

# Implementation of Enhanced Image processing Tool with efficient solutions and Comparisons for Image Smoothing using Fuzzy Filter

Kiran B <sup>1</sup>, J Satish Babu <sup>2</sup>, Ramprasad M <sup>3</sup>, Srikanth D <sup>4</sup>

M.TECH (DCN), Department of Electronics and communication Engineering, SJCIT, Chickballapur, Karnataka, India<sup>1</sup>

Associate Professor, Department of Electronics and communication Engineering, SJCIT, Chickballapur, Karnataka, India<sup>2</sup>

P.G. Student (LSP), Department of Electronics and communication Engineering, SJCIT, Chickballapur, Karnataka, India<sup>3</sup>

P.G. Student (LDS), Department of Electronics and communication Engineering, SJCIT, Chickballapur, Karnataka, India<sup>4</sup>

**ABSTRACT:** This paper describes the “Multi Platform Image Processing tool”, using the multi platform language: java. The application is implemented using java’s graphic and Advanced Imaging libraries (“JAI”). The main features are, MDI (Multiple Document Interface) style application, which allow the user the view and edit multiple images simultaneously, Support of the standard image file formats. Saving images in the following formats: JPEG, TIFF, BMP, PNM and PNG. Geometric operations on images: Rotate, Scale, Mirror (flip).

Image Enhancement, Sharpen, Median filter, curve filter, diffusion, gain filter, grayscale filter, inverter filter, quantizer filter, kelidosc open filter, rescale filter, rgb filter, threshold filter and many more. Color dithering, Arithmetic operations on images: Add, Subtract, Divide, Multiply, Image analysis, Edge detection, Adding graphics to images: Lines, Circles and filled circles Rectangles and filled rectangles, Sketching (drawing free handed), Zooming on images: Zoom in Zoom out, Zoom in on selected area.

**Keywords:** Multiple document interface, diether, video java advanced image libraries.

## I. INTRODUCTION

The project's goal is to design and implement an enhanced image processing with number of standard image processing functions (dithering, edge detection etc.). The image processing will be implemented in Java using Swing and Java Advanced Imaging libraries components. The main features of the application are: Convenient user interface. Support of standard image files formats (BMP, JPG etc.). Saving in different formats. Zoom in/out, resizing, rotation and mirroring. Basic image processing algorithms. Adding graphics and/or text to an image. During image acquisition or transmission, digital Images are often contaminated by impulse noise due to a number of non-idealities in the imaging Process. The noise usually corrupts images by replacing some of the pixels of the original image with new pixels having luminance values near or equal to the minimum or maximum of the allowable dynamic luminance range. In the most applications, it is very important to remove impulse noise from image data, since the performances of subsequent image processing tasks are strictly dependent on the success of image noise removal operation. However, this is a difficult problem in any image processing system because the restoration filter must not distort the useful information in the image and preserve image details and texture while removing the noise. The days of retake after retake are over with the continuing evolution of photo editing software. The minor errors that would've sent your pictures to the trash bin years before can be fixed with a few clicks of the mouse using one of the programs in this category. There are going to be some times when a photo is too damaged to fix, but those are going to be few and far between if you have photo editing software to back you up. The amount of pictures you're going to get rid of or have to retake will decrease dramatically, which means you will be able save those precious moments that would've otherwise been lost. Some image processing comes with a little bit of a learning curve, but that's to be expected when you're dealing with software that's geared more towards seasoned professionals. Luckily, there are programs with streamlined feature sets for those of us that haven't been in the industry for a number of years. Whether you choose one or the other depends greatly on what you need your photo editing software to do and you're level of experience. You will find programs that will meet the needs of the seasoned professionals while gently ushering new users through the process.

