IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.301

Volume 3, Issue 7, April 2023

COVID-19 Detection using Deep Learning

Shani Deo Pandey[1], Gautam Sharma[2], Chandra Shekhar[3], Aditya Chauhan[4], Ms. Shailja Varshney[5]

Department of Computer Science andEngineering[1,2,3,4,5]

ITS Engineering College, Greater Noida, Uttar Pradesh, India

Abstract: *The COVID-19 pandemic has had a significant impact on public healthcare worldwide, and medical imaging techniques like computed tomography (CT) have emerged as a potential alternative to RT-PCR as a screening method. However, the limited availability of COVID-19 imaging data has made it challenging to develop effective automated picture segmentation methods for quantitative assessment and disease monitoring. To address this issue, deep learning techniques have been employed for picture segmentation and classification on CT scans of the lungs. The proposed method utilizes an infection segmentation model that uses the U-NET model to identify infected areas and classify patients as COVID-19 positive or negative. The segmentation of infections and lungs in the suggested method is achieved by preprocessing the images to enhance contrast and remove irrelevant background elements. The dice similarity coefficient is used to evaluate the performance of two different U-NET models. The results demonstrate that the proposed method outperforms existing alternatives, advances the state of the art in COVID-19 segmentation, and improves medical image analysis with limited data availability. Therefore, deep learning techniques offer a promising approach for automated picture segmentation in medical imaging, particularly in the context of COVID-19*.

Keywords: Convolutional Neural Network, CLAHE Contrast Enhancement, Dice Similarity Coefficient, Infection Segmentation, lung segmentation, U-NET Model Architecture

I. INTRODUCTION

In this project of lungs classification and segmentation for COVID-19 CT-scans,The project objective is to differentiate between COVID-19 positive and negative patients using machine learning and then give their respective infection segmentation mask as well so that we can clearly see the infected areas of COVID-19 positive patients. The project mainly focuses on image classification and segmentation. For that purpose we have used different models for training the data set. The objective of the project is to approach or build an effective model which can classify the patients in COVID-19 positive and negative using their lungs' CT Scan analysis. The project also focuses on lung and infection segmentation to estimate the percentage of infection or the harm caused to lungs in the COVID-19 positive patients. In order to achieve desired results it is necessary to achieve following objectives in order:

Selecting Appropriate Dataset: Firstly an appropriate CT scan dataset needs to be selected which matches the conditions of our project, i.e the dataset size should not be too small. It should cover all age groups for better accuracy. The dataset should have equal representation of positive and negative CT scans.

Selecting best performing techniques to pre-process the given dataset: Next objective is to basically use the best performing techniques to analyze and pre-process the dataset such that the processed data does not contain any noise and gives good results.

Selecting the best algorithm for image classification: In this objective we need to find the best classification model/algorithm to differentiate between COVID-19 positive and negative patients by making two groups so that we can apply lung and infection segmentation.

Selecting best model for lung and infection segmentation: This objective is to find the best algorithm or model to segment the lungs from the given CT scan images and further highlight the infected area from the lungs segmentation image.

II. BACKGROUND

Research has suggested the identification of COVID-19 against numerous radiological techniques, with a variety of restrictions. For example, X-ray, CT, and MRI have all been used as imaging modalities in the diagnosis of COVID-19. There has been many research done on COVID-19 detection using machine learning techniques, some of which are using SVM, Regression Approaches, Random Forest Algorithm and deep learning as described below.

Various machine learning models have been employed for the classification and detection of COVID-19 cases. Zhang et al. [1] used Support Vector Machine (SVM) model to validate the performance of clinical data, blood test results, and urine test results. Their simulation results showed that SVM model was effective and achieved accuracy, sensitivity, and specificity values of 81.48%, 83.33%, and 100%, respectively. Brinati et al. [2] proposed seven machine learning methods to detect COVID-19 using routine blood tests taken from 279 individuals. The Random Forest algorithm was found feasible and successful with accuracy, precision, sensitivity, specificity, and AUC of 82%, 83%, 92%, 65%, and 84%, respectively. Perumal et al. [3] used deep CNN models and Haralick features for classification and achieved an average accuracy of 93%, precision of 91%, and sensitivity of 90% using a dataset of CT pictures gathered from multiple sources.The deep learning methods, including various types of neural networks such as CNN, RNN, and GAN, are considered the best performing techniques for classification. For the diagnosis and evaluation of COVID-19 on CT scans, Wang et al. [4] recommended using COVID-19Net, which performed well with an AUC of 87%, accuracy of 78.32%, sensitivity of 80.39%, F1 score of 77%, and specificity of 76.61% in a simulation using data from chest CT scans collected from six cities or provinces, including Wuhan, China.

