International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 12 (2015) pp. 31131-31141

http://www.ripublication.com

Speech Signal Denoising Using Global Threshold Based Cellular Automata

Ahmed Sajjad Khan and * E.G. Rajan

Department of Electronics and Telecommunication
Anjuman College of Engineering and Technology Nagpur, India.
091-9823284086 Email: acetsajjadsir@yahoo.co.in
Professor Of Signal Processing
Professional Member, ACM Hyderabad, India
091-9849164747 E-mail: rajaneg@yahoo.co.in

Abstract

For speech signal denoising we use cellular automata concept. For denoising we use two different methods that is hard thresholding and soft thresholding. Here we will corrupt the speech signal by babble noise at different signal to noise ratio levels. typically 0 dB, 5 dB, 10 dB and 15 dB.
Using matlab we will simulate and find the results. Here we used hard thresholding and soft thresholding methods to calculated & compared Output Signal to Noise Ratio and Mean Square Error, and we found that at all input SNR levels the Soft thresholding method performs far better than hard thresholding method. Soft thresholding method shows a good 35.17 dB improvement in output SNR while using hard thresholding method the improvement in output SNR is very low just 21.79 dB.

Abbreviation — CA Cellular Automata, FT Fourier Transform, STFT Short Time Fourier Transform

Key words *:* Cellular Automata, Denoising, Thresholding, MSE Mean square error, SNR signal to noise ratio

1. INTRODUCTION

There are different types of noises present in the environment. These are white noise, pink noise, babble noise etc. these noise signals corrupt the speech signal. When these noisy signal is received at the receiver, the noises are removed to get back the original signal. The process of removing noise signal from the speech signal received is known as speech denoising.

In Speech processing, the noise signal deteriorates the quality of speech signal, so removing noise from speech signal is an area of interest for the researchers of this decade.

For speech denoising we generally use Wavelet transform methods. There are different applications of wavelet transform, such as 1D or 2D biomedical signal analysis, in video coding and forecasting, wavelet modulation in communication channels, producing & analyzing irregular signals or images etc

For speech denoising we can also use concepts of Cellular automata. By using cellular automata we can analyze according to scale. CA can decompose a signal into several neighbors and each neighbor represents different frequency bands. these neighbors also determines the approximate value the position of signals instantaneous structures. Such a property can be used for removing the noise. the information of the signal to be analysed is not there,but CA select information from speech signal without reducing its quantity.

Other way, CA takes all neighbor hood component of speech signal and transforms convert a signal into a series and it provides a way for analyzing the waveforms, the waveforms are bounded with frequency and duration. This helps us for storing signal more efficiently than by Fourier transform. Cellular Automata is preferred over Short Time Fourier Transform and Fourier Transform, as it provides multi-resolution. for the purpose of speech denoising various CA based methods have been proposed One of the method is based on the thresholding of the signal, here each Cellular automata coefficient of the signal is compared with a set threshold value,if the Cellular automata coefficient value is less than the set threshold value, then the output is made zero, else it is slightly reduced in amplitude. For denoising we use Soft Thresholding and Hard Thresholding methods. We use Cellular Automata to remove noise from a speech signal, for this we have to identify the components which contain the noise, and then we have to reconstruct the signal without those components.

2. CELLULAR AUTOMATA

cellular automata concept is based on cell, state of different cell, their neighbors and it is governed by different rules. As time advances at discrete steps, the state of the new cell is determined by the state of the present cell along with the state of its neighboring cell, that too according to a specific rule.

The rules of cellular automata are local and uniform. There are different models of cellular automata, they are one dimensional C.A, two dimensional C.A and three dimensional C.A. One dimensional C.A with two states consist of line of cells with values “0” and “1”.

If we are using One dimensional C.A with n- states, than each cell can have any integer value from 0 to n-1. The local rule will determine the status of next cell and the value of cells are updated at each discrete time steps. It means that the local rule will control the evolution of cellular automata model.

