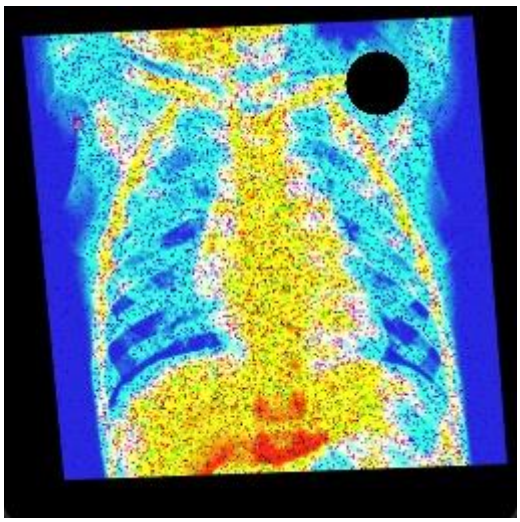
**Introduction**

This project aims to extract 100 X-Ray images to perform image processing on them. The expected result of the task is to increase the classify rate of these images when classifying them from being a healthy patient and a patient with pneumonia. I have created a python program to process the images and increase the accuracy of the classifier from 50% to 94%. I will detail the implementation process below.

**Program evaluation**

The program evaluation dissects the program step by step to give insight into what image enhancement have been implemented to reach the final images of the result images.



Firstly, I will show the original image of one of the images given by the client. There are few problems that can be identified from here. The image is clearly warped as there are blanks at the edges, noises such as gaussian and salt and pepper are significant, color is imbalanced, and a missing region can be seen at right top of the image. Theses problems are solved by following the steps below

*F*

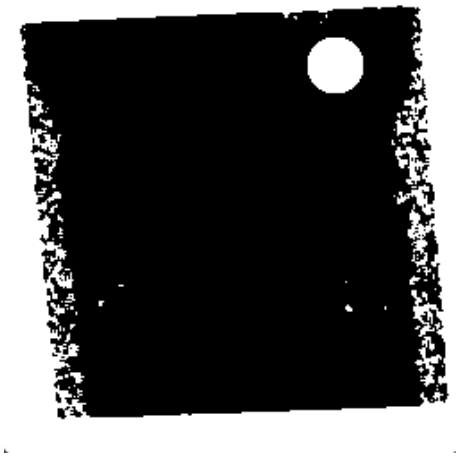


*Figure 2 image mask*

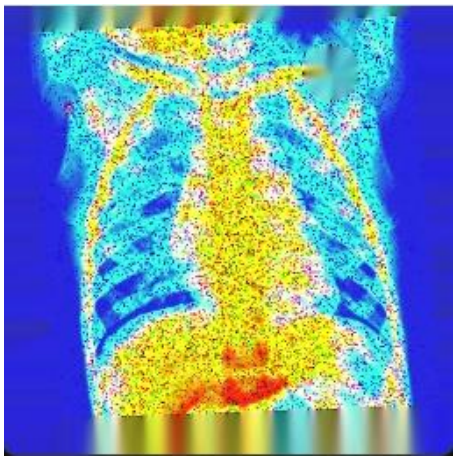


*Figure 1 image after imprinting*

The first step of the program is to solve the issue of missing regions by using imprinting techniques. From figure 1, we can see the mask that is identified in the program that will be used in the imprint process. From this figure you can see that the white areas is the targeted place that is going to imprint by the algorithm. I have specified in the program to have a threshold value of 60 to target pixels that have 60 pixel value higher than pure black (that has 255 as the value). This is to eliminate some noises in the image that is dark so that later during the denoising process it will have a clearer image to denoise. After creating this mask, the program will start the imprinting process to imprint the image, the results can be seen in figure 2.

