Image Upscaling and Denoising with Gaussian filter in Coloured Images -A Performance Analysis

Amanjot Singh[1,2] and Jagroop Singh[3]
[1]I.K.G. P.T.U. – 144603, Punjab, India;
[2]School of Electronics and Electrical Engineering,
Lovely Professional University, Phagwara - 144411, Punjab, India;
er.ajotsingh@gmail.com
[3]Department of Electronics and Communication Engineering,
DAVIET, Jalandhar – 144008, Punjab, India;
roopasidhu@yahoo.com

***Abstract*— The paper has been focused on the comparison of the different upscaling methods for images. Along with upscaling Gaussian filter based denoising has also been considered for images contaminated with various noises. Gaussian filter is a usual filter used in Digital image processing [1][4]. Its functioning can be controlled by its kernel size. Interpolation is one of the basic methods used in image up scaling and Super-resolution. The response varies with interpolation method variation and that has been presented in the analysis. In this paper, image quality matrices like PSNR and MSE have been compared for the various noises in images. The behavior of different methods for image upscaling along with image denoising has also been included in the analysis. Moreover, Gaussian filter based de-noising method has been compared along with different types of interpolations. The paper gives a review of interpolation methods and their behavioral analysis under the influence of different noises.**

Keywords: Interpolation, Noise, PSNR, MSE.

I. INTRODUCTION

In today's time there are many challenges which are present in communication systems and one of them is the handling of the multimedia data. In the presence of limited resources like channel bandwidth usually images are downscaled and may need upscaling at the other end [14]. However, In the process of acquisition or transmission of images, these images often get corrupted by noises. Many acquisition techniques also introduce the various sorts of noises and artifacts. These noisy parts introduce the undesired visual effects [1]. Denoising of images is an essential part of digital image processing which focus on removal of noise while retaining important features [11]. Therefore, needless to say that both upscaling as well as denoising play a significant role in digital image processing [14].

In literature there are various methods of denoising where traditional methods focus on linear techniques and recent on non-linear techniques [1,12,13]. Gaussian filtering is a usual method of noise removal. It has already been used for many noise removal techniques. It could be used as filtering technique along with upscaling methods. Image upscaling can be achieved by the interpolation of pixels. There are various types of mathematical functions present in literature which are used for the interpolation of images, those will also be discussed in paper. In this paper, Section 2 is dealing with details of Gaussian Denoising, Section 3 with Interpolation based upscaling methods and Section 4 is providing the details of noise model, section 5 is providing the MATLAB based analysis and in the end Section conclusion has been presented.

II. GAUSSIAN DE-NOISING

Noise removal is the essential part of post-processing of the images. In the literature one of the important method is present known as Gaussian method of filtering. It is based on the mathematical function of Gaussian distribution. It could be having one dimensional or two dimensional structure based on analytical functions [5] [6]. In the image processing two dimensional Gaussian function is required which is given by:

$$G(a,b) = \frac{1}{2\pi\sigma^2} e^{-\frac{a^2+b^2}{2\sigma^2}}$$

The functioning of Gaussian filter is controlled by its point spread function. However, in image processing it further depends upon the kernel size. Images are stored in the form of pixels and they are having the discrete values. Image filtering is also implemented in discrete form [7]. Many times Gaussian filter could also blur the image it happens if proper value of kernel is not selected. Basically Gaussian filter provides the weighted average of its nearest pixels with more weightage towards the central pixels [15] [10].

III. IMAGE INTERPOLATION and UP-SCALING

Image interpolation is basically a method of artificially increasing the number of pixels in an area inside an image. It is one of the methods which are usually present in the image processing applications like facial reconstructions, image super resolution and multiple image description. Image interpolation is one of the traditional methods used in Super resolution. In order to execute the interpolation there are various techniques present in literature. However, there are three basic categories i.e. nearest neighbor, bilinear and bi-cubic interpolation which are used in image interpolation. These techniques are very useful to increase the quality of the image visualization. All these techniques have their analytically defined functions which are discussed in further part.

