International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 15 (2018) pp. 12264-12271

Investigation of Optimal Denoising Filter for MRI Images

Ch. Rajasekhara Rao, M N V S S Kumar, G Sasi Bhushana Rao

Department of Electronics and Communication Engineering, AITAM, Tekkali, India.
Department of Electronics and Communication Engineering, Andhra University, Visakhapatnam, India.

Abstract

Any real world equipment is affected by a certain degree of noise, whether it is thermal, electrical or otherwise. This noise will corrupt the true measurement of the signal, such that any resulting data is a combination of signal and noise. In medical image processing, medical images are corrupted by different type of noises. In case of medical imaging, the environment is very complex as the conditions such as operator performance, equipment and the environment, which vary spatially as well as temporally in turn leads to errors. In this paper, several spatial (Mean filter, Gaussian filter and Median filter) and frequency domain filters (Lowpass, Highpass and Bandpass filter) for MRI image denoising are discussed along with the results. A comparison for performance evaluation is made among the filters based on the error metrics, PSNR and MSE.

Keywords: MRI, Tumour, Filters, Denoising

INTRODUCTION

Most research in developed countries has exposed that the death rate of people affected by brain tumor has increased over the past three decades. Today, one of the major causes for the increase in fatality among children and adults is brain tumor. Brain tumor is a pathology appearing in the intracranial anatomy due to abnormal and unstructured augmentation of cells. It is a very aggressive and life-threatening condition, which must be promptly diagnosed and cured to prevent mortality [1]. Brain tumors are of different sizes, locations and positions. They also have overlapping intensities with normal tissues. Tumor can be benign or malignant can occur in different parts of the brain and may or may not be primary tumors. So it is very essential to identify tumors before reaching uncontrollable stage. The mechanism used to identify the tumors is MRI. MRI is an advanced medical imaging technique providing rich information about the human soft-tissue anatomy. It is mostly used in radiology in order to visualize the structure and function of the human body. It produces the very detailed images of the body in any direction. Particularly, MRI is useful in neurological (brain), musculoskeletal, and oncological (cancer) imaging because it offers much greater contrast between the diverse soft tissues of the body. These images contains lot of noise along with information. In medical image processing, the denoising of signal or image is very important to provide accurate information for perfect diagnosis. Therefore, it is essential to remove the noise from medical images. In this paper, several spatial and frequency domain filters for MRI image denoising are discussed along with the results. A comparison for performance evaluation is made among the filters based on the error metrics, PSNR and MSE.

ANALYSIS OF SPATIAL AND FREQUENCY DOMAIN FILTERS FOR DENOISING OF MRI IMAGES

In image processing, there are two basic types of filtering methods: spatial domain methods and frequency domain methods. The term spatial domain refers to the image plane itself and methods in this category are based on direct manipulation of pixels in an image. Spatial domain methods are the methods that directly modify pixel values possibly using intensity information from the neighborhood of the pixel. Frequency domain methods are the methods that modify the Fourier Transform (FT) of the image. First, compute the FT of the image. Then alter the FT of the image by multiplying a filter transfer function. Finally, use inverse transform to get the modified image [2]. The key is the filter transfer function.

Spatial domain filters

Spatial domain filters are the filters in which the operations involve directly on the image pixels itself i.e. pixels under the mask perform convolution or any non linear operation with the image pixels on which the filter is placed. The various spatial domain filters are Average filter, Gaussian filter, Laplacian of Gaussian filter, Maximum filter, Minimum filter, etc.

Mean filter

The mean or average filter smoothes image data, thus eliminating noise. This filter performs spatial filtering on each individual pixel in an image using the gray level values in a square or rectangular window surrounding each pixel. One important linear filtering is to use a 3×3 mask and take the average of all nine values within the mask. This value becomes the grey value of the corresponding pixel in the new image.

There is an obvious problem in applying a filter i.e when applying a mask, it partly falls outside the image so there will be a lack of grey values to use in the filter function at the edge of the image. There are a number of different approaches to deal with this problem

i) **Ignore the edges:** The mask is only applied to those pixels in the image for which the mask will lie fully within the image. This means all pixels except for the edges are included and results in an output image which is smaller than the original. If the mask is very large, a significant amount of information may be lost by this method.

ii) **"Pad" with zeros:** All necessary values outside the image are assumed to be zeros. This gives us all values to work with and will return an output image of the same size as the original, but may have the effect of introducing unwanted artifacts (for example, edges) around the image.

iii) **Replicate**: The size of the image is extended by replicating the values in its outer border.

iv) **Symmetric:** The size of the image is extended by mirror reflecting it across its border.

v) **Circular:** The size of the image is extended by treating the image as one period a 2-D periodic function.

