A

OEA Report

on

"Design and implement an image processing technique for low light image enhancement"

Our Idea: "To enhance low light image by using various techniques like HSV, Histogram equalization and Gamma correction"

Course: 20PEEC501LC Digital Image Processing LAB

Prepared By

1.Swaroopa Dubal - UEC2021210

2. Mayuri Lomate - UEC2021229

3. Kasturi Mandlik -UEC2021231

4. Nikita Miraje -UEC2021232

TY BB1

Department of Electronics and Telecommunication Engineering,
MKSSSs, Cummins College of Engineering for Women, Pune,
Academic Year – 2023-24

CONTENTS

INTRODUCTION:

Low light enhancement is the process of improving the quality and visibility of images or videos that are captured under conditions of insufficient lighting. In low light conditions, images often suffer from various issues such as poor visibility, high noise, reduced contrast, and distorted colors. Low light enhancement techniques are used to address these challenges and produce clearer and more visually appealing results.With the rapid improvements in camera sensor quality, smartphones have given point-and-shoot cameras and digital single-lens reflex cameras (DSLRs) a run for their money. Although smartphones do hold their own against DSLRs in most aspects, the one area where they fared badly is low-light photography. Due to limited computational resources and the requirement of the fastest possible processing time, image processing algorithms are under significant performance pressure to produce high-quality media in low-light environments.This challenge in low light for mobile devices is well known in the computational photography community but remains open. By now, one of the most mature solutions in the industry is based on multiple-exposure frames fusion, which requires end-users to hold 2s-4s to capture multiple exposure frames with the same contents later to be blended. But it is hard to hold still long enough to take a good picture in dim light, and such multiple-exposure frames fusion solutions cannot be applied to real-time tasks, such as preview and video in smartphones.

The primary objectives of low light enhancement include:

1. Noise Reduction: Low light images tend to have a high level of noise, which appears as random grain or speckles. Reducing noise is a crucial aspect of enhancement.
2. Brightness and Exposure Adjustment: Increasing the overall brightness of the image to make it more visible without overexposing bright areas.
3. Contrast Enhancement: Enhancing the contrast to make objects and details more distinguishable.
4. Color Correction: Correcting color shifts and distortions that often occur in low light conditions.
5. Detail Preservation: Ensuring that important details in the image are preserved and not lost during enhancement.

Low light conditions can have a significant impact on the quality and clarity of images, leading to several undesirable effects. Here are some common ways low light conditions affect images:

1. Reduced Visibility: In low light, objects and scenes are often poorly illuminated, leading to reduced visibility. This can make it challenging to discern details and objects in the image.
2. Increased Noise: Low light images tend to have higher levels of noise, which appears as random variations in brightness and color. This noise can degrade image quality and make it look grainy.
3. Loss of Detail: Low light conditions can result in a loss of fine details in the image. This is particularly noticeable in textures, patterns, and small objects that may become less distinct.