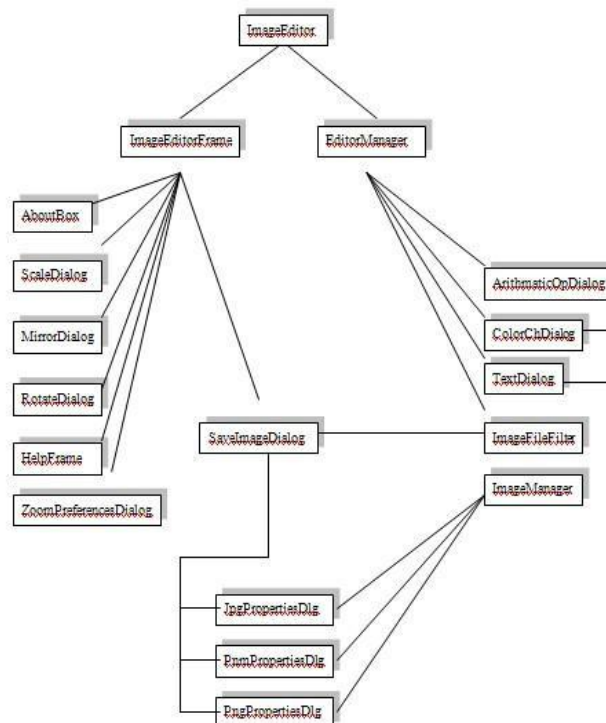
## II. METHODOLOGY

### Classes Description

The main classes of the application are the following: **Image Editor** – This class is the main class that is called by the Java virtual machines and is responsible of running the application. **Image Editor Frame** – This class represents the application main frame. It is responsible of handling the main UI of the application. **Editor Manger** – This class is responsible of the logic and handling of the multiple document interfaces. It is also responsible of performing the image manipulation actions that are done on more than one image. **Image Manger** – This class represents an image. It handles the opening and saving of an image, displaying an image on an internal frame, performing the image manipulation action, and handling the events executed on the image. **Mirror Dialog** – This class implements the user interface dialog for the mirror operation. It is responsible of getting the type of mirror action the user wants to perform. **Scale Dialog** - This class implements the user interface dialog for the scale operation. It is responsible of getting the values for the scale action. **Rotate Dialog** - This class implements the user interface dialog for the rotate operation. It is responsible of getting the rotation angle.

**ArithmeticOpDialog** – This class implements the user interface dialog for the arithmetic operations. It allows the user to choose the source and destination images and the arithmetic operation to perform. **ColorChDialog** – This class implements the user interface dialog for choosing the color for the drawing operations. **Text Dialog** –This class implements the user interface dialog for the adding text operation. It allows the user to enter the text he would like to add, to choose the font, the font size and the font color. **SaveImageDialog** – This class implements the user interface dialog for saving an image. It allows the user to enter the file name to save, the image file format to save and to launch the advance properties dialog (if exist for this image file format). **Jpg Properties Dialog** – This class implements the user interface dialog for defining the advanced save properties for JPEG image file format. **Png Properties Dialog** – This class implements the user interface dialog for defining the advanced save properties for PNG image file format. **Pnm Properties Dialog** - This class implements the user interface dialog for defining the advanced save properties for PNM image file format. **Help Frame** – This class is responsible of displaying the help pages.

Fig 1: Class Diagram



## III. FILTER

**Filters** play a significant role in the image de-noising process. It is a technique for modifying or enhancing an image. The basic concept behind reducing noise in noisy images using linear filters is digital convolution and moving window principle. Linear filtering is filtering in which the Value of an output denoised pixel is a linear combination of the values of the pixels in the input pixel's neighborhood.

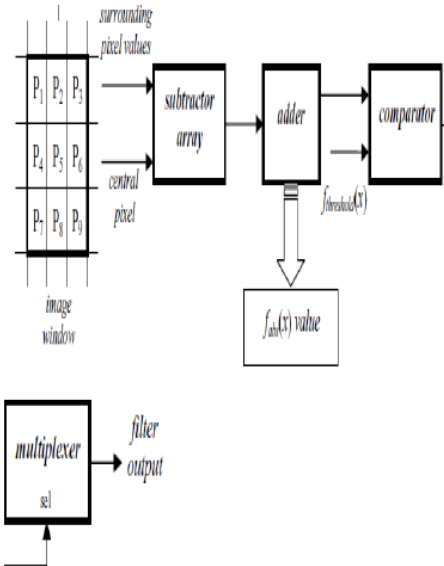


Fig 2: General architecture of a filter

**3.1 Architecture of a Mean Filter**

A mean filter acts on an image by smoothing it. i.e., it reduces the variation in terms of intensity between adjacent pixels. The mean filter is a simple moving window spatial filter, which replaces the centre value in the window with the average of all the neighbouring pixel values including that centre value. An example of mean filtering of a single 3x3 window of values is shown below. Let us consider a 512x512 image and 3x3 mask and let the filter mask is