Gaussian filter

Gaussian filter removes the high frequency components in an image. So it could be used as low pass filtering. A Gaussian filter removes the Gaussian noise (noise that has a frequency distribution which follows the Gaussian curve) in an image [3]. It is a linear spatial filter. Gaussian filtering is performed by convolving the Gaussian function with the image. A Gaussian filter is a filter whose impulse response is a Gaussian function. It can also be used in frequency domain. It may be considered to be the smoothest of all the filters. The 1-D Gaussian filter has an impulse response given by

$$G_{1D}(x_1) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x_1^2}{2\sigma^2}}$$

For the 2-D Gaussian filter, it is the product of two such Gaussians and is given by

$$G_{2D}(x_1, y_1) = \frac{1}{2\pi\sigma^2} e^{-\frac{x_1^2 + y_1^2}{2\sigma^2}}$$

where, x_1 is the distance from the origin in the horizontal axis,

y_1 is the distance from the origin in the vertical axis and

σ is the standard deviation of the Gaussian distribution.

When applied in two dimensions, this formula produces a surface whose contours are concentric circles with a Gaussian distribution from the center point. Values from this distribution are used to build a convolution matrix which is applied to the original image. Each pixel's new value is set to the weighted average of that pixel's neighbourhood. The original pixel's value receives the heaviest weight (having the highest Gaussian value) and the neighbouring pixels receive smaller weights as their distance to the original pixel increases. This results in a blur that preserves boundaries and edges better than the others.

Properties of the Gaussian filter

i) An important property of the Gaussian function is that the Fourier of the Gaussian is itself a Gaussian

ii) The width of the Gaussian increases as standard deviation (σ) increases.

iii) Gaussian filter is separable

$$G_{2D}(x_1, y_1) = \left(\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x_1^2}{2\sigma^2}} \right) \times \left(\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{y_1^2}{2\sigma^2}} \right)$$
$$= G_{1D}(x_1) \times G_{1D}(y_1)$$

Median filter

The median filter consider an area of an image (3x3, 5x5, 7x7, etc.), arranges all the pixel values in that area in an either ascending or descending manner and replaces the center pixel with the median value of all the neighbouring pixels in that area including the center pixel. The median filter does not require convolution [4]. It does, however, require sorting the values in the image area to find the median value. This can be done by repeating the above process for each pixel in the image. It is a non linear spatial filtering technique, particularly useful to reduce speckle noise/ salt and pepper noise

Frequency domain filtering

The idea in frequency domain filtering is to select a filter transfer function that modifies F(u,v), the FT of the image in a specified manner by multiplying the centered F(u,v) with the filter transfer function.

The procedure for filtering in the frequency domain is summarized below

(i) Multiply the input image by $(-1)^{x+y}$ to center the transform

(ii) Compute the FT, F(u,v) of the resulting image

(iii) Multiply F(u,v) by a filter transfer function, H(u,v)

(iv) Compute the inverse FT of the result in step (iii)

(v) Obtain the real part (take the magnitude)

(vi) Multiply the result in (v) by $(-1)^{x+y}$

The well known filters such as Lowpass filter, Highpass filter and Bandpass filters are implemented in this chapter. These filters are circularly symmetric and are specified as various functions of distance from the origin of the transform.

Lowpass filter (LPF)

The ideal lowpass filter suppresses all the frequencies higher than the cutoff frequency, r_0 and leaves smaller frequencies unchanged [8]. The transfer function of an ideal LPF (ILPF) is given as

$$H(u,v) = \begin{cases} 1, & if \quad D(u,v) \le r_0 \\ 0, & if \quad D(u,v) > r_0 \end{cases}$$

where, r_0 is called the cutoff frequency (non negative quantity), and *D(u,v)* is the distance from point *(u,v)* to the center of the filter. If the image is of size M×N, then

$$D(u,v) = \sqrt{\left(u - \frac{M}{2}\right)^2 + \left(v - \frac{N}{2}\right)^2}$$

The low pass filter considered is radially symmetric about the origin. The locus of points for which $D(u,v)=r_0$ is a circle. Keeping in mind that the filter transfer function multiplies the FT of an image (F(u,v)), it can be seen that an ideal filter "cuts off" (multiplies by 0) all components of F(u,v) outside the circle and leaves unchanged (multiplies by 1) all components on, or inside the circle. Although the filter is not realizable in analog form using electronic components, it certainly can be simulated in a computer using the preceding transfer function [5]. The lowpass filter is displayed as an image in the results and discussion section. The cutoff frequency, r_0 of ILPF determines the amount of frequency components passed by the filter. Smaller the value of r_0, more the number of image components eliminated by the filter. The value of r_0 is chosen such that most components of interest are passed through, while most components not of interest are eliminated.

