Impulse Noise Reduction for Texture Images Using Real Word Spelling Correction Algorithm and Local Binary Patterns

Shervan Fekri-Ershad[1], Seyed Fakhrahmad[2], and Farshad Tajeripour[2]
[1]Faculty of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran
[2]Department of Computer Science and Engineering, Shiraz University, Shiraz, Iran

Abstract: *Noise Reduction is one of the most important steps in very broad domain of image processing applications such as face identification, motion tracking, visual pattern recognition and etc. Texture images are covered a huge number of images where are collected as database in these applications. In this paper an approach is proposed for noise reduction in texture images which is based on real word spelling correction theory in natural language processing. The proposed approach is included two main steps. In the first step, most similar pixels to noisy desired pixel in terms of textural features are generated using local binary pattern. Next, best one of the candidates is selected based on two-gram algorithm. The quality of the proposed approach is compared with some of state of the art noise reduction filters in the result part. High accuracy, Low blurring effect, and low computational complexity are some advantages of the proposed approach.*

Keywords: *Image noise reduction, local binary pattern, real word spelling correction, texture analysis.*

Received June 22, 2015; accepted March 9, 2016

1. Introduction

Noise reduction is one of the challengeable research problems of Image processing. This problem appears to be very simple however that is not so when considered under practical situations, where the type of noise, amount of noise and the type of images all are variable parameters, and the single algorithm or approach can never be sufficient to achieve satisfactory results. In this respect, many approaches are pure applied on spatial domain, where non-linear filters are invented to reduce noise such as Min-Max Median filter [25], center weighted median filter [27], tri-state median filter [5], decision based filter [26] and etc. Some researchers tried to propose noise reduction algorithms which are applied in frequency domain [18] or combination of spatial and frequency [15]. A comparative study on noise types and noise reduction approaches is proposed in [9]. Noise reduction can be used in many applications as a preprocess step such as medical image processing [24] or visual inspection systems.

Texture is a repeating pattern of local variation in image intensities. Texture is a variation of data at scales smaller than the scales of interest [21]. Texture images are covered a huge number of images where are collected as database in very image processing applications. In Figure 1, an image is shown which consisting five different texture regions. Noise reduction approaches may be used for texture images as like as non-textures, but the results are not good enough, especially when the background knowledge of textural information is considered. The number of approaches, which has been proposed to reduce noises especially on textures are not much yet.

The main aim of this paper is to propose an efficient approach for noise reduction in texture images. The proposed approach is based on Real Word Spelling correction algorithms (RWSP), which are used in natural language processing problems.

Figure 1. An Example of texture image.

Real-word spelling errors are words in a text that occur when a user mistakenly types a correctly spelled word when another was intended [11]. Errors may be caused by the writer's ignorance of the correct spelling of the intended word or by typing mistakes. Such errors generally go unnoticed by most spellcheckers as they deal with words in isolation, accepting them as correct if they are found in the dictionary, and flagging them as errors if they are not.

For example, suppose a desired sentence, "the Barcelona football team may wind the game today".

The word “wind” is a real word in dictionary meaning "single turn", but it's meaning isn't correct in this sentence based on whole sentence meaning. It seems the word “win” should be replaced by “wind”.

Since now many approaches are proposed to solve RWSC problem accurately. Instead of their difference and diversity, common point of these approaches is using neighbor words meaning and its relation with desired miss spelling word.

The definition of Real Word Spelling Correction (RWSC) problem may be similar to texture noise reduction, when the texture image is considered as sentence and noisy pixel as miss-spelled word. The proposed approach includes two steps. First one is generating a candidate list of pixels, which are similar to noisy pixel based on local binary pattern features. The second step is choosing best one using bigram language modeling technique.

In the result part, the proposed approach is compared by some of state-of-the-art noise reduction filters in terms of Peak Signals to Noise Ratio (PSNR). The results are shown that the proposed approach provides more accuracy than others in texture images. The proposed approach is a general one for each type of noise but the results are shown its quality especially for impulse noises.

This paper is organized as follows: section 2 is related to the RWSC motivations and algorithms. In section 3 the proposed noise reduction approach is included which consists sub section about local binary patterns In section 4, the iterative version of proposed approach is described. Finally, experimental results and conclusion are included.