We can define cellular automata model as follows [1]. A cellular automata is a 4 – tuple (L,S,N,F), here L → regular lattice of cells, S → finite state of cells, N→ finite set of neighbors which indicates the position of a cell in relation to other cells of the lattice N, and F → is a function which will assign a new state to a cell.
where $F: S^{|N|} \to S$. There are 256 $(2^8 = 256)$ kinds of different local rules. Therefore, S. Wolfran numbered for elementary cellular automata by its local rules and studies it deeply. The results shows that even through elementary cellular automata is so simple, their space configuration which it presents is extraordinary complex.

Table. 1 shows a 1D binary state nearest neighbor cellular automaton. The lattice configuration is 7 cells wide and is shown at two successive time steps that is at t=0 and at t=1.

In this example, the local neighborhood configuration of the third cell (1) at time step t = 0 is "1-1-1" (the current values of the second, third, and fourth cells), and from the lookup table we can that this cell will be in state "0.66" at the next time step t = 1. All cells in the lattice are updated in a similar way and simultaneously.

Table 1: Initial contribution and after updates

0	1	1	1	1	1	0
0.33	0.66	1	1	1	0.33	0.66

3. STRUCTURE OF THE NEIGHBORHOODS

[As the speech signal is a two dimensional, here we use two dimensional cellular automata model. In two dimensional cellular automata model, we can use triangular lattice structure or square lattice structure or hexagonal lattice structure, but for speech signals the results using square lattice is far convincing as compare to other lattice structures, so here we have used square lattice for analysis. Each speech sample is of 16 bits. Initially we consider original signal at time t=0.

There are two types of neighborhood structures, first von Neumann neighborhood structure and second moore neighborhood structure. Also there are number of rules of cellular automata. So for denoising the speech signal we have to find the type of neighborhood structure and depending on this we will select the rules of cellular automata.

The cell on right and left, and the cell above and below are called von Neumann neighborhood of this cell. as only the next layer is considered therefore the radius of this definition is 1. As there are four neighboring cells and including itself, the total number of cells become 5 [2] as given in equ (1)

$$N(I, j) = \{(k, l) \in L : |k - i| + |l - j| \ \leq 1\} \ (1)$$

where, k is the number of states for the cell and l is the space of speech signal.

As in von Neumann neighborhood, there are total 5 cells including itself (1 + 4 neighbours), but in moore neighborhood there are total 9 cells including itself (1 +

8 neighbours), The cell on right and left, and the cell above and below plus 4 diagonal cells. This is represented by the equation (2)

$$N(I,j) = \{(k,l) \in L : max|k-i|,|l-j| \leq 1\} \; (2)$$

The state of the target cell at time $t+1$ depends on the states of itself and the cells in the neighbourhood at any time t, this is represented in equation (3)

$$S_{i,j}(t+1) = f(S_{i-1,j-1}(t), S_{i+1,j}(t), S_{i-1,j+1}(t), S_{i,j-1}(t),$$

$$S_{i,j}(t), S_{i,j+1}(t), S_{i+1,j-1}(t), S_{i+1,j}(t), S_{i+1,j+1}(t) \; (3)$$

The concept of structuring element is used to compare the central amplitude of speech signal with the amplitude of neighboring cells, as shown in the equation (4)

$$SE = Strel('square', w) \; (4)$$

This creates a square structuring element with a width of w samples.the integer w must be a positive integer scalar as shown in the table 2:

Table 2: Nine neighbourhood structure

i-1, j-1	i-1, j	i-1, j+1
i, j-1	i, j	i, j+1
i+1, j-1	i+1, j	i+1, j+1

4. MODIFIED UNIVERSAL THRESHOLD

By using multi resolution concept we will remove the babble noise from the noisy signal. For removing the noise from a noisy signal, selection of threshold value is very important, if the threshold value is very low, it will be very inefficient in removing the noise that is at the output we will be getting speech signal with large amount of noise and if we select threshold value somewhat large, than it will remove noise signal efficiently but at the same time some part of speech signal will also be removed. So Donoho and Jonstone [4] developed the method for selection f the threshold value and it is called universal threshold. Different values of universal threshold was proposed[5]

$$\mathrm{hr} = \sigma\mathrm{n}\sqrt{2\mathrm{Log2(N)}} \; (5)$$

Where σn is standard deviation of noise and N means number of samples of noise. The threshold value obtained by using equation (1) is very high.