A. Nearest Neighbor Interpolation

Nearest neighbor interpolation is a replicating type of interpolation. It is very basic method and has its advantage of having very less computation. Moreover, because of lesser computation its time consumption is also very less. So it is a good candidate for faster processing for image upscaling. It is just repeating the values of neighbors so it is not changing the any original data. There is an interpolation kernel which is defined for each direction; mathematically it is defined as [9]:

$$s(z) = \begin{cases} 0 & |z| > 0.5 \\ 1 & |z| < 0.5 \end{cases}$$

Where z is the distance between interpolated point and grid point.

B. Bilinear Interpolation

Bilinear interpolation is the another technique used for interpolation which is based on the weighted average of pixels. It calculates the average in the horizontal and vertical directions. it is computationally heavier than the nearest neighbor however results are expected to be more favorable. Its linear kernel in mathematic form is given by [9]:

$$s(z) = \begin{cases} 0 & |z| > 1 \\ 1 - |z| & |z| < 1 \end{cases}$$

Where z is distance between interpolated point and grid point.

C. Bicubic Interpolation

Bicubic interpolation is based on weighted average of 16 closest neighbors. It is more complex than other methods. However, in certain cases it gives the better results than other techniques. It consumes more time as compared to other methods. The kernel for the cubic interpolation is given by the expression as below [9]:

$$s(z) = \begin{cases} 3/2|z|^3 - 5/2|z|^2 + 1 & 0 <= |z| < 1 \\ -1/2|z|^3 + 5/2|z|^2 - 4|z| + 2 & 1 <= |z| < 2 \\ 0 & 2 < |z| \end{cases}$$

Where z is the distance between interpolated point and grid point.

IV. NOISE MODELS

It has been seen in image processing that images might have contamination with various types of noises. The various noises can be classified into categories like Gaussian noise, Salt and Pepper noise, Poisson noise or Speckle noise [14]. In order to have the comparison between upscaling and de-noising various noise models need to be considered. The effectiveness of the noise removal method has been compared based on these models. All these types of noises are present in usual images. Gaussian noise is sort of blurring noise. It usually follows the Gaussian distribution. Salt and pepper noise is noise which is introduced due to the errors in transmissions [8]. It is having two extreme values either white or black. Another type of noise like Poisson noise could be present in images; it is basically dependent upon the data. Speckle type of noise basically occurs in medical images like SAR images or ultrasound images. It is basically a multiplicative noise. It follows the gamma distribution [3]. In medical image processing removal of such a type of noise is essential [14].

V. SIMULATIONS AND ANALYSIS

In order to give the comparative analysis, the performance of Gaussian filtering along with different interpolation methods has been shown. The performance of system has been compared with different interpolation techniques. The performance matrices like PSNR (Peak signal to noise ratio) and MSE (Mean square error) have been compared for different noises. The ‘Lena.jpg’ (512 X 512) has been taken as a test image. It is firstly converted to 256 X 256 size and then various types of noises have been added to it, (as shown in figures). MATLAB [2] has been used to simulate the complete system and for implementing the interpolation.

