Enhancement of Night Vision IR Images

Aadesh Desai 19110116
Eshan Gujarathi 19110082
Saagar Parikh 19110199
Sanjay Venkitesh 19110200

Project Brief

Aim: To apply image enhancement algorithms on Night vision IR images using Vivado.

Algorithms:

1. Histogram Equalisation
2. Histogram Matching
3. Double Plateau Histogram Equalization
4. Top Hat Transform

These algorithms perform contrast enhancement on the image.

Dataset: ir:iricra2014 – ASL Datasets

Weekly Plan

1. **Week 1:** Implementing Histogram Equalisation and Histogram Matching on Python.
2. **Week 2:** Implementing Double Plateau Histogram Equalization and Top Hat Transform on Python.
3. **Week 3:**
    a. Compare the results of the different algorithms and select the ones that need to be implemented on Verilog.
    b. Reading images on Vivado
4. **Week 4:** Complete the Verilog implementation of the selected algorithms.

Sequence of tasks implemented

1. Implementing all four algorithms on Python.
2. Obtain an enhanced image using an inbuilt, advanced image processing algorithm on MATLAB.
3. Comparing the results obtained from the four algorithms with the enhanced image obtained on Python using PSNR values.
4. Reading the image on to Vivado.
5. Performing the selected algorithms (Histogram Equalization and Double Plateaus Histogram Equalization) on Vivado.
6. Convert the outputs obtained from Vivado back to an image using Python.

Division of Labour

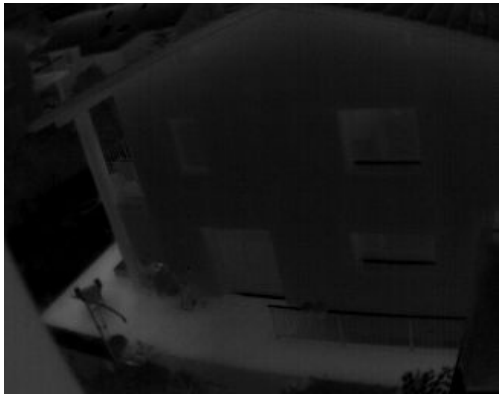
1. Converting images to binary files and reading them on to Vivado- **Sanjay**
2. Converting the output from Vivado into an image using Python- **Sanjay**
3. Debugging the code for both algorithms on Vivado- **Saagar**
4. Implementing Histogram Equalization on Vivado- **Aadesh**
5. Implementing Double Plateaus Histogram Equalization on Vivado- **Eshan**
6. Comparing the PSNR values for all images and finding the best performing algorithm - **Aadesh and Sanjay**
7. Performing Histogram Equalization and Double Plateaus Histogram Equalization on Python - **Saagar**
8. Performing Top Hat Transform and Histogram Matching on Python - **Eshan**

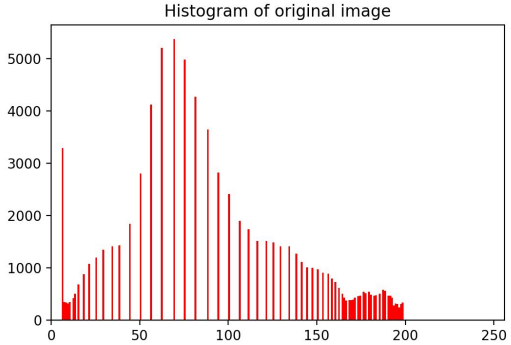
Histogram Equalization

- A contrast stretching algorithm with a mathematical function that uniformly stretches the image histogram .
- Calculate the probability and cumulative distribution function(CDF) of gray scale values, normalize the values by multiplying CDF values with the greatest gray scale level.
- Stretches the intensity values present in the image to the entire range of intensity values, thus increasing the contrast of the image.