$h_1$	$h_2$	$h_3$
$h_4$	$h_5$	$h_6$
$h_7$	$h_8$	$h_9$

Fig 3: Pixels in a image

And the neighborhood of pixel (5, 5)

$W_{44}$	$W_{45}$	$W_{46}$
$W_{54}$	$W_{55}$	$W_{56}$
$W_{64}$	$W_{65}$	$W_{66}$

$$h_1W_{44} + h_2W_{45} + h_3W_{46} + h_4W_{54} + h_5W_{55} + h_6W_{56} + h_7W_{64} + h_8W_{65} + h_9W_{66}$$

**3.2 Architecture of a Median Filter**

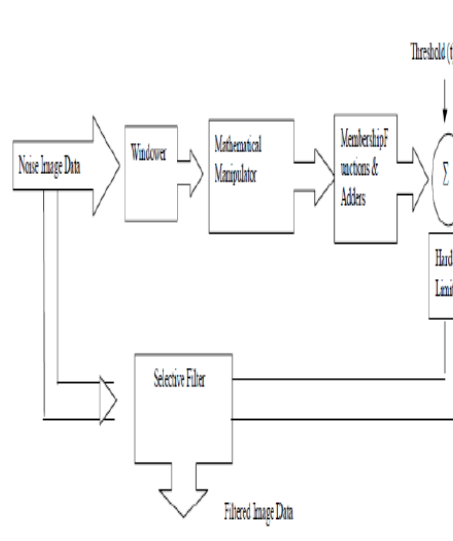
A median filter comes under the class of nonlinear filter. It also follows the moving window principle, like mean filter. A 3×3, 5×5, or 7×7 kernel of pixels is moved over the entire image. The below process illustrates the methodology of median filtering. Let us take 3x3 mask and the pixel values of image in the neighborhood of considered noisy pixel are.

125	147	175	111	150
120	115	150	108	118
122	132	<b>140</b>	107	112
112	152	128	134	112
134	155	155	198	145

Let us consider pixel at (3, 3) i.e., pixel value of 140. Neighbourhood of this pixel are 115,150,108,132,107,152,128,134.

After sorting these pixels (in ascending order).  
 We will get 107, 108, 115, 128, 132, 134, 140, 150, 152.

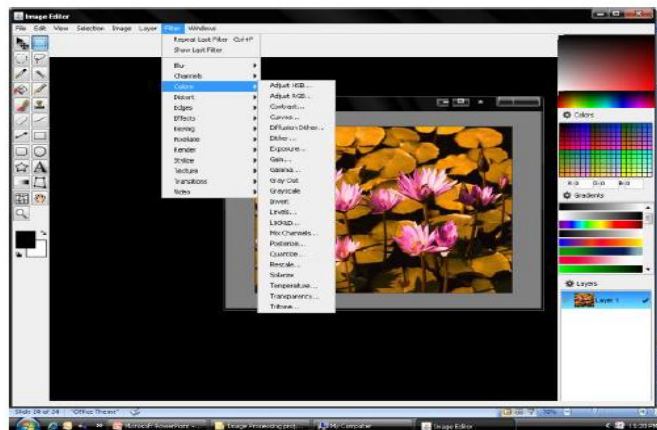
**3.3 Architecture of Fuzzy filters**



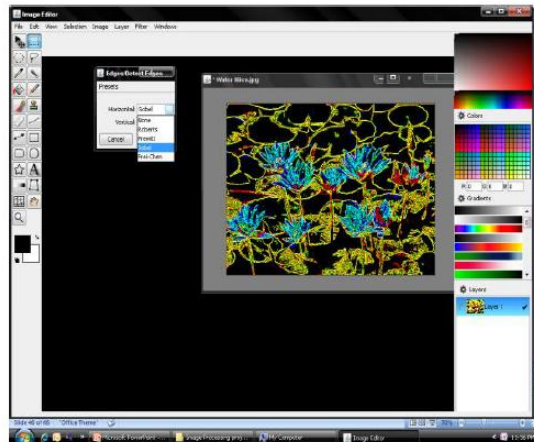
**Fig 4: Fuzzy filters implemented.**

Logic is a form of many-valued logic, it deals with reasoning that is approximate rather than fixed and exact. In contrast with traditional logic theory, where binary sets have two-valued logic: true or false, Fuzzy logic variables may have a truth value that ranges in degree between 0 and 1. Fuzzy logic has been extended to handle the concept of partial truth, where the truth value may range between completely true and completely false. Furthermore, when linguistic variables are used, these degrees may be managed by specific Functions. Various others Filters Implemented Diffusion filter, Gain filter, Gray scale filter, Inverter, Rescale filter, RGB filter, Threshold filter, keledoscope filter and many more.