2. Real Word Spelling Correction

Practically, spelling errors in type written text vary between 1% and 3% [13], where 80% of them are usually caused by trivial editing operations such as insertion, deletion, substitution, and transposition [7].

Spelling error words can be categorized in two groups, non-words and real-words.

Words that can’t be found in dictionary are known as non-words. Some typing spelling errors may be occurred which are meaningful words. These errors are knows as real-words. Most of the RWSC algorithms are context-sensitive which means error correction is applied based on their context in the sentence. Several linguistic models and algorithms were proposed and experimented to solve this problem. The most prominent ones are Noisy Channel model, *n*-gram model, edit distance algorithm, and the context-sensitive error correction.

Most of them follows a twice steps as follows:

a) Candidate generating.
b) Choose the best.

a) *Candidate generating.* The aim of this step is to collect a candidate list of words from dictionary with minimum edit distance to miss-spelling word. The Minimum Edit Distance algorithm was first conceived by Wagner and Fischer [29]. It is defined as the minimum number of edit operations needed to transform a string *x* into a string *y*. These operations are insertion, deletion, and substitution. In spelling correction, the purpose of the Minimum Edit Distance algorithm is to reduce the number of candidate spellings by eliminating the candidates with maximum edit distance as they are considered to share fewer characters with the spelling error than other candidates. For example, some real words with 1-edit distance to "wind" are "mind", "win". 2-edit distance candidates may be "mink", "wild", and etc. There exist different edit distance algorithms: Levenshtein [14], Hamming [10] Longest Common Subsequence [1] and etc.

b) *Choose the best.* In order to correct the miss-spelled word, the best candidate word which is collected in previous step should be chosen. N-Gram models are used as a popular approach to done it accurately [3]. In short, an *n*-gram is simply a collocation of words that is *n* words long. For instance, "the dog" is a 2-gram sequence, "the dog smells" is a 3-gram sequence. The *n*-gram model calculates the conditional probability $P(w|s)$ of a word, *w* given the previous sequence of words *s*, that is, predicting the next word based on the preceding *n-1* words. For example, the conditional probability of $P(dog|the)$ consists of calculating the probability of the whole sequence "the dog" in the context. Put differently, for the word "the", the probability that the next word is "dog" is to be computed.

Since it is too complicated to calculate the probability of a word given all previous sequence of words, the 2-gram model is rather used most of the time. It is denoted by $P(w_n|w_{n-1})$ denoting the probability of a word w_n given the previous word w_{n-1}.

$$P(W_n \mid W_{n-1}) = Count(W_{n-1}, W_n) \,/\, Count(W_{n-1}) \qquad (1)$$

Where, count (w_{n-1},w_n) shows the occurrence frequency of sequence "$w_{n-1}w_n$" in corpus. Count (w_{n-1}) means the occurrence frequency of word w_{n-1}. For a sequence of n-grams, the probability is calculated as follows, Where S_n shows the sequence which includes n-grams.

$$P(S_n) = \prod_{k=1}^{n} P(W_k \mid W_{k-1}) \qquad (2)$$

Some algorithms had proposed to improve the *n*-gram model from different perspectives such as smoothing techniques [6], the weighted *n*-gram model [12], and the variable length *n*-gram model [17]. In order to choose the best candidate, 2-gram algorithm can be used. In this respect, suppose, W_i as miss spelled word that occurs in a context collocated some previous (W_{i-1}) and next words (W_{i+1}).

Candidate list which contains similar words are generated using previous step. The best one of the candidates can be chosen as follows:

$$P(C_j) = P(C_j \mid W_{i-1}) \times P(W_{i+1} \mid C_j) \quad (3)$$

Where, C_j means the j_{th} candidate. $P(C_j|W_{i-1})$ and $P(W_{i+1}|C_j)$ are 2-gram probabilities which are computed using a big corpus of real documents. Finally, the candidate with maximum probability Equation (3) is chosen as best.

In many cases, the error word which is typed may be a real one that is existed in dictionary (real words).

This problem is known as RWSC. In order to do RWSC accurately, test word should be included in candidate list and the same algorithm is applied. The schematic design of RWSC process is shown in Figure 2.

