

# Noise Reduction in camera captured images in android application

Sumit Debnath<sup>[1]</sup> Aditya Waghdhare<sup>[2]</sup> Dashrath Mane<sup>[3]</sup>

VES Institute of Technology,  
Chembur, Mumbai

*Abstract- This paper presents innovative approach towards design, development and implementation of android application which performs the noise reduction operation compatible to android mobile devices or tablets. Noise reduction is not much been done in android as it requires the help of a lot of native libraries like OpenCV, ImageJ etc. Android application development is comparatively new in the existing field for developers. The processing of these images for noise reduction is done in a systematic approach described in this paper.*

## Introduction

Digital "noise" is a common problem in digital cameras today. Noise in a digital camera is that often annoying grainy look that makes your photos seem like they were shot in a dust storm or printed on an old dot matrix printer. Getting unwanted noise in your images is something that most of us have struggled with. Digital noise usually represents itself as speckled pixels of colour in your images or as a grainy effect, and is generally considered undesirable.

If you have a digital camera, whether you have an advanced top of the line DSLR or a simple point and shoot, you will at some point get images with small dots all over the image. Those small dots might not be very noticeable when you look at the image on the back of the camera, but when you zoom in and view the image on your PC, or the a print out of the image then, they all of a sudden become quite visible.

**Image noise** is random (not present in the object imaged) variation of brightness or colour information in images, and is usually an aspect of electronic noise. In digital photographs, "noise" is the commonly-used term to describe visual distortion.

While most people turn to post-production software such as Photoshop to reduce the negative effects of noise, there are also a lot of things you can do before that point to help solve the problem.

## Causes of Noise

Noise is added to an image usually at the time of the image capturing process. There are different causes for the adding of noise in the image. The amount of pixel corrupted in an image represents the amount of noise included in an image.

There are a few major factors responsible for noise in a digital image:

- ✓ The environmental situation affects the camera sensors resulting in adding Thermal Noise in the image.
- ✓ Low light around the camera at the time of image capturing.
- ✓ Change in the temperature around the camera sensors causes the addition of noise in the image.
- ✓ Dust particles around the camera lenses in the environment causes noisy image.

So we divided our process of noise reduction into two parts

1. Pre-Processing operation.
2. Post-Processing operation.

## Pre-Processing Operation

As the unwanted effects in the captured image is the noise in the image.

The noise is the undesirable effects produced in the image during image acquisition of the image. At the time of the image capturing process if there are not adequate conditions available there are chances of the noise being elevated in the image.

So if the image is not taken with the perfect setting of the camera this may make things worse for the image.

Thus providing with the perfect setting according to the environment is a very essential process even before the image is acquired. The following changes in the camera setting has to be made for good image acquisition

1. Spread the focus area all across the document.
2. Set the shutter speed of the camera to the minimum.
3. Set the focus of the camera all still capture of images without being effected of motion blur.
4. Set the camera to its maximum exposure.
5. Set the ISO of the camera to the maximum.
6. Make sure that the environment has enough light spread continuously across the document.
7. The flash must be set off as it flashes light only to a particular part of the document, which makes the other parts of the document dim.

So we provide a custom camera with the above setting already done in it, as expecting the user to do the above setting to the camera is not expected n more over all the mobile phone cameras come with a lot of setting in it.

The above setting of the camera is possible with the help of the API provided by the android called the Camera API. This Camera API allows us to pass the appropriate camera parameters while the camera is being opened.

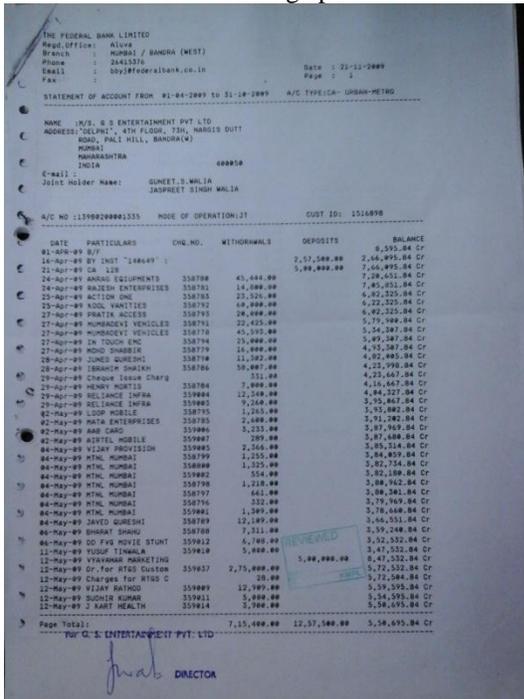


Fig. 1- Original captured image

## Post-Processing Operation

The Image acquired at the pre-processing operation now has to be operated so as to reduce the noise in the image.

