IMAGE PROCESSING REPORT

Kush Kapur
Durham University

This Project uses several Image Processing Algorithms to output a final video. The aim of this study is to create a video from the processed images, which when processed through the Yolo.py and compare images.py algorithm gives us highest possible Yolo and SSIM scores.

1. METHODOLGY

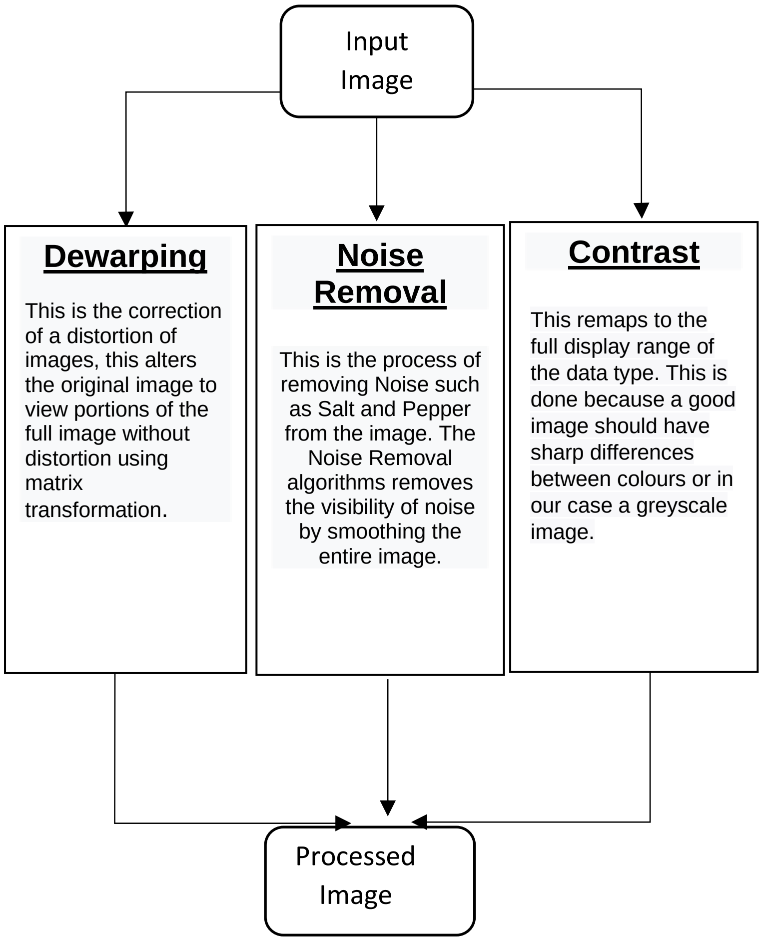
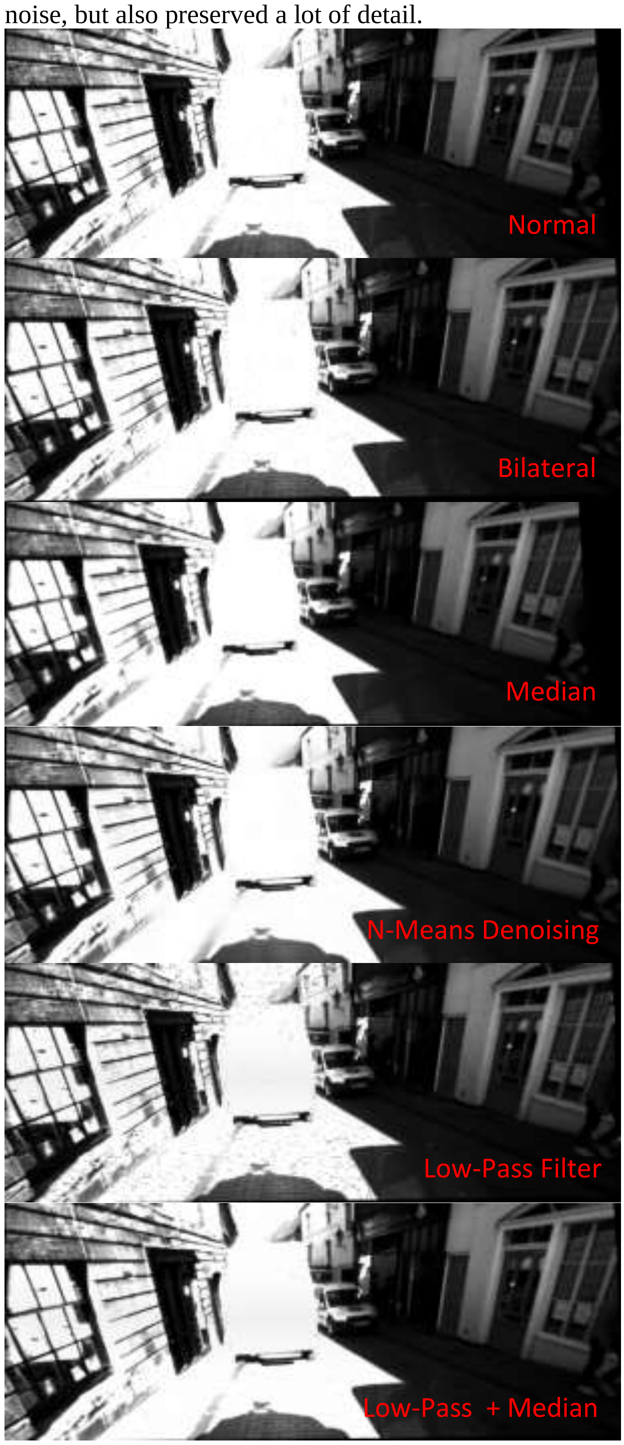