Highpass filter (HPF)

A highpass filter (HPF) function can be obtained by inverting the corresponding lowpass filter, i.e. an ideal highpass filter (IHPF) blocks all frequencies smaller than r_0 and leaves the others unchanged. Using the transfer function of a lowpass filter, the transfer function of a HPF can be derived as :

$$H_{hp}(u,v) = 1 - H_{lp}(u,v)$$

where $H_{hp}(u,v)$ and $H_{lp}(u,v)$ are the transfer functions of highpass and lowpass filter respectively. The transfer function of an IHPF with the cutoff frequency, r_0

$$H(u,v) = \begin{cases} 0, & if \quad D(u,v) \le r_0 \\ 1, & if \quad D(u,v) > r_0 \end{cases}$$

D(u,v) is the same as shown above

A HPF yields edge enhancement or edge detection in the spatial domain, because edges contain many high frequencies. Areas of rather constant gray level consist of mainly low frequencies and are therefore suppressed.

Bandpass filter (BPF)

Bandpass filters are a combination of both lowpass and highpass filters. They attenuate all frequencies smaller than a frequency, r_2 and higher than a frequency, r_1 while the frequencies between the two cut-offs remain in the resulting output image [6]. We obtain the filter function of a bandpass by multiplying the filter functions of a lowpass and of a highpass in the frequency domain, where the cut-off frequency of the lowpass is higher than that of the highpass [9]. The transfer function of a BPF is given as

$$H(u,v)_{bp} = \begin{cases} 1, & if \quad r_1 \ge D(u,v) \le r_2 \\ 0, & elsewhere \end{cases}$$

A bandpass attenuates very low and very high frequencies but retains a middle range band of frequencies. Bandpass filtering can be used to enhance edges (suppressing low frequencies) while reducing the noise at the same time (attenuating high frequencies).

RESULTS AND DISCUSSION

Several MRI images are considered for filtering analysis. The spatial and frequency filters which are described in this paper are applied on the images and results describing the performance of each filter with respect to other are given. Each spatial filter is applied to the sample input image by varying window size from 3×3, 9×9, 15×15 to 25×25. By varying the window size, the signal to noise ratio, execution time of filtering process and the response pixel values are computed from the filtered images and are tabulated Table 1 and graphs are shown in Fig 2, Fig 3, Fig 4. Similarly, the frequency domain filters have been implemented with respect to the MRI images. Each frequency domain filter is shown with its magnitude and phase information which is obtained after performing FT on the filter transfer function [7]. The results with relevant plots have been discussed. The Peak Signal to Noise Ratio (PSNR) is an important parameter to measure the quality of the filtered images [10]. The PSNR is the ratio between the image's maximum intensity and the mean square error of that image. Fig. 1 shows the input original MRI image and the corresponding average filtered image.

Table 1. Filtering of MRI images using an average filter

Window size	Pixel position	Pixel value	Response pixel value	PSNR (dB)	Execution time (sec)
3×3	(310,119)	105	111	29.1647	0.215419
5×5	(310,119)	105	118	27.7981	0.220677
7×7	(310,119)	105	121	27.1430	0.239341
9×9	(310,119)	105	121	26.7546	0.247215
13×13	(310,119)	105	112	26.2747	0.270145
15×15	(310,119)	105	108	26.1143	0.284329
17×17	(310,119)	105	105	25.9824	0.322480
21×21	(310,119)	105	100	25.7788	0.342904
25×25	(310,119)	105	94	25.6116	0.361039
29×29	(310,119)	105	87	25.4702	0.366175