The need for accurate image segmentation for COVID-19 diagnosis and assessment is critical. However, the limited availability of annotated imaging data makes it challenging to identify various lesion shapes, textures, and localizations. To address this issue, different strategies have been proposed, including segmentation and classification. Deep learning architectures such as Inception-v3, ResNet, DenseNet, and VB-Net have shown promising results, but U-Net and its variations are the most widely used and accurate deep learning models for segmentation. U-Net versions have demonstrated acceptable performance on 2D datasets of sufficient size, but semi-supervised learning methods have been employed to overcome small dataset sizes. Attention mechanisms have also been integrated into the traditional U-Net architecture. These techniques have shown potential to enhance both supervised and unsupervised training on labeled and unlabeled data, and the development of unique neural network designs for small dataset sizes.

III. PROPOSED WORK

In our proposed model we first demonstrated the databases used in it before going over the associated jargon and techniques. The pre-processing method we utilized in this research, the convolutional neural network we used for the classification problem, and the U-net model architecture unit we used for segmentation. We have discussed feature extraction as well as the particulars of model construction and application. Fig:1

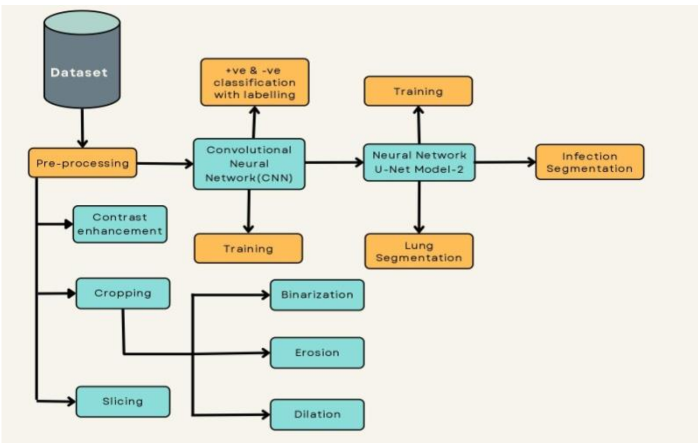

Fig: 1: Model

Step1 Selecting Appropriate Dataset

Step2 Selecting best performing techniques to pre-process the given dataset

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 7, April 2023

Step3 Selecting best algorithm for image classification
Step4 Selecting best model for lung and infection segmentation

A. Dataset

The dataset utilized in this study is a publicly available dataset from Kaggle [8][9], consisting of 20 annotated COVID-19 chest CT volumes. These volumes are critical for assessing the severity of the patient's disease and assisting in COVID-19 diagnosis. Accurate identification of COVID-19 indications and characterization of its findings can be particularly beneficial in areas with a shortage of qualified radiologists for effective diagnosis and treatment. The dataset includes properly segmented images of the lungs and infections in 20 CT scans of individuals diagnosed with COVID-19. Figure 2 provides a sample of the CT images from the dataset, with clear segmentation of the COVID-19 infection.

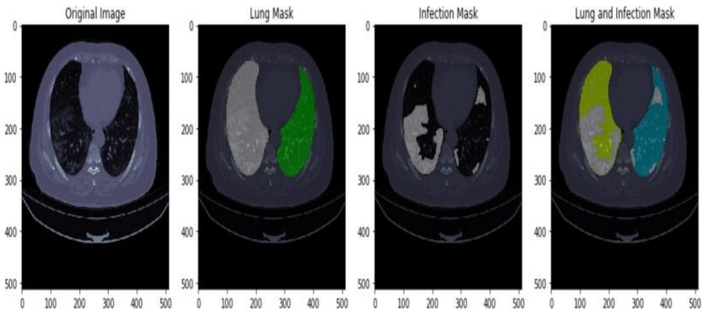

Fig:2 :Chest CT's Images from datasets

B. Data Pre-processing

In order to speed up the process of pattern discovery and model fitting, we applied a number of preprocessing approaches to the dataset. Two steps in the data pre-processing process include contrast enhancement and trimming the CT-scan pictures to only show the lung area. To do this, we used CLAHE enhancer for contrast improvements. Before using biomedical imaging techniques like erosion and dilation to crop the images, we first binarized the images. The CT scan was cropped using K-means, and clusters were found