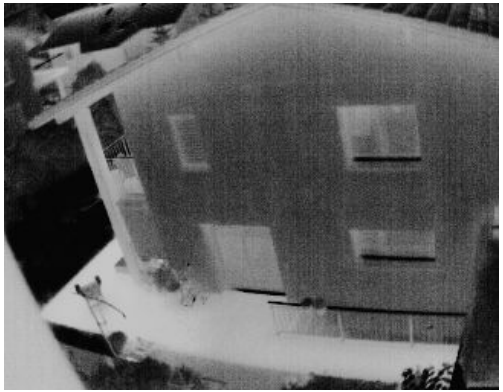
Sample Results

Original Image





Enhanced Image

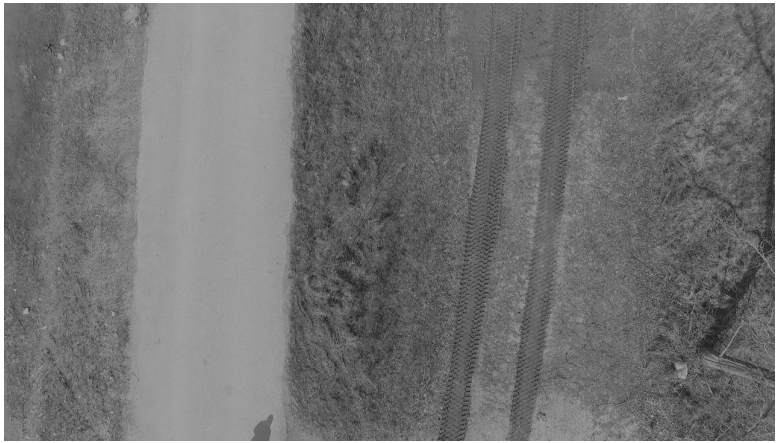


Histogram Matching

- Extension of histogram equalization.
- This algorithm transforms a target image so that it’s histogram matches with the histogram of a given reference image.
- Firstly, histogram equalization is performed on both, the target and the reference image, which is then followed by the matching part.
- The target image histogram is manipulated such that it matches the histogram of the reference image.