Fig 1: Image Processing Procedure

2. NOISE REMOVAL

For Noise Removal, a range of algorithms were tested. As the original image had Gaussian and Salt and Pepper Noise, it meant that one Filter would not be enough. From the images using only one algorithm, produces one of two results either it removes most of the noise, or it results in a loss of detail. For example, N means denoising removes almost all the noise, however in this process it also smoothens the image thus decreasing sharpening significantly. Similarly Bilateral preserves the sharpness but doesn't remove significant noise. Hence to ensure maximum sharpness and noise removal, Fourier Transform is used in conjunction with Median Noise Removal. Fourier Transform Is used because of high frequency in spatial noise. As this employs a moving window operator which affects one pixel of the image at a time, it preserves more detail than other algorithms which would affect neighbours of pixels resulting in over smoothing. This had to be used in conjunction with the median filter, as a low pass filter doesn't remove Salt and Pepper noise. This resulted in an image which had no noise, but also preserved a lot of detail.



3. BRIGHTNESS

An image is 'good' if it has pixel with intensities from all regions of the image. In this case histogram equalisation is used in CLAHE and Histogram equalization, where it stretches the histogram to either ends to improve contrast. Here it is evident that CLAHE provides best contrast, much better than Gamma and HE, providing more details especially in darker areas. Although a disadvantage is that CLAHE introduces noise.

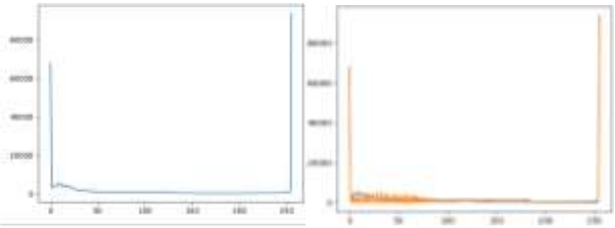
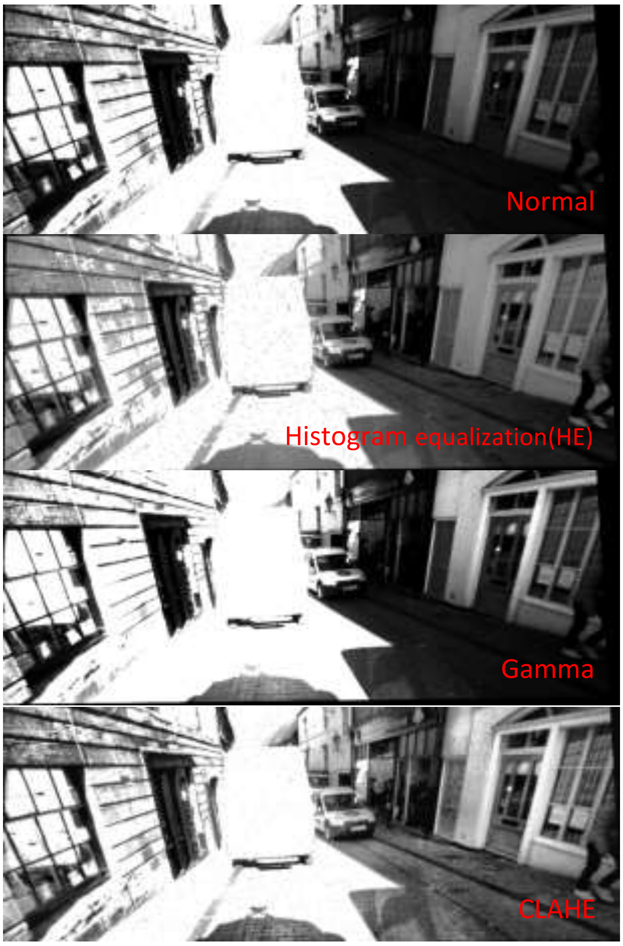