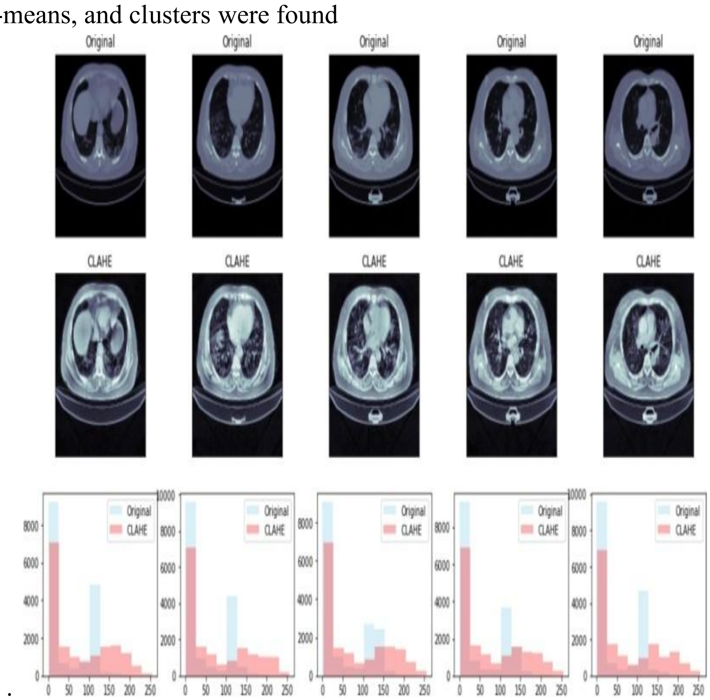

Fig 3 : Improve contrast CT images

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 7, April 2023

In order to improve the contrast of the CTs images, we have used a function called create CLAHE from open CV library to balance out the histogram in a uniform way. Basically, we'll strive to increase the contrast so that dark areas can be seen and lighter areas can be clearly seen in a dynamic fashion. We can clearly see the result of create CLAHE function and the difference between original image and after improving contrast in Fig:3.

The region of the lung that is of interest for our project. If there is any region outside of it that is useless, we will crop the image and only keep what we are interested in Fig:4.

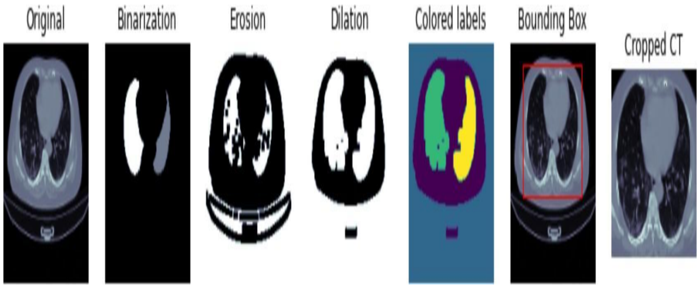

Fig:4:Methods used to get cropped CT scans

Now we will apply the same technique of improving contrast and cropping image to get the desirable region of interest on lung and infection mask given in the dataset. After that we will get our final processed data that you can see in Fig: 5.

It's crucial to remove the blank infection mask so that our model can accurately forecast the appropriate infection mask. Here, we will remove all of the black infection masks, ct scans, and lung masks that go with them. To do this, we'll use np unique to find the distinct elements; if its size is 1, the mask is empty; and then we'll put the outcome in an index array list then use del to delete the index on CT scans, lung masks, and blank infection masks after sorting the index arr array in descending order

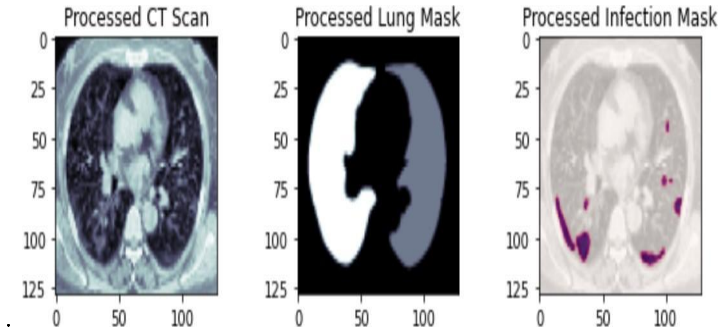

Fig 5: Final visualization of CT scans dataset

Slicing is done in order to convert a 3D image dataset to 2D images. For Focusing on Lungs we have maintained the gap of 20 Percent from front as well as back.

C. Image Classification

The next step in our project involves image classification to group the data into positive and negative patients, along with labels. This classification will help differentiate between positive and negative patients and aid in infection segmentation of positive patients. To perform this classification, we have developed a CNN model and trained it using our pre-processed data. CNNs are a type of neural network specifically used for processing image data in the form of a 2D matrix. In our case, the CNN model is used for the detection and classification of COVID-19 images. Based on the infection masks, the pre-processed 2D CT images are classified as COVID-19 positive or negative.

The trained CNN model produces a value between 0 and 1, which is transformed into binary values using a threshold determined using the ROC curve. The confusion matrix is used to compute precision, recall, and the F1 score. The steps used in classification include loading the pre-processed data, creating labels, splitting the data, defining the neural network, compiling the model, training the model, plotting the accuracy and loss graph, predicting on the test dataset, and calculating precision, recall, and F1 score. The model details are shown in Fig. 6.