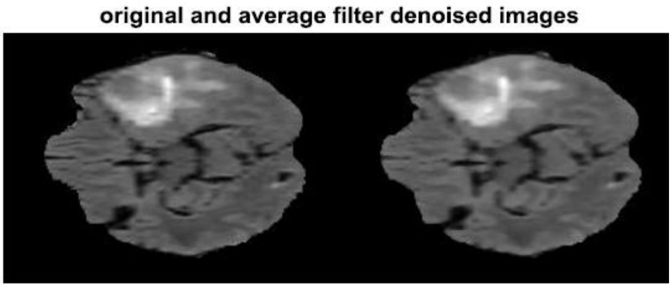

Figure 1. Original image and average filtered image

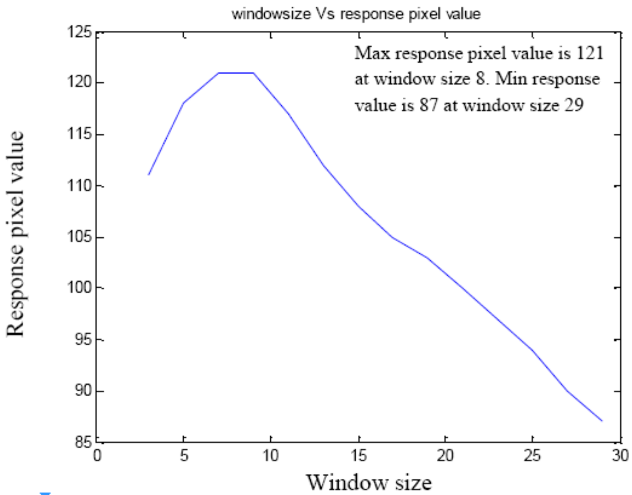

Figure 2. Graph between window size and response pixel value for average filter

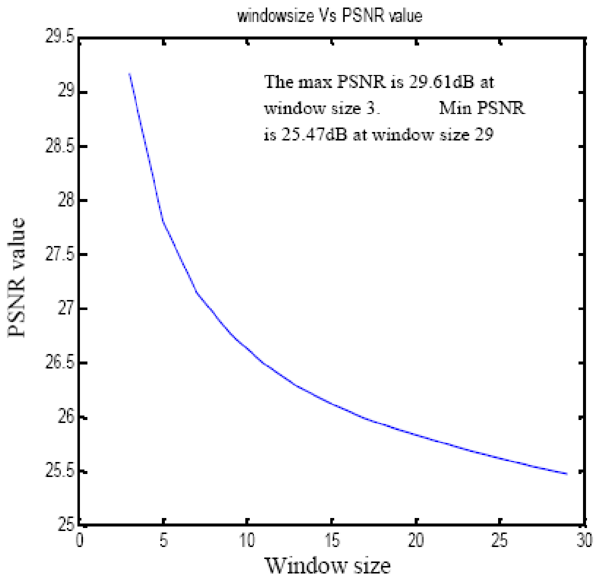

Figure 3. Graph between window size and PSNR value for average filter

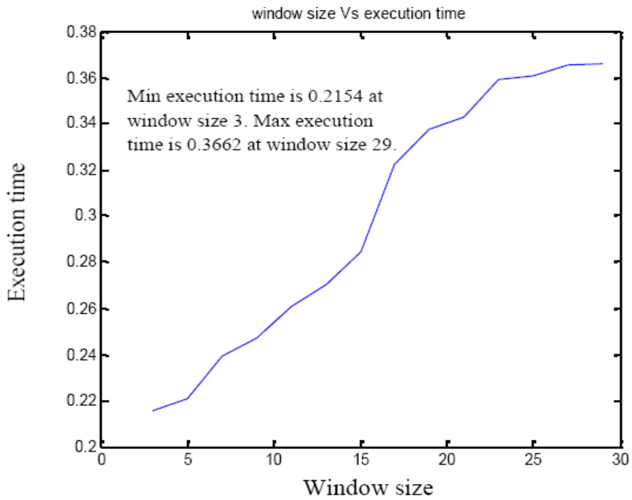

Figure 4. Graph between window size and execution time For average filter

Fig. 5 shows the original input MRI image and the corresponding Gaussian filtered image.

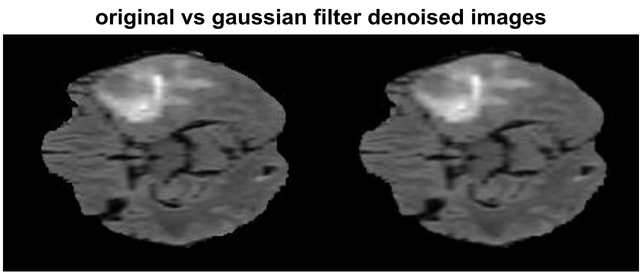

Figure 5. Original image and gaussian filtered image

Table 2 tabulates the values of several parameters such as response pixel value, PSNR and execution time for the window sizes varied from 3×3 to 29×29. Fig. 6 shows the graph drawn between window size and response pixel value. From the graph, it is noticed that as the window size increases, the response pixel value increases till the window size is 5, after which it remains constant.

Table 2. Filtering of MRI images using Gaussian filter

Window size	Pixel position	Pixel value	Response pixel value	PSNR (dB)	Execution time (s)
3×3	(310,119)	105	110	30.2634	0.220669
5×5	(310,119)	105	112	29.6915	0.276770
7×7	(310,119)	105	112	29.6486	0.235164
9×9	(310,119)	105	112	29.6473	0.239695
13×13	(310,119)	105	112	29.6473	0.260010
15×15	(310,119)	105	112	29.6473	0.268932
17×17	(310,119)	105	112	29.6473	0.321998
21×21	(310,119)	105	112	29.6473	0.327598
25×25	(310,119)	105	112	29.6473	0.337226
29×29	(310,119)	105	112	29.6473	0.342730

