Advance Approach towards Key Feature Extraction Using Designed Filters on Different Image Format for Providing Security

Ms. Reena Gupta
Research Scholar
Department of Computer Science & IT
JRNRV University
Udaipur, India
er.reena@gmail.com

Dr. Manju Mandot
Associate Professor
Department of Computer Science & IT
JRNRV University
Udaipur, India
manju.mandot@gmail.com

Abstract— In fast growing database repository system, image as data is one of the important concern despite text or numeric. Still we can't replace test on any cost but for advancement, information may be managed with images. Therefore image processing is a wide area for the researcher. Many stages of processing of image provide researchers with new ideas to keep information safe with better way. Feature extraction, segmentation, recognition are the key areas of the image processing which helps to enhance the quality of working with images. Paper presents the comparison between image formats like .jpg, .png, .bmp, .gif. This paper is focused on the feature extraction and segmentation stages with background removal process. There are two filters, one is integer filter and second one is floating point Filter, which is used for the key feature extraction from image. These filters applied on the different images of different formats and visually compare the results.

Keywords- *Segmented Filter, Background Removal, Image Key Feature Extraction, Edge Enhancement, Integer Filter, Floating Point Filter*

_____*****_____

I. INTRODUCTION

Since we are acquainted with that there is numerical data repository behind the any image storage or image processing as well. In general a digital image is an array of real and complex numbers represented by a finite number of bits [1]. Digital images are saved in various data storage layout or formats like BMP, JPG, GIF, PNG, and TIFF etc. Each and every format is different in a number of manners. These differentiations are present in the given table 1. The features of various types of images are compared as per their special properties [2] [3] [5] [10] [11].

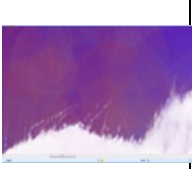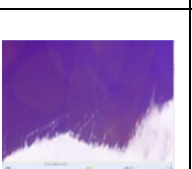TABLE I.   FEATURES COMPARISON OF THE GENERAL IMAGE FORMATS
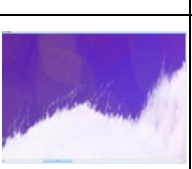
Categories	.JPEG	.PNG	.GIF	.BMP
Full Form	Joint Photographic Experts Group [4]	Portable Network Graphics [4]	Graphics Interchange Format [4]	Bitmap Image
Area	Associated with photography	Associated with Internet Graphics File and support Transparency in Browsers with an elegance [4]	Associated with Internet Graphics File & Animation File	Associated with Windows Metafile (WMF) [11] and near universal compatibility
Use	use to find redundancies in files to compress data	Allow to work on each pixel of image and reproduction of pixels by itself and provide full transparencies [4]	Use in creating animation files with very limited color or creating rough transparencies.	Use in image editing
Working	Image Compressio	Compression of Image	GIF use 8-bit color palette,	Data Compressio
	n by reducing sections of images to blocks of pixels or tiles [4]	without degrading image quality [4]	it pick color from RGB Color model and saved to a color Look Up Table (CLUT) [4]	n and alpha channel is use for image editing
Compression Nature	Lossy nature [6]	Loss-less Compression [6]	lossless LZW Compression [6]	Lossless Compression [10]
Upgrade Version	Lossless JPG technology-JPG-LS			
Supported Color Palette	JPG use 24-bit RGB, CMYK and 8-bit Grayscale [9]	PNG use 8-bit color palette as well as 24-bit RGB [4]	GIF use 8-bit color palette i.e. color Look Up Table (CLUT) [4]	Color table specified in 4-byte per entry RGBA32 format [11]

To identify with more celerity we captivating an example about measure up to the features of common image with above explained image formats. As shown in Table 2, there are four different image formats which applied on same image with different image size. For minute data study, little part of image is representation with zoom which marked with red color outlined rectangle. It was observed that the extracted zoom image result of JPEG form is blurred, GIF form is blurred with pixel disturbance, result of the PNG form and BMP form is almost same as the original one.

