YOLOv5 Model for COVID-19 Detection in X-ray Images

1st Rikhat Akizhanov
SEDS, Robotics Department
Nazarbayev University
Nur-Sultan, Kazakhstan
rikhan.akizkhanov@nu.edu.kz

2nd Kamila Zhumakhanova
SEDS, Robotics Department
Nazarbayev University
Nur-Sultan, Kazakhstan
kamila.zhumakhanova@nu.edu.kz

3rd Sanzhar Kabitkanov
SEDS, Robotics Department
Nazarbayev University
Nur-Sultan, Kazakhstan
sanzhar.kabitkanov@nu.edu.kz

4th Saparkhan Kassymbekov
SEDS, Robotics Department
Nazarbayev University
Nur-Sultan, Kazakhstan
saparkhan.kassymbekov@nu.edu.kz

***Abstract*—COVID-19 is a disease that is causing massive number of deaths today. There are a number of methods to diagnose the virus in human body. Although, they are proven to be effective, they have some limitations in the quality and availability of the tests. Therefore, the use of deep learning models were proposed to detect the virus. This paper focuses on the detection of Coronavirus from X-ray lung images using state of the art detector YOLOv5 model. A dataset of 5910 labeled X-ray lung images were used for training the model. Confusion matrices, precision recalls, loss graphs were used to evaluate the results of the trained model. The evaluation shows that the model performs with the average accuracy of 74%, which is comparable to other models like Vgg19, Resnet18, etc. As the accuracy is lower than expected, better prepossessing and better labelinng of the data is suggested for future works.**

I. Introduction

Covid-19 is a severe respiratory disease that appeared in 2019 in China's city of Wuhan and spread all over the world in matter of weeks causing disruptions in the economies and damage to the health of hundreds of millions of people [1].. Because of such high speed of spread, a number of countries has suffered heavy human lives losses, 5 million deaths, regardless of the level of development and wealth they possessed. The primary examples are the U.S.A., India, China, Germany, the U.K., etc [1]. Various vaccines such Phizer/BioNTech, CoronaVac, Moderna, JonsonJonson, and AstraZeneca were invented in the end of the 2020 and the beginning of 2021 easing the situation [2]. However, the virus has proven to be mutating into different variants such as Alpha, Beta, Delta, Gamma, etc., as it reaches more and more population [3]. That is why, the best preventive measure against the spread of the virus is its early detection.

The most widespread method is re-verse transcription polymerase chain reaction or RT-PCR method. Although, it provides some appropriate testing of the disease, it has some limitations due to its focus on the antibodies to the virus, rather than virus itself, which can lead to false-negative results of the testing [4]. Moreover, many countries cannot afford the equipment and personnel for mass-scale testing within their respective populations. Thus, some other affordable ways of detecting the virus are required. One of the highly discussed methods is applying Deep Learning models on the data of X-ray chest images, as the tools for collecting such data are more widespread in hospitals and clinics.

In recent years, the increase in the use of machine learning algorithms in clinical diagnosis is observed. The famous example is detecting tumors in X-ray images for early treatment of brain/breast cancer [5]. As the number of people suffering from the COVID-19 increased dramatically, the number of available personnel to cope with the pandemic has become insufficient. That is why, the use of accurate and fast models, which decrease the work-load for medical workers, has become a highly necessary tool for clinical practice. This demand were fulfilled with a number of research that used various models for classification of the coronavirus infected and healthy X-ray images.

In our paper, we are using YOLOv5 model for detection of the coronavirus. It is trained on the dataset with 5910 X-ray chest images [6]. Although there are a number of other models such as ResNet18, ResNet50, Vgg16, DenseNet, etc. which already have good results [7], YOLOv5 is a state of the art object detection model [8], which promises better or comparable results.

The paper has the following structure. First, we briefly cover the literature review of related works in section 2. Then, we explain the details of model implementation in the third section. Next, we present our results in section 4. The concluding section will finish the paper with recommendations for future work.

II. Related Work

A. General info about paper

The research done by Asnaoui and Chawki with the tittle ”Using X-ray images and deep learning for automated detection of coronavirus disease” has served as major inspiration for this paper. It was conducted right after the beginning of the pandemic in early 2020, in order to provide better understanding of how deep learning algorithms can be used for virus detection. In their work, they compared the performance of different state of the art models such as VGG16, VGG19, DenseNet201, Inception ResNet V2, Inception V3, Resnet50, and MobileNet V2 on classification of healthy, bacteria infected and coronavirus infected X-ray chest images [7].

