. Other context-based filters have been also proposed. In [11], the authors suggested a probabilistic filter which, based on random walks on small image neighborhoods, can provide a good denoised estimation for a given pixel. In their work, the neighborhood dimension and shape are adjusted run-time. In our work, we use a fixed sized and shaped neighborhood and we search for similar neighborhoods. We replace a pixel affected by noise with the most frequent intensity occurred in similar neighborhoods. In [12], the authors presented a probabilistic denoising technique consisting in Markov-Chain Monte Carlo sampling. A method employing a dissimilarity measure for the local neighborhood of the noisy pixel was presented by Berkovich et al. [13]. The content-based kernel uses a statistical model to exclude dissimilar intensities from the weighted average. The kernel was adjusted to the image content to preserve the edges or textures.

Universal filtering algorithms, which can be used on different types of noise, have been also proposed. Besides the very common impulse noise, a Poisson type noise distribution was analyzed by Mishra et al. [14]. This variety of noise is present especially in medical x-Ray imaging and affects low intensity regions. A modified version of the Bilateral Filter was introduced, followed by a performance comparison. In [15], Smolka and Kusnik presented a robust local similarity filter to reduce mixed Gaussian and impulse noise from the affected images. In order to determine the distortion level of a pixel, they compute the similarity of the pixels from the processing region and a small filtering window centered on the pixel being restored, as a sum of the smallest distances.

The denoising operation is generally affecting areas with discontinuities, producing an unwanted smoothing effect. In the paper of Rouf and Ward [16], the fact that chromatic discontinuities have lower gradients than luminance was used in order to restore image areas affected by noise. The method can be employed to recover deleted information and improve the denoising process. In the same direction, the Sorted Switching Median Filter presented in [17], is a three stage filtering process that classifies the pixels and avoids the smoothing effect on uncorrupted areas. The multistage filtering process has been also employed by Liu et al. [18], with a new statistical process called ROD-ROAD and a fuzzy logic rule for the pixel classification at first, followed by a weighted mean filtering.

In [19], Bingham and Mannila used random projection as dimensionality reduction tool on high-dimensional image or text data sets and concluded that random projection is a promising alternative for noise reduction.

The rest of the paper is organized as follows: Section 2 describes the proposed Markov filter. Section 3 presents the experimental methodology whereas Section 4 the results. Finally, Section 5 summarizes the conclusions and suggests further work opportunities.

2. Filtering Impulse Noise Images with Markov Chains

Markov chains can be applied to compute the probability of a certain value in a sequence, as its number of occurrences in a considered context. Markov chains have been used in different computer science fields like bioinformatics [20], web access mining [21], pervasive computing [22], image retrieval [23], computational linguistics [24], etc. In an R^{th} order Markov model, the probability of the current state is computed based on R previous states [25], as follows:

$$P[q_t | q_{t-1},\ q_{t-2},\ ...] = P[q_t | q_{t-1},\ ...,\ q_{t-R}] \quad (1)$$

where q_t is the state at time t and R is the order of the Markov chain. A general prediction algorithm with Markov models, determining the next state of a 1D sequence based on the transition frequencies from the current state, was described in [26].

In [1], we reconstructed the grayscale images corrupted by impulse noise using Markov chains adapted for pixel intensities from 2D areas. The probability of pixel intensity in a certain context is computed as the number of its occurrences in similar contexts. The noisy pixel must be replaced with the predicted next state. The surrounding pixel values constitute the context and the search area is encoding the previous states. Thus, in grayscale images, the states are pixel intensities from the [0, 255] interval. The adjusted R^{th} order Markov model is given in (2), where CS is the context size (the width of the context square, as it is depicted in Figures 1 and 2), SR is the search radius (used to limit the search area), and W and H specify the image width and height. A certain pixel intensity $q_{x,y}$ depends on the neighbor context intensities. We have considered noisy the black and white pixels, as in [2].