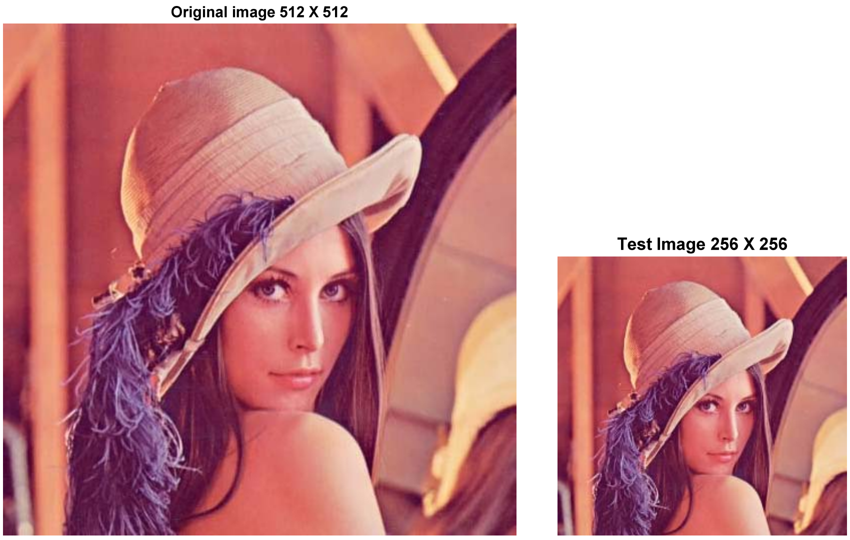

Fig. 1. Original and Test image 'lena.jpg'

Fig.2. Noisy Images

This figure shows the noisy image of small size which required to be up-scaled. In the figure it can been seen that various noises have their own effects.

A. Gaussian Filter Performance with Nearest Neighbor Interpolation

In this part of simulation all type of noises have been considered and along with that nearest neighbor interpolation has been applied for all cases. As per figure it is observed that with the kernel size of range 0.6-1.0 filter is giving its best performance along with upscaling. In the figures PSNR and MSE have been shown graphically. Tables further gives the exact values of PSNR and MSE for the Gaussian filtering and nearest neighbor based upscaling along with variation of kernel size of the filter from 0.4 to 2.0.

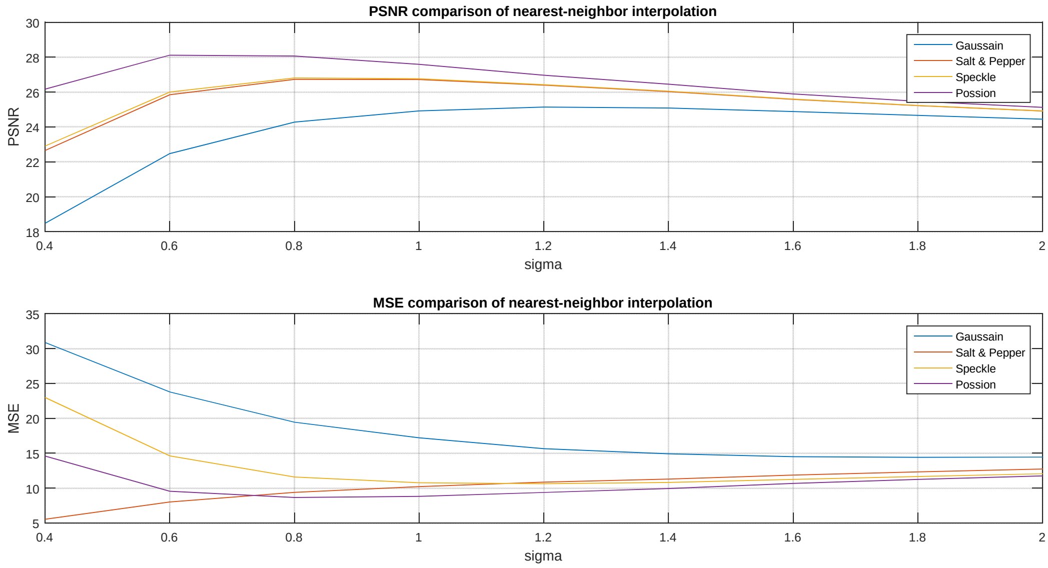

Fig. 3.PSNR and MSE for different noises with Kernel variations

TABLE.1. PSNR variations for different noises

Kernel/Sigma	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
Gaussian noise	18.47444	22.47457	24.2777	24.92218	25.1395	25.08848	24.88747	24.66614	24.44725
Salt & Pepper noise	22.648	25.84658	26.72732	26.71699	26.39472	26.03428	25.58177	25.22076	24.91091
Speckle noise	22.90975	26.00466	26.80738	26.76283	26.4234	26.05403	25.59732	25.23282	24.92164
Poisson noise	26.16673	28.11602	28.06906	27.59106	26.96656	26.45488	25.89026	25.47032	25.12353

