Illumination Correction Techniques

for images in android platform

Harshit Kumar Gupta
Department of Electrical Engineering
Indian Institute of Technology Delhi
New Delhi , India
harshit.knit@gmail.com

Project Superviser

Prof Santanu Chaudhury
Department of Electrical Engineering
Indian Institute of Technology Delhi
New Delhi , India

Abstract—**The present work is a part of an initiative to build a android application for image illumination correction.**

The goal of illumination correction is to remove uneven illumination of the image caused by sensor defaults , non uniform illumination of the scene, or orientation of the objects surface.

I. Introduction

Illumination is the deliberate use of light to achieve a practical or aesthetic effect. Lighting includes the use of both artificial light sources like lamps and light fixtures, as well as natural illumination by capturing daylight . Extra exposure of light on the surface of object , orientation of object surface and uneven illumination due to sensor and non-uniform illumination of scene are various forms of noises in images. Our objective is to remove these noises from the image.

II. Category Of Noises

1) *Gaussian noise*

Principal sources of Gaussian noise in digital images arise during acquisition e.g. sensor noise caused by poor illumination and/or high temperature, and/or transmission e.g. electronic circuit noise.

2) *Shot noise*

The dominant noise in the lighter parts of an image from an image sensor is typically that caused by statistical quantum fluctuations, that is, variation in the number of photons sensed at a given exposure level. This noise is known as photon shot noise.

3) *Quantization noise (uniform noise)*

The noise caused by quantizing the pixels of a sensed image to a number of discrete levels is known as quantization noise. It has an approximately . uniform distribution Though it can be signal dependent, it will be signal independent if other noise sources are big enough to cause dithering, or if dithering is explicitly applied

4) *Anisotropic noise*

Some noise sources show up with a significant orientation in images. For example, image sensors are sometimes subject to row noise or column noise

III. Image noise reduction

Most algorithms for converting image sensor data to an image, whether in-camera or on a computer, involve some form of noise reduction. There are many procedures for this, but all attempt to determine whether the actual differences in pixel values constitute noise or real photographic detail, and average out the former while attempting to preserve the latter. However, no algorithm can make this judgment perfectly, so there is often a trade off made between noise removal and preservation of fine, low-contrast detail that may have characteristics similar to noise. Many cameras have settings to control the aggressiveness of the in-camera noise reduction.

IV. Illumination Correction

Illumination correction is based on background subtraction. This type of correction assumes the scene is composed of an homogeneous background and relatively small objects brighter or darker than the background. There are two major types of background subtraction techniques depending on whether the illumination model of the images can be given as additional images or not:

1. prospective correction,

2. retrospective correction

V. Prospective correction

Prospective correction uses additional images obtained at the time of image capture. Two types of additional images can be used.

- A dark image is an image of the scene background acquired with no light.
- A bright image is an image of the scene background acquired with light but without objects.

It is always advantageous to capture several occurrences of bright images and dark images in order to remove noise and attenuate lighting defects. The model of each type of image is then the average of these images:

- dark = ∑ darki
- bright = ∑ brighti

1.Correction from a Dark Image and a Bright Image

The corrected image g(x,y) is obtained using the following transformation:

$$g(x,y) = \frac{f(x,y) - d(x,y)}{b(x,y) - d(x,y)} . C$$

where f(x,y) is the original image, d(x,y) is the dark image, b(x,y) is the bright image, and C is a

$$C = mean(f(x,y)) . \frac{1}{mean\left(\frac{f(x,y) - d(x,y)}{b(x,y) - d(x,y)}\right)}$$

normalization constant used to recover the original colors. where mean(i(x,y)) is the mean value of the image i(x,y).

2.Correction from a Bright Image

If only the bright image is available, the method uses division of the source image by the bright image if the image acquisition device is linear, or the subtraction of the source image with the bright if the acquisition device is logarithmic with a gamma of 1.

In case of a linear acquisition device, the corrected image g(x,y) is obtained using the following transformation:

$$g(x,y) = \frac{f(x,y)}{b(x,y)} . C$$

where f(x,y) is the original image, b(x,y) is the bright image, and C is a normalization constant that is used to recover the initial colors:

$$C = mean(f(x,y)) . \frac{1}{mean\left(\frac{f(x,y)}{b(x,y)}\right)}$$

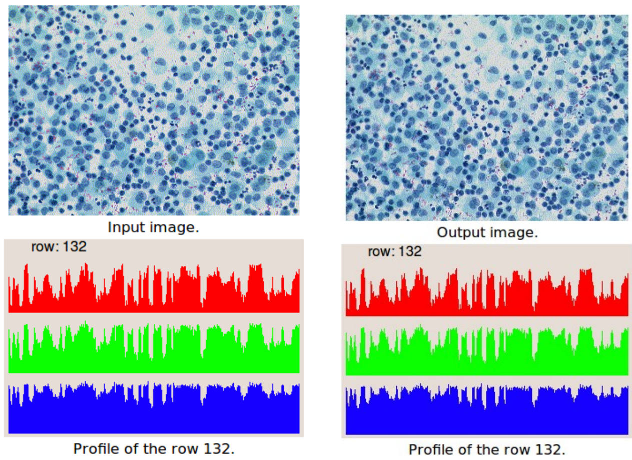
where mean(i(x,y)) is the mean value of the image i(x,y).

3.Correction from a Dark Image

If only the dark image is available, the method consists in subtraction of the dark image with the original image. The corrected image g(x,y) is then obtained using the following transformation:

$$g(x,y) = f(x,y) - d(x,y) + mean(d(x,y))$$

where f(x,y) is the original image, d(x,y) is the dark image and mean(d(x,y)) is the mean value of the dark image.