Sample Results

Target Image



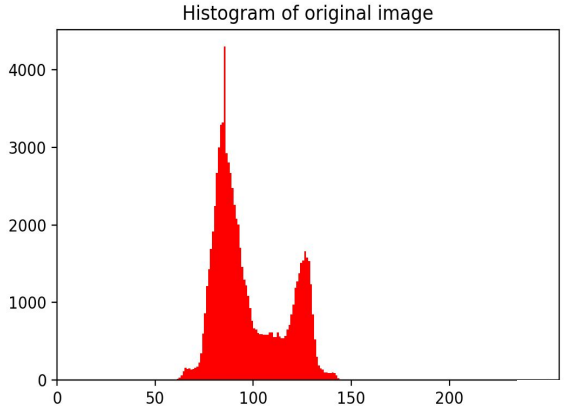
Target Image after Matching

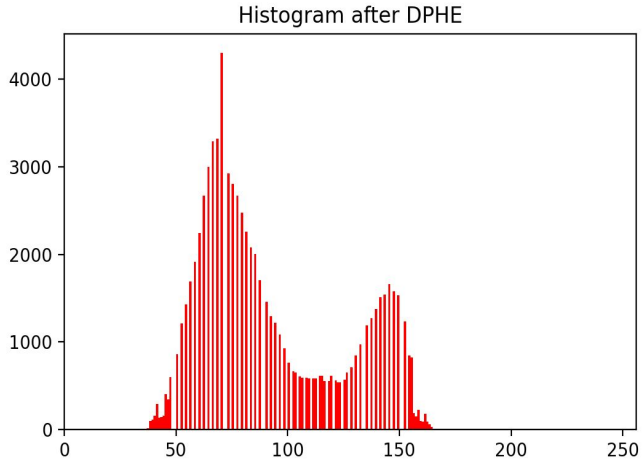
Reference Image

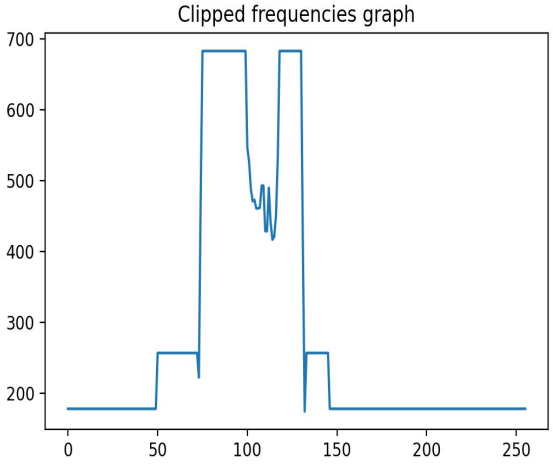
Double Plateaus Histogram Equalization

- In the image histogram, there are certain grayscale values (bins) which have exceptionally high frequencies while some have very low.
- Thus, we select two threshold values of frequencies, and clip the bins according to these thresholds.
- The excess pixels in a high frequency bin are redistributed into the bins having low frequencies.

Histograms

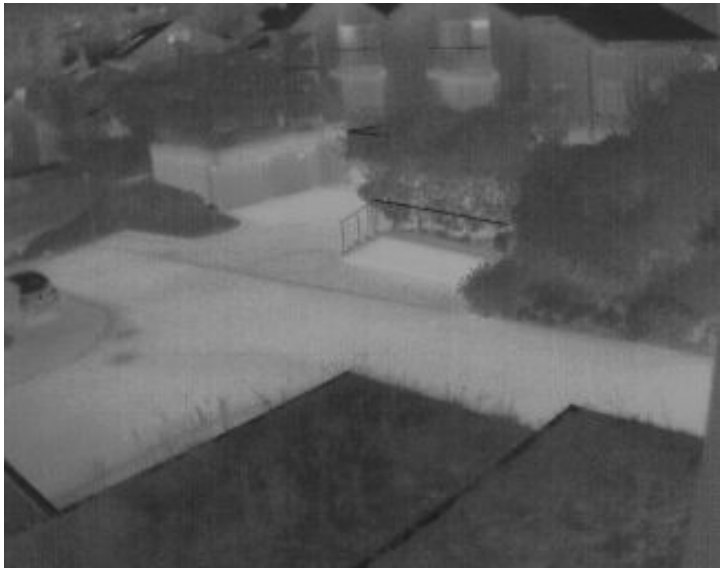

Histogram of original image
4000
3000
2000
1000
0
0
50
100
150
200


Histogram after DPHE
4000
3000
2000
1000
0
0
50
100
150
200
250


Clipped frequencies graph
700
600
500
400
300
200
0
50
100
150
200
250

Sample Results

Original Image



Enhanced Image

Top-Hat Transform

- In digital image processing, the algorithm extracts minute elements and details from the images.
- Considering structural elements, the filter enhances bright objects of interests in a dark background.
- Hence, it focuses on enhancing the light pixels in dark background.

Sample Results

Original Image

Enhanced Image

Comparing the performance of the Algorithms

- **Peak Signal to Noise Ratio (PSNR):** It uses the mean squared error of all the pixels between the two images to compute an expression.
- The **higher the PSNR value,** the **better the image enhancement**.
- We compare the PSNR value of all the 4 new images with an enhanced image obtained using an advanced, inbuilt image processing function on MATLAB.
- Based on the PSNR values, Double Plateaus Histogram Equalization and Histogram Equalization were the best performing algorithms.

Sample Results

- PSNR Value for Histogram Equalisation 14.13208046534746
- PSNR Value for Histogram Matching 13.721811068771785
- PSNR Value for Double Plateaus Histogram Equalisation 21.522595397489383
- PSNR Value for Top Hat Transform 9.8543439512946

Reading the images on Vivado

- Images need to be converted to binary text files to be read onto Vivado.
- The entire image is flattened to a 1 D array, each element contains the pixel’s intensity value in binary 8 bit format using Python.
- The file can be read using the readmemb function on Vivado.