TABLE.2. MSE variations for different noises

Kernel/Sigma	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
Gaussian noise	30.86529	23.78039	19.44646	17.21182	15.65513	14.91788	14.50839	14.40527	14.44781
Salt & Pepper noise	5.548368	8.013769	9.397716	10.22426	10.86083	11.29888	11.86754	12.32342	12.73113
Speckle noise	22.98693	14.62707	11.59517	10.7595	10.62126	10.8097	11.2534	11.66853	12.0637
Poisson noise	14.59259	9.562961	8.649673	8.81955	9.38254	9.933128	10.67966	11.25546	11.75527

B. Gaussian Filter Performance with Bilinear Interpolation

In this part of simulation all type of noises have been considered and along with that bilinear interpolation has been applied for all cases. The kernel size of range 0.6-1.0 almost is giving its best performance along with upscaling. In the figures below PSNR and MSE have been shown graphically. Tables further give the exact values of PSNR and MSE for the Gaussian bilinear upscaling along with variation of kernel size of the filter from 0.4 to 2.0.

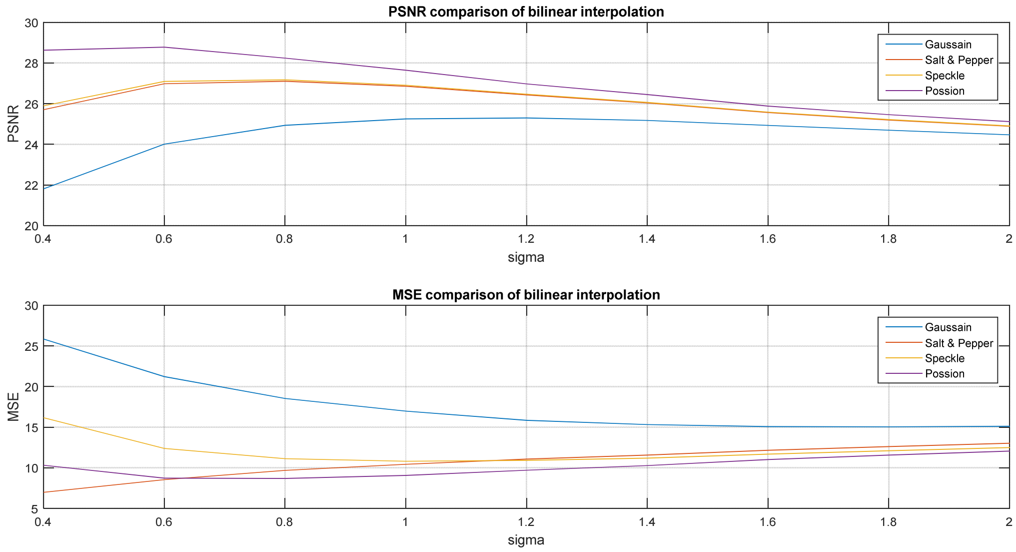

Fig. 4. PSNR and MSE for different noises with Kernel variations

TABLE.3. PSNR variations for different noises

Kernel/Sigma	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
Gaussian noise	21.81183	24.01075	24.93433	25.25215	25.29787	25.1765	24.93323	24.69501	24.46907
Salt & Pepper noise	25.70569	26.9814	27.10085	26.85905	26.43318	26.03681	25.56422	25.19845	24.88953
Speckle noise	25.89402	27.09781	27.17423	26.91102	26.46896	26.06597	25.58723	25.21773	24.90494
Poisson noise	28.63143	28.78483	28.24199	27.65	26.97455	26.45118	25.88088	25.4615	25.11623