The images in the second row are the profiles of the row 132, where the gray level values are represented as vertical bars. We notice that in the input image the illumination is higher on the right side than on the left side .

VI. Retrospective correction

When additional image are not available, an ideal illumination model has to be estimated to define the bright image. Therefore , retrospective correction can use the same background subtraction method than the prospective correction with the estimated bright image. There are different algorithms for estimating the bright image. All of them assume the scene background corresponds to the low frequencies and the objects to the high frequencies. The retrospective correction techniques consist in removing the objects from the background to build the bright image, and then apply the same techniques as the prospective correction.

1. Retrospective Correction using Low-pass Filtering

The background is estimated by using a low-pass filtering with a very large kernel. The background is then subtracted from the input image to compensate the illumination.

The corrected image g(x,y) is obtained from the input image f(x,y) by:

$$g(x,y) = f(x,y) - LPF(f(x,y)) + mean(LPF(f(x,y)))$$

where LPF(f(x,y)) is the low-pass filtering of image f(x,y), and mean(LPF(f(x,y))) is the mean value of the low pass image.

2. Retrospective Correction using Homomorphic Filtering

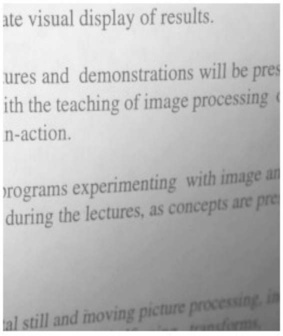
The background is estimated by using homomorphic filtering with a low-pass filter. Homomorphic principle is to remove high frequencies (considered as reflectance) and keep the low frequencies (considered as illumination). The background is removed by high pass filtering the logarithm of the image and then taking the exponent (inverse logarithm) to restore the image.
The corrected image g(x,y) is obtained from the input image f(x,y) by:

$$g(x,y) = \exp(LPF(\log(f(x,y)))) \cdot C$$

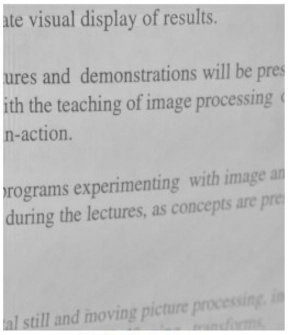
where LPF(f(x,y)) is the low-pass filter of f(x,y) and C is the normalization coefficient given by:

$$C = \frac{mean(f(x,y))}{mean\left(\frac{f(x,y)}{\exp(LPF(\log(f(x,y))))}\right)}$$

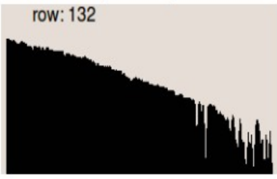
where mean(i(x,y)) is the mean value of the image i(x,y).



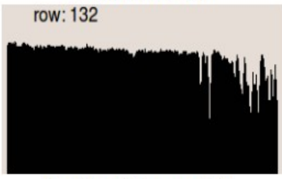
Input image.



Output image.



Profile of the row 132.



Profile of the row 132.

3.Retrospective Correction using Morphological Filtering

The background is estimated by mathematical morphology opening or closing. The estimated background is then subtracted from the original image. The total sequence of operations corresponds to a top hat of the image. Top hat removes high frequencies (considered as reflectance) and keeps low frequencies (considered as illumination).

Black top hat is used for clear background and white top hat is used for dark background. If the background is clear, the corrected image g(x,y) is obtained using:

g(x,y) = BTH[f(x,y)] + mean(closing(f(x,y)))

g(x,y) = [f(x,y) - closing(f(x,y))] + mean(closing(f(x,y)))

where mean(closing(f(x,y))) is the mean value of the closed image. We should use a disc structuring element with size , greater than the size of the objects (characters).

B. Figures and Tables

1) PSNR: **Peak signal-to-noise ratio**, often abbreviated **PSNR**, is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel scale.

PSNR is most easily defined via the mean squared error (*MSE*). Given a noise-free *m*×*n* monochrome image *I* and its noisy approximation *K*, *MSE* is defined as:

$$MSE = \frac{1}{m\,n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2$$

The PSNR is defined as:

$$\begin{aligned} PSNR &= 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \\ &= 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \\ &= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE) \end{aligned}$$

TABLE I. PSNR COMPARISON

No	Value PSNR		
	PSNR	***min***	***max***
1	Correction from a Bright Image and Dark Image	48	50
2	*Correction from Bright Image*	13	20
3	*Correction from Dark Image*	60	63
4	*Retrospective Correction using Low-pass Filtering*	20	25
5	Retrospective Correction using Homomorphic Filtering	16	20
6	Retrospective Correction using Morphological Filtering	3	10

VII. Future Work

Visual Quality of images can be further be improved by using image enhancement techniques.

Above method can also be used for increasing efficiency of character recognition from textual images.

Above techniques can also be used in retinal and MRI scan images to remove illumination.

References

[1] Android Developers ”http://www.developer.android.com/”

[2] Hsu, Mertens, Paris, Avidan, Durand. "Light mixture estimation for spatially varying white balance". In: ACM Transactions on Graphics, 2008

[3] Daniel Sauter ”Rapid Android Development,Build Rich, Sensor-Based Applications with Processing”

[4] Leprojet Pantheon ”https://www.clouard.users.greyc.fr/Pantheon/index-en.html”

[5] Wikipedia ”http://en.wikipedia.org/wiki/Illumination”