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 15 (2018) pp. 12264-12271

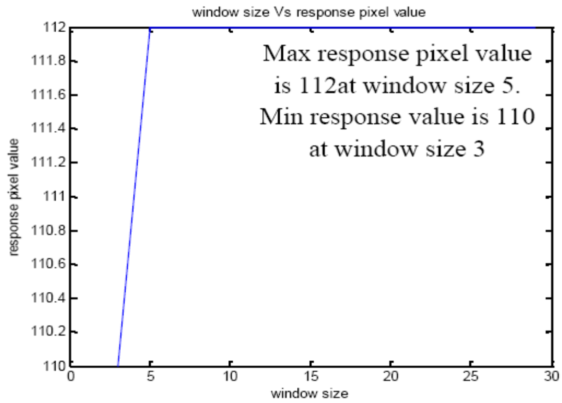

Figure 6. Graph between window size and response pixel value for Gaussian filter

In Fig. 6, the maximum response pixel value is 121 at window size 5. And the minimum response pixel value is 110 at window size 3. Fig. 7 and Fig. 8 describe the variations in PSNR and execution time with respect to the window size and it is observed that PSNR value decreases and execution time increases as window size increases.

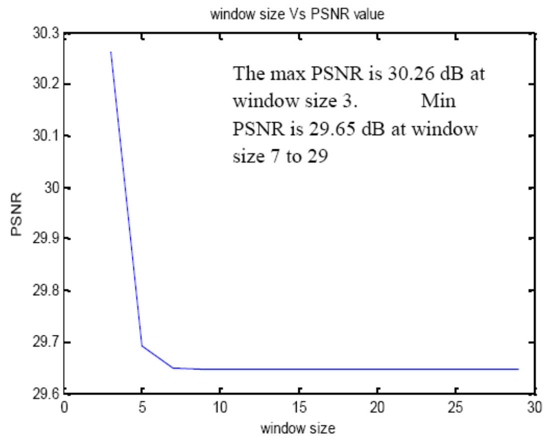

Figure 7. Graph between window size and PSNR value for Gaussian filter

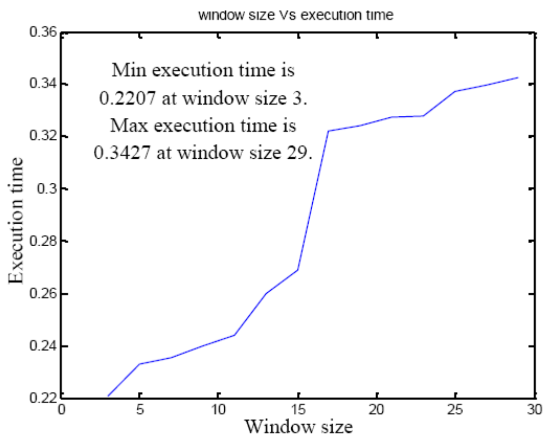

Figure 8. Graph between window size and execution time for Gaussian filter

In Fig. 7, the PSNR value decreases as the window size increases because of decrease in the mean square error between the original and the resultant image. The maximum PSNR is 30.26dB at window size 3. And minimum PSNR is at 29.65dB at window size 7 to 29. So the optimum window size is 3×3. In Fig. 8, the execution time increases with increase in window size because time taken for the convolution process increases. The time taken for the execution of the program at window size 3 is 0.2207 sec and at window size 29 is 0.3427 sec.

The Median filter is applied to the sample input image shown in Fig. 9 with varying window size and the result is also shown. Variation of PSNR, execution time, response pixel value with respect to the window size are tabulated and are shown in Table 3.