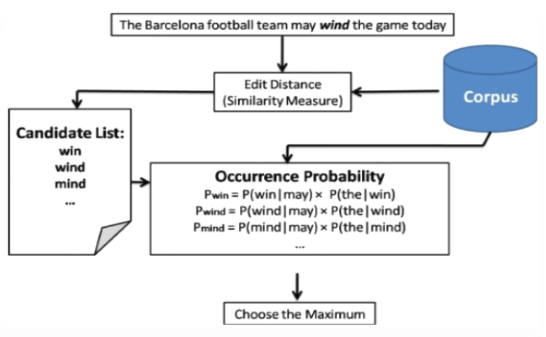

Figure 2. The schematic design of RWSC algorithm.

3. Proposed Noise Reduction Approach

The definition of RWSC problem may be very similar to texture image noise reduction problem, when the texture image is considered as sentence and noisy pixel as desired miss spelled word. In this respect, an approach is proposed in this section which consist two steps. The first one is generating a candidate list of pixels in test image, which are similar to noisy pixel based on local binary pattern features. The second step is choosing best one of them, using bigram language modeling technique.

The noise reduction must be done by repeating the algorithm for all of the pixels individually.

3.1. Local Binary Patterns

The Local Binary Pattern (LBPs) is a non-parametric operator which describes the local spatial structure and local contrast of an image. Ojala *et al.* [19] first introduced this operator and showed its high discriminative power for texture classification. At a given pixel position (X_c, Y_c), LBP is defined as an ordered set of binary comparisons of pixel intensities between the center pixel and its surrounding pixels. Usually to achieve the rotation invariant, neighborhoods would be assumed circular. So, points which the coordination's are not exactly located at the center would be found by interpolation. Some circular neighborhoods by radius (R) and (*P*) neighborhoods pixels are shown in Figure 3.

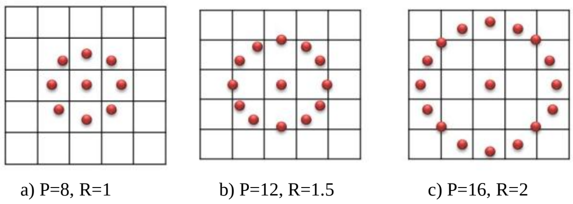

Figure 3. Some examples of circular neighborhoods.

Now, the LBPs are defined at a neighborhood of image by Equation (4).

$$LBP_{P,R} = \sum_{n=0}^{P-1} S(g_n - g_c)2^n \quad (4)$$

Where, "g_c" corresponds to the grey value of the centered pixel and "g_n" to the grey values of the neighborhood pixels. Also, P is the number of neighborhoods of center pixel, and function $s(x)$ is defined as:

$$S(x) = \begin{cases} 1 & if \quad x \geq 0 \\ 0 & if \quad x < 0 \end{cases} \quad (5)$$

According to [19], The $LBP_{P,R}$ operator produces (2P) different output values, corresponding to the 2P different binary patterns that can be formed by the P pixels in the neighbor set. The author's practical experience in [22], showed high complexity of basic LBP. To solve this, Ojala *et al.* [20] defined an uniformity measure "U", which corresponds to the number of spatial transitions (bitwise 0/1 changes) in the "pattern". It is shown in Equation (6). For example, patterns 00000000 have U value of 0, while 00011101 have U value of 3.

$$U(LBP_{P,R}) = \left|S(g_{p-1} - g_c) - S(g_0 - g_c)\right| + \sum_{i=1}^{P-1}\left|S(g_i - g_c) - S(g_{i-1} - g_c)\right| \quad (6)$$

In this version of LBP, the patterns which have uniformity amount less than U_T are categorized as uniform patterns and the patterns with uniformity amount more than U_T are categorized as non-uniform patterns. Finally, the LBP is computed as follows:

$$LBP^{riuT}_{P,R} = \begin{cases} \sum_{i=0}^{P-1} S(g_i - g_c) & if \quad U(LBP_{P,R}) \leq U_T \\ P+1 & Otherwise \end{cases} \quad (7)$$

Superscript "riu_T" reflects the use of rotation invariant "uniform" patterns that have U value of at most U_T. According to Equation (7), applying LBP will assign a label from 0 to P to uniform patterns and label P + 1 to non-uniform patterns.