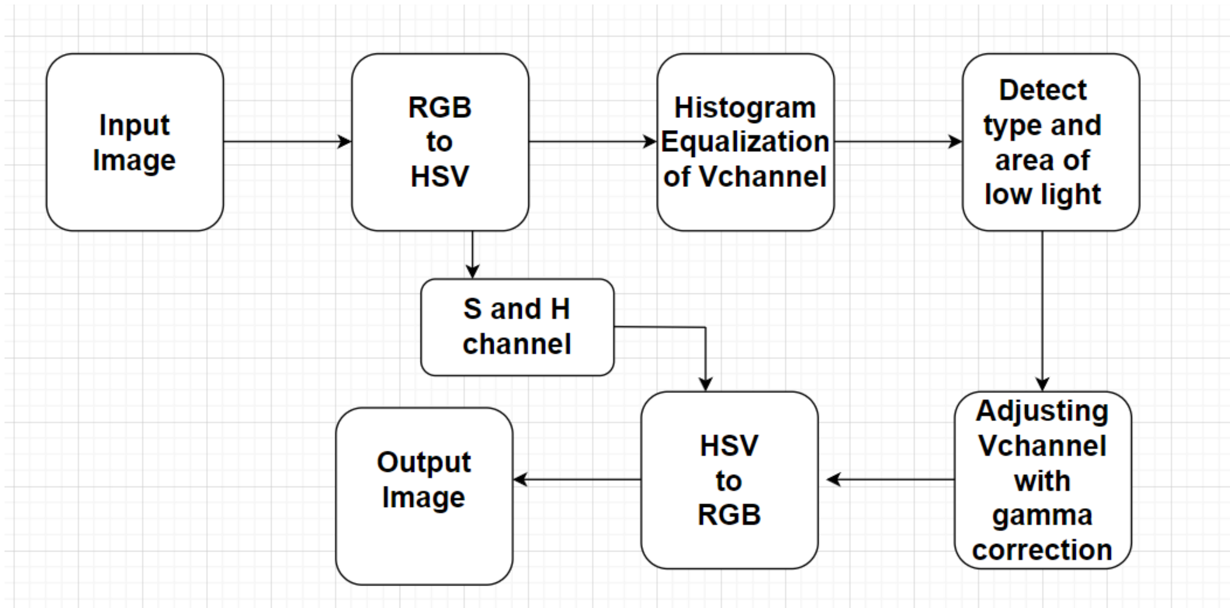
4. Contrast Reduction: Contrast is essential for defining the edges and features of objects in an image. Low light often leads to a reduction in contrast, making it difficult to distinguish objects from their backgrounds.
5. Color Shifts:Colors in low light images can appear distorted or shifted, leading to inaccuracies in the representation of the scene. This is due to variations in the spectrum of light available in low light.
6. Blurriness: Low light conditions can increase the likelihood of motion blur because longer exposure times may be required to capture enough light. This results in images that appear soft or blurry.
7.Loss of Color Saturation: Colors can appear desaturated and dull in low light, as the limited illumination affects the vibrancy of colors.
8. Artifacts: Low light images may exhibit artifacts like lens flare, glares, or halos, which can further degrade image quality.
9. Limited Dynamic Range: Low light conditions often reduce the dynamic range of an image, making it challenging to capture both bright and dark areas in the same frame.
10. Increased Sensitivity to Camera Settings: Low light images are more sensitive to camera settings like ISO, shutter speed, and aperture. Suboptimal settings can exacerbate the above-mentioned issues.[4]

LITERATURE TABLE:

Author	Name of paper	Journal	Published year
P. Banik, R.Saha, K. Kim	Contrast enhancement of low-light image using histogram equalization and illumination adjustment	International Conference on Electronics, Information, and Communication (ICEIC)	January 2018
W. Wang, X.Wu, X. Yuan , Z. Gao	An Experiment-Based Review of Low-Light Image Enhancement Methods	IEEE ACCESS	2022
H. Tang, H. Zhu, L. Fei, T. Wang, Y. Cao, C. Xie	Low-Illumination Image Enhancement Based on Deep Learning Techniques: A Brief Review	Photonics	2023
W. Kim	Low-Light Image Enhancement: A Comparative Review and Prospects	IEEE Access	2022

Z. Fu, M. Song, C. Ma, J. Nasti, V. Tyagi, G.Lloyd, W. Tang	An Efficient Hybrid Model for Low-light Image Enhancement in Mobile Devices		

BLOCK DIAGRAM:



[1]

METHODOLOGY:

1. Load the Image:
- First load an RGB image, 'church_low_light.png,' using the `imread` function. This image is the initial input for all the subsequent processing steps.

2. Display the Original Image and its Histogram:
- A graphical user interface (GUI) figure is created to display images and plots.
- The original low-light image is displayed on the left side of the figure, and its histogram is displayed on the right side. The histogram provides information about the distribution of pixel intensities in the image.

3. Histogram Equalization (HE) on Original Image:
- Histogram equalization is a process that redistributes the pixel intensities in an image to achieve a more uniform distribution. This step improves the contrast and overall visibility of the image.
- The result of this step is a new image where the pixel intensities have been adjusted through histogram equalization.

4. Display the Histogram-Equalized Image and its Histogram:
   - A new figure is created to display the histogram-equalized image.
   - The histogram of the histogram-equalized image is displayed. This allows you to visualize the changes in pixel intensity distribution that occurred due to the histogram equalization process.

5. Convert RGB Image to HSV Color Space:
   - The RGB image is converted to the HSV (Hue, Saturation, Value) color space. HSV separates an image into three components that represent the color and brightness information.
   - The HSV color space is particularly useful for image processing tasks that involve modifying color and brightness separately.

6. Display HSV Image and its Channels:
   - A new figure is created to display the entire HSV image.
   - Within the HSV color space, there are three channels: Hue (H), Saturation (S), and Value (V).
   - Each of these channels is visualized separately:
     - Hue (H) Channel: Displays the color information in the image.
     - Saturation (S) Channel: Represents the intensity of color in the image.
     - Value (V) Channel: Represents the brightness or lightness of the image.