TABLE II.   IMAGE IN DIFFERENT FILE FORMAT WITH EFFECT

Image type & Properties	Image	Cropped Part	Zoom Effect

Type: JPEG File Size: 520 KB Dimensions: 1024 x 768			Blurred
Type: PNG File Size: 0.99 MB Dimensions: 1024 x 768			Almost Same as Original
Type: GIF File Size: 379 KB Dimensions: 1024 x 768			Blurred and Pixel Disturbance
Type: BMP File Size: 3 MB Dimensions: 1024 x 768			Same as Original

II. PROPOSED WORK OUTLINE

The anticipated work comprises the threefold action on the image which is
   (a) extracts the edges of the key features of the image,
   (b) background removal operation and
   (c) boundary extraction process [8].

These operations are although gives results just like a single operation but as mention is built of three miner actions in a single time.

III. METHODOLOGY

The key aspect of the study is key feature extraction of the image without image preprocessing stage. In that reference two filters, named as Integer Filter and Floating Point Filter are designed. These filters' code values with filter diagram and process flow are defined in table 3 and figure 3 respectively. These filters are applied on a range of images be fond of:
   a)   JPEG Image in Grayscale mode
   b)   JPEG Image in RGB mode
   c)   PNG Image in Grayscale mode
   d)   GIF Image in Grayscale mode
   e)   BMP Image in Grayscale mode

A. *Integer Filter*

This filter is designed by only integer number with either negative or positive, which is reflected in their name. It is (9 * 9) Matrix of integer numbers. This filter is applied on pixels set of image as shown in figure 1.
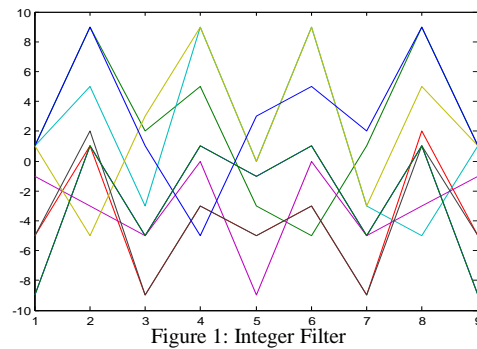
$$Integer\_Filter \in I$$

Figure 1: Integer Filter

B. *Floating Point Filter*

This filter is designed by only real number with four precision with either negative or positive, which is reflected in their name. It is (9*9) matrix of real numbers, which applied on pixels set of image as shown in figure 2.
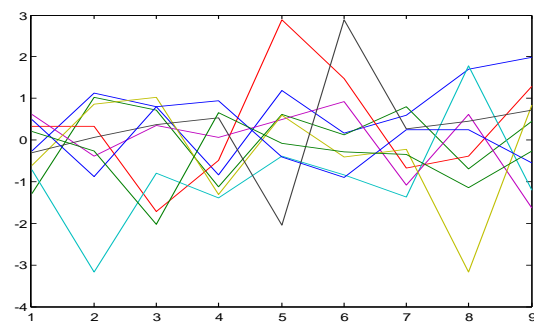
$$Floating\_Point\_Filter \in R$$

Figure 2: Floating Point Filter

The feature extraction of the image without image preprocessing was processed by the help of MATLAB software.

IV. IMPLEMENTATION

To implement proposed methodology in MATLAB following algorithm that given in section A is pursue.

A. *Algorithm*

Step 1: Capture or Select or Read Image.

   $I = Imread(Image)$

Step 2: If Image is in RGB Mode then convert it into Two Color Mode (Gray Scale Mode) [7].
I in RGB Mode ? Change in Two Color mode: Two Color mode

Step 3: Apply the Integer Filter and Floating Point Filter on the Image.

   $Filtered\_Image =$
$Integer\_Filter(I) \& Floating\_Point\_Filter(I)$

Step 4: Result will be computed with Threshold value.