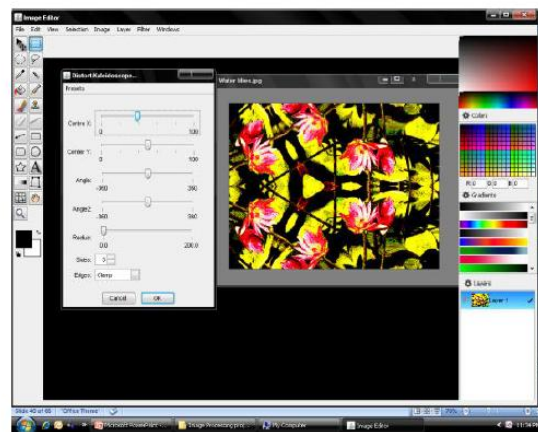
#### IV. RESULT



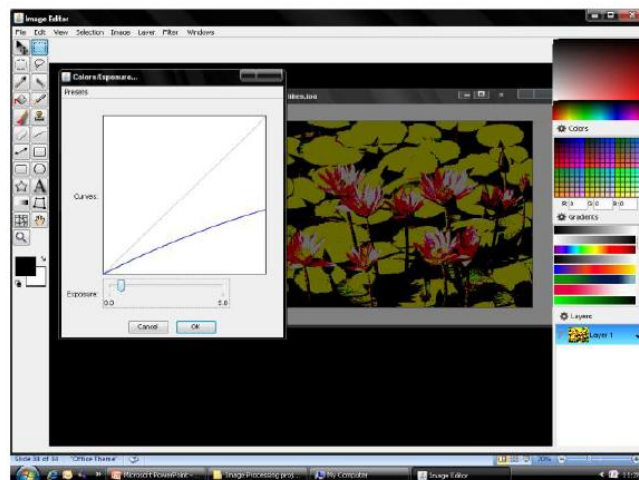
**Fig 5: various filters implemented.** Here in the above figure shows all the options present in the tool and the various filters implemented in the paper.



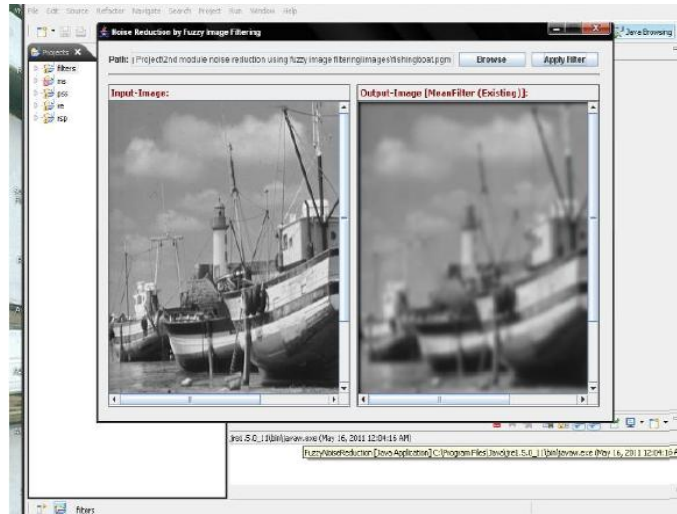
**Fig 6: Edge detection filter:** The above fig shows the Edge detection filter in which edges of the given image is detected by the above implemented filter



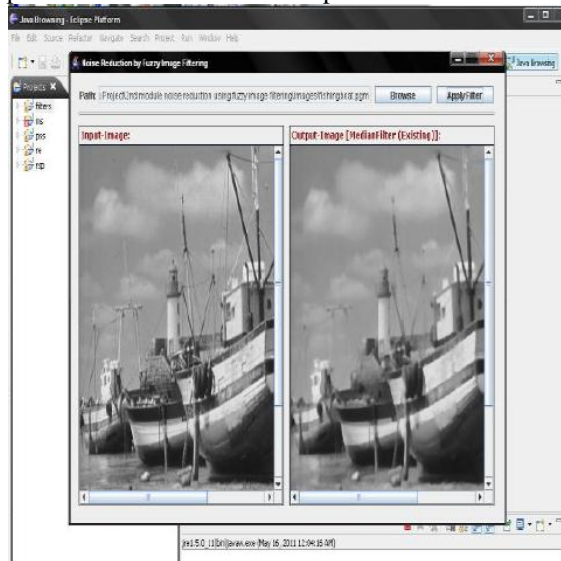
**Fig 8: The keledoscope filter.**  
 The above fig shows the Kelidoeoscope filter in which the given image X-axis,y-axis ,radius and angles can be changed .