$$\begin{aligned} &P[q_{x,y} | q_{x+i,y+j}, i, j = -SR, ..., SR,\ 0 \le x+i < W,\ 0 \le y+j < H, without\ i = j = 0] = \\ &= P\left[q_{x,y} \left| q_{x+i,y+j},\ i, j = -\frac{CS}{2}, ..., \frac{CS}{2},\ 0 \le x+i < W,\ 0 \le y+j < H, without\ i = j = 0 \right. \right] \end{aligned} \quad (2)$$

$$
\begin{aligned}
P[q_{x,y} \Big| & q_{x,y+j},\ j=-SR,...,SR,\ 0 \le y+j < H,\ without\ \ j=0; \\
& q_{x+i,y},\ i=-SR,...,SR,\ 0 \le x+i < W,\ without\ \ i=0; \\
& q_{x+k,y+k},\ k=-SR,...,SR,\ 0 \le x+k < W, 0 \le y+k < H,\ without\ \ k=0; \\
& q_{x-l,y+l},\ l=-SR,...,SR,\ 0 \le x-l < W, 0 \le y+l < H,\ without\ \ l=0] = \\
= P\Bigg[& q_{x,y} \Big| q_{x,y+j},\ j=-\frac{CS}{2},...,\frac{CS}{2},\ 0 \le y+j < H,\ without\ \ j=0; \\
& q_{x+i,y},\ i=-\frac{CS}{2},...,\frac{CS}{2},\ 0 \le x+i < W,\ without\ \ i=0 \Bigg]
\end{aligned}
\tag{3}
$$

The adjusted Markov filter given in (2) is depicted in Figure 1, where the noisy pixel *N* is colored with black, the context pixels *C* are dark gray and the search area is light gray. A noisy pixel is replaced with the most frequent noise-free intensity occurred in similar contexts within a larger surrounding area limited by *SR* (without leaving the image boundaries). As it can be observed in Figure 1, in [1] we have applied a full search within the search area (limited by SR) of a full context consisting in all the neighboring pixels. Further, we denote that filter *S0_C0*.

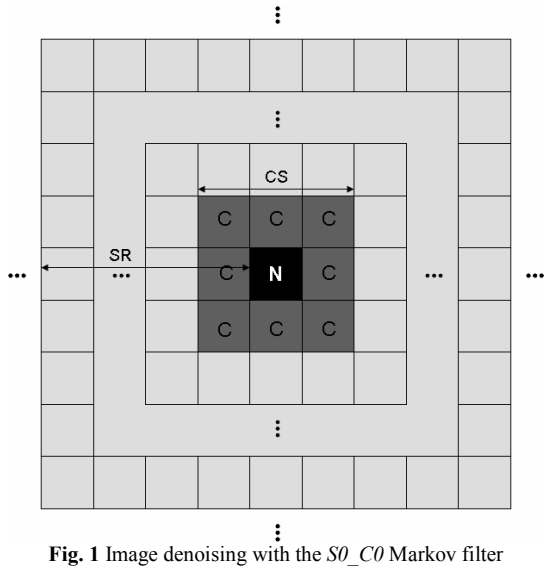

Fig. 1 Image denoising with the *S0_C0* Markov filter

In this work, we have investigated other simpler search rules and also different simpler context shapes. The goal is to improve the denoising performance and speed of the context-based filter. We tried to replace the full search (used in [1]) with a search in form of "+", "X" and also their combination in form of "*". We tried also to replace the full context (used in [1]) with different context shapes: the context in form of "+", "X" and their combination in form of "*".

The most efficient combination (determined on the test images), search in form of "*" of contexts in form of "+", is given in (3). This Markov filter is denoted *S*_C+*. Despite equation (3) seems more complicated than (2), in fact it is significantly simpler because it implies processing fewer pixels and thus we expect a faster filtering. We will also evaluate comparatively (2) and (3) and other variants in terms of denoising performance. Figure 2 presents the *S*_C+* Markov filter. The considered context pixels are highlighted with dark gray (forming a "+") and the search rule with light gray (forming an "*").