Because of, just one label (*P* + 1) is assigned to all of the non-uniform patterns, so uniform labels should cover mostly patterns in the image. Tajeripour and Fekri-Ershad [28], and Fekri-Ershad and Tajeripour [8] show that if in the definition of LBP operator the value of U_T is selected equal to (P/4), only a negligible portion of the patterns in the texture takes label P + 1.

In order to extract the feature vector, first the $LBP_{P,R}$ is applied on the image and the labels are assigned each pixels. Then the occurrence probability of each label in the image is regarded as one of the dimensions of the feature vector. The occurrence probability of a specific label in the image can be approximated by the ratio of the number of that label to the number of all labels Equation (8).

$$d_i = N_{P_i} / N_{total} \qquad 0 \le i \le P+1 \tag{8}$$

Where, N_{Pi} is the number of pixels that labeled as P_i, and N_{total} is the number of pixels. So, for every image, a "P+2" dimensional feature vector can be extracted as follows:

$$F = < d_0, d_1, ..., d_{P+1} > \tag{9}$$

3.2. First Step (Candidate List Generating)

The aim of this part is to generate a candidate list of pixels which are more similar to noisy pixel. In RWSC problem, the candidates are selected from the dictionary with minimum edit distance, but in our application, candidates should be chosen from the textural image based on discriminative features.

In order to find similar pixels to noisy one, LBP are applied. Then feature vector (F_N) is extracted for noisy pixel considering a window by center of noisy pixel and size of N×N. According to our results N=8, provides maximum accuracy. Next, the feature vector (F_{ij}) should is computed for pixel I_{ij}. By repeating this process for all of the pixels, a vector is extracted for each one of them separately. Finally, the distance is computed between each one of the feature vectors and noisy vector using log-likelihood ratio. Other similarity/distance measures can be used, but Tajeripour and Fekri-Ershad practical analysis in [8, 28] are shown the ability of log-likelihood to provide discrimination. Log-likelihood is a non-similarity measure which is shown in Equation (10).

$$L_{i,j} = (F_{i,j}, F_N) = \sum_{k=0}^{P+1} F_{i,jk} \, log\left(\frac{F_{i,jk}}{F_{N_k}}\right) \tag{10}$$

Where, $L_{i,j}$ shows the non-similarity ratio between $I_{i,j}$ and noisy pixel and k shows the number of vector dimensions. In order to generate candidate list, the log-likelihood ratios should be sorted and M pixels which have minimum value is chosen as candidates.

According to our results, increasing M may improve computational complexity and decreasing M may decrease accuracy. In the result part, various number of M are trained and finally, M=10 provided maximum accuracy beside low computational complexity.

3.3. Second Step (Choosing Best One)

The aim of this section is to select best one of the candidates which are generated in previous section. In RWSC problem, N-gram algorithm is used by N=2 to done this step accurately as follows:

First of all, previous and next pixel's intensities of the desired noisy pixel are considered as "PRE" and "NEX". If the noise coordination is considered as (i,j), hence, PRE is (i,j-1) and NEX is (i,j+1).

Next, the co-occurrence probability of each candidate pixels and PRE is computed using Equation (11), which is generated based on Equation (1).

$$P(G_i \mid PRE) = Count(PRE \mid C_i) \,/\, Count(PRE) \tag{11}$$

Where, C_i shows the i_{th} candidate and Count (PRE,C_i) shows co-occurrence frequency of PRE and C_i intensities in texture image according to their coordination. C(PRE) shows the occurrence frequency of the intensity PRE in the test texture. The co-occurrence probability of each candidate pixel and NEX is computed in the same way.

Next, the Equation (12) is computed for each candidate to measure 2-gram as follows:

$$P(C_i) = P(C_i \mid PRE) \times P(NEX \mid C_i) \tag{12}$$

Notice, Equation (12) is defined based on Equation (3). Finally, the candidate with maximum probability P(C_i) is chosen as the best and the intensity of the noisy pixel will be changed by intensity of the best candidate in test image. The noisy pixels are not clear, hence, the noise reduction algorithm should be applied for each one of the pixels individually. In this respect, the test desired pixel must be included in candidate lists. In Figure 4, a textural image is shown with 4 gray-levels, where black pixel is considered as noise. Green pixels shows generated candidates.