   $Ori = Threshold(Filtered\_Image)$

50

Step 5: Key feature extracted image will get as resultant image.

Result = Ori

Step 6: Visualization

Step 7: Stop

B. *Pseudo Code*
Begin

- Read Image
- Grayscale conversion
- Apply filters
- Apply global threshold
- Collect result
- Visualized

End

C. *Resultant Image*

At this juncture we find the results of proposed algorithm shows the variation in the results of different image form in figure 4 to figure 29. Figure 4 to figure 8, shows results of jpeg form in grayscale mode. Figure 9 to figure 14, shows results of jpeg form in RGB mode. Figure 15 to figure 19 shows result of PNG form in grayscale mode. Figure 20 to figure 24, shows result of GIF form in grayscale mode. Figure 25 to Figure 29, shows result of BMP form in gray scale mode.
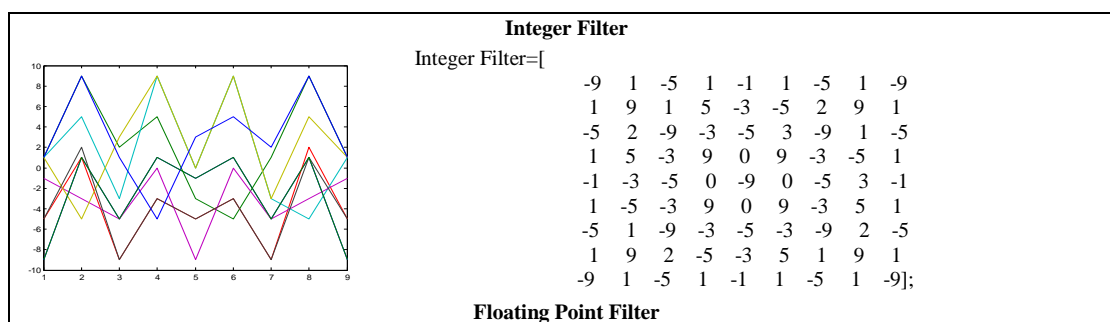
V.   CONCLUSION