D. Lung segmentation

Now our next task is to mark or highlight the lungs part in the pre-processed CTs images so that we can clearly see the lung. A CNN model is trained using the preprocessed 2d ct-scans and lung masks to identify the lung area in the ct-scans. For this task, a U-net is used. This model is used for segmentation problems and it is very effective and accurate. It's a basically advanced version of a traditional convolution neural network having some extra functionality. The model's effectiveness is assessed using the dice coefficient. In order to alter the learning rate during training, exponential decay is used. The steps we used in segmentation are - 1) Loading the pre-processed data, Creating labels, splitting the data, Defining loss and accuracy function, Defining the U-NET neural network, compiling the model ,Training the model 7) Plot the accuracy and loss graph 8) Predicting on the test dataset.

Infection segmentation The primary goal now is to mark or emphasize the infection portion of the patients with a COVID positive report. To achieve this, we applied the same U-NET model to the previously processed images. In addition, we defined certain metrics to assess the model's performance, such as the dice coefficient and the Cosine Annealing Learning Rate Schedule for cutting-edge segmentation.

```
Model: "2dcnn"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_5 (InputLayer)         [(None, 128, 128, 1)]     0
_________________________________________________________________
conv2d_13 (Conv2D)           (None, 126, 126, 64)      640
_________________________________________________________________
max_pooling2d_12 (MaxPooling (None, 63, 63, 64)        0
_________________________________________________________________
batch_normalization_12 (Batc (None, 63, 63, 64)        256
_________________________________________________________________
conv2d_14 (Conv2D)           (None, 61, 61, 64)        36928
_________________________________________________________________
max_pooling2d_13 (MaxPooling (None, 30, 30, 64)        0
_________________________________________________________________
batch_normalization_13 (Batc (None, 30, 30, 64)        256
_________________________________________________________________
conv2d_15 (Conv2D)           (None, 28, 28, 128)       73856
_________________________________________________________________
max_pooling2d_14 (MaxPooling (None, 14, 14, 128)       0
_________________________________________________________________
batch_normalization_14 (Batc (None, 14, 14, 128)       512
_________________________________________________________________
conv2d_16 (Conv2D)           (None, 12, 12, 256)       295168
_________________________________________________________________
max_pooling2d_15 (MaxPooling (None, 6, 6, 256)         0
_________________________________________________________________
batch_normalization_15 (Batc (None, 6, 6, 256)         1024
_________________________________________________________________
global_average_pooling2d_1 ( (None, 256)               0
_________________________________________________________________
dense_4 (Dense)              (None, 512)               131584
_________________________________________________________________
dropout_2 (Dropout)          (None, 512)               0
_________________________________________________________________
dense_5 (Dense)              (None, 1)                 513
=================================================================
Total params: 540,737
Trainable params: 539,713
Non-trainable params: 1,024
_________________________________________________________________
```

Fig 6: CNN Model for classification

1. Algorithms and Terminologies

A. Convolution Neural Network and it's variation

Deep learning has proven to be a highly effective tool, especially for pattern detection with large volumes of data. Among the most popular deep neural networks are convolutional neural networks (CNNs). To understand how CNNs work, let's first review some basics of how images are represented. While grayscale images are composed of a single matrix of pixel values, RGB images have multiple planes. In our case, we are working with CT scans.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 7, April 2023

CNNs consist of layers upon layers of artificial neurons that compute weighted sums of inputs to produce activation values, mimicking biological neurons. Each layer of a CNN extracts specific features from an image. For example, the first layer might extract horizontal or diagonal edges, while the next layer detects corners and more complex edges. Deeper layers recognize increasingly intricate aspects such as objects and faces. Given that our dataset includes CT images and infection masks, we will use CNNs to classify the data into positive and negative groups and apply labels accordingly. The various layers in a CNN produce multiple activation functions that are passed on to subsequent layers. The basic architecture of a CNN is shown in Figure 7.