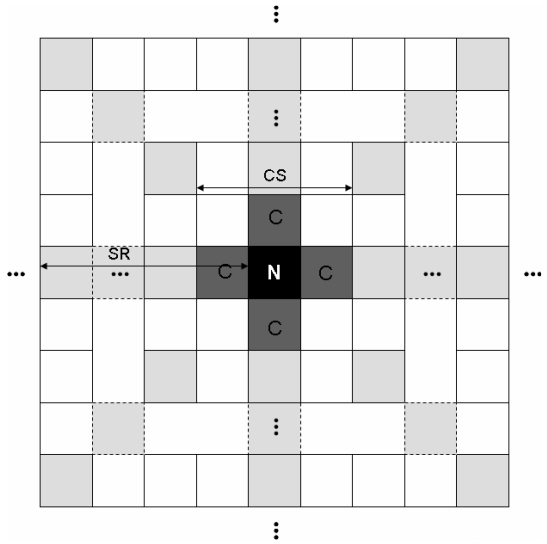

Fig. 2 Image denoising with the *S*_C+* Markov filter

Obviously, we have implemented and tested all the following combinations in which *S* denotes the search type and *C* the context type: *S0_C0* from [1], and from this work *S0_C+*, *S0_CX*, *S+_C0*, *S+_C+*, *S+_CX*, *SX_C0*, *SX_C+*, *SX_CX*, *S*_C+*.

The algorithm which replaces a noisy pixel through the *S*_C+* Markov model is described in the following pseudocode:

```
Markov(x, y, SR, CS, T)
  For j:=y-SR to y+SR, 0≤j<H
    If j=y then Continue
    If SAD(x, y, x, j, CS)<T
      AND NOT Salt_Pepper(x, j) then
        Q[Color(x, j)]:=Q[Color(x, j)]+1
  For i:=x-SR to x+SR, 0≤i<W
    If i=x then Continue
    If SAD(x, y, i, y, CS)<T
      AND NOT Salt_Pepper(i, y) then
        Q[Color(i, y)]:=Q[Color(i, y)]+1
  For k:=-SR to SR, 0≤x+k<W, 0≤y+k<H
    If k=0 then Continue
    i:=x+k
    j:=y+k
    If SAD(x, y, i, j, CS)<T
      AND NOT Salt_Pepper(i, j) then
```

```
      Q[Color(i, j)]:=Q[Color(i, j)]+1
  For k:=-SR to SR, 0≤x-k<W, 0≤y+k<H
    If k=0 then Continue
    i:=x-k
    j:=y+k
    If SAD(x, y, i, j, CS)<T
      AND NOT Salt_Pepper(i, j) then
        Q[Color(i, j)]:=Q[Color(i, j)]+1
  If Q[Max(Q)]=0 then Return Color(x, y)
  Return Max(Q)
```

The parameters of the *Markov* function are: the line and the column of the current pixel, the search radius *SR*, the context size *CS* and the similarity threshold value *T*. The first two *for* instructions are performing the "+" search and the last two the "X" search for similar contexts. These two search rules used together constitutes the "*" search. We considered two image areas similar if the sum of absolute differences is less than *T*.

The following pseudocode presents how we compute the similarity degree in the *S**_*C*+ Markov filter, the context having a "+" shape:

```
SAD(x1, y1, x2, y2, CS)
  S:=0
  For j:= -CS/2 to CS/2, 0≤j+y1<H, 0≤j+y2<H do
    If j=0 then Continue
    S:=S + |Color(x1, j+y1)-Color(x2, j+y2)|
  For i:= -CS/2 to CS/2, 0≤i+x1<W, 0≤i+x2<W, do
    If i=0 then Continue
    S:=S + |Color(i+x1, y1)-Color(i+x2, y2)|
  Return S
```

The first *for* instruction is processing the pixels from the vertical line and the second one from the horizontal line of the "+" context shape, both avoiding the middle pixel.