As per the algorithm used for present research work, it is very much clear that work were done for boundary extraction or key feature extraction. The work and algorithm executed successfully and provides satisfied result in context of various images extensions. All the results are helpful to secure the images from the security point of view. The process includes background removal process which is also an important feature for securing images. So in the nut shell the results obtained are fully utilized for the security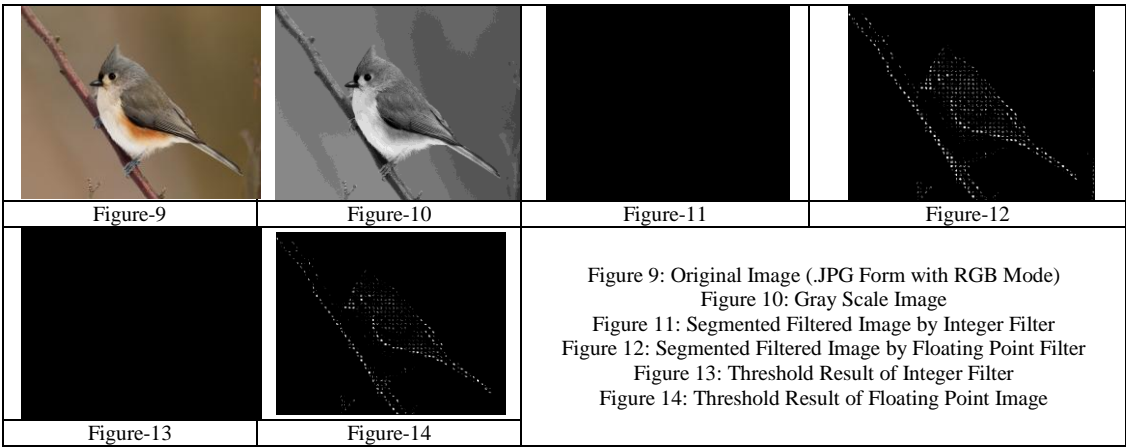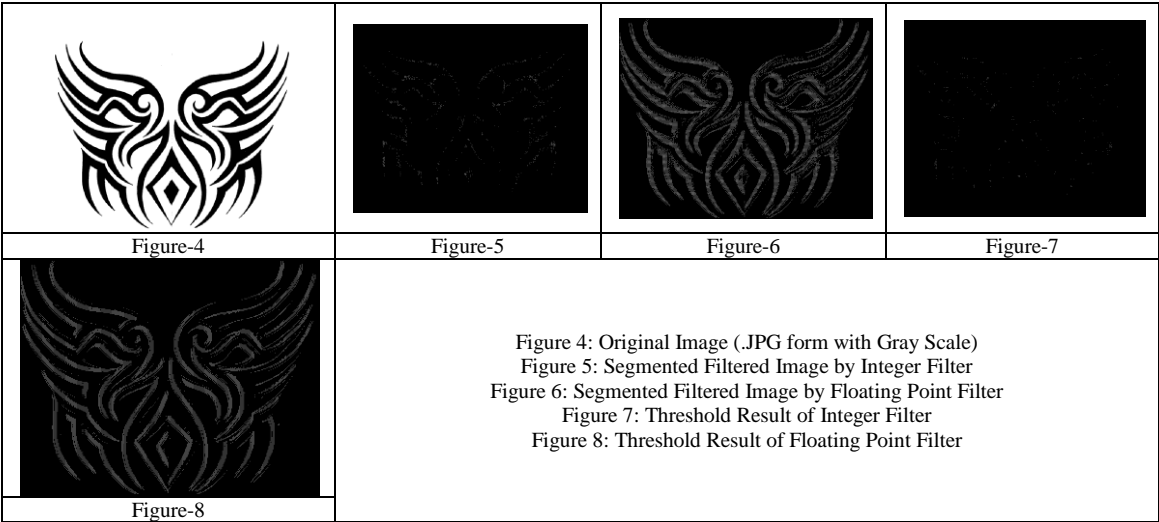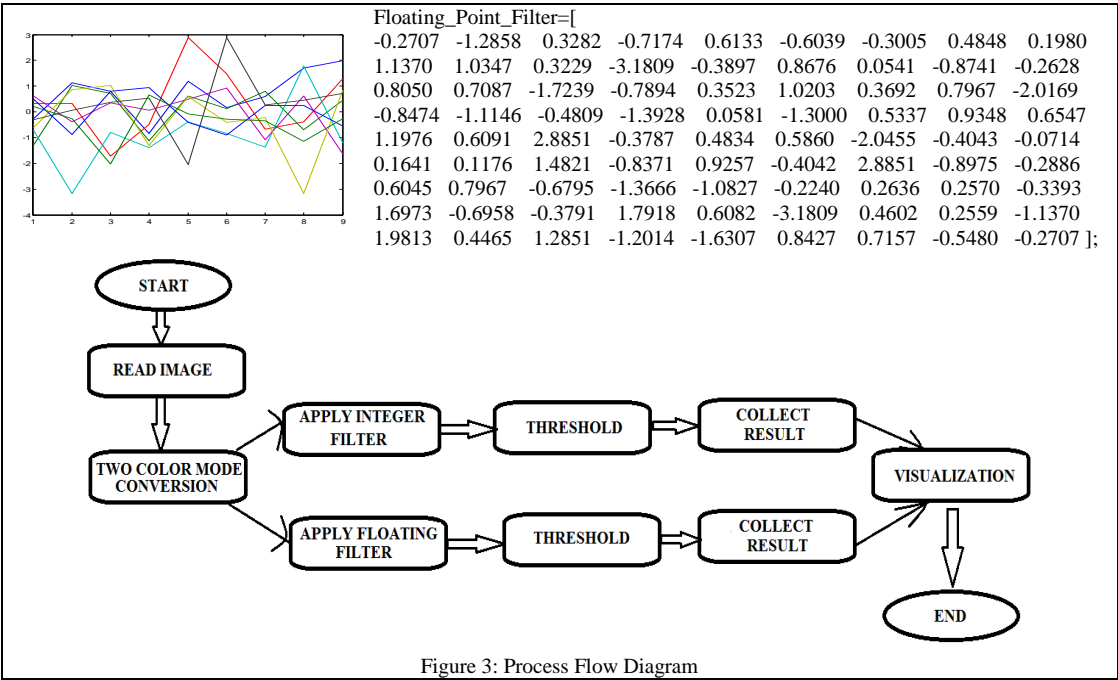 purpose of the image. In future these filters are modifying for getting more sharp edges of the objects without background.
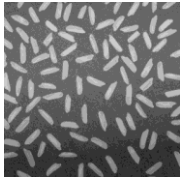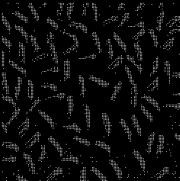
REFERENCES
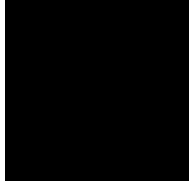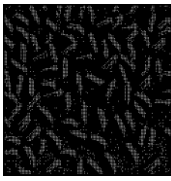
[1] Fundamentals of Digital Image Processing, Anil K. Jain, 1989-Prentice Hall Information and System Sciences Series, Thomas Kailath, Series Editor, ISBN: -0-13-336165-9, ch-1, pg-1

