Low Light Image Enhancement Project

Introduction

In recent years, deep learning has revolutionised the field of image processing, particularly in enhancing low-light images. This report presents a project that implements a Convolutional Blind Denoising Network (CBDNet) to address the challenges posed by noisy images captured in low-light conditions and a image post processing method to tackle low light situations and colour correction. CBDNet is designed to first estimate the noise present in the image and then perform non-blind denoising to enhance the image quality. The model's performance is evaluated using the Peak Signal-to-Noise Ratio (PSNR) metric, a widely used indicator of image quality.

Project Details

Phase 1: Denoising Low Light Image

Data Preparation

The dataset used for training the model is the LOL (Low-Light) dataset, which includes paired low-light and normal-light images. The images are preprocessed to align and resize them to a uniform dimension of 512x512 pixels.

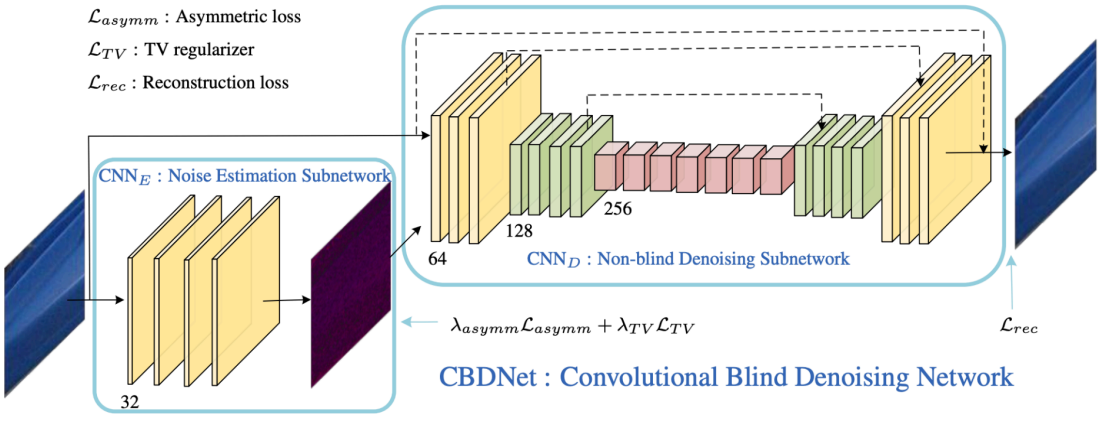
A function dim is defined to simulate low-light conditions by dimming the normal-light images. The dim function prepares the target dataset to the CBDNet model , as we have noisy low light image so for target image we use normal light image and reduce the value component of the hsv format of the image by a factor of 0.15 which gives us a clean low light image.

The images are then converted into arrays for model training.

Architecture and Specifications

The CBDNet architecture consists of two main subnetworks: the noise estimation subnetwork and the non-blind denoising subnetwork.

The noise estimation subnetwork is responsible for predicting the noise present in the input image. It consist of 5 convolution 2D layers with kernel size (3,3) and activation “relu”.



The non-blind denoising subnetwork uses this noise estimate along with the input image to produce a cleaner output.
The model is implemented using TensorFlow and Keras, leveraging a bottleneck architecture with skip connections to retain both low level and high level feature of the image.

As described in the diagram the model is composed of series of 2D convolution layers followed by average pooling with increasing feature maps and then transposed 2D convolutional layers with decreasing feature map to match the size of imput/target image.

The model is trained for 30 epochs and with batch_size 32 , using the Adam optimizer (with learning rate 10^-3) and Mean Squared Error (MSE) as the loss function. The training process involves generating inputs from the prepared dataset and feeding them into the model over multiple epochs.

Phase 2: Increasing Lighting and Colour Correction

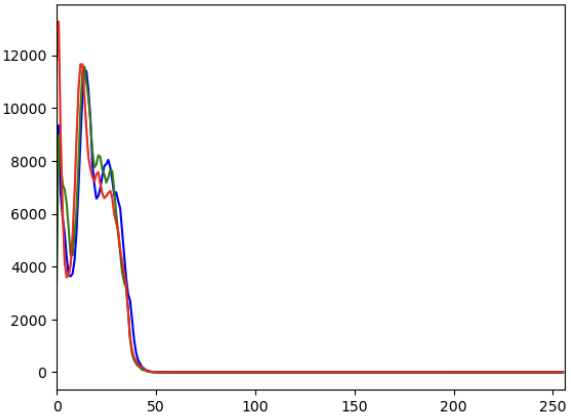
Image Preparation and Enhancement

The imgPrep function is designed to further process the denoised images by adjusting the pixel values to enhance visibility. This step is necessary to ensure that the resulting images have a more natural and clear appearance, especially in cases where the initial denoising might not completely address low-light conditions.

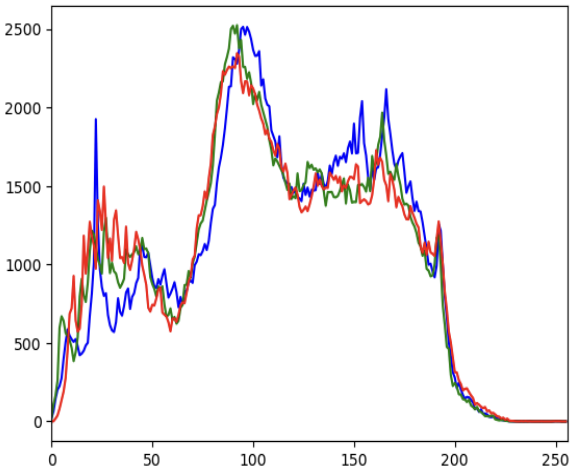
This function operates as follows:

1. **Reading and Resizing the Image**: The high-light image is read, converted from BGR to RGB, and resized to 512x512 pixels.
2. **Pixel Range Adjustment**: The function finds the pixel value thresholds for the red, green, and blue channels such that 99.7% of the pixels lie within these thresholds. It then scales the pixel values to enhance the image.
3. **Combining Channels**: After scaling and clipping the data to range [0 , 255] , the function merges the channels back together to form the enhanced RGB image.