TABLE.4. MSE variations for different noises

Kernel/Sigma	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
Gaussian noise	25.83884	21.223	18.53419	16.97857	15.84681	15.32118	15.06818	15.04354	15.11576
Salt & Pepper noise	6.984626	8.54404	9.688213	10.43709	11.08044	11.56161	12.15615	12.61689	13.01815
Speckle noise	16.16959	12.38357	11.12318	10.81004	10.91653	11.19901	11.68251	12.10442	12.48808
Poisson noise	10.31931	8.72774	8.69613	9.062617	9.70599	10.27825	11.01407	11.57264	12.05707

C. Gaussian Filter Performance with Cubic Interpolation:

In this part of simulation all type of noises have been considered and along with that cubic interpolation has been applied for all cases. As per figure it is observed that the kernel size of range 0.6-1.0 (approximately) for Gaussian filter is giving its best performance along with upscaling. In the figures below PSNR and MSE have been shown graphically. Tables further give the exact values of PSNR and MSE for the Gaussian Cubic upscaling along with variation of kernel size of the filter from 0.4 to 2.0.

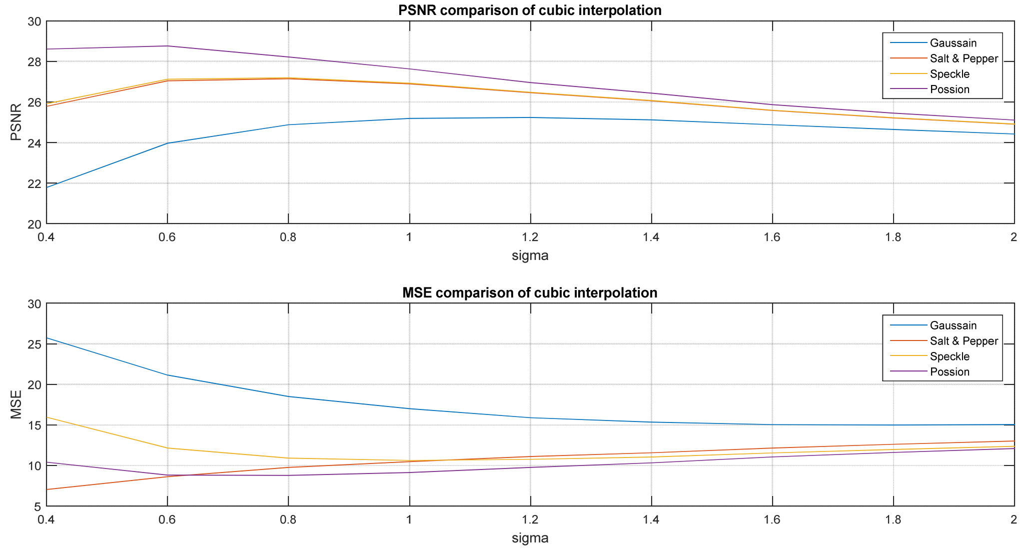

Fig. 5. PSNR and MSE for different noises with Kernel variations

Table.5. PSNR variations for different noises

Kernel/Sigma	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
Gaussian noise	21.7857	23.96871	24.88049	25.19305	25.23882	25.11944	24.88068	24.64603	24.42362
Salt & Pepper noise	25.78559	27.045	27.14881	26.89583	26.45902	26.05866	25.58288	25.21502	24.90416
Speckle noise	25.91779	27.12421	27.19585	26.9283	26.48187	26.07669	25.59721	25.22628	24.91389
Poisson noise	28.61151	28.76349	28.22246	27.63231	26.96042	26.44022	25.8717	25.45327	25.10943

TABLE.6. MSE variations for different noises

Kernel/Sigma	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
Gaussian noise	25.7363	21.1432	18.51213	17.00695	15.8912	15.34597	15.04487	14.99063	15.05045
Salt & Pepper noise	7.053727	8.62922	9.766372	10.48613	11.10792	11.57323	12.15696	12.61909	13.02201
Speckle noise	15.95501	12.15445	10.92053	10.63079	10.7645	11.05718	11.55696	11.98665	12.37816
Poisson noise	10.40951	8.821357	8.781882	9.143866	9.774626	10.33274	11.06046	11.61646	12.09015

D. Interpolation function based comparison:

One of the most important evaluations is to compare the upscaling methods. Here the performance of each one has been compared based on PSNR and MSE. Table 7 shows the comparison of PSNR while table 8 compares the MSE.