So a new Universal threshold was proposed in [6] and it is modified by factor ‘k’, this is done in order to get higher quality of output signal:

$$thr = k.\sigma n\sqrt{2Log2(N)} \quad (6)$$

In the course of our research we find that we use two factors k and m, then we get a new threshold value which shows better results in retrieving the original speech signal from the noisy signal. This topic is discussed in next section

5. SOFT AND HARD THRESHOLDING

There are two methods, soft thresholding method and hard thresholding method to estimate the cellular automata coefficients in speech signal threshold denoising.[7] by using hard thresholding method we zeros out the small coefficients, this results in efficient representation of signal. By using soft thresholding we can soften the coefficient value which are greater than threshold value, by decreasing them to the set threshold value. But it is found that by applying thresholding, perfect reconstruction of speech signal (original) is not possible.

Hard thresholding is a process where we set to "0" the elements whose absolute values are less than threshold value (thr). Hence we can say that hard threshold signal are those signal whose values are more than threshold value. that is hard threshold signal (x) is, if $x \geq thr$ and it is 0 if the value of $x < thr$.

In Soft thresholding, it first sets to zero the elements whose absolute value is less than threshold and then it shrinks the non zero coefficients to zero value. if $x \geq thr$, soft threshold signal is $(sign(x).(x - thr))$and if $x < th$, soft threshold signal is 0. Hard thresholding is very simple but but by using soft thresholding, denoising performance is improved. soft thresholding has also nice mathematical properties.

$$T_{HARD}(x) = \begin{cases} x \; |x| \geq thr \\ 0 \; |x| < thr \end{cases} \quad (7)$$

$$T_{Soft}(x) = \begin{cases} sign(x).(x - thr) \; x \geq thr \\ 0 \; -thr \leq x < thr \\ sign(x).(x + thr) \; x < -thr \end{cases} \quad (8)$$

6. EXPERIMENTAL RESULTS

By using MATLAB we implemented babble noise removal algorithm. For speech analysis there is a large collection of functions in cellular automata. At the input we have applied a noisy signal (ie signal corrupted by noise). Here the speech signal is corrupted by a babble noise at 0 dB, 5 dB, 10 dB, 15 dB and 20 dB SNR levels. The noisy signal is taken in the wave format and it is sampled at a frequency of 16 khz.

Babble noise removal algorithm is very useful when the original speech signal is unknown to us. Here we assume that signals having high amplitude Cellular Automata coefficients represents original speech signal and signals having low amplitude Cellular Automata coefficients represents noise signal.

As we are using multi-resolution analysis, the selection of proper level is very important. The approximated signal which is the output of LPF and then decimation are splitted into certain levels. Figure 1 shows the original speech signal, figure 2 shows the corrupted speech signal and figure 3 shows reconstructed speech signal.