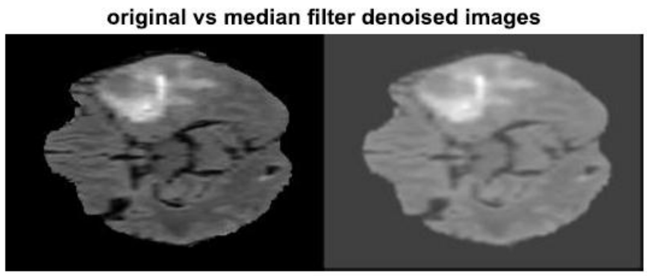

Figure 9. Original image and median filtered image

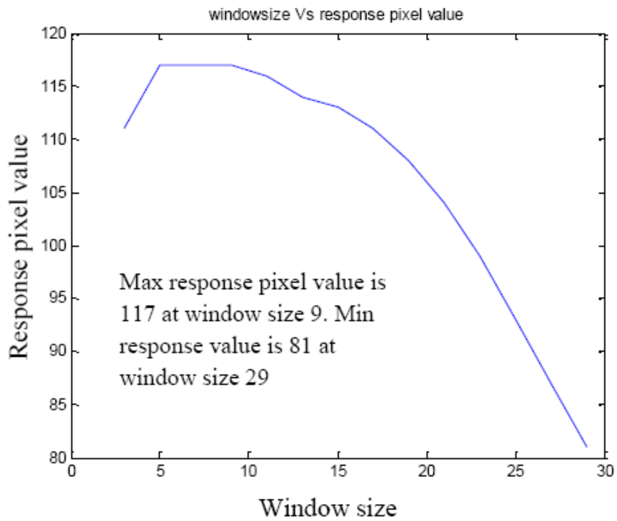

Figure 10. Graph between window size and response pixel value for Median filter

Fig. 10 shows the graph drawn between window size Vs response pixel value. The variation in the pixel value depends on the neighbouring pixel values. From Table 3, the maximum response pixel value is 117 at window size 9. And the minimum response pixel value is 81 at the window size 29. In Fig. 11, the PSNR value decreases as the window size increases because of decrease in the mean square error between the original and the resultant image. The maximum PSNR is 27.82dB at window size 3. And minimum PSNR is at 25.2dB at window size 29. So the optimum window size is 3×3.

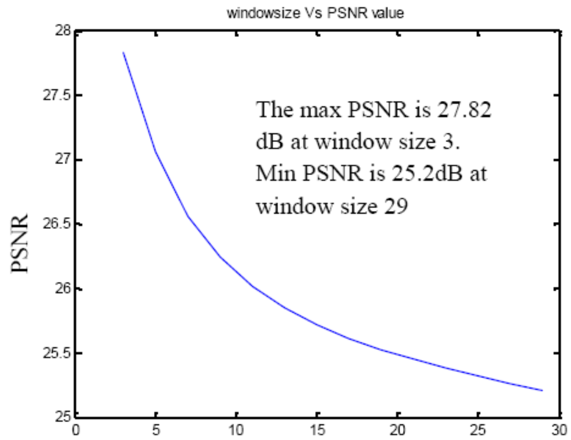

Figure 11. Graph between window size and PSNR value for Median filter

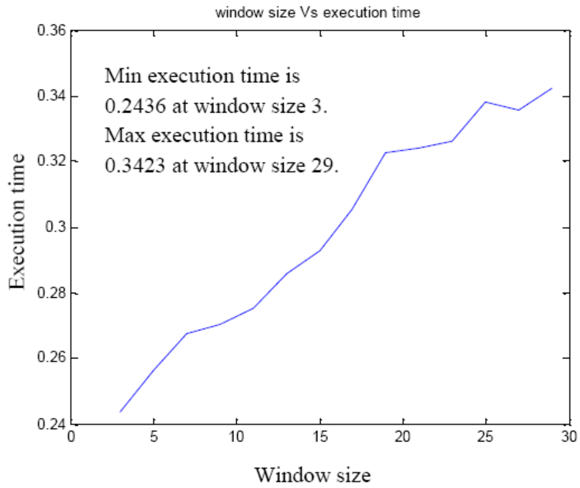

Figure 12. Graph between window size and execution time for Median filter

Table 3. Filtering of MRI image using median filtering for various window sizes

Window size	Pixel position	Pixel value	Response pixel value	PSNR (dB)	Execution time (s)
3×3	(310,119)	105	111	27.8249	0.215419
5×5	(310,119)	105	117	27.0544	0.220677
7×7	(310,119)	105	117	26.5591	0.239341
9×9	(310,119)	105	117	26.2380	0.247215
13×13	(310,119)	105	114	25.8463	0.270145
15×15	(310,119)	105	113	25.7170	0.284329
17×17	(310,119)	105	111	25.6121	0.322480
21×21	(310,119)	105	104	25.4517	0.342904
25×25	(310,119)	105	93	25.3228	0.361039
29×29	(310,119)	105	81	25.2029	0.366175
z					

In Fig. 12, the execution time increases with increase in window size as the number of neighbouring pixels increases. The time taken for the execution of the program at window size 3 is 0.2436 sec and at window size 29 is 0.3423 sec. As the size of the window increases, the time to complete the neighborhood processing increases.

In a similar way, the frequency filters are designed and implemented for MRI images.. The log magnitude of each filter along with its phase information is shown in the form of images. Fig. 13(a) and Fig. 14(b) represent the corresponding magnitude and phase information of the lowpass filter.