TABLE.7 PSNR variations for different noises

Noise type	**Nearest Neighbor**	**Bilinear**	**Bi-cubic**
Gaussian	22.4592	23.99901	23.16708
Salt & Pepper	25.88607	26.97495	26.60512
Speckle	25.9995	27.07355	26.71758
Poisson	28.15278	28.78777	28.91243

TABLE.8 MSE variations for different noises

Noise type	**Nearest Neighbor**	**Bilinear**	**Bi-cubic**
Gaussian	23.83386	21.03495	22.57118
Salt & Pepper	7.89564	8.584107	7.700946
Speckle	14.56014	12.32852	13.18743
Poisson	9.644241	8.852678	8.872797

On graphical representation as per figure below bilinear interpolation gives the best performance for the Gaussian noise, salt and pepper noise and also with the speckle noise. Cubic interpolation performs well with Poisson noise. Similar sort of results has been seen in MSE based comparison.

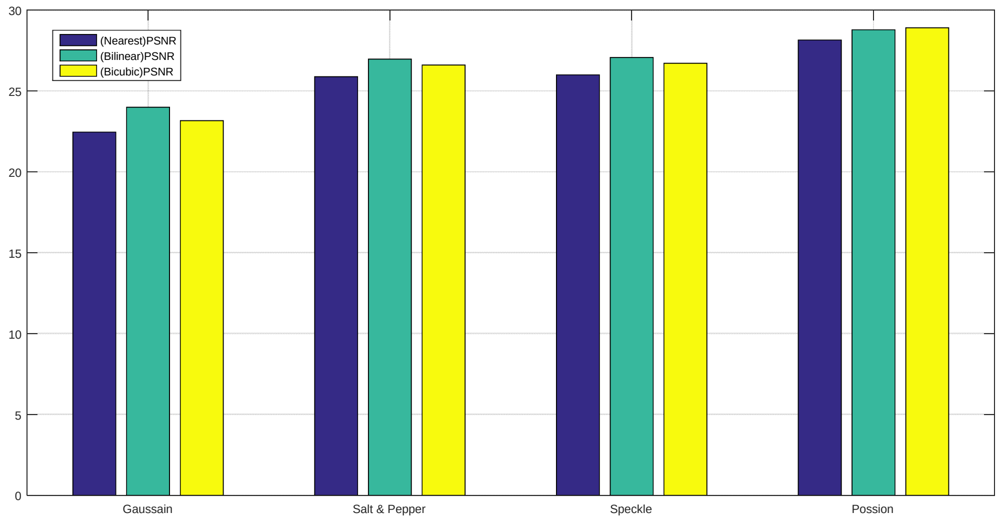

Fig. 6.PSNR variations for different noises

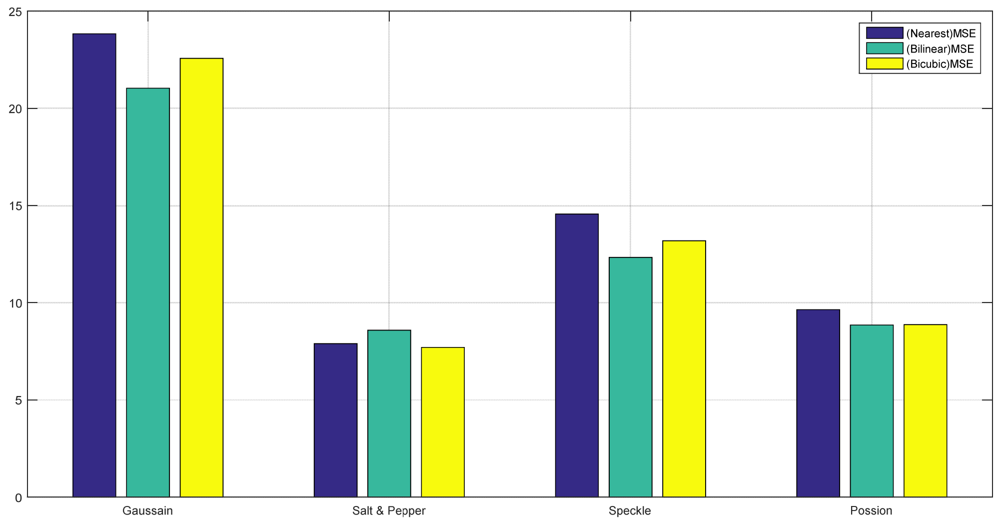

Fig. 7.MSE variations for different noises

VI. CONCLUSION

In this paper performance analysis has been done for the various upscaling methods along with the Gaussian filtering of images, applied on various images contaminated with different noises. It has been seen that there are variations in PSNR and MSE values with different kernel values of Gaussian filter. In the similar way PSNR and MSE have been varying for different interpolation methods for the same type of noise. Based on overall comparison bilinear and cubic based interpolations have given the better response along with Gaussian filtering for the same type of noise. However Gaussian based filtering can also be improved with proper selection of kernel size. Moreover, with suitable interpolation function as well as with proper selection of Gaussian kernel size, the performance for image upscaling can be further enhanced.

REFERENCES

[1] Rafael C. Gonzale z, Richard E. Woods, “Digital Image Processing”, Second edition upper saddle River, NJ: Prentice Hall, 2006.
[2] Rafael C. Gonzalez & Richard E. Woods, Digital Image Processing using MATLAB. Third edition: Pearson education, 2005.