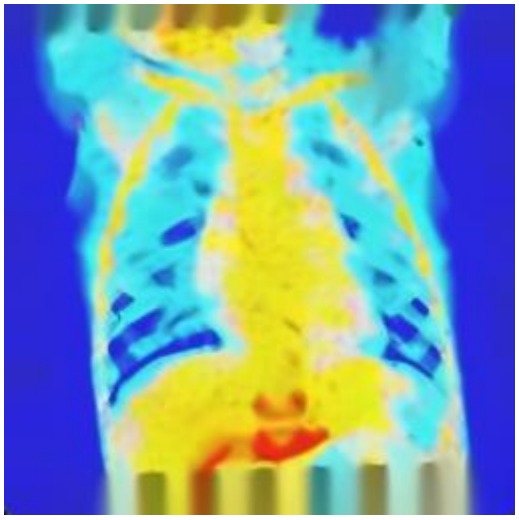


*Figure 3 image after denoise*

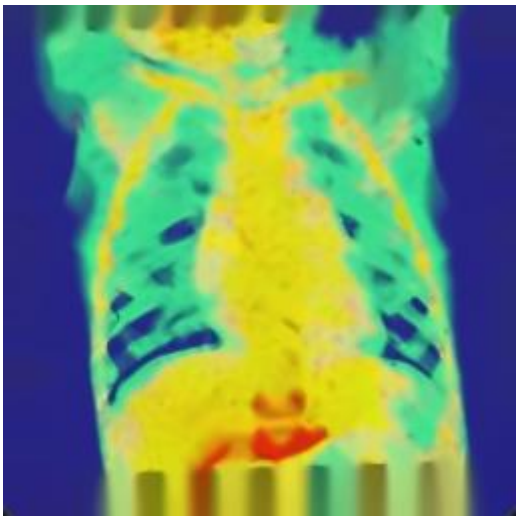


*Figure 4 image after colour balance*

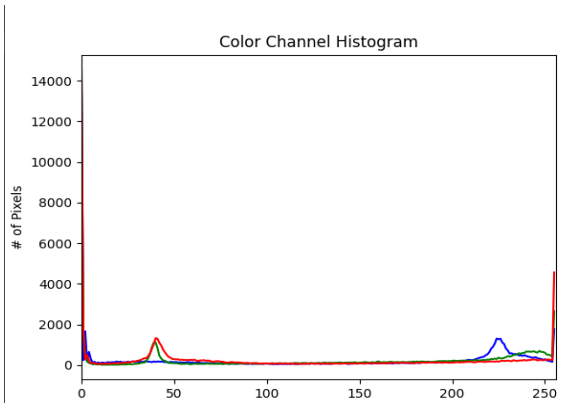



*Figure 5 colour channel before*

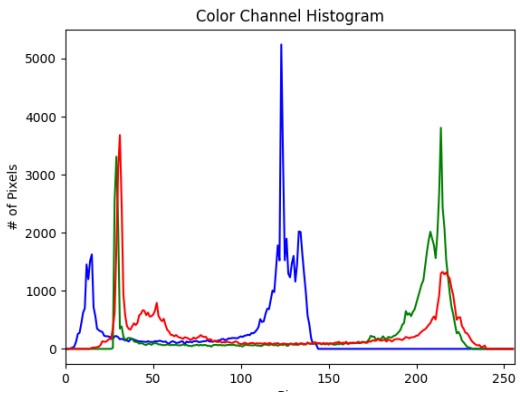



*Figure 6 colour channel after*

After applying imprinting, the program will then take the image through a denoise filter. It combines two denoising method, median blur and fast non-local means denoising. The median blur filter aims to remove most of the salt and pepper noises by taking the median of all the pixels in the neighbourhood of 3 pixels and replaced them with the median value. After that, it uses a non-local means denoising to remove the Gaussian noises. The idea behind this technique is to take similar patches of image together and average to remove the gaussian noises from the image. From figure 3, the program has successfully removed most noises from the image.

The next step of the program is to resolve the colour imbalance in the image, refer to figure 3 we can see that the image is too bright, and we cannot clearly see the details of the image such as the bone structure. In order to remedy this, the program will then adjust the colour channel (Red, Green, Blue) to achieve the darker version in figure 4. By looking into the histogram of the original image (Figure 5), we can see that the colour channels are out of range in both spectrums, by reducing the value of the red and green colour channel, the image is darker. However, as the blue in the background helps carry out the details of the image, the blue channel is brought down significantly more to do so. Figure 6 shows the colour channels after the colour channel changes, the colour channels are more brought together and the colour pixel out of range issue has been resolved.

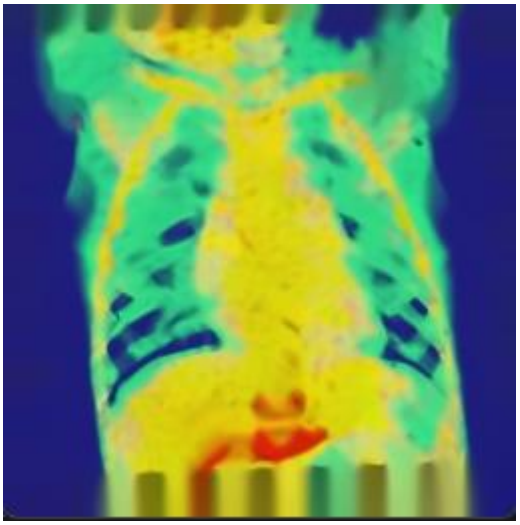


*Figure 7 gamma changes*

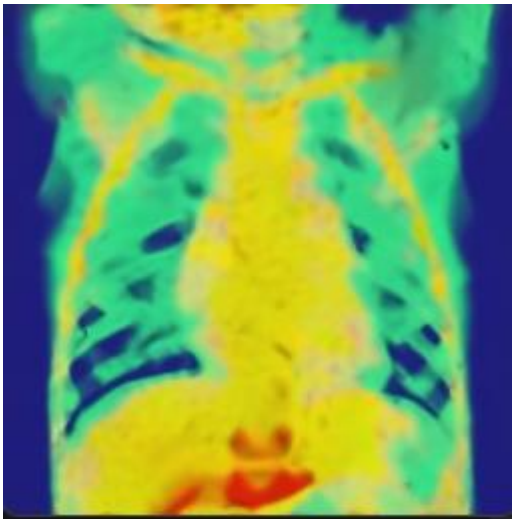


*Figure 8 warped image*

In Figure 7, the program changes the gamma of the image. This is mostly a function added for better visualization to the human eyes. Gamma controls the general intensity of the pixel value of the image, meaning that the brightness is affected by the gamma value. I have set the gamma level to a low value as a high value will significantly affect the classifier performance. Finally, figure 8 represents the final state of the image as the program warped the image by warping the perspective of the image in specified coordinates where is found by using Microsoft Paint.

**Additional information**

Program instructions and implementation details are available in the README text file. More information can be found the program specification.