Fig 2: Histogram Equalisation

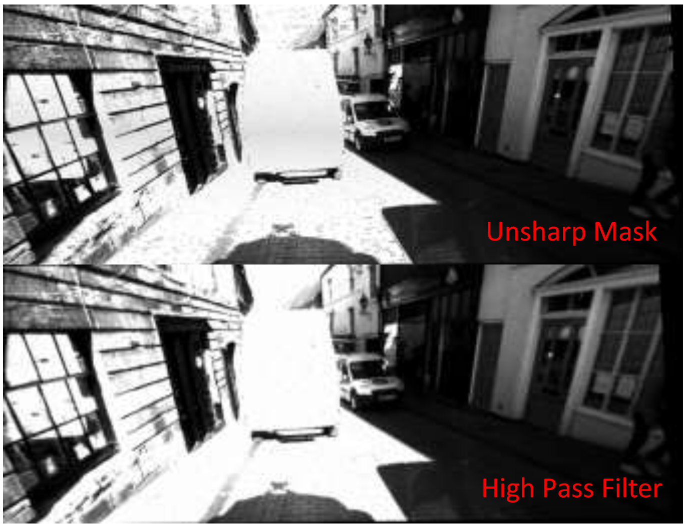


4. DEWARPING

Our images were slightly distorted; hence the full portion of the image was not visible. Hence, using get Perspective a Matrix was obtained to transform the original image to a full image without distortion.

5. SHARPENING

For sharpening, the superiority of High Pass Filter was clearer than Unsharp Mask, as unsharp mask subtracts a gaussian blue from the original image it is not as affective Fourier transformation.



6. CHOICE OF ALGORITHMS

For the final Algorithm, the sequence is Noise Removal, Sharpening, Dewarping and Contrast. In these methods Noise removal and Sharpening use Fourier Transform, to get a Discrete Fourier Transform after which a high and a low pass filter is applied. These Low and high pass filter images are then added to receive an image which has only salt and pepper noise and is sharpened. Due to the salt and pepper noise a median filter is applied after which the denoised image is Dewarped and Contrasted using a Gamma processing with r = 0.95. Gamma processing instead of CLAHE is performed as CLAHE reintroduces a lot of noise, also some images such as test102, had pixels with very high intensities, therefore r = 0.95 is used to decrease intensities.