[2] R. Gonzalez, R.Woods, S. Eddins, "Digital Image Processing Using MATLAB ®, ISBN 978-0070-7026-22, Pearson Education, South Asia, Second Edition, 2010.

[3] T. Acharya, A.K. Ray, "Image Processing: Principles and Applications", Wiley Publication, ISBN- 978-0-471-71998-4, Aug 2005.

[4] https://en.wikipedia.org/wiki/Portable_Network_Graphics

[5] http://gizmodo.com/5656669/whats-the-difference-between-jpg-png-and-gif

[6] http://www.mvnet.fi/index.php?osio=Tutkielmat&luokka=Yliopisto&sivu=Image_compression

[7] https://edoras.sdsu.edu/doc/matlab/toolbox/images/rgb2gray.html

[8] http://in.mathworks.com/help/images/what-is-image-filtering

[9] https://en.wikipedia.org/wiki/JPEG

[10] Article on "Compare the features of the most common image formats" Source in: https://en.wikipedia.org/wiki/Comparison_of_graphics_file_formats, last updated on 06-11-2017

[11] https://en.wikipedia.org/wiki/BMP_file_format

[12] Arsath Natheem, "Image Segmentation Using MATLAB: Digital Image Processing with Source Code", Kindle Edition, Copyright © 2017 by futurebme.com

[13] Scott E Umbaugh, "Digital Image Processing And Analysis, Human and Computer Vision Applications with CVIPtools", Second Edition, published by CRC Press- Taylor & Francis Group, Boca Raton, ISBN: 13:978-1-4398-0206-9(eBook-PDF), 2010.

[14] S V D Walt, J.L. Schonberger, J.N. Lglesias, F. Boulogne, J.D. Warner, N. Yager, E. Gouillart, T. Yu, "Scikit-Image: Image processing in Phython", published in 'PeerJ Picks 2015 Collection', 19-06-2014.