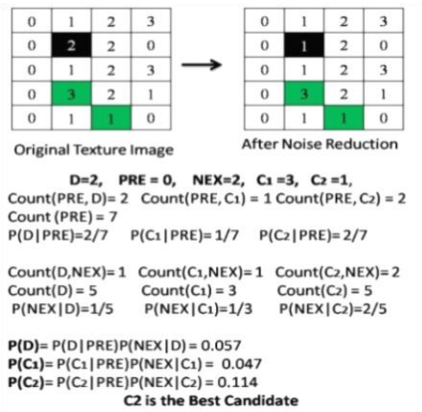

Figure 4. An Example of best candidate selection step.

4. Iterative Proposed Approach

In section3, an approach is proposed to reduce impulse noise in texture images. Noise reduction methods don't use anything as train image. The proposed approach should use noisy image to generate candidate list, which may decrease the accuracy. In section 3.2, the log-likelihood distance is computed between each extracted vector and desired one. Finally M pixels are chosen as candidates. Using this method may cause including other really noise pixels in candidate list which may decrease the accuracy. In this respect, an iterative enhanced algorithm is proposed as follows:

a. Appling first step on input image and using a threshold to choose candidates with distance lower than threshold.
b. Appling second step to select best candidate.
c. Change best one with desired pixel.
d. Increase threshold number and go to step A.
e. Repeat this algorithm for N times.

Using the proposed algorithm, pixels which are very similar to desired noisy pixel are selected in primary iterations. In this respect, the probability of choosing noisy pixels in second step may be decreased.

5. Experimental Results

In this paper, an approach was proposed for reducing impulse noises in texture images. Brodatz [4] is one of the state of the art texture albums. In order to evaluate performance, first 3 classes were chosen from Brodatz Album randomly, which are Grass, Wall, and Textile. Images were in size of 256×256.

Each image is cropped to 4 sub images in size of 64×64 without any overlap.

A salt and pepper noise is applied on images with different strength which are shown in Table 1. The proposed iterative version is applied with different numbers of iterations. Finally, the number of iterations is selected as 5 because of providing maximum accuracy. Lower numbers may decrease accuracy and upper numbers increases computational complexity. In order to apply $LBP_{P,R}$, different numbers of radius are tested, finally, $LBP_{8,1}$ provides maximum accuracy. The Peak Signal to Noise Ratio (PSNR) is used as measure to evaluate noise reduction performance [2, 23]. It is defined in decibel (db) and for gray scale images as follows:

$$PSNR = 10Log10 \times \frac{255 \times 255}{MSE} \tag{13}$$

$$MSE = \sum_{i=1}^{M}\sum_{j=1}^{N}(X_{i,j} - R_{i,j}) / M \times N \tag{14}$$

Where, X is the original image and R is restored image. MSE shows mean square error. Hence, M and N are size of image. X_{ij} shows the intensity of pixel (i,j) in original image. The higher PSNR shows the better quality for de-noised image. For comparison, some efficient noise reduction filters such as average and median are evaluated.

In average filter a square window of size 2k+1 is used. Here value of k changes from 1 to n. pixel (k+1, k+1) is considered as center. Using window original image is scanned row wise and column wise.

Each time of scan value of central pixel of window is replaced by the average value of its neighboring pixels comes within the window [16].Working of Median Filter is same as Average filter but here central pixel value is replace by the median value of its neighboring pixels comes within the window [16].

As it shown in Table 1, the PSNR ratio is higher in restored images with proposed approach than filters.

In low noise ratio levels, the PSNR of the proposed approach and filters are near, but it goes far in high noise levels.

Results show the quality of proposed approach for reducing impulse noises in texture images especially in high noise ratios.

Table1. Comparison results on brodatz album.