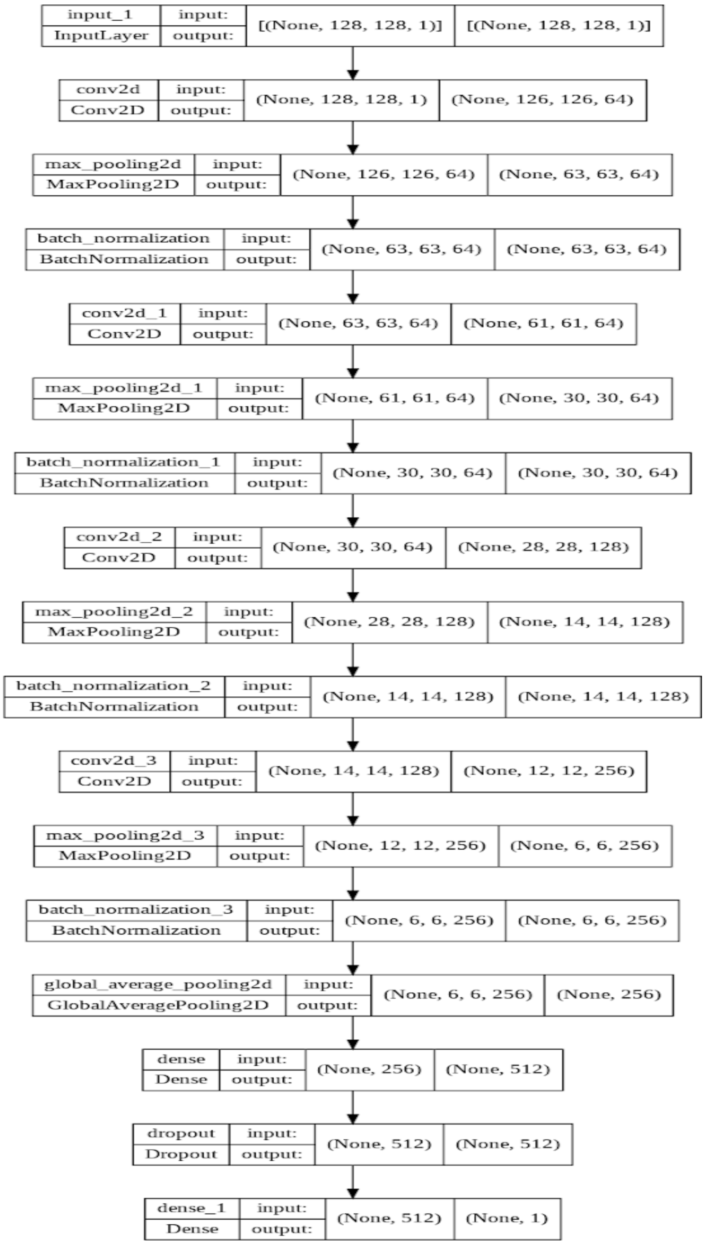

Fig 7: CNN architecture

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 7, April 2023

B. U-NET Model Architecture

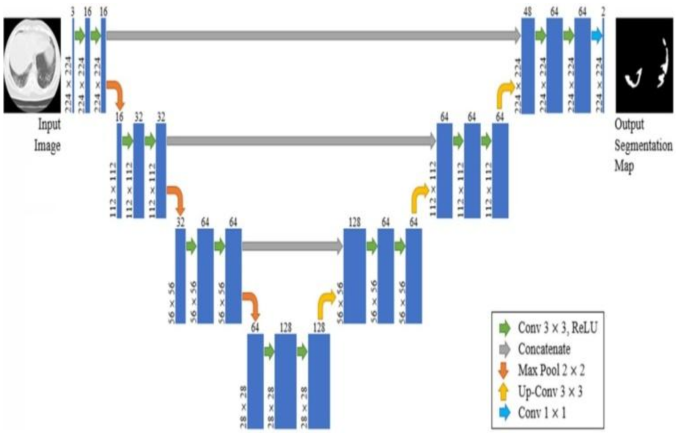

Fig 8 :U-NET model Architecture

The U-Net is a convolutional neural network variant that was introduced in 2015 for processing biomedical images. Unlike traditional CNNs that focus on image classification, U-Nets perform global-local processing to identify and localize areas of abnormality in medical images. The network architecture consists of two pathways: the contraction (encoder) pathway and the symmetric expanding (decoder) pathway. The encoder pathway captures the image context using simple convolutional and max-pooling layers, while the decoder pathway performs accurate localization using transposed convolutions. The two pathways are connected via a bottleneck that contains two convolutional layers with batch normalization and dropout. The U-Net differs from other FCN methods by using skip connections that employ concatenation instead of summation, allowing local information to flow to global information during upsampling. The U-Net also features a symmetric design and a large number of feature maps in the upsampling pathway, enabling effective information flow. The U-Net has been used for lung and infection segmentation in biomedical applications.