RGB histogram for low light image
(Fig 1)



RGB histogram for normal light image
(Fig 2)



RGB histogram for low light image
after applying imgPrep
(Fig 3)

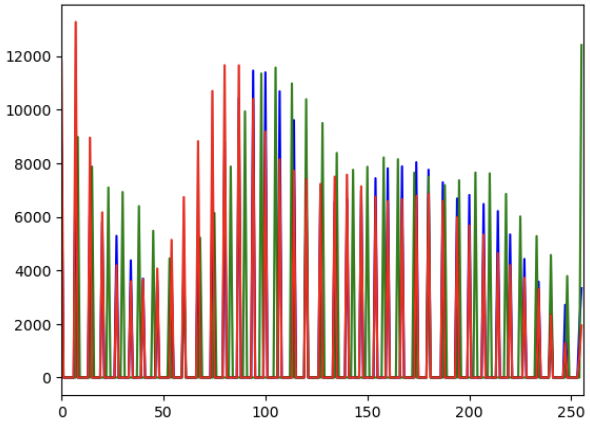


Figure 1 illustrates the RGB histogram for a low-light image, showing a concentration of pixel values towards the lower end of the spectrum, indicating underexposure. Figure 2 presents the RGB histogram for a normal light image, which has a more even distribution of pixel values across the spectrum, indicating a well-lit image. Figure 3 shows the RGB histogram for the same low-light image after applying the imgPrep function, which preprocesses the image to enhance visibility. Notably, the histogram in Figure 3 resembles that of the normal light image in Figure 2, demonstrating that the imgPrep function effectively adjusts the pixel values of the low-light image to better match those of a normally lit image. This transformation suggests a compression of the pixel values towards the middle of the spectrum, improving the overall visibility and quality of the low-light image.

Final Enhanced image

The processbef function plays a crucial role in enhancing low-light images using the trained CBDNet model. It involves multiple steps to preprocess the image, predict the denoised output, and further refine the enhanced image.

It performs two final procedures as follows:

Histogram Equalisation:

- The low-light image (xte1) is converted to the HSV colour space.
- The V (value) channel, which represents brightness, is separated and equalized using cv.equalizeHist to enhance the contrast.
- The HSV image is then converted back to the RGB colour space.

Image Blending:

- The refined denoised image (ypd) is blended with the histogram-equalized image (rgb) using cv.addWeighted. This step combines the benefits of both the denoising process and the contrast enhancement.

Weighted adding of histogram-equalized image gives the final colour correction to the enhanced image and hence provide us with final enhanced image.

Results

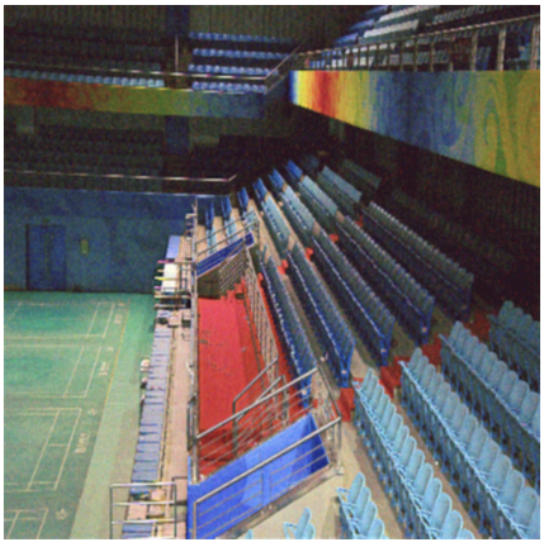
PSNR: 27.03 dB

Predicted Image

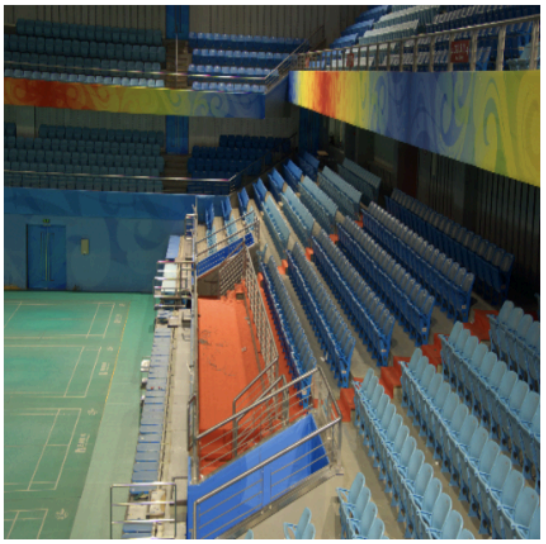
Target Image

PSNR: 23.05 dB

Predicted Image



Target Image



PSNR: 22.15 dB

Predicted Image

Target Image

PSNR: 24.24 dB

References:

- Guo, S., Yan, Z., Zhang, K., Zuo, W., & Zhang, L. (2019). Toward Convolutional Blind Denoising of Real Photographs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 1712-1722).

  Link to the paper can be found here