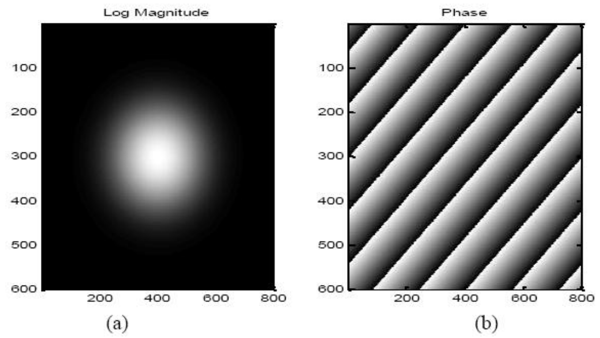

Figure 13. Lowpass filter

The lowpass filtered image reconstructed from both magnitude and phase information of the image is shown in Fig. 14.

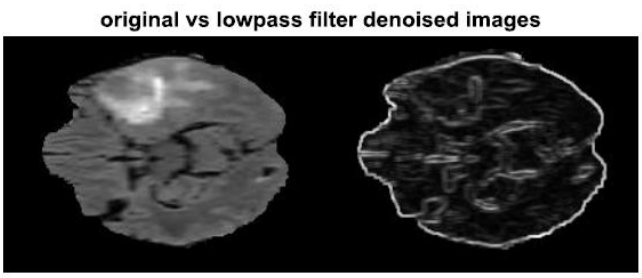

Figure 14. Original image and lowpass filtered image

Similarly, Fig. 15(a) and Fig. 15(b) represent the corresponding magnitude and phase information of the HPF. The highpass filtered image reconstructed from both magnitude and phase information of the image is shown in Fig. 16.

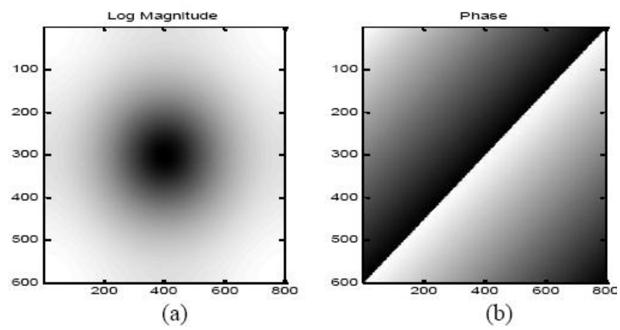

Figure 15. Highpass filter

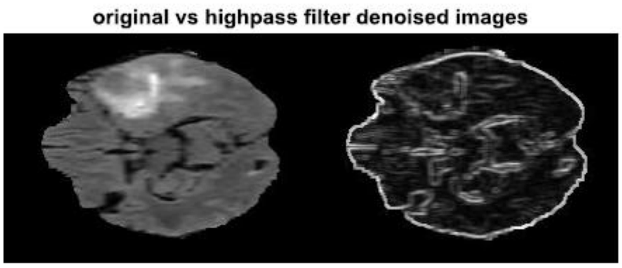

Figure 16. Original image and highpass filtered image

In a similar way, Fig. 17(a) and Fig. 17(b) represent the corresponding magnitude and phase information of the bandpass filter. The bandpass filtered image reconstructed from both magnitude and phase information of the image is shown in Fig. 18.

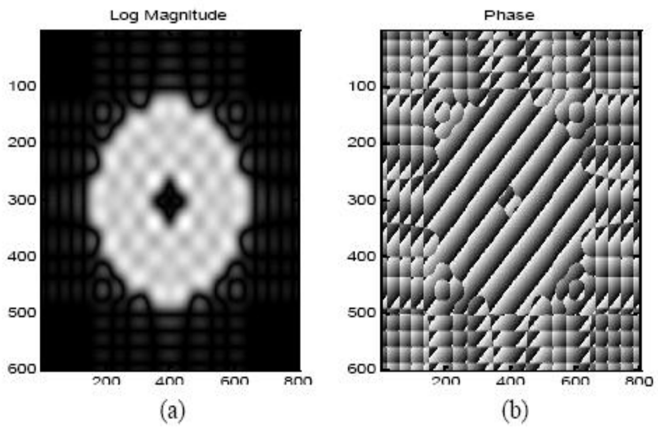

Figure 17. Band pass filter

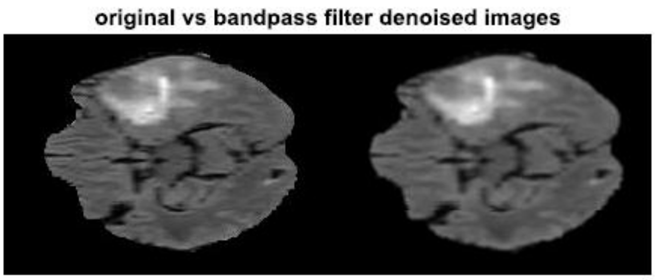

Figure 18. Original image and bandpass filtered image

Table 4 gives a comparison of all the filters with respect to the error metrics: PSNR values, MSE and execution time. From the table, it is quite evident to choose the lowpass filter as an optimal filter for removing the noise present in the MRI images both in respect of the PSNR and the MSE.

