DETECTION OF COVID-19

COMPUTER NETWORK- DR. M PRABU

MARCH 26, 2022

Group – 3
Sheenam Waris M200717CA
Janvi Agrawal M200695CA
Amanpreet Kaur M200680CA
Ankit Hammad M200683CA
Chetan Patidar M200688CA
Jayant Parganiha M200696CA

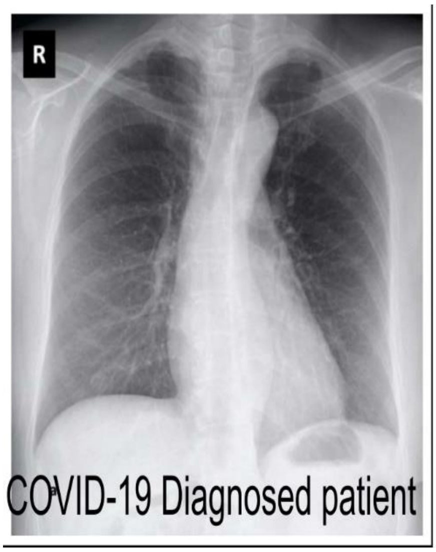
1. Introduction

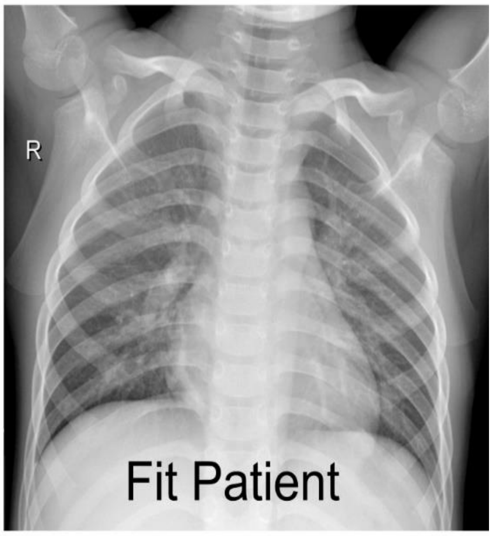
A novel coronavirus, COVID-19, is the infection caused by SARS-CoV-2. Reverse transcription Polymerase chain reaction (RT-PCR), gene sequencing for respiratory or blood samples are now the main methods for COVID19 detection. Research shows RT-PCR correlation with Chest CT, while others study its correlation with X-ray chest images.

There are only a few large open access datasets of COVID-19 X-ray images; most of the published studies use as a foundation the COVID-19 Image Data Collection, which was constructed with images from COVID-19 reports or articles, in collaboration with a radiologist to confirm pathologies in the pictures taken.

2. Methodology

Our methodology consists of three main experiments to evaluate the performance of the models and assess the influence of the different stages of the process. Each experiment follows the workflow shown in below diagram. The difference between experiments is the dataset used. In all instances, the same images for COVID-19 positive cases were used. Meanwhile, datasets for negative cases were used.





3. COVID-19 classification datasets

A total of positive and negative Chest X-ray images datasets was used in different stages. For positive cases, the COVID-19 Image Data Collection by GitHub, and negative cases of Normal chest X-ray by Kaggle.

4. CNN Processing

4.1 Image projection filtering

The images from the COVID-19 datasets have a label corresponding to the image projection: frontal (posteroanterior). In order to automate the process of filtering the images according to the projection, a classification model CNN was trained on a subset of covid positive and normal chest X-ray dataset, with 284 frontal images.

4.2 Lung segmentation

Two datasets were used to train the CNN models for these segmentations: GitHub dataset with 142 images, Kaggle normal dataset with 142. Despite the apparent small amount of data, the quantity and variability of the images was enough to achieve a useful segmentation model.

4.3 Image separation

For the classification task, data were divided into a train (60%), validation (20%), and test (20%) partitions, following the clinical information to avoid images from the same subject in two different partitions, which could generate bias and overfitting in the models.

The image quantity was considerably less for the segmentation task, so creating a test dataset was avoided, leaving the distribution of 80% (225 images) for the train set and 20% (57 images) for validation data.

4.4 Pre-processing

As the images come from several datasets with different image sizes and acquisition, a pre-processing step is applied to reduce or remove effects on the performance of the models due to data variability.