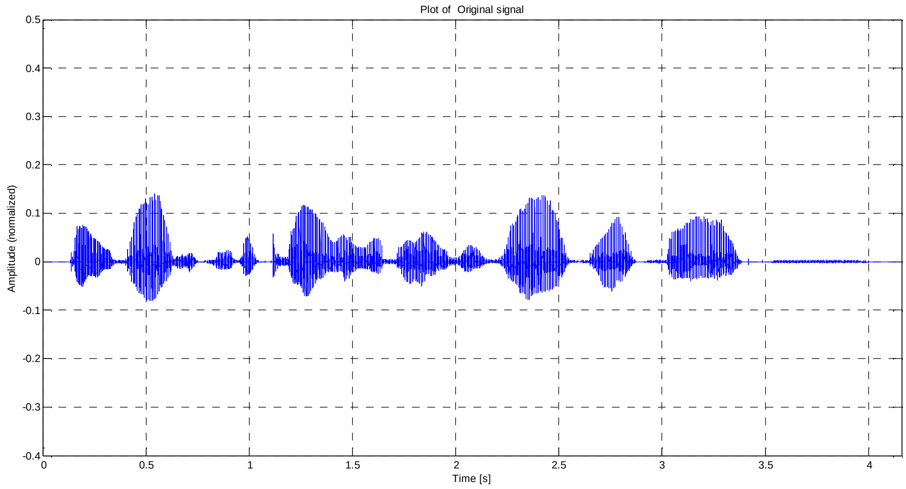

Figure. 1: Original speech signal

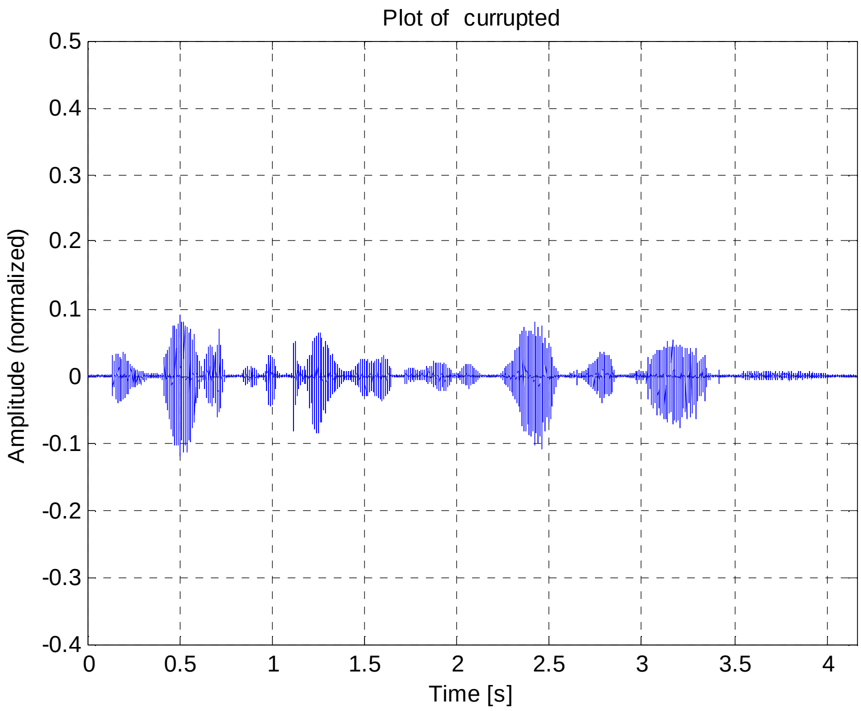

Figure. 2: Corrupted speech signal

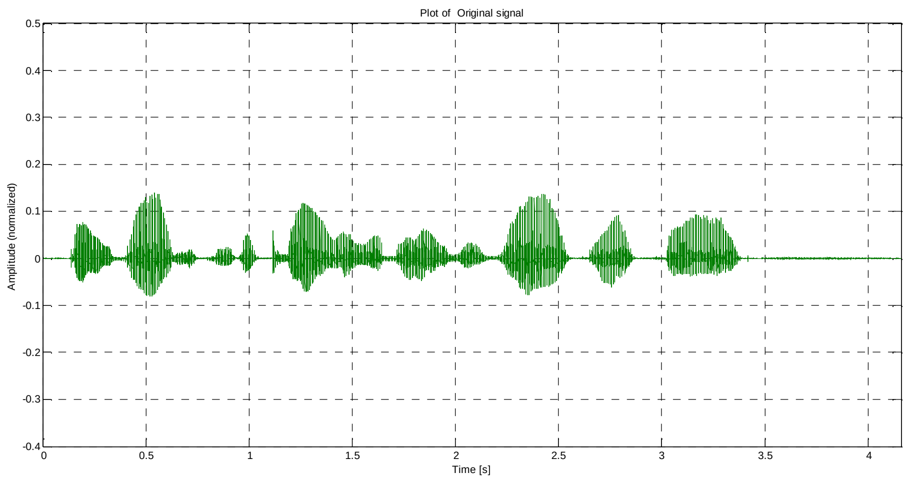

Figure.3: Reconstructed speech signal for babble noise of 15 db using soft threshold