- I. Separating the image into 4x4 parts.
- II. Calculating the greyscale frequency.
- III. Calculate the Moving Average the greyscale frequency.
- IV. Calculate the Rate of Change of the Moving Average Frequency.
- V. Calculate the 2<sup>nd</sup> Derivative of the Moving Average of the Frequency
- VI. Calculate the Crossover of the Two Derivatives.
- VII. Clean Up the Image
- VIII. Overlaying all the 4x4 Images as One Image

### I Separating the image into 4x4 parts.

The image is divided into a matrix of 4x4 parts, as the noise is not equally spread across the whole image so each part of the image has different levels of noise in it. So for linear affect of the noise reduction of the whole image this operation is applied to 4x4 parts of the image instead of one single image.

### II Calculating the greyscale frequency.

Now each 4x4 part of the image is considered as one single image. The Gary value intensity of every pixel of the image is stored and its frequency is calculated.

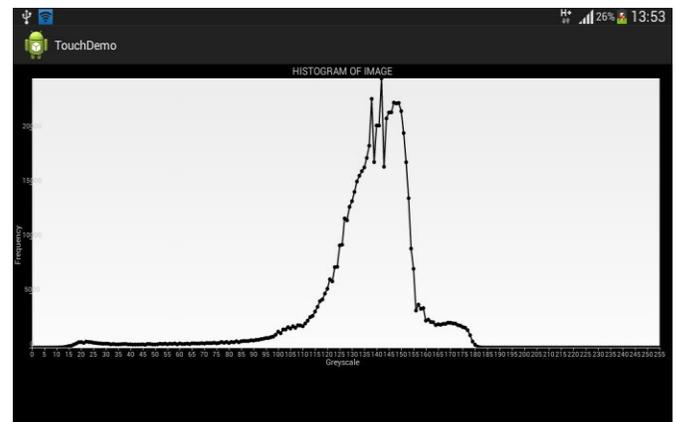


Fig. 2- Greyscale frequency graph(Histogram of Image)

### III Calculate the Moving Average the greyscale frequency.

The calculated frequency is the applied with a moving average algorithm.

$$Sf_i = (f_{(i-4)}+f_{(i-3)}+f_{(i-2)}+f_{(i-1)}+f_{(i)}+f_{(i+1)}+f_{(i+2)}+f_{(i+3)}+f_{(i+4)})/9$$

Where

$Sf_i$  is the Smooth Frequency of the  $i^{\text{th}}$  gray value.

$i$  is the gray values ranging from 0-255.

$f$  is the frequency of the gray value.

This is done so the frequency is represented in a smooth way, so as the noise parts is easily distinguished in the image histogram.

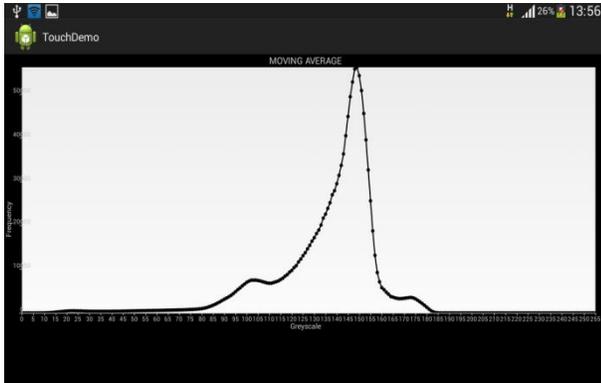


Fig. 3- Moving Average

#### IV Calculate the Rate of Change of the Moving Average Frequency.

Rate of change of the Smoothened frequency is calculated.

$$DyDiff1_{(i)} = f_{(i)} - Sf_{(i)}$$

Where

$DyDiff1$  is the rate of change of the Moving Average Frequency ie the 1<sup>st</sup> Derivative of it.

$Sf_i$  is the Smooth Frequency of the  $i^{\text{th}}$  gray value.

$i$  is the gray values ranging from 0-255.

This calculated so as the change in the Moving Average Frequency is clearly represented



Fig. 4- 1st Derivative of Moving Average

#### V Calculate the 2<sup>nd</sup> Derivative of the Moving Average of the Frequency

This is the rate of change of the rate of change of the Moving Average.

$$DyDiff2_{(i)} = DyDiff1_{(i)} - DyDiff1_{(i-1)}$$

Where

$DyDiff2$  is the rate of change of the rate of change of the Moving Average Frequency ie the 2<sup>nd</sup> Derivative of it.