Performing the algorithms on Vivado

- The image is read on the testbench and passed to the design module.
- The design consists of a single module:
  - It inputs the image as a 1 D array and converts it to a 2 D array of dimensions same as that of the original image.
  - The algorithm is now performed on this 2 D array that represents the image.
  - This image is again flattened to a 1 D image and sent as an output to the testbench.
- The output is received at the testbench and written into a binary file.
- This binary text file is converted to obtain the new image using python.

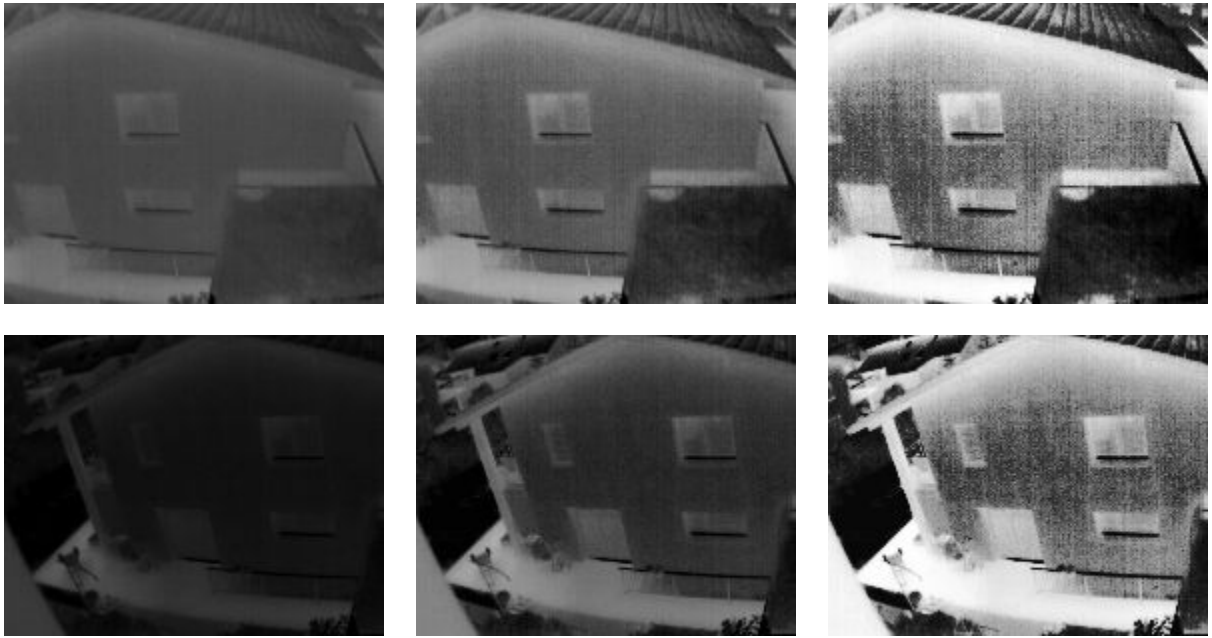
Comparing the algorithms on Vivado

Table containing the PSNR values of images obtained from both the algorithms for different images:

- After performing the algorithms on different images, we can see that the PSNR values are higher for Double Plateaus Histogram Equalization in all cases.
- Thus, Double Plateaus Histogram Equalization is the best amongst the ones that were used.

Image Name	PSNR after Histogram Equalization	PSNR after Double Plateaus Histogram Equalization
Image 1	12.703	23.120
Image 2	9.144	24.936
Image 3	14.333	18.820
Image 4	9.864	22.864

Sample Results



Original DPHE HE

Sample Results

Original DPHE HE

Challenges faced

- For the PSNR value calculations, we faced challenges in finding correct reference images to compare with our results. We decided to perform inbuilt MATLAB high level algorithms on the original image and used them as our reference images.
- We had to find and use a proper convention while converting images to binary files containing the stream of all bits and vice versa.
- We were not able to find proper functions for reading images onto Vivado

Future Scope

- If we are able to perform these algorithms on each frame in a video fast enough, we can get real time video enhancement.
- Applying these algorithms on Night vision goggles can improve the enhancement of images.