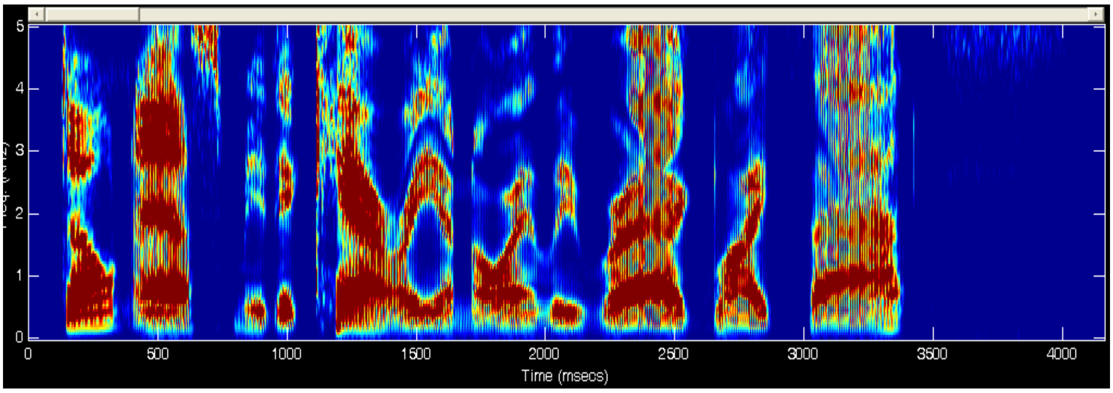

Figure.4 Original speech signal Spectrogram

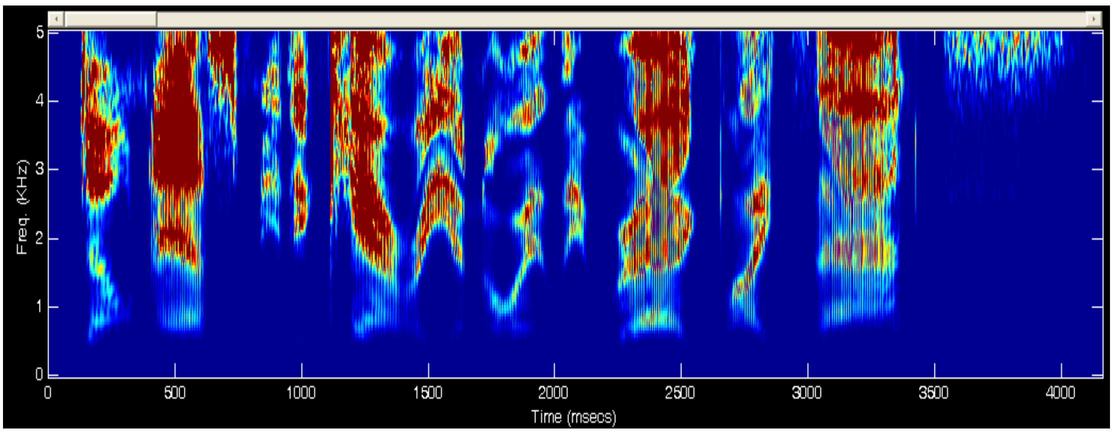

Figure.5 Corrupted speech signal Spectrogram

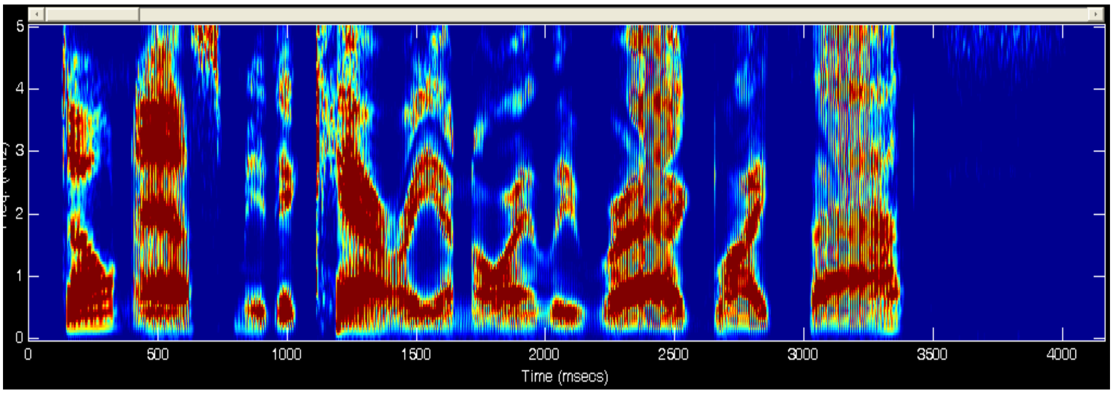

Figure.6 Reconstructed speech signal Spectrogram for babble noise of 15 db using soft threshold

Figure 4 shows the original speech signal spectrogram, figure 5 shows the corrupted speech signal spectrogram and figure 6 shows reconstructed speech signal spectrogram.

For getting the Improved threshold value we replaced threshold thr in equation (6) with

$$thr = m.k.\sigma n\sqrt{2Log2(N)} \quad (9)$$

Where, 0 < k < 1 and 0 < m < 1. ”N” denotes number of noise samples and "σn" is standard deviation of noise.

Here two factors k and m are used. If we fix one factor and change the other factor, than we get a range threshold values which helps to recover original speech signal for low level noise.

By using a range of threshold values to approximation coefficients and detail coefficients. We use soft thresholding and hard thresholding separately. These new coefficients are used to construct the signal. It is found that soft thresholding gives better result than hard thresholding.