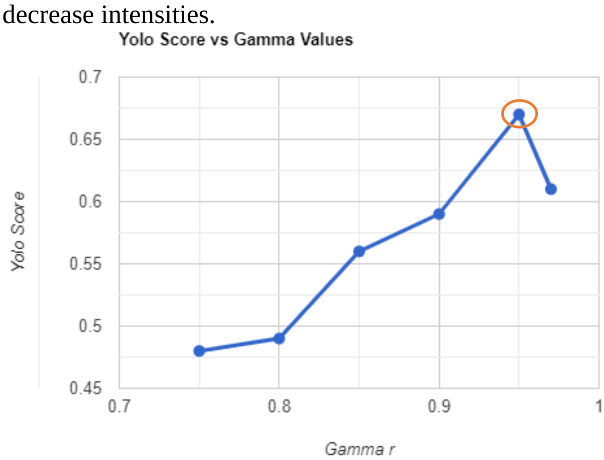


Fig 3: Gamma (R parameter) vs Yolo Score on Test Set

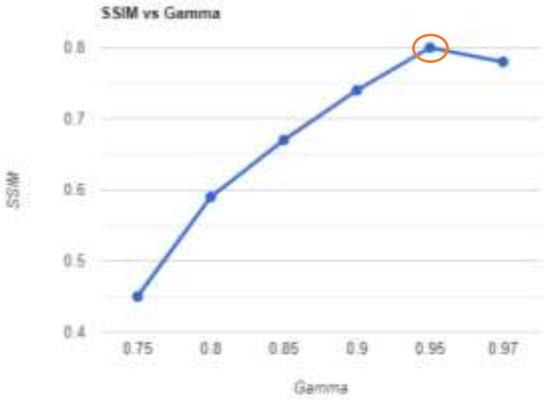


Fig 4: Gamma (R parameter) vs SSIM Score on Validation Set

Fig 5: CLAHE (Clip Size parameter) vs Yolo Score on Test Set

Fig 6: CLAHE (Clip Size parameter) vs SSIM Score on Validation Set

From this using Gamma is better than CLAHE, as it gives better SSIM scores on the validation set, and better Yolo Score on the test set.

5. TUNING

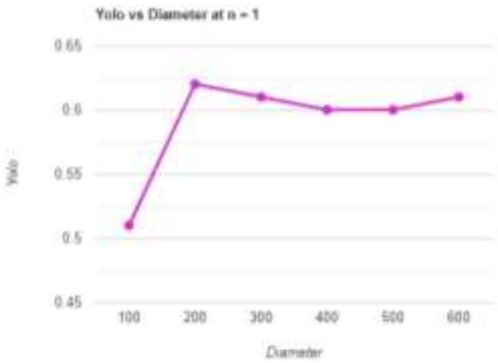


Fig 7: Yolo vs Low Pass Filter Diameter at N=1, Low pass filter For Noise Reduction

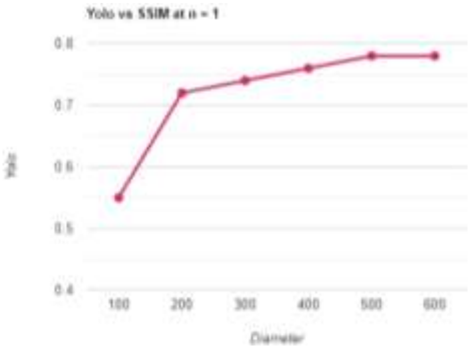


Fig 8: SSIM Score vs Low Pass Filter Diameter at N=1, Low pass filter For Noise Reduction

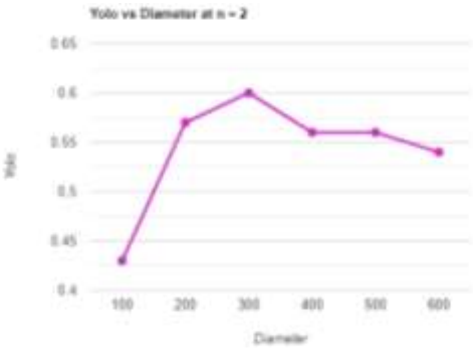


Fig 9: Yolo Score vs Low Pass Filter Diameter at N=2, Low pass filter For Noise Reduction

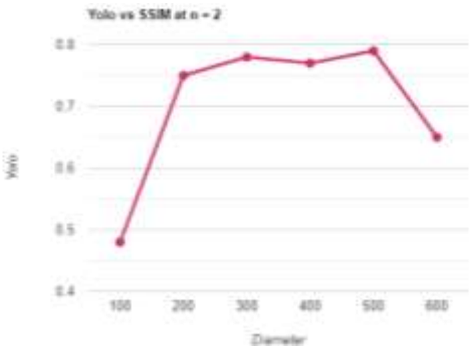


Fig 10: SSIM Score vs Low Pass Filter Diameter at N=2, Low pass filter For Noise Reduction

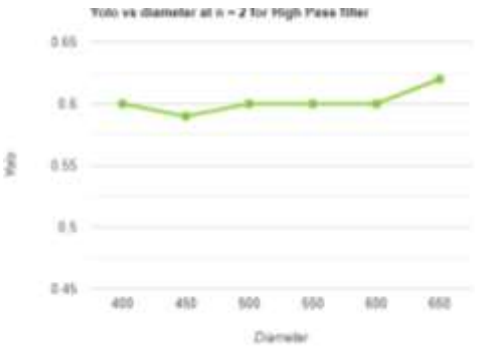


Fig 11: Yolo Score vs High Pass Filter Diameter at N=2, High Pass filter For Image Sharpening