7. Histogram Equalization on V Channel:
   - Histogram equalization is applied specifically to the Value (V) channel of the HSV image. This operation enhances the brightness of the image without affecting its color.

8. Convert Equalized HSV Image Back to RGB:
   - The HSV image with the equalized V channel is converted back to the RGB color space. This results in an RGB image with enhanced brightness while preserving the original colors.

9. Create Low-Light Mask:
   - A threshold value of 0.6 is defined to distinguish low-light areas from the rest of the image. A lower threshold value indicates a more restrictive definition of low-light areas.

10. Visualize Low-Light Areas on Original HSV Image:
   - The low-light areas, as defined by the threshold, are highlighted on the original HSV image. These areas are set to black, making it easier to identify regions that were considered low-light.

11. Convert Modified HSV Image Back to RGB:
   - The modified HSV image with the highlighted low-light areas is converted back to the RGB color space. This results in an RGB image with the low-light areas made black.

12. Gamma Correction for Brightness Enhancement:
   - Gamma correction is applied to the Value (V) channel of the HSV image using different gamma values (0.3, 0.4, and 0.5).
   - Gamma correction is a non-linear operation that can enhance or reduce image brightness based on the chosen gamma value. Lower gamma values enhance brightness more aggressively.

13. Calculate SSIM and PSNR:

- SSIM (Structural Similarity Index) and PSNR (Peak Signal-to-Noise Ratio) are metrics used to evaluate image quality.
- The code compares the gamma-corrected images (with different gamma values) to a reference image ('Bright_captured.jpeg') and calculates their SSIM and PSNR values. Higher SSIM and PSNR values indicate better image quality. [1][2][3]

ALGORITHM:

1. Input:
   - Load the original low-light RGB image 'church_low_light.png'.

2. Display Original Image and Histogram:
   - Create a GUI figure for visualization.
   - Display the original image on the left side of the figure.
   - Display the histogram of the original image on the right side of the figure.

3. Histogram Equalization (HE) on Original Image:
   - Apply histogram equalization to the original image.
   - Store the histogram-equalized image in a variable.

4. Display Histogram-Equalized Image and Histogram:
   - Create a new GUI figure for visualization.
   - Display the histogram-equalized image on the left side of the figure.
   - Display the histogram of the histogram-equalized image on the right side of the figure.

5. Convert RGB Image to HSV Color Space:
   - Convert the RGB image to the HSV color space.
   - Split the HSV image into three channels: Hue (H), Saturation (S), and Value (V).

6. Display HSV Image and Channels:
   - Create a new GUI figure for visualization.
   - Display the entire HSV image.
   - Display the individual channels: Hue (H), Saturation (S), and Value (V).

7. Histogram Equalization on V Channel:
   - Apply histogram equalization specifically to the Value (V) channel of the HSV image.
   - Update the Value channel with the equalized values.

8. Convert Equalized HSV Image Back to RGB:
   - Convert the HSV image with the equalized V channel back to the RGB color space.
   - The result is an RGB image with enhanced brightness.

9. Create Low-Light Mask:
   - Define a threshold value to identify low-light areas in the V channel.
   - Create a binary mask where low-light areas are marked as 1.

10. Visualize Low-Light Areas on Original HSV Image:
   - Highlight the low-light areas by setting the corresponding pixels to black (0) in the original HSV image.

11. Convert Modified HSV Image Back to RGB:
   - Convert the modified HSV image with low-light areas highlighted back to the RGB color space.
   - This results in an RGB image with blackened low-light areas.

12. Gamma Correction for Brightness Enhancement:
   - For different gamma values (e.g., 0.3, 0.4, and 0.5):
     - Apply gamma correction to the Value (V) channel of the HSV image.
     - Create a new HSV image with the adjusted V channel.
     - Convert the adjusted HSV image back to RGB.
     - Repeat this process for each gamma value.

13. Evaluate Image Quality:
   - Load a reference image ('Bright_captured.jpeg').
   - Calculate the SSIM (Structural Similarity Index) and PSNR (Peak Signal-to-Noise Ratio) for each gamma-corrected image compared to the reference image.
   - Print the SSIM and PSNR values to assess the image quality.