The frequencies of the noise-free pixel values occurring in similar contexts are kept in *Q*. The *Max* function returns the most frequent intensity which will replace the noisy pixel. The noisy pixel is not changed if the *Markov* function cannot find any similar context. The *Salt_Pepper* function checks if a pixel is noisy, by returning TRUE for black and white pixels. The *Markov_Filter* function, which calls the previously presented *Markov* function, is the same as in [1] and it is presented in the following pseudocode:

```
Markov_Filter(CS, SR, T)
  For i:=0 to W-1 do
    For j:=0 to H-1 do
      If Salt_Pepper(i, j) then
        Set_Color(i, j, Markov(i, j, CS, SR, T))
```

where the *Set_Color* function changes the intensity of the noisy pixel (*i*, *j*) with the value returned by the *Markov* function.

3. Evaluation Methodology

The proposed Markov filter was implemented in C# and we used for comparisons the available Matlab source codes of several filters. We performed the evaluations on the Cameraman, Boat and Airplane 512×512 grayscale PNG images having salt-and-pepper noise levels between 10% and 90%. The proposed Markov filter has been configured on the Cameraman image with 30% noise level and, after that, we compared the optimal model with the other existing techniques on all the three test images with all the noise levels.

The denoising performance has been determined using the MSE metric whose computation is given in (4):

$$MSE = \frac{\sum_{i=0}^{W-1}\sum_{j=0}^{H-1}\left(F(i,j)-O(i,j)\right)^2}{W \cdot H} \tag{4}$$

where *W* and *H* are the image width and height. The goal is to obtain the MSE as low as possible.

4. Evaluation Results

First, we have checked again the *SR* and *CS* parameters and the optimal values are the same as in [1]: *CS*=3 and *SR*=4. The optimal value of *T* for a full context was 500 in [1]. Since the number of pixels is reduced to the half in the "+" and "X" contexts, we expect a reduction of the optimal T to around 250 in the filters implying such contexts. In Figure 3 we have measured the MSE by varying the similarity threshold *T* around the expected optimal value. For this first parametrical setup we have chosen the *S*+_*C*+ Markov filter and the Cameraman test image with 30% noise.

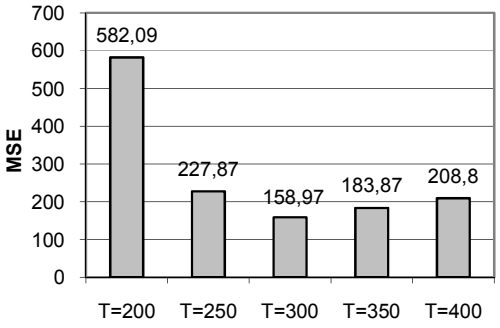

Fig. 3 The MSE of the 30% noised Cameraman image filtered with *S*+_*C*+ using different similarity thresholds

Figure 3 has shown that in the case of a "+" context the best value for *T* is 300. As we checked, for an "X" context the best *T* value is the same, which is obvious, since it implies the same number of pixels. Further we will use T=500 for a full context and T=300 for the "+" and "X" contexts.

Next, we have compared different search rule and context shape combinations. The MSE values obtained on the Cameraman, Boat and Airplane images are presented in Figures 4, 5 and 6, respectively. Since *S0_CX* was less performing than *S0_C*+ and also *SX_C0* was less performing than *S*+_*C0*, we have checked but not included the other models that imply "X" search or "X" contexts (*S*+_*CX*, *SX_CX* and *SX_C*+) in these figures.