Noise Ratio	Grass		
	Proposed	**Average**	**Median**
10%	**27.04**	22.69	27.33
20	**23.69**	20.06	25.35
30	**21.81**	18.42	21.81
40	**21.80**	17.10	18.19
50	**19.26**	16.29	14.99
60	**19.31**	15.37	12.39
70	**17.36**	15.25	12.36
Noise Ratio	**Wall**		
	Proposed	**Average**	**Median**
10%	**29.90**	22.08	25.22
20	**27.30**	20.02	24.38
30	**25.32**	18.56	21.62
40	**23.88**	17.62	18.63
50	**18.72**	16.45	15.01
60	**18.61**	15.63	12.32
70	**17.16**	14.91	10.19
Noise Ratio	**Textile**		
	Proposed	**Average**	**Median**
10%	**29.02**	23.82	24.72
20	**28.45**	23.27	23.88
30	**26.73**	20.62	21.27
40	**24.80**	20.11	20.08
50	**20.93**	19.53	19.47
60	**20.46**	18.41	18.92
70	**19.83**	15.92	16.51

6. Conclusions

The aim of this paper was to propose a noise reduction algorithm for texture images. In this respect, an innovative algorithm was proposed based on RWSC theory. As it is shown in the result part, the proposed approach reduces impulse noises in texture images better than well known filters. Some advantages of the proposed approach are as follows:

- General Method, which can be used for all image types.
- Lower Blurring effect than well known filters.
- Creative fusion of Natural language processing (NLP) and Image Processing (IP) theories for the first time.

References

[1] Allison L. and Dix T., "A Bit-String Longest Common-Subsequence Algorithm," *Information*

Processing Letters, vol. 23, no. 5, pp. 305-310, 1986.

[2] Anjali P., Ajay S., and Spare S., "A Review on Natural Image Denoising using Independent Component Analysis Technique," *Advances in Computational Research*, vol. 2, no. 1, pp. 6-14, 2010.

[3] Bassil Y. and Alwani M., "Context-Sensitive Spelling Correction Using Google Web 1T 5-Gram Information," *Computer and Information Science*, vol. 5, no. 3, pp. 37-48, 2012.

[4] Brodatz P., *Textures: A Photographic Album for Artists and Designers*, Dover Press, 1966.

[5] Chen T., Ma K., and Chen L., "Tri-State Median Filter for Image Denoising," *IEEE Transactions on Image processing*, vol. 8, no. 12, pp. 1834-1838, 1999.

[6] Church K. and Gale W., "A Comparison of the Enhanced Good-Turing and Deleted Estimation Methods for Estimating Probabilities of English Bigrams," *Computer Speech and Language*, vol. 5, no. 1, pp. 19-54, 1991.

[7] Damerau F., "A Technique for Computer Detection and Correction of Spelling Errors," *Communication of the ACM*, vol. 7, no. 3, pp. 171-176, 1964.

[8] Fekri-Ershad S. and Tajeripour F., "A Robust Approach for Surface Defect Detection Based on One Dimensional Local Binary Patterns," *Indian Journal of Science and Technology*, vol. 5, no. 8, pp. 3197-3203, 2012.

[9] Gupta K. and Gupta S., "Image Denoising Techniques-A Review Paper," *International Journal of Innovative Technology and Exploring Engineering*, vol. 2, no. 4, pp. 6-9, 2013.

[10] Hamming R., "Error Detecting and Error Correcting codes," *The Bell System Technical Journal*, vol. 29, no. 2, pp. 147-160, 1950.

[11] Islam A. and Inkpen D., "Real Word Spelling Correction Using Google Web 1T 3-Grams," *in Proceedings of Conference on Empirical Methods in Natural Language Processing*, Singapore, pp. 1241-1249, 2009.

[12] Kuhn R. De M., "A cache-based Natural Language Model for Speech Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 6, pp. 570-583, 1990.

[13] Kukich K., "Techniques for Automatically Correcting Words in Text," *ACM Computing Surveys*, vol. 24, no. 4, pp. 377-439, 1992.

[14] Levenshtein V., "Binary Codes Capable of Correcting Deletions, Insertions, and Reversals," *Cybernetics and Control Theory*, vol. 10, no. 8, pp. 707-710, 1966.

[15] Malini S. and Moni R., "A Combined Spatial and Frequency Domain Approach for Removal of Impulse Noise from Images," *International Journal of Engineering and Advanced Technology*, vol. 4, no. 1, pp. 63-67, 2014.