[15] J. Schindelin, C. T. Rueden, M.C. Hiner, K. W. Eliceiri, "The ImageJ Ecosystem: An Open Platform for Biomedical Image Analysis", Published in Journal 'Advances in Biological Imaging: From Photons to Pixels', vol- 82, Issue- 7-8, pp- 518-529, July- August, 2015.
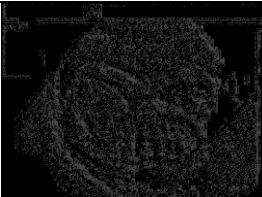
TABLE III.   MATRIX OF INTEGER & FLOATING POINT FILTER

Integer Filter	
	Integer Filter=[-9 1 -5 1 -1 1 -5 1 -9 1 9 1 5 -3 -5 2 9 1 -5 2 -9 -3 -5 3 -9 1 -5 1 5 -3 9 0 9 -3 -5 1 -1 -3 -5 0 -9 0 -5 3 -1 1 -5 -3 9 0 9 -3 5 1 -5 1 -9 -3 -5 -3 -9 2 -5 1 9 2 -5 -3 5 1 9 1 -9 1 -5 1 -1 1 -5 1 -9];
Floating Point Filter	

```
Floating_Point_Filter=[
-0.2707  -1.2858   0.3282  -0.7174   0.6133  -0.6039  -0.3005   0.4848   0.1980
 1.1370   1.0347   0.3229  -3.1809  -0.3897   0.8676   0.0541  -0.8741  -0.2628
 0.8050   0.7087  -1.7239  -0.7894   0.3523   1.0203   0.3692   0.7967  -2.0169
-0.8474  -1.1146  -0.4809  -1.3928   0.0581  -1.3000   0.5337   0.9348   0.6547
 1.1976   0.6091   2.8851  -0.3787   0.4834   0.5860  -2.0455  -0.4043  -0.0714
 0.1641   0.1176   1.4821  -0.8371   0.9257  -0.4042   2.8851  -0.8975  -0.2886
 0.6045   0.7967  -0.6795  -1.3666  -1.0827  -0.2240   0.2636   0.2570  -0.3393
 1.6973  -0.6958  -0.3791   1.7918   0.6082  -3.1809   0.4602   0.2559  -1.1370
 1.9813   0.4465   1.2851  -1.2014  -1.6307   0.8427   0.7157  -0.5480  -0.2707 ];
```


Figure 3: Process Flow Diagram

| Figure-4 | Figure-5 | Figure-6 | Figure-7 |

Figure-8

Figure 4: Original Image (.JPG form with Gray Scale)
Figure 5: Segmented Filtered Image by Integer Filter
Figure 6: Segmented Filtered Image by Floating Point Filter
Figure 7: Threshold Result of Integer Filter
Figure 8: Threshold Result of Floating Point Filter

| Figure-9 | Figure-10 | Figure-11 | Figure-12 |

| Figure-13 | Figure-14 |

Figure 9: Original Image (.JPG Form with RGB Mode)
Figure 10: Gray Scale Image
Figure 11: Segmented Filtered Image by Integer Filter
Figure 12: Segmented Filtered Image by Floating Point Filter
Figure 13: Threshold Result of Integer Filter
Figure 14: Threshold Result of Floating Point Image

Figure-15	Figure-16	Figure-17	Fihure-18

	Figure 15: Original Image (.PNG Form with Grayscale Mode) Figure 16: Segmented Filter Image by Integer Filter Figure 17: Segmented Filtered Image by Floating Point Filter Figure 18: Threshold Result of Integer Filter Figure 19: Threshold Result of Float Point Filter
Figure-19	

Figure-20	Figure-21	Figure-22	Figure-23	Figure-24

Figure 20: Original Image (.GIF with Grayscale Mode)
Figure 21: Segmented Filtered Image by Integer Filter
Figure 22: Segmented Filtered Image by Floating Point Filter
Figure 23: Threshold Result of Integer Filter
Figure 24: Threshold Result of Floating Point Result

Figure-25	Figure-26	Figure-27	Figure-28

	Figure 25: Original Image (.BMP with Grayscale Mode) Figure 26: Segmented Filtered Image by Integer Filter Figure 27: Segmented Filtered Image by Floating Point Filter Figure 28: Threshold Result of Integer Filter Figure 29: Threshold Result of Floating Point Result
Figure-29	