C. CLAHE Enhancer

To increase the visibility of hazy photos or videos, contrast limited adaptive histogram equalization, or CLAHE, is used. In contrast to traditional histogram equalization, adaptive histogram equalization (AHE) redistributes the image's brightness value using a variety of methods, each of which corresponds to a distinct region of the image. As opposed to adaptive histogram equalization, CLAHE defines the shape of the histogram using a distribution parameter, producing results of greater quality (AHE). For this reason, in the pre-processing stage, we used CLAHE to enhance the quality or contrast of CT images. We used OpenCV's create CLAHE function to equally balance the histogram. In short, we'll work to improve contrast so that, in a dynamic format, darker portions are more noticeable and lighter areas are more clear. Fig:9

2. Scales and Metrics

During the fitting phase, we evaluated the segmentation performance for each epoch by using randomly chosen and data-augmented patches from the validation dataset. This helped us to determine how well the training set was fitting. Once the training was completed, we evaluated the inference performance by comparing the segmentation overlap between the predicted and actual image. We used three commonly employed evaluation metrics in the field of medical image analysis. The Dice similarity coefficient is the most widely used computer vision statistic and is calculated using the cosine annealing learning rate to ensure accurate results, as shown in the equation below.

The confusion matrix was used to calculate precision, recall, F1 score, and specificity, which are the evaluation metrics for our classification model. These measures are commonly used in the medical industry and are based on the rates of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN), respectively.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 7, April 2023

$$DSC = \frac{2.TP}{2.TP + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1score = \frac{2 * precision * recall}{precision + recall}$$

IV. EXPERIMENT RESULTS

In case of image classification the CNN model is train which gives a value between 0 and 1.The prediction is transformed to binary values and the threshold is determined using the ROC curve. Precision, recall, and the F1 score are computed using a confusion matrix, Similarly for segmentation part U-net model is train which marks the lung area in the ct-scans, dice-coefficient used exact results for all models mentioned below

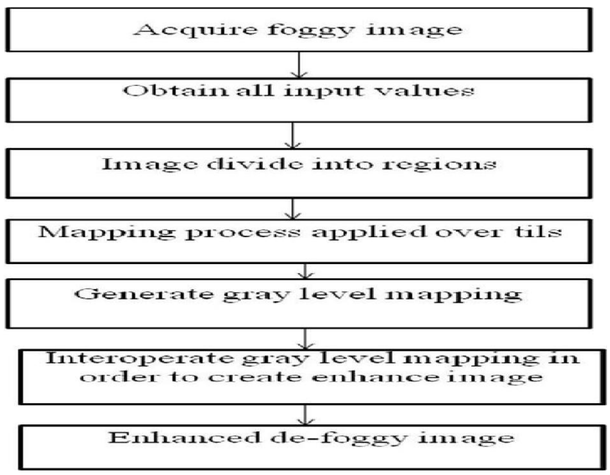

Fig 9: Flowchart for CLAHE algorithm

A. Data Pre-processing

In data pre-processing we improved the contrast using CLAHE enhancer in order to see the lungs CT scan clearly, further in order to remove the unnecessary portion we cropped the images after applying these techniques on original CT scan, we repeated it on lung masks and on infection masks Fig:10 & Fig 11.

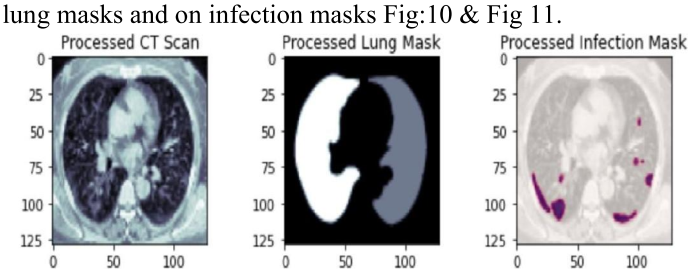

Fig:10: Final processed dataset images

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 7, April 2023

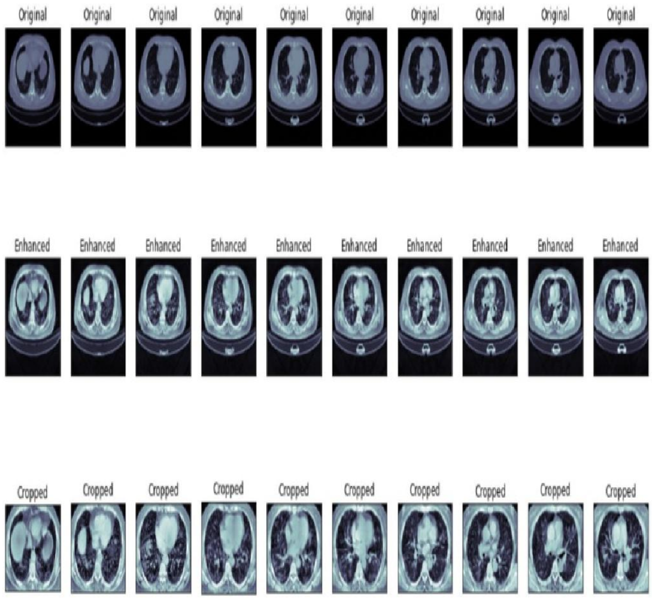

Fig:11: Original images along with improve contrast and cropped

B. Classification Model Results

During the training of classification CNN model we plot accuracy and loss graphs over number of epochs(the number of times the machine learning algorithm has successfully iterated through the whole training dataset) to see the performance of our model on training. During the training the average loss we obtain is around 0.3083 and accuracy is 0.93358. Final results of the model were obtained using precision, recall and F1 score as performance measuring parameters of our CNN model. We get the values of precision 0.99, recall 0.9310, F1 score 0.9596.