Table 1 Comparison of the computation times in seconds

Noise density	Cameraman		Boat		Airplane	
	S0_C0	S*_C+	S0_C0	S*_C+	S0_C0	S*_C+
10%	9.38	2.29	10.63	2.63	10.51	2.41
20%	18.27	4.33	19.92	4.80	18.68	4.48
30%	26.56	6.27	27.88	6.74	26.64	6.39
40%	33.08	7.91	35.30	8.45	33.90	8.08
50%	40.02	9.49	41.87	9.96	41.00	9.68
60%	46.76	10.97	48.28	11.32	47.69	11.13
70%	53.21	12.45	54.43	12.59	53.81	12.50
80%	59.60	13.79	60.00	13.82	59.64	13.71
90%	65.95	14.88	66.51	14.84	66.38	14.82

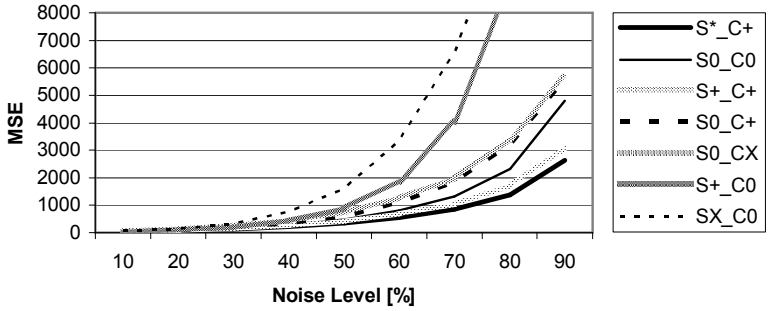

Fig. 4 The MSE of the Cameraman image denoised with different Markov models

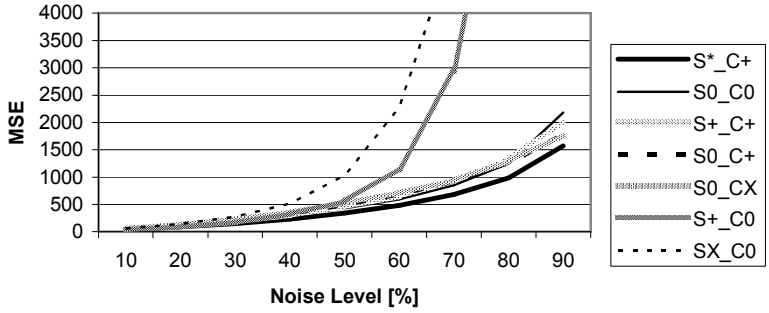

Fig. 5 The MSE of the Boat image denoised with different Markov models

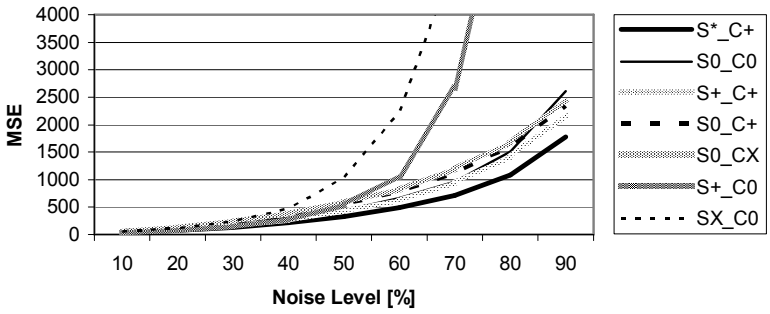

Fig. 6 The MSE of the Airplane image denoised with different Markov models

As Figures 4-6 show, the *S**_*C*+ is the best Markov filter on all the three test images. Thus, even if an "X" search is less performing than a "+" search, their combination into "*" search provides the best results. The proposed *S**_*C*+ model is significantly outperforming the initial *S0_C0* filter from [1], on all the noise levels. The denoising speed also decreased on all noise levels. As Table 1 shows, the *S**_*C*+ Markov filter is about four times faster than *S0_C0*.

The second best model is *S*+_*C*+, which is outperforming the initial *S0_C0* on the Cameraman and the Airplane images but it is worse on Boat with noise between 40-80%.