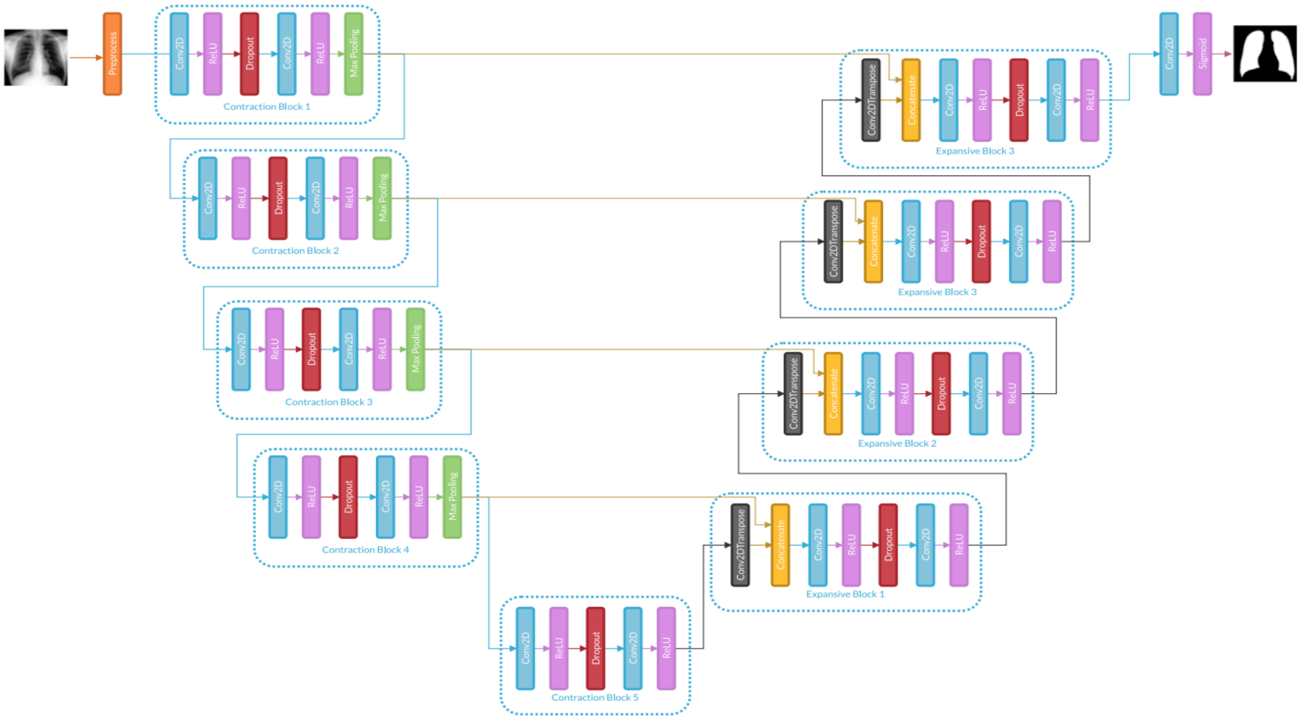
The pre-processing layer is shown orange in below figure, it consists of three steps: resize all images to 224 × 224 pixels in one channel (grayscale). When applying standardization to the validation and test sets, the mean and standard deviation (std) from the training set were used to unify the data distribution.

4.5 Segmentation

There are multiple ways to perform image segmentation; this report uses a Deep Learning model based on neural network model. We tested three different amounts of filters on convolutional layers to find the optimal for this task.

4.6 Hyperparameters

Kernel size in convolutional layers is 3 × 3 with a kernel initialization and padding same. In Max-pooling layers, the pool size is 2 × 2, the Dropout rate in the first two Expansion and contraction blocks is 0.1, while in three and four of 0.2, and for contraction block five is 0.3. Transposed convolutional layers use kernel size of 2 × 2, strides of 2 × 2, and padding same. Finally, the last convolutional layer uses one filter and a kernel size of 1 × 1.



4.7 Classification

There is a classification task in this report, one to distinguish COVID-19 positive cases from negative ones. For this task, Deep Learning model Convolution neural network were used. The networks were trained using transfer learning with pre-trained weights. The use of pre-trained models takes advantage of features learned on a larger dataset so that a new model converges faster and performs better on a smaller dataset. Pre-trained models come from the Keras library.

5. Conclusions

As shown, classification models using this technique need between 10 and 30 epochs to converge, while segmentation models without transfer learning need about 200.COVID-19 Disease in Chest X-ray images with a general accuracy of 92.72%, classifying COVID and NO-COVID images. Meanwhile, only for the COVID label, the approach has a 95.63% accuracy in the test dataset.

6. Future works

For more accurate results, we identified two main future work opportunities, first consists of extending COVID datasets to generalize information and overcome the problems of using different image sources.

THANK YOU 😊