$DyDiff1$  is the rate of change of the Moving Average Frequency ie the 1<sup>st</sup> Derivative of it.

$i$  is the gray values ranging from 0-255.

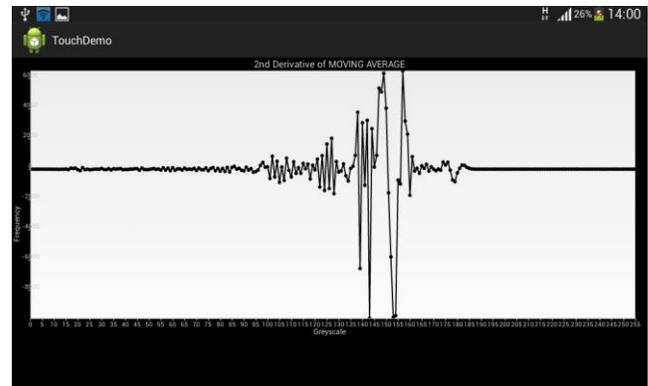


Fig. 4- 2nd Derivative of Moving Average

#### VI Calculate the Crossover of the Two Derivatives

The crossover point is the point where there is drastic change in the movements of the frequency and can be recognised as noise in the image. So the crossover point of the two derivatives ie DyDiff2 and DyDiff1 is found out by the following way:

```
if((((DyDiff2 (i))>0) AND
      (DyDiff2(i+1))>0) AND
      (DyDiff2(i+2))>0) AND
      (DyDiff1(i)* DyDiff1(i+2))<0.0)
```

Where  
DyDiff2 is the rate of change of the rate of change of the Moving Average Frequency ie the 2<sup>nd</sup> Derivative of it.  
DyDiff1 is the rate of change of the Moving Average Frequency ie the 1<sup>st</sup> Derivative of it.  
i is the grey values ranging from 0-255.

Then the start point of the noisy section of the image has been found.

### VII Clean Up the Image

The cleanup of the image is an algorithm of making the image sharper as the important and the good pixels of the images are shifted to the 0 side of the Greyscale and the noise is shifted to the 255 side of the Greyscale, without losing any pixel.

```
if (pix[x][y] <= (float) (CrossoverPoint))
{
  red   = (( (pixel >> 16) & 0xff) *.05);
  green = (( (pixel >> 8) & 0xff) *.05);
  blue  = (( (pixel >> 0) & 0xff) *.05);
}
else
{
  red   = (Math.min(255,(( (pixel >> 16) & 0xff) *2) ));
  green = (Math.min(255,(( (pixel >> 8) & 0xff) *2) ));
  blue  = (Math.min(255,(( (pixel >> 0) & 0xff) *2) ));
}
```

Where  
pix is the pixel value of the image at a specific point.  
x is the point on the x axis of the image ranging from 0-image Width.  
y is the point on the y axis of the image ranging from 0-image Height.  
Crossover Point is the point at which the noise of the image starts.

### VIII Overlaying all the 4x4 Images as One Image

All the 4x4 images are then taken together and shown as one single image.