In the example 1 as given in table 1, when we fix the value of factor m= 0.9 & changes the value of factor ‘k’ from 0.1 to 0.9, the value of Post signal to noise ratio also increases.

Table 3. Results obtained by using noise removal algorithm for the modified soft thresholding method (example 1)

Std Dev σ	k	m	Input MSE (10-4)	Output MSE (10-4)	SNR Pre (db)	SNR Post (db)
0.006	0.1	0.9	0.3535	0.2038	15	17.40
0.006	0.2	0.9	0.3535	0.1253	15	19.30
0.006	0.3	0.9	0.3535	0.0772	15	21.20
0.006	0.4	0.9	0.3535	0.046	15	23.24
0.006	0.5	0.9	0.3535	0.026	15	25.53
0.006	0.6	0.9	0.3535	0.0141	15	27.99
0.006	0.7	0.9	0.3535	0.0077	15	30.42
0.006	0.8	0.9	0.3535	0.0042	15	32.81
0.006	0.9	0.9	0.3535	0.0024	15	35.16

In example 2 as given in table 4, the value of “k” and “m” are fixed, and analysis is done on the values of post SNR, by taking 0, 5,10,15 db of noisy signal.

For estimating the quality of signal we use Mean square error (MSE). Table 4. gives relation between input MSE and Output MSE at different signal to noise ratio level.