CODE:

```
% Load the RGB image
I = imread('church_low_light.png');
% Plotting original image and its histogram
figure(1);
subplot(1, 2, 1);
imshow(I);
title('Original Image');
subplot(1, 2, 2);
imhist(I);
title('Histogram of Original Image');
% HE(histogram equalization) on Original image
histImg = histeq(I);
% Plotting HE image and its histogram
figure(2);
subplot(1, 2, 1);
imshow(histImg);
title('Histogram Equalisation');
subplot(1, 2, 2);
imhist(histImg);
title('Histogram of Histogram Equalisation');
% Convert the RGB image to HSV
hsvImage = rgb2hsv(I);
% Display the entire HSV image
figure(3);
subplot(2,2,1);
imshow(hsvImage);
title('HSV Image');
```

```
hChannel = hsvImage(:, :, 1); % Hue channel
subplot(2,2,2);
imshow(hChannel);
title('hChannel');
sChannel = hsvImage(:, :, 2); % Saturation channel
subplot(2,2,3);
imshow(sChannel);
title('sChannel');
vChannel = hsvImage(:, :, 3); % Brightness channel
subplot(2,2,4);
imshow(vChannel);
title('vChannel');
% Perform histogram equalization on the V channel
equalizedVChannel = histeq(vChannel);
% Create a new HSV image with the equalized V channel
equalizedHSVImage = hsvImage;
equalizedHSVImage(:, :, 3) = equalizedVChannel;
% Convert the equalized HSV image back to RGB
equalizedRGBImage = hsv2rgb(equalizedHSVImage);
figure(4);
subplot(2,2,1);
imshow(equalizedRGBImage);
subplot(2,2,2);
imshow(equalizedHSVImage);
% Define a threshold to identify low-light areas
threshold = 0.6;
% Create a binary mask where low-light areas are 1
lowLightMask = vChannel < threshold;
% Visualize the low-light areas on the original HSV image
lowLightAreas = hsvImage;
lowLightAreas(lowLightMask) = 0;
% Set low-light areas to black
figure(5);
subplot(2,2,1);
imshow(lowLightAreas);
title('Modified HSV Image');
% Convert the modified HSV image back to RGB for visualization
resultRGBImage = hsv2rgb(lowLightAreas);
subplot(2,2,2);
imshow(resultRGBImage);
title('Modified HSV to RGB image');
% Increase gamma for TypeA (enhance brightness)
gamma = 0.4;
% Apply gamma correction to the V channel
gammaCorrectedV = vChannel.^gamma;
% Create a new HSV image with the adjusted V channel
adjustedHSVImage = hsvImage;
adjustedHSVImage(:, :, 3) = gammaCorrectedV;
% Convert the adjusted HSV image back to RGB
adjustedRGBImage_4 = hsv2rgb(adjustedHSVImage);
gamma = 0.3;
gammaCorrectedV_3 = vChannel.^gamma;
adjustedHSVImage = hsvImage;
adjustedHSVImage(:, :, 3) = gammaCorrectedV_3;
adjustedRGBImage_3 = hsv2rgb(adjustedHSVImage);
gamma = 0.5;
gammaCorrectedV_5 = vChannel.^gamma;
```

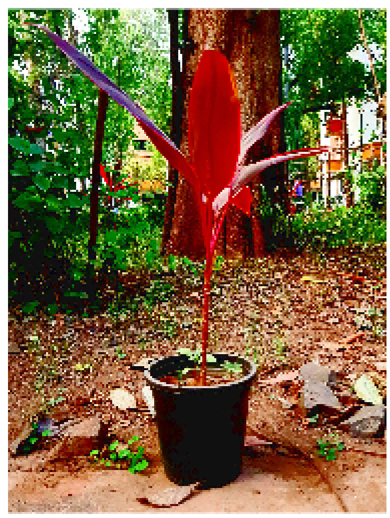
```
adjustedHSVImage = hsvImage;
adjustedHSVImage(:, :, 3) = gammaCorrectedV_5;
adjustedRGBImage_5 = hsv2rgb(adjustedHSVImage);
% figure(6);
%montage({I,histImg,equalizedRGBImage,adjustedRGBImage_3,adjustedRGBImage_4,adj
ustedRGBImage_5},Size=[1 6],BorderSize=5,BackgroundColor="w")
figure(6);
subplot(2, 3, 1);
imshow(I);
subplot(2, 3, 2);
imshow(histImg);
subplot(2, 3, 3);
imshow(equalizedRGBImage);
subplot(2, 3, 4);
imshow(adjustedRGBImage_3);
subplot(2, 3, 5);
imshow(adjustedRGBImage_4);
subplot(2, 3, 6);
imshow(adjustedRGBImage_5);
%%
I_grd=imread('Bright_captured.jpeg');
I_grd = im2uint8(I_grd);
adjustedRGBImage_3 = im2uint8(adjustedRGBImage_3);
adjustedRGBImage_4 = im2uint8(adjustedRGBImage_4);
adjustedRGBImage_5 = im2uint8(adjustedRGBImage_5);
%
% % Calculate SSIM
ssimValue_3 = ssim(adjustedRGBImage_3, I_grd);
ssimValue_4 = ssim(adjustedRGBImage_4, I_grd);
ssimValue_5 = ssim(adjustedRGBImage_5, I_grd);
fprintf('SSIM of 0.3 gamma: %.2f\n', ssimValue_3);
fprintf('SSIM of 0.4 gamma: %.2f\n', ssimValue_4);
fprintf('SSIM of 0.5 gamma: %.2f\n', ssimValue_5);
%
%%Calculate NIQE
A = imread('ground.png');
fprintf('NIQE Score of original image: %.2f\n', niqe(A));
fprintf('NIQE Score of equalized image: %.2f\n', niqe(equalizedRGBImage));
fprintf('NIQE    Score    of    gamma    corrected(0.3)    image:    %.2f\n',
niqe(adjustedRGBImage_3));
fprintf('NIQE    Score    of    gamma    corrected(0.4)    image:    %.2f\n',
niqe(adjustedRGBImage_4));
fprintf('NIQE    Score    of    gamma    corrected(0.5)    image:    %.2f\n',
niqe(adjustedRGBImage_5));
```