DATE	PARTICULARS	CHQ. NO.	WITHDRAWALS	DEPOSITS	BALANCE
01-Apr-09	B/F				8,595.84 Cr
16-Apr-09	BY INST "148649"		2,57,500.00		2,66,895.84 Cr
21-Apr-09	CA 128			5,00,000.00	7,66,895.84 Cr
24-Apr-09	ANRAG EQUIPMENTS	358780	45,444.00		7,28,451.84 Cr
24-Apr-09	RAJESH ENTERPRISES	358781	14,000.00		7,14,451.84 Cr
25-Apr-09	ACTION ONE	358783	23,526.00		6,92,925.84 Cr
25-Apr-09	KODL VANITIES	358792	98,000.00		6,22,925.84 Cr
27-Apr-09	PRATIK ACCESS	358785	28,000.00		6,42,925.84 Cr
27-Apr-09	MUMBADEVI VEHICLES	358791	22,425.00		5,79,500.84 Cr
27-Apr-09	MUMBADEVI VEHICLES	358778	45,595.00		5,34,205.84 Cr
27-Apr-09	IN TOUCH ENC	358794	25,000.00		5,49,205.84 Cr
27-Apr-09	MHD SHABIR	358779	16,000.00		4,93,205.84 Cr
28-Apr-09	JUNED QURESHI	358790	11,902.00		4,82,803.84 Cr
28-Apr-09	IBRAHIM SHAIKH	358786	58,007.00		4,23,996.84 Cr
29-Apr-09	Cheque Issue Chrg		331.00		4,23,667.84 Cr
29-Apr-09	HENRY MOTIS	358784	7,000.00		4,16,667.84 Cr
29-Apr-09	RELIANCE INFRA	359004	12,340.00		4,04,327.84 Cr
29-Apr-09	RELIANCE INFRA	359003	9,266.00		3,95,061.84 Cr
02-May-09	LOOP MOBILE	358795	1,265.00		3,93,796.84 Cr
02-May-09	MATA ENTERPRISES	358785	2,689.00		3,91,207.84 Cr
02-May-09	AND CARD	359006	3,232.00		3,87,975.84 Cr
02-May-09	AIRTEL MOBILE	359007	289.00		3,87,686.84 Cr
04-May-09	VIJAY PROVISION	359005	2,366.00		3,85,320.84 Cr
04-May-09	MTNL MUMBAI	358799	1,255.00		3,84,065.84 Cr
04-May-09	MTNL MUMBAI	358800	1,325.00		3,82,740.84 Cr
04-May-09	MTNL MUMBAI	359002	554.00		3,82,186.84 Cr
04-May-09	MTNL MUMBAI	358798	1,218.00		3,80,968.84 Cr
04-May-09	MTNL MUMBAI	358797	661.00		3,80,307.84 Cr
04-May-09	MTNL MUMBAI	358796	332.00		3,79,975.84 Cr
04-May-09	MTNL MUMBAI	359001	1,389.00		3,78,586.84 Cr
04-May-09	JAVED QURESHI	358789	12,199.00		3,66,387.84 Cr
06-May-09	BHARAT SHAHU	358788	7,311.00		3,59,276.84 Cr
06-May-09	DO FOR MOVIE STUNT	359012	6,740.00		3,52,536.84 Cr
11-May-09	YUSUF TINWALA	359010	5,000.00		3,47,536.84 Cr
12-May-09	YVAVAHAR MARKETING			5,00,000.00	8,47,536.84 Cr
12-May-09	Dr. for RTOS Custom				5,72,536.84 Cr
12-May-09	Charges for RTOS C				5,72,504.84 Cr
12-May-09	VIJAY RATHOD	359009	12,999.00		5,59,505.84 Cr
12-May-09	SUDHAR KUMAR	359011	5,000.00		5,54,505.84 Cr
12-May-09	J KART HEALTH	359014	3,900.00		5,50,605.84 Cr
Page Total:			7,15,400.00	12,57,500.00	5,58,695.84 Cr

Fig. 6- final processed image

### CONCLUSION

Different image processing tasks can be done in java as well as open cv, it will be new research field to develop and test these applications on mobile phones. Since more new hardware platforms for the android operating system are being released every month, it will also be important to test the

### References

- [1] EE368 Digital Image Processing, "Tutorial On Using Android for Image Processing Projects", Spring 2012, Windows Version.
- [2] J.R.Parker. Algorithms for Image Processing and Computer Vision, John Wiley and Sons, 1997.
- [3] Mehmet Sezgin and Bulent Sankur, "Survey Over Image Thresholding Techniques and Quantitative Performance Evolution.", Journal of Electronic Imaging 13(1), 146 165 (January 2004).

- [4] Wrox Programmer to Programmer,"*Beginning Android Application Development*".
- [5] L.K.Huang and M.J.Wang. "Image Thresholding by Minimizing The Measures of Fuzziness", *Pattern Recognition*, vol 28, p. 41-51,1995
- [6] C. A. B. Mello *et al.*, "Image Segmentation of Ovitrams for Automatic Counting of *Aedes Aegypti* Eggs", in 30th Annu. International Conf. of the IEEE Engineering in Medicine and *Biology Society*, Vancouver, 2008, pp. 3103-3106.
- [7] Carlos A. B. Mello, Wellington P. dos Santos, Marco A. B. Rodrigues, Ana Lucia B. Candeias, Cristine M. G. GusmãoImage,"Segmentation of Ovitrams for Automatic Counting of *Aedes Aegypti* Eggs", 30th Annual International IEEE EMBS Conference Vancouver, British Columbia, Canada, August 20-24, 2008.
- [8] O Reilly's "Learning Open CV" by Gary Bradski and Adrian Kaehler