Table 4. Results obtained by using noise removal algorithm for the modified soft thresholding method (example 2)

Std Dev σ	k	m	Input MSE (10-4)	Output MSE (10-4)	SNR Pre (db)	SNR Post (db)
0.036	0.8	0.9	12.6942	0.1071	0	13.90
0.018	0.8	0.9	3.1583	0.1013	5	16.99
0.010	0.8	0.9	1.0822	0.0290	10	23.46
0.006	0.8	0.9	0.3535	0.0042	15	32.81

The value of input MSE is given by the equation 10 as given below

$$\frac{1}{N}\sum_{i}(X_i - Y_i)^2 \quad (10)$$

Here X_i represents original signal and Y_i represents the noisy signal.
The value of output MSE is given by the equation 11 as given below

$$\frac{1}{N}\sum_{i}\left(X_i - \overline{X_i}\right)^2 \quad (11)$$

Where, $\overline{X_i}$ represents a reconstructed signal.

By checking SNR we can find the amount of noise that has been removed from the signal. Here we have to compare Post SNR and Pre SNR, if value Post SNR is greater then the value of Pre SNR, then we can say that the denoising of signal took place.

Signal to Noise Ratio is defined as:

$$SNR_{db} = 10Log_{10}\left(\frac{P_{signal,db}}{P_{noise,db}}\right) = P_{signal,db} - P_{noise,db} \quad (12)$$

In example 3 as shown in table 5, by using hard thresolding method it is found that results are not good as compare to analysis which is performed by making use of soft thresholding method.

Table 5. Results obtained by using noise removal algorithm for the modified hard thresholding method (example 3)

Std Dev σ	k	m	Input MSE (10-4)	Output MSE (10-4)	SNR Pre (db)	SNR Post (db)
0.036	0.8	0.9	12.6942	0.1071	0	4.93
0.018	0.8	0.9	3.1583	0.1013	5	10.61
0.010	0.8	0.9	1.0822	0.0290	10	15.91
0.006	0.8	0.9	0.3535	0.0042	15	21.79

7. CONCLUSION

Here cellular automata based filter is used for denoising speech signal corrupted with babble noise. Removing noise from sepech signal is performed in CA filter by thresholding coefficients. It is found that by using MUT, denoising is better at low level noise. After performing number of different analysis it is found that the results of soft thresholding is better than hard thresholding. If Higher threshold is used than the noise removal is very good, but some portion of original speech signal is also removed along with the noise. It is found that it is impossible to filter noise signal without affecting the original speech signal. We have done analysis of denoising the speech signal by two ways one by using signal to noise ratio (SNR) and second by using mean square error (MSE) analysis.

REFERENCES

[1] M. A. Peer, N. A. Shah and R. A. Khan, 2005, " image processing systems based on cellular automata", proceeding of the national conference on recent advances and future in IT, (RAFIT).

[2] L. Songtao, C. Hongguan, Y. Shaoging, 2008,"An effective filtering algorithm for salt-peper noises based on cellular automata", IEEE congress on image and signal processing.

[3] MATKO SARIC, LUKI BILICIC, HRVOJE DUJMIC,2005, "White Noise Reduction of Audio Signal using Wavelets Transform with Modified Universal Threshold", University of Split, R. Boskovica b. b HR 21000 Split.

[4] D.L. Donoho, Stanford University, 1992, "De-noising by Soft Thresholding", IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 41, NO. 3.

[5] Alexandru Isar, Dorina Isar, 2003, "Adaptive denoising of low SNR signals", Third International Conference on WAA, Chongqing, P. R. China, 29-31, p.p. 821-826.

[6] Lim, J. S., 1990, Two-Dimensional Signal and Image Processing, Prentice-Hall.

[7] S. Wolfram, 1985, "Cryptography with Cellular Automata", Proceedings of Crypto'85, 429–432.