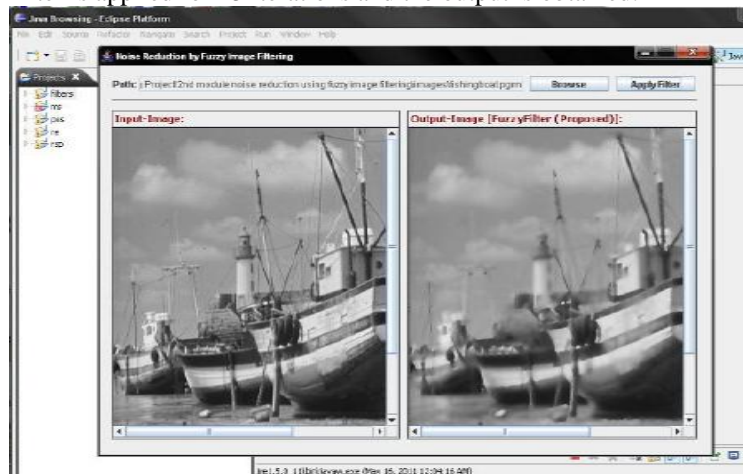
**Fig 7: Exposure filter.** The above fig shows the colour image exposure and the the RGB adjust according to our needs



**Fig 9: Mean filter after “15” iterations**  
 In the above figure mean filter is applied for 15 iterations and output is obtained.



**Fig 10: Median filter after “15” iterations**  
 In the above figure median filter is applied for 15 iterations and the output is obtained.



**Fig 11: Fuzzy filter after “1”iterations**

In the above figure fuzzy filter is applied and output is obtained.

#### **V. CONCLUSION**

In this paper, we have presented the implementation of various filters and the smoothing of image using mean, median, fuzzy filter and their results.

#### **REFERENCES**

- [1] R.C. Gonzalez and R.E. Woods, Digital Image Processing, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall; 2002.
- [2] A.K. Jain, Fundamentals of Digital Image Processing. Englewood Cliffs, NJ: Prentice Hall; 1989.
- [3] S.Sudha, G.R.Suresh and R.Sukanesh, "Speckle Noise Reduction in Ultrasound Images by Wavelet Thresholding based on weighted Variance", International Journal of Computer Theory and Engineering, Vol. 1, No. 1, April 2009.
- [4] N.Bhoi, S. Meher, "Circular Spatial Filtering under High Noise Variance
- [5] M Mastriani and A.E. Giraldez, "Smoothing of coefficients in wavelet domain for speckle reduction in synthetic aperture radar images", Journal of Graphics Vision and Image Processing, vol. 7(special issue), pp. 1-8, 2007.
- [6] X. Wang, "Lee filter for multiscale image denoising," Proceedings of international conference on signal processing, vol. 1, 2006.
- [7] Oleg V. Michailovich and Allen Tannenbaum, "Despeckling of Medical Ultrasound Images", IEEE transactions on ultrasonics, ferroelectrics, and frequency control, Vol. 53, no. 1, January 2006
- [8] M. Kazubek, "Wavelet domain image denoising by thresholding and Wiener Filtering," IEEE Signal Processing Letters, vol.10, no. 11, pp. 324-326, 2003.
- [9] D.L. Donoho, I.M. Johnstone, "Denoising by soft-thresholding," IEEE Transactions on Information Theory, vol. 41, no. 3, pp. 613-27, 1995.
- [10] N.Bhoi, S. Meher, "Gaussian noise removal using anisotropic diffusion," Proc. In National Conference on Nascent Technologies in the Engineering, NCNTE-2008.
- [11] S. Meher, Development of Some Novel Nonlinear and Adaptive Digital Image Filters for Efficient Noise Suppression, Doctoral Dissertation, National Institute of Technology, Rourkela, India.