The model's used for image segmentation, effectiveness is assessed using the dice coefficient. In order to alter the learning rate during training, exponential decay is employed Fig:12.

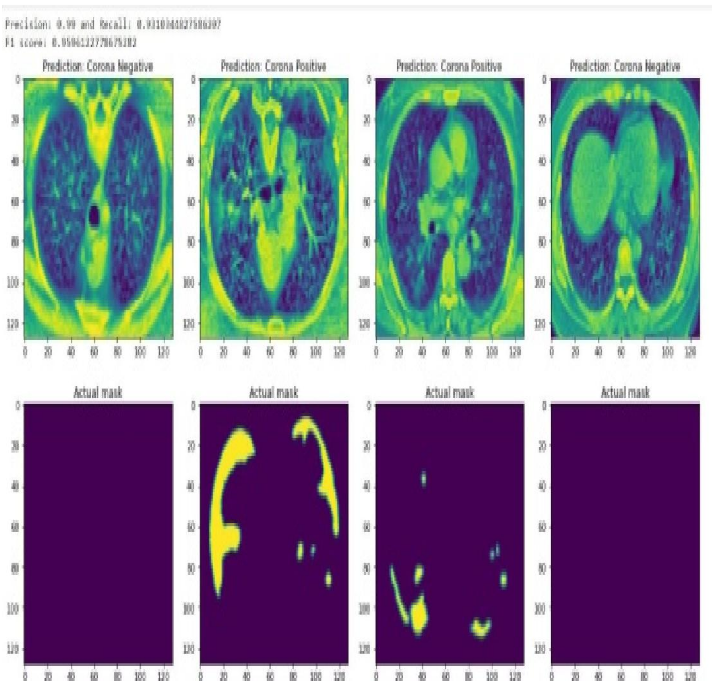

Fig:12: Precision, recall , F1 score , classified and visual results

IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.301

Volume 3, Issue 7, April 2023

C. Infection Segmentation Result

To evaluate the effectiveness of the U-Net model used for infection segmentation, we used the dice coefficient along with the Cosine Annealing Learning Rate Schedule, which is a state-of-the-art method for segmentation. The model achieved a dice coefficient of 0.8217 and a validation dice coefficient of 0.7821. To visualize the performance of the model during training, we plotted the accuracy and loss graphs, which can be seen in Fig:13. These graphs help us to monitor the performance of the model and to make adjustments if necessary.

Finally, we generated a visualization of the final prediction of the model, which can be seen in Fig:14. This helps us to understand how well the model is able to segment the infection from the CT scans, and to identify any areas where the model may be struggling.