[16] Nain A., Singhania S., Gupta S., and Bhushan B., "A Comparative Study of Mixed Noise Removal Techniques," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 7, no. 1, pp. 405-414, 2014.

[17] Niesler T. and Woodland P., "A Variable-length Category-based N-gram Language Model," *in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, Atlanta, pp. 164-167, 1996.

[18] Nisha S. and Mohideen K., "A Novel Approach for Image Mixed Noise Reduction Using Wavelet Method," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 11, pp. 795-799,2013.

[19] Ojala T., Maeenpaeae T., and Pietikaeinen M., "Texture Classification by Multi Predicate Local Binary Pattern Operators," *in Proceedings of 15th International Conference on Pattern Recognition*, Barcelona, pp. 951-954, 2000.

[20] Ojala T., Pietikainen M., and Maenpaa T., "Multi Resolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971-987, 2002.

[21] Petrov M. and Silva P., *Image Processing Dealing with Texture*, Willey and Sons Press, 2006.

[22] Pietikäinen M., Ojala T., and Xu Z., "Rotation-Invariant Texture Classification using Feature Distributions," *Pattern Recognition*, vol. 33, no. 1, pp. 43-52, 2000.

[23] Rahman M., Kumar M., and ShorifUddin M., "Optimum Threshold Parameter Estimation of Wavelet Coefficients Using Fisher Discriminant Analysis for Speckle Noise Reduction," *The International Arab Journal of Information Technology*, vol. 11, no. 6, pp. 573-581, 2014.

[24] Riji R., Rajan J., Sijbers J., and Nair M., "Iterative Bilateral Filter for Rician Noise Reduction in MR Images," *Signal, Image and Video Processing*, vol. 9, no. 7, pp. 1543-1548, 2014.

[25] Satpathy S., Panda S., Nagwanshi K., and Ardil C., "Image Restoration in Non-Linear Filtering Domain using MDB Approach," *International Journal of Information and Communication Engineering*, vol. 6, no. 1, pp. 45-49, 2010.

[26] Srinivasan K. and Ebenezer D., "A New Fast and Efficient Decision-Based Algorithm for Removal of High-Density Impulse Noises," *IEEE Signal Processing Letters*, vol. 14, no. 3, pp. 189-192, 2007.

[27] Subg-Jea K. and Hoon Y., "Center Weighted Median Filters and Their Applications to Image Enhancement," *IEEE Transactions on Circuits and systems*, vol. 38, no. 9, pp. 984-993, 1991.

[28] Tajeripour F. and Fekri-Ershad SH., "Developing a Novel Approach for Stone Porosity Computing Using Modified Local Binary Patterns and Single Scale Retinex," *Arabian Journal for Science and Engineering*, vol. 39, no. 2, pp. 875-889, 2014.

[29] Wagner R. and Fischer M., "The String-to-string Correction Problem," *Journal of the Association for Computing Machinery*, vol. 21, no. 1, pp. 168-173, 1974.

Shervan Fekri-Ershad received his M.Sc. degree from Shiraz University, Iran in 2012, majored in Artificial Intelligence. He is currently a PhD student in the School of computer engineering, Shiraz University, Iran. He joined the department of computer engineering at Najafabad Branch, Islamic Azad University, Isfahan, Iran as assistant Professor in 2015. His research interests are visual inspection systems, texture analysis, surface defect detection, etc.

Seyed Fakhrahmad received his BSc in Computer Engineering from Kharazmi University of Tehran (Iran) in 2003. His MSc and PhD degrees were received in Computer Engineering both from Department of Computer Science and Engineering, Shiraz University (Iran), in 2006 and 2011, respectively. He is currently an assistant professor in Department of Computer Science and Engineering, at Shiraz University. His research interests include Natural Language processing, Data Mining, Social Network Analysis and Fuzzy systems.

Farshad Tajeripour received the B.Sc. and M.Sc. degrees in electrical engineering from Shiraz University, in 1994 and 1997, and the Ph.D. degree in electrical engineering from the Tarbiat Modarres University of Tehran, in 2007. In 2007, he joined the Department of Computer Engineering at Shiraz University, Shiraz, as an Assistant Professor. His research interests include digital image processing, machine vision, medical image processing, signal processing, and vision based inspection systems.