RESULTS:

Ground image

Input image

Histogram Equalized



HSV Equalized

Gamma corrected image (Gamma=0.3)

Gamma corrected image (Gamma=0.4)

Gamma corrected image (Gamma=0.5)

Low light image

Histogram equalized image

Equalized image

Gamma corrected image (Gamma=0.3)

Gamma corrected image (Gamma=0.4)

Gamma corrected image (Gamma=0.5)

APPLICATIONS:

1. Night Photography:
Enhancing photos taken at night, like cityscapes, starry skies, or outdoor events.
Example: Suppose you took a picture of a city skyline at night, and it appears dark and grainy. Low light enhancement can brighten the image, reduce noise, and reveal more details in the buildings and lights.[5]

2. Security Cameras:
Improving the quality of surveillance camera footage captured in low light environments.
Example: Security cameras often struggle to capture clear images at night. Enhancement can make it easier to identify people or objects in the dark.[5]

3. Astronomy:
Enhancing astronomical images of stars, planets, and deep space objects.
Example: Amateur astronomers take photos of the night sky. With low light enhancement, these images can bring out faint stars or distant galaxies that were barely visible before.[5]

4. Smartphone Photography:
Improving the quality of photos taken with smartphone cameras in low light conditions.
Example: You take a selfie in a dimly lit room, and the photo is too dark. Low light enhancement can brighten your face and surroundings, making the photo look better.[5]

5. Medical Imaging:
Enhancing medical images like X-rays or MRI scans taken in low light settings.
Example: A doctor needs to examine a low-contrast X-ray. Enhancing the image can make it easier to spot abnormalities or fractures.[5]

CONCLUSION :

Low-light image enhancement involves techniques to improve the quality and visibility of images taken in dim or poorly lit conditions. Through methods like histogram equalization, gamma correction, and identifying low-light areas, this process strives to enhance the overall brightness, and improve contrast in the images. By applying these methods we can reveal finer details and enrich the visual content. Evaluating these enhancements can be done using metrics like SSIM (Structural Similarity Index) and PSNR (Peak Signal-to-Noise Ratio) to quantitatively measure the improvement in image quality. This allows us to validate the effectiveness of the enhancements and ensures the resulting images are more visually appealing and comprehensible, even when initially captured in challenging low-light conditions.

REFERENCES:

[1] P. Banik, R.Saha, K. Kim, “Contrast enhancement of low-light image using histogram equalization and illumination adjustment” ,in *International Conference on Electronics, Information, and Communication (ICEIC)* , January 2018

[2] W. Wang, X.Wu, X. Yuan , Z. Gao, “An Experiment-Based Review of Low-Light Image Enhancement Methods”,in *IEEE ACCESS,* Vol 8,2022.

[3] H. Tang, H. Zhu, L. Fei, T. Wang, Y. Cao, C. Xie, “Low-Illumination Image Enhancement Based on Deep Learning Techniques: A Brief Review”,in *Photonics 2023*, Vol 10, 2023

[4] Z. Fu, M. Song, C. Ma, J. Nasti, V. Tyagi, G.Lloyd, W. Tang, “An Efficient Hybrid Model for Low-light Image Enhancement in Mobile Devices” .

[5]W. Kim, "Low-Light Image Enhancement: A Comparative Review and Prospects," in *IEEE Access*, vol. 10, pp. 84535-84557, 2022.