Tuberculosis Detection Using Image Processing and Neural Network

Rand Barnawi*, **Taif Nono****, **Zainab Melaibari*****
*1911336, University of Jeddah, 1911336@uj.edu.sa
**1910922, University of Jeddah, 1910922@uj.edu.sa
***1911643, University of Jeddah, 1911643@uj.edu.sa

Abstract

Medical images - including tuberculosis(TB) x-ray images - usually need image pre-processing to facilitate disease diagnosis by doctors. This is done by displaying the regions of interest more clearly using different image pre-processing techniques. The aim of this project is to create an artificial intelligence model that can give initial predictions about the existence of TB. We use a dataset named "TB Chest Radiography Database" and use image pre-processing techniques to clarify any symptoms in the patient's lung. Then we create a model and trained it to diagnose TB. Also, we create an Application Programming Interface for our model to be used easily by doctors. We find that our model has achieved a high accuracy of 0.9884 in its prediction, and can be used as a trusted initial diagnosis for TB.

1 Introduction

For centuries, medicine has used medical imaging techniques to assist physicians in diagnosing diseases. They develop many types such as MRI and X-ray radiography. X-ray radiography is used to produce images of the body structures and tissues. The solid tissues, such as bone, appears white and the air appears black.[1] These images can be enhanced for example: resizing, orienting, and color corrections by "Image preprocessing".
One of the medical conditions that used X-ray radiography and need image preprocessing is Tuberculosis (TB). It is an infectious disease that is caused by a bacterium called Mycobacterium tuberculosis which usually attack the lungs[2].
Although it is easy to distinguish between normal and infected lungs in radiographs with the naked eye, sometimes it can be challenging and time-consuming. Radiographs could be unclear due to many reasons including salt and pepper noise, blurriness, etc. Furthermore, doctors have to look at different radiographs of different patients who have different conditions, which in turn can lead to missing important details because of exhausting their eyes. They need preprocessing techniques to show the regions of interest more clearly. Also, It is much easier to have an artificial intelligence model that can do all the job for them and gives an initial diagnosis for the presence of TB. To do so, the model needs to be trained on a large dataset that contains radiographs of normal and infected lungs. Thus, we decided to work on a project to detect TB through x-rays of the of the chest.

2 Dataset

This dataset contains CXR images of Normal (3500) and patients with TB (700 TB images in publicly accessible and 2800 TB images can be downloaded from NIAID TB portal by signing an agreement).

The TB database is collected from the source:

- NLM dataset: National Library of Medicine (NLM) in the U.S. [3] has made two lung X-ray datasets publicly available: the Montgomery and Shenzhen datasets.
- Belarus dataset: Belarus Set [4] was collected for a drug resistance study initiated by the National Institute of Allergy and Infectious Diseases, Ministry of Health, Republic of Belarus.
- NIAID TB dataset: NIAID TB portal program dataset [5], which contains about 3000 TB positive CXR images from about 3087 cases.

- RSNA CXR dataset: RSNA pneumonia detection challenge dataset [6], which is comprised of about 30,000 chest X-ray images, where 10,000 images are normal and others are abnormal and lung opacity images.

This database has been used in the paper titled “Reliable Tuberculosis Detection using Chest X-ray with Deep Learning, Segmentation and Visualization” published in IEEE Access in 2020.

3 Methodology

To preprocess the images for making good tuberculosis detection, we had followed these steps:

1. plot the histograms of the images with the cv2.calcHist function which has the following parameters:

   - images: This is the image that we want to compute a histogram for.
   - channels: A list of indexes, where we specify the index of the channel we want to compute a histogram for. To compute a histogram of a grayscale image, the list would be [0]. To compute a histogram for all three red, green, and blue channels, the channels list would be [0, 1, 2].
   - mask: Remember learning about masks in my Image Masking with OpenCV guide? Well, here we can supply a mask. If a mask is provided, a histogram will be computed for masked pixels only. If we do not have a mask or do not want to apply one, we can just provide a value of None.
   - histSize: This is the number of bins we want to use when computing a histogram. Again, this is a list, one for each channel we are computing a histogram for. The bin sizes do not all have to be the same. Here is an example of 32 bins for each channel: [32, 32, 32].
   - ranges: The range of possible pixel values. Normally, this is [0, 256] (that is not a typo — the ending range of the cv2.calcHist function is non-inclusive so you’ll want to provide a value of 256 rather than 255) for each channel, but if you are using a color space other than RGB [such as HSV], the ranges might be different.)

2. Make histograms equalization to see the equalized images. By using cv2.equalizeHist() function. Which is an OpenCV function that takes the grayscale image as an input

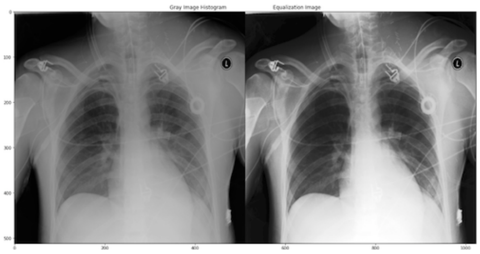

Figure 1: Gray Image on the Left and Equalized Image on the Right

   Now we can see the difference between the two images. The histogram equalization works well when you focus on a specific area. The above equalization gives a good improvement but not the best because it’s considering the global contrast of the image, as we can see in the upper right area of the chest x-ray the white lines, or as they are called lymph nodes have been softened, which we need to highlight to determine if the patient suffers from tuberculosis or not. So histogram equalization will not work well in places where there are large differences in density and where the graph covers a large area. To solve the histogram equalization problem we will use CLAHE(Contrast Limited Adaptive Histogram Equalization). CLAHE divides the image into small regions called (tiles) to avoid the global contrast of the image and then applies the Histogram Equalization for each tile to enhance the contrast. if there’s a noise in one of the tiles it will amplify. to avoid this problem OpenCV uses a contrast limit. "If any histogram bin is above the specified contrast limit, those pixels are clipped and distributed uniformly to other bins before applying histogram equalization. After equalization, to remove artifacts in tile borders, bilinear interpolation is applied". [7]