Further, we have compared this best *S**_*C*+ Markov filter with other existing filters: our previous context-based prediction filter (CBPF) [1], the Noise Adaptive Fuzzy Switching Median Filter (NAFSMF) [5], the Decision Based Algorithm (DBA) [2], the Median Filter (MF), the Progressive Switching Median Filter (PSMF) [4], the Relaxed Median Filter (RMF) [3] and the Analysis Prior Algorithm (APA) [27], whose source codes were available. Figures 7-9 are presenting comparatively the *MSE* for all the considered methods on the Cameraman, Boat and Airplane test images.

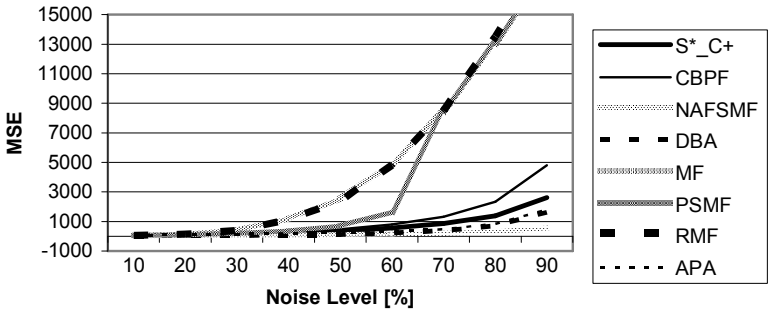

Fig. 7 Comparing the MSE on the Cameraman image denoised with different existing methods

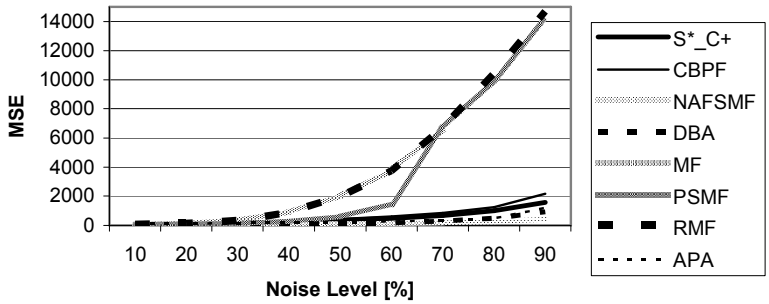

Fig. 8 Comparing the MSE on the Boat image denoised with different existing methods

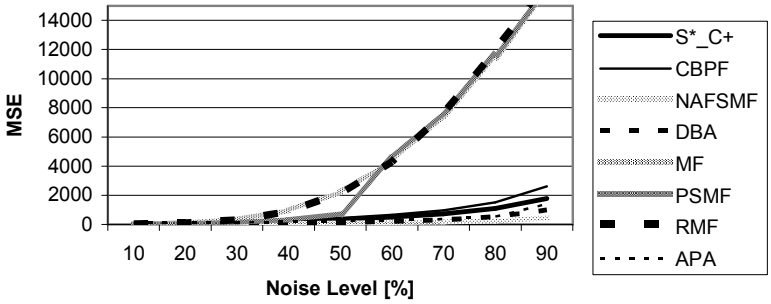

Fig. 9 Comparing the MSE on the Airplane image denoised with different existing methods

Fig. 10 Denoising the Cameraman image having 60% noise **(a)** using the *S**_*C*+ Markov filter **(b)**, CBPF **(c)**, NAFSM **(d)**, DBA **(e)**, MF **(f)**, PSMF **(g)**, RMF **(h)**, APA **(i)**

Figures 7-9 show that the proposed *S**_*C*+ Markov filter is better than the MF, PSMF, RMF and CBPF on all the noise levels. It is also better than APA on noise densities up to 30%. It is just slightly worse than the NAFSMF and DBA.