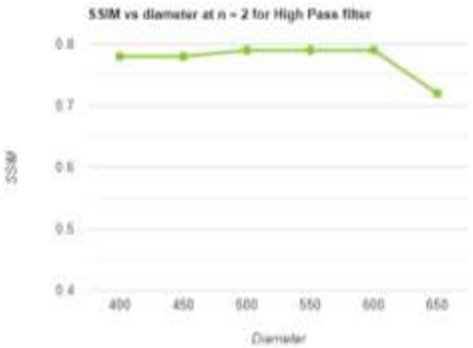


Fig 12: SSIM Score vs High Pass Filter Diameter at N=2, High Pass filter For Image Sharpening

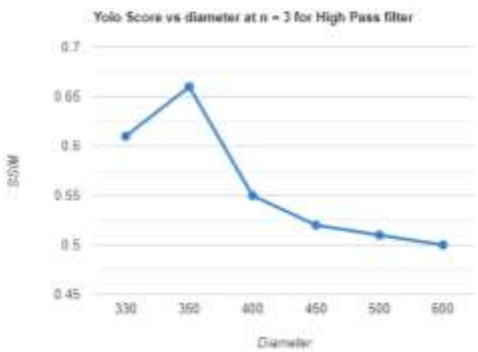
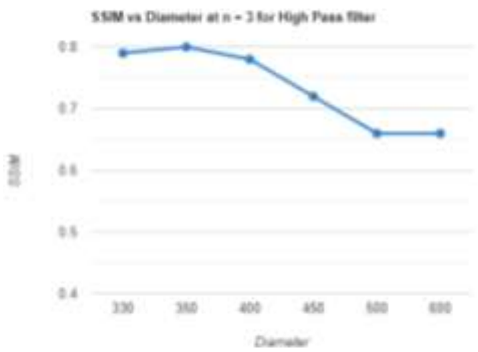


Fig 13: Yolo Score vs High Pass Filter Diameter at N=3, High Pass filter For Image Sharpening



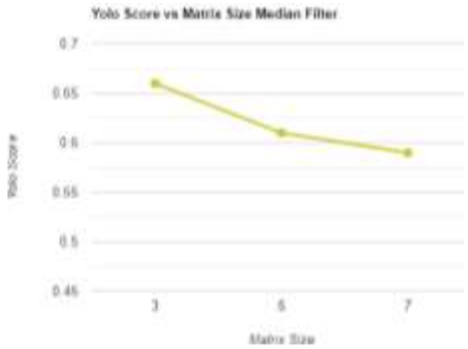


Fig 15: Yolo Score vs Median Filter Matrix Size, Median Filter for Salt and Pepper Noise Removal

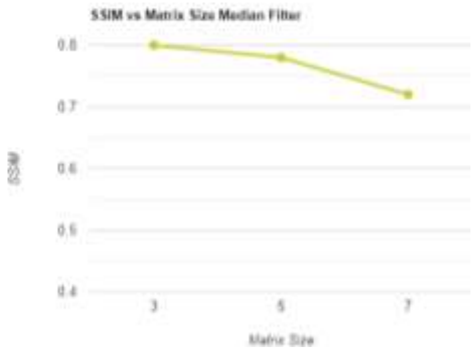


Fig 16: SSIM Score vs Median Filter Matrix Size, Median Filter for Salt and Pepper Noise Removal

BEST PARAMETERS

Low Pass Filter	Diameter = 400 N = 2
High Pass Filter	Diameter = 350 N = 3
Median Filter	Matrix Size = 3
Gamma	R = 0.95

OVERALL SCORE

Yolo Score (Test Set)	0.67
SSIM Score (Validation Set)	0.81

References

[1] Atapour GitHub https://github.com/atapour/ip-python-opencv/blob/main/low-high-pass-filter-fourier.py
[2] Open CV Documentation
[3] Dewarping https://eocortex.com
[4] Stack Overflow
[5] Skimage Documentation for unsharp mask