[3] William K. Pratt, Digital Image Processing Third edition, A Wiley Inter Science Publication 2001.
[4] Jain, Anil K. Fundamentals of Digital Image Processing. Prentice Hall of India, 2002.
[5] S.Sridhar, Digital Image Processing. Oxford University Press, 2011.
[6] Bernd Jähne, Digital Image Processing Sixth edition, Springer 2005.
[7] Huiyu Zhou,Jiahua Wu and Jianguo Zhang, Digital Image Processing. Part I, 2010.
[8] Raymond H. Chan, Chung -Wa Ho, and Mila Nikolova, “Salt -and-pepper noise removal by median -type noise detectors and detail preserving regularization,” IEEE Transactions on Image Processing, vol. 14, no. 10, October 2005.
[9] Ankit prajapati, Sapan Naik , Sheetal Mehta “Evaluation of Different Image Interpolation Algorithms”, International Journal of Computer Applications (0975 – 8887) Volume 58– No.12, November 2012
[10] Celia A. Zorzo Barcelos and Marcos Aure´lio Batista, “Image restoration using digital inpainting and noise removal,” Image and Vision Computing 25, page 61–69, 2007.
[11] Ce Liu, Richard Szeliski, Sing Bing Kang, C. Lawrence Zitnick and William T.Freeman, “Automatic estimation and removal of noise from a single image,” IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 30, no. 2, February 2008.
[12] Anutam and Rajni, “Performance analysis of image denoising with wavelet thresholding methods for different levels of decomposition” The International Journal of Multimedia & Its Applications (IJMA) Vol.6, No.3, June 2014
[13] Rupinderpal , “Survey of De-noising Methods Using Filters and Fast Wavelet Transform” International Journal of Advanced Research in Computer Science and Software Engineering 3(2), February - 2013, pp. 133-136
[14] Amanjot Singh, Jagroop Singh, “Comparative Analysis of Gaussian Filter with Wavelet Denoising for Various Noises Present in Images”, Indian Journal of Science and Technology, Vol 9(47), DOI: 10.17485/ijst/2016/v9i47/106843, December 2016.
[15] P. Moulin and J. Liu, “Analysis of multiresolution image denoising schemes using generalized Gaussian and complexity priors”, IEEE Infor. Theory, Vol. 45, No 3, Apr. 1999, pp. 909-919.