Table 4. Comparison of error metrics of all the filters

S.No	Name of the filter	PSNR value (dB)	MSE	Execution time (sec)
1	Average filter	23.16	17.7165	0.5141
2	Gaussian filter	23.18	17.6892	0.5578
3	Median filter	40.86	2.3102	0.3526
4	Lowpass filter	78.08	0.0319	1.1193
5	Highpass filter	69.11	0.0899	1.1903
6	Bandpass filter	73.57	0.0561	1.2313

CONCLUSION

The concept of linear and nonlinear spatial filtering, frequency domain filtering are explained. Several frequency domain filters such as lowpass, highpass and bandpass filters and spatial domain filters such as mean filter, Gaussian filter and median filter are implemented. The error metrics (PSNR, MSE and execution time) for various filtering techniques discussed are computed for analyzing their performances. To find the optimum window size, image processing filtering analysis is carried out for different configurations of window sizes (3×3, 5×5, 7×7, 9×9 etc.). For each configuration PSNR, execution time of filtering process and the response pixel values are computed. With increase in the window size, there is an increase in the execution time for filtering the image and decrease in the PSNR value. From the analysis it is found that the optimum window size is 3×3 for all the spatial domain filters. The PSNR values computed for the Average, Median, Gaussian filtered images are 23.16 dB, 40.86 dB and 23.18 dB respectively. The MSE values computed for the Average, Median, and Gaussian filtered images are 17.7165, 2.3102, and 17.6892 respectively. Similar analysis is undertaken for the frequency domain filters also. In all frequency filters, the optimum cutoff frequency is found as 5. The PSNR values computed for the LPF, HPF and BPF are 78.08 dB, 69.11 dB and 73.57 dB respectively. Similarly, the MSE values obtained for LPF, HPF and BPF are 0.0319, 0.0899 and 0.0561 respectively. From the spatial and frequency domain filters analysis it is concluded that the LPF is suggested for the medical image processing as this lowpass filter gives a better PSNR and MSE values of 78.08 dB and 0.0319 respectively So, to remove the noise from the MRI images, the lowpass filter which is better than the remaining filters giving high PSNR and low MSE can be suggested.

RFERENCES

[1]. Muhammad Nasir, Asim Baig and Aasia Khanum "Brain Tumor Classification in MRI Scans Using Sparse Representation", Springer International Publishing Switzerland 2014, A. Elmoataz et al. (Eds.): ICISP 2014, LNCS 8509, pp. 629–637, 2014.

[2]. Sayali SavajiPP, Parul AroraP "Denoising of MRI Images using Thresholding Techniques through Wavelet" International Journal of Science, Engineering and Technology Research (IJSETR) Volume 1, Issue 3, September 2012.

[3]. Shashikant Agrawal, Rajkumar Sahu "Wavelet Based MRI Image Denoising Using Thresholding Techniques" International Journal of Science, Engineering and Technology Research (IJSETR) Volume 1, Issue 3, September 2012.

[4]. Aleksandra Pi_zurica, Alle Meije Wink, Ewout Vansteenkiste,Wilfried Philips and Jos B.T.M. Roerdink "A review of wavelet denoising in MRI and ultrasound brain imaging" CURRENT MEDICAL IMAGING REVIEWS 2005.

[5]. Nelly Gordillo , Eduard Montseny , Pilar Sobrevilla "State of the art survey on MRI brain tumor segmentation" Magnetic Resonance Imaging (2013).

[6]. Saurabh Shah and N C Chauhan "An Automated Approach for Segmentation of Brain MR Images using Gaussian Mixture Model based Hidden Markov Random Field with Expectation Maximization" VOLUME 2 ISSUE 4 17th August 2015.

[7]. H. Tiwari, V. Bhateja, A. Srivastava, "Estimation based non-local approach for pre-processing of MRI", 2nd international conference on computing for sustainable global Development (lNDIACom), 2015.

[8]. Stanislav Pyatykh, Lei Zheng, Jurgen Hesser, "Image noise level estimation by principal component analysis", IEEE Transactions on image processing, vol. 22, no. 2, Feb 2013.

[9]. M. Raghav, Sahil Raheja, "Image denoising techniques: Literature review", International Journal of engineering and computer science, vol. 3, no. 5, pp. 5637-5641, May 2014, ISSN 2319-7242.

[10]. M. J. Krishanaveni, V. Guo, "A survey on the magnetic resonance image denoising methods", Biomedical Signal Processing and Control, vol. 9, pp. 56-69, 2014.