Figure 10 shows the Cameraman image having 60% salt-and-pepper noise (a) and the outputs obtained with our proposed *S**_*C*+ Markov filter (b), as well as using CBPF (c), NAFSM (d), DBA (e), MF (f), PSMF (g), RMF (h), APA (i).

As Figure 10 depicts, our proposed *S**_*C*+ Markov filter is better than the CBPF, MF, PSMF, RMF and APA techniques. We can observe again that the quality of the image denoised with the *S**_*C*+ Markov filter is very close to the quality of the images filtered with NAFSM and DBA.

5. Conclusions and Further Work

In this work, we have improved a context-based filter proposed in [1], to denoise grayscale images corrupted by impulse noise. Our filter is using Markov chains to replace the noisy pixel with the pixel value having the highest number of occurrences in similar contexts. The context of a noisy pixel consists in its neighbor pixels and is searched in a larger but limited surrounding area. The original contribution of this paper consists in analyzing different search rules and different context shapes.

We have replaced the full search used in [1] with a search in form of "+", "X" and also their combination in form of a "*". We have also replaced the full context used in [1] with different context shapes: "+", "X" and "*". The MSE results obtained on the Cameraman, Boat and Airplane test images show that the most efficient model is the proposed *S**_*C*+ Markov filter which applies the search in form of "*" of contexts in form of "+". This filter is better than our previous CBPF on all the noise levels, but also than the MF, PSMF, RMF and partially than APA and it is just slightly worse than the NAFSMF and DBA denoising methods. Beside the better denoising performance, the computational time has been also significantly improved with respect to the previous CBPF. The context information is a great advantage of our

method, whereas the computational time, despite it was significantly improved, is still a slight disadvantage compared with some existing techniques.

A further work direction could try to adjust dynamically the search radius or the context size. Other research directions are the run-time computation of the similarity threshold proportionally with the context size and the utilization of the Markov filter together with fuzzy and neural techniques.