3. Use CLAHE function for contrast limit function Parameters: clipLimit Threshold for contrast limiting. tileGridSize Size of the grid for histogram equalization. Input images will be divided into equally sized rectangular tiles. tileGridSize defines the number of tiles in rows and columns.[8]

   As we can see the CLAHE function gives us a better result and its highlights the lymph nodes that the normal Histogram Equalization obliterated.

4. inverting the pixels of the images

   Now the Images are ready for detection

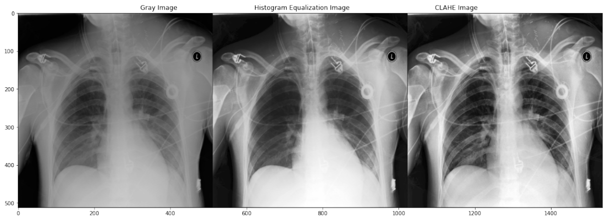

Figure 2: Gray Image on the Left, Equalized Image on the Middle, and Contrast Limited Histogram Equalization on the Right

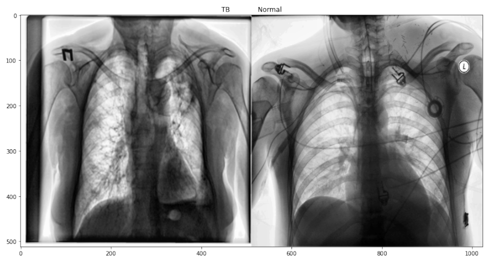

Figure 3: After Inverting the Infected chest on the Left, and healthy one on the Right

5. Use VGG16 pre-trained model to make binary classification.

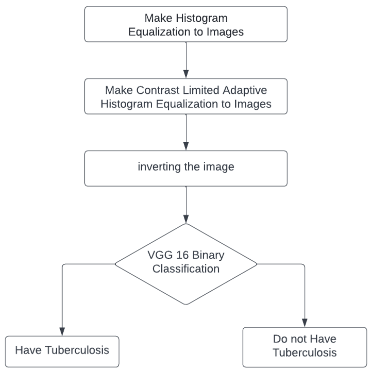

Figure 4: Methodology Flowchart

4 Neural Network Model and the Result

We used VGG16 pre-trained model which is one of the most excellent vision model architecture nowadays. The model has 21 layers such as Convolution layer, MaxPooling and Flatten layer at the end with Softmax activation function [1] github repository [2].

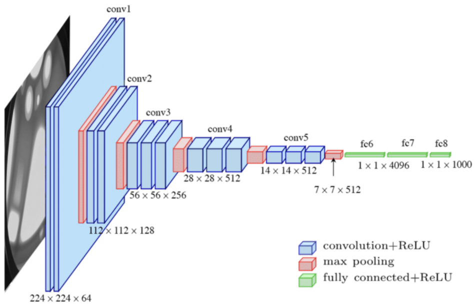

Figure 5: VGG16 model architecture

$\text{train}_{l}oss$	$\text{train}_{a}ccuracy$	$\text{val}_{l}oss$	$\text{val}_{a}ccuracy$
0.1414	0.9884	0.0398	0.9976

5 Conclusion

In conclusion, we hope that our model will help to reduce the time and effort needed in diagnosing TB. Depending on the accuracy that we obtained after testing, the model achieved high accuracy. In the future, the scope of the project can be expanded, the model may be re-trained to diagnose different lung diseases.

References

[1] E Seleþchi and OG Duliu. Image processing and data analysis in computed tomography. In *7th International Balkan Workshop on Applied Physics*, pages 5–7, 2006.

[2] Adnan Fojnica, Ahmed Osmanović, and Almir Badnjević. Dynamical model of tuberculosis-multiple strain prediction based on artificial neural network. In *2016 5th Mediterranean Conference on Embedded Computing (MECO)*, pages 290–293. IEEE, 2016.

[3] Stefan Jaeger, Sema Candemir, Sameer Antani, Yì-Xiáng J Wáng, Pu-Xuan Lu, and George Thoma. Two public chest x-ray datasets for computer-aided screening of pulmonary diseases. *Quantitative imaging in medicine and surgery*, 4(6):475, 2014.

[4] B. P. Health. (2020). Belarus tuberculosis portal [online]. available:. URL: `https://ieee-dataport.org/documents/tuberculosis-tb-chest-x-ray-database`.

[1] https://share.streamlit.io/rand-rb/tbdetection/main/TB$_{s}treamlitApp.py$

[2] https://github.com/Rand-RB/TBDetection

[5] Niaid tb portal program dataset [online]. available:. URL: https://tbportals.niaid.nih.gov/download-data.

[6] URL: https://tbfacts.org/dying-tb/.

[7] URL: https://docs.opencv.org/4.x/d5/daf/tutorial_py_histogram_equalization.html.

[8] URL: https://docs.opencv.org/3.4/d6/db6/classcv_1_1CLAHE.html.