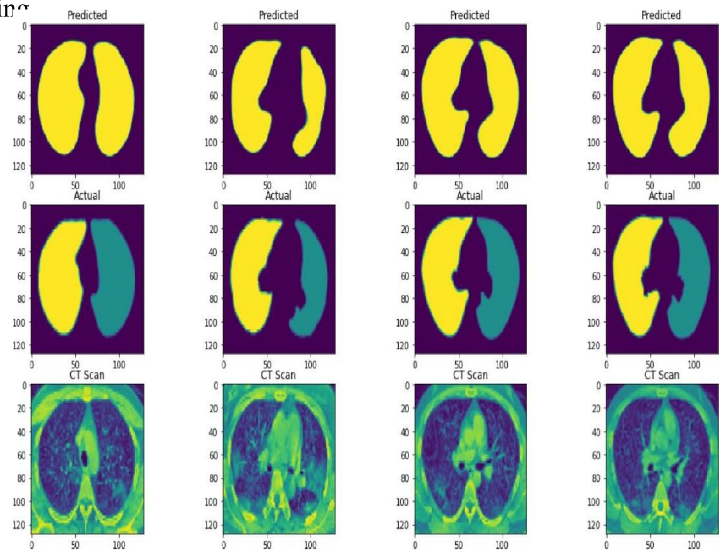

Fig 13: Lungs Segmentation visualization

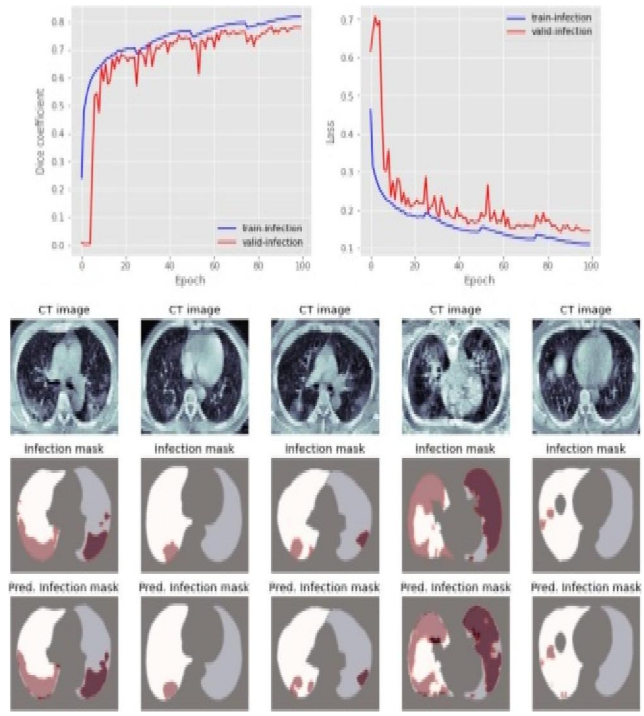

Fig 14: Infection segmentation visualization

DOI: 10.48175/IJARSCT-9489

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 7, April 2023

V. CONCLUSION

In this study, we have developed and evaluated a method for COVID-19 lung CT-scan image segmentation and classification using UNET, which has been implemented in TensorFlow and Keras. Our approach primarily focuses on determining whether a person is COVID-19 positive or negative, and if positive, identifying the infection mask. To simplify the process of training our models, we have utilized various preprocessing techniques and data augmentation, which have allowed us to manage small data sets that serve as different databases. Instead of using new and complex neural network topologies, we have used a common 2D U-Net. Our study has demonstrated that our pipeline for segmenting and classifying medical images can effectively train accurate models without overfitting on sparse data. Additionally, we have been able to outperform existing segmentation methods for COVID-19-infected areas, including the lungs. This research has significant potential for use as a clinical decision support system for COVID-19 quantitative evaluation and disease monitoring. However, further research is necessary to evaluate the clinical performance and robustness of COVID-19 semantic segmentation in clinical trials.

VI. FUTURE SCOPE

The project still has room for improvement. The next section discusses several potential directions for the project's expansion or improvement.

- Focusing on severity identification from 3D CT volumes might be a future improvement or expansion of this study.
- We apply algorithms and techniques on not quite a huge data set, so another future work is to find or build a quality data set and try to implement the same technique on it.
- It is possible to do research to improve our understanding of how one attribute influences the other for prediction.

REFERENCES

[1] Zhang Ruochi Duan Meiyu Xie Tianqi Pan Jiahui Peng Ejun Huang Juanjuan Zhang Yingli Xiaoming Xu Yao Haochen, Zhang Nan. Severity detection for the coronavirus disease, et al. (covid-19) patients using a machine learning model based on the blood and urine tests.

[2] Ferrari D Locatelli M Banfi G Cabitza F. Brinati D, Campagner A. Detection of covid-19 infection from routine blood exams with machine learning: a feasibility study. j med syst. page 44(8):1–12, 2020.

[3] Rajasekar SJS. Perumal V, Narayanan V. Detection of covid-19 using cxr and ct images using transfer learning and haralick features. Appl Intell, page 51(1):341–58, 2021.

[4] Weimin Li Qingxia Wu Xiaohu Li Meng Niu Meiyun Wang Xiaoming Qiu Hongjun Li He Yu Shuo Wang, Yunfei Zha. A fully automatic deep learning system for covid-19 diagnostic and prognostic analysis. European Respiratory Journal, page 56(2), 2020.

[5] Bingbing Zheng, Yaoqi Liu, Yu Zhu, Fuli Yu, Tianjiao Jiang, Dawei Yang, and Tao Xu. Msd-net: Multi-scale discriminative network for covid-19 lung infection segmentation on ct. Ieee Access, 8:185786–185795, 2020.

[6] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.

[7] Feng Shi, Jun Wang, Jun Shi, Ziyan Wu, Qian Wang, Zhenyu Tang, Kelei He, Yinghuan Shi, and Dinggang Shen. Review of artificial intelligence techniques in imaging data acquisition, segmentation, and diagnosis for covid-19. IEEE reviews in biomedical engineering, 14:4–15, 2020.

[8] 2020 .CORONACASES.ORG Paiva, O. Helping radiologists to help people in more than 100 countries! | coronavirus cases - . Coronacases.org, 2020.

[9] Y. Glick. Viewing playlist: Covid-19 pneumonia. Coronacases.org, 2020