References

1. Gellert, A., Brad, R.: 'Context-Based Prediction Filtering of Impulse Noise Images', IET Image Processing, 10(6), 429-437 (2016).
2. Srinivasan, K.S., Ebenezer, D.: 'A New Fast and Efficient Decision Based Algorithm for Removal of High Density Impulse Noise', IEEE Signal Processing Letters, 14(3), 189-192 (2007).
3. Hamza, A.B., Luque-Escamilla, P., Martínez-Aroza, J., Román-Roldán, R.: 'Removing Noise and Preserving Details with Relaxed Median Filters', Journal of Mathematical Imaging and Vision, 11(2), 161-177 (1999).
4. Wang, Z., Zhang, D.: 'Progressive Switching Median Filter for the Removal of Impulse Noise from Highly Corrupted Images', IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, 46(1), 78-80 (1999).
5. Toh, K.K.V., Isa, N.A.M.: 'Noise Adaptive Fuzzy Switching Median Filter for Salt-and-Pepper Noise Reduction', IEEE Signal Processing Letters, 17(3), 281-284 (2010).
6. Chan, R.H., Ho, C.-W., Nikolova, M.: 'Convergence of Newton's Method for a Minimization Problem in Impulse Noise Removal', Journal of Computational Mathematics, 22(2), 168-177 (2004).
7. Chan, R.H., Ho, C.-W., Nikolova, M.: 'Salt-and-Pepper Noise Removal by Median-type Noise Detectors and Detail-preserving Regularization', IEEE Transactions on Image Processing, 14(10), 1479-1485 (2005).
8. Xu, Z., Wu, H.R., Yu, X., Qiu, B.: 'Adaptive progressive filter to remove impulse noise in highly corrupted color images', Signal, Image and Video Processing, 7(5), 817-831 (2013).
9. Nasri, M., Saryazdi, S., Nezamabadi-pour, H.: 'A fast Adaptive salt and pepper noise reduction method in images', Circuits, Systems, and Signal Processing, 32(4), 1839-1857 (2013).
10. Krejcar, O., Frischer, R.: 'Non Destructive Defect Detection by Spectral Density Analysis', Sensors, 11(3), 2334-2346 (2011).
11. Estrada, F., Fleet, D., Jepson, A.: 'Stochastic Image Denoising', British Machine Vision Conference, London, p. 117 (2009).
12. Wong, A., Mishra, A., Zhang, W., Fieguth, P., Clausi, D.A.: 'Stochastic Image Denoising Based on Markov-Chain Monte Carlo Sampling', Signal Processing, 91(8), 2112-2120 (2011).
13. Berkovich, H., Malah, D., Barzohar, M.: 'Non-local means denoising using a content-based search region and dissimilarity kernel', 8th International Symposium on Image and Signal Processing and Analysis (ISPA), Trieste, pp. 10-15 (2013).
14. Mishra, B.K., Bharadi, V.A., Nemade, B., Thakur, K.V., Damodare, O.H., Sapkal, A.M.: 'Poisson Noise Reducing Bilateral Filter', Proceedings of International Conference on Communication, Computing and Virtualization (ICCCV) 2016, Procedia Computer Science, Vol. 79, pp. 861-865 (2016).
15. Smolka, B., Kusnik, D.: 'Robust local similarity filter for the reduction of mixed Gaussian and impulsive noise in color digital images', Signal, Image and Video Processing, 9(1), 49-56 (2015).
16. Rouf, M., Ward, R.K.: 'Retrieving information lost by image denoising', 2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP), Orlando, FL, pp. 1066-1070 (2015).
17. Horng, S.J., Hsu, L.Y., Li, T., Qiao, S., Gong, X., Chou, H.H., Khan, M.K.: 'Using sorted switching median filter to remove high-density impulse noises', Journal of Visual Communication and Image Representation, 24(7), 956-967 (2013).
18. Liu, L., Chen, C.P., Zhou, Y., You, X.: 'A new weighted mean filter with a two-phase detector for removing impulse noise', Information Sciences, 315, 1-16 (2015).
19. Bingham, E., Mannila, H.: 'Random projection in dimensionality reduction: applications to image and text data', 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, USA, pp. 245-250 (2001).
20. Jääskinen, V., Parkkinen, V., Cheng, L., Corander, J.: 'Bayesian clustering of DNA sequences using Markov chains and a stochastic partition model', Statistical Applications in Genetics and Molecular Biology, 13(1), 105-121 (2014).
21. Gellert, A., Florea, A.: 'Web Prefetching through Efficient Prediction by Partial Matching', World Wide Web: Internet and Web Information Systems, 19(5), DOI: 10.1007/s11280-015-0367-8, 921-932 (2016).
22. Gambs, S., Killijian, M.O., del Prado Cortez, M.N.: 'Next Place Prediction using Mobility Markov Chains', Proceedings of the First Workshop on Measurement, Privacy, and Mobility, New York, USA, p. 3 (2012).
23. Waghchawre, A.J., Shinde, J.V.: 'Query Mining for Image Retrieval System Using Markov Chain Model', International Conference on Emerging Trends in Computer Engineering, Science and Information Technology, India, pp. 109-112 (2015).
24. Mushtaq, A., Lee, C.-H.: 'An Integrated Approach to Feature Compensation Combining Particle Filters and Hidden Markov Model for Robust Speech Recognition', IEEE International Conference on Acoustics, Speech, and Signal Processing, Kyoto, Japan, pp. 4757-4760 (2012).
25. Rabiner, L.R.: 'A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition', Proceedings of the IEEE, 77(2), 257-286 (1989).
26. Gellert, A, Florea, A.: 'Investigating a New Design Pattern for Efficient Implementation of Prediction Algorithms', Journal of Digital Information Management, 11(5), 366-377 (2013).
27. Majumdar, A., Ward, R.K.: 'Synthesis and Analysis Prior Algorithms for Joint-Sparse Recovery', IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 3421-3424 (2012).