B. Used methodology

The purpose of that paper was obtained using the following methodology. First, they have prepared the dataset with chest 6087 X-ray CT images (2780 images of bacterial pneumonia, 1493 of corona- virus, 231 of Covid19, and 1583 normal). Then, they have used channel identification, intensity normalization and CLAHE to preprocess the data before feeding it to the network. Next, they trained the mentioned models with those images and tested its classification for normal, pneumonia and coronavirus infested classes [7].

C. Obtained results

At the end, they provided the classification accuracies, in which Inception ResNet V2 and Densnet201 had the best results of 92.18% and 88.09% respectively [7].

III. Methodology

A. Dataset

We used in our approach CoronaHack Chest X-Ray-Dataset that consists of 5910 chest X-ray images with several diagnoses such as normal, Covid19, bacterial, and viral pneumonia. Considering a proportion of those labels, 27 % from the total numbers of images are labeled as ”Normal” and the other 73 % have a label of ”Pneumonia”. Regarding the data split, approximately one-tenth of the images were chosen for a test part, while others were used in the train process.

We trained our model to classify:

1) Two classes: Normal and Pneumonia: The basic division of data into two categories of healthy and ill

2) Three classes: Normal, Pneumonia and Virus: A further division which could give us a better outcome in comparison with the basic division

B. Preprocessing and labels

1) Bounding boxes: The original CoronaHack dataset is well-suited for classification task and labels are just class name. However the object detection task that we are proposing requires to have bounding boxes as labels. In this paper we propose and use the simple way of getting the bounding box for each image. The whole X-Ray image is taken as one bounding box. Thus, each image in the dataset contains exactly one bounding box. In this way the model will learn to detect not only the class of the X-Ray image but also a borders of this scan.

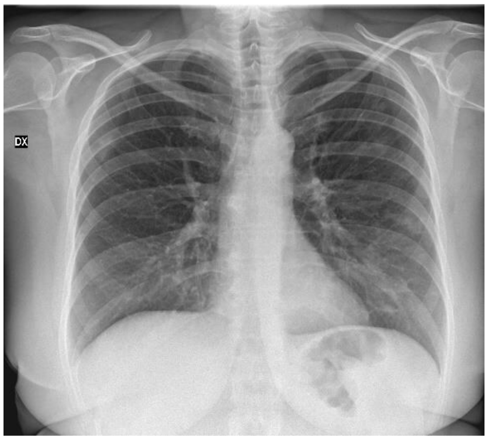

Fig. 1. An example of the images in the dataset

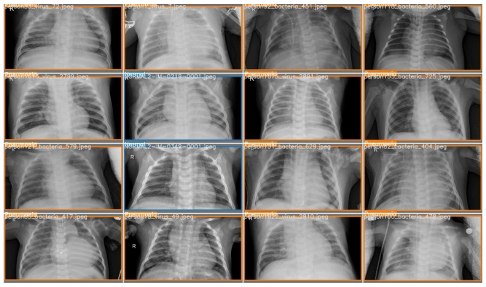

Fig. 2. A batch of labelled images with bounding boxes

2) Image preprocessing: Dataset contains X-Ray images obtained from different places with a different machines and in general X-Ray images have a low contrast. Preprocessing procedure helps to decrease the noise, eliminate high and low frequencies, and in general improve the quality and visual information contained in each image. Thus, in order to increase the results and make it easier for the model to learn differences between classes the preprocessing before training was done. The first step is intensity normalization, which basically changes the range of pixel values and it is typical to use it for X-Ray images.The second step is Contrast Limited Adaptive Histogram Equalization (CLAHE) [9] which changes the histogram of the images and makes it more distributed.

C. YOLOv5 model

1) Model architecture: YOLO (you only look once) is a family of state-of-the-art object detection models. The working principle of YOLO is quite straightforward: it breaks down the images into a grid system and the detection of objects is performed inside the grid [12]. We used the latest, yet unofficial version YOLOv5. It was released on May 29, 2020, by Glenn Jocher from ultralytics [10]. In figure 1 the architecture of the model is depicted. YOLOv5 has several advantages. To begin, it is a single shot object detection model, and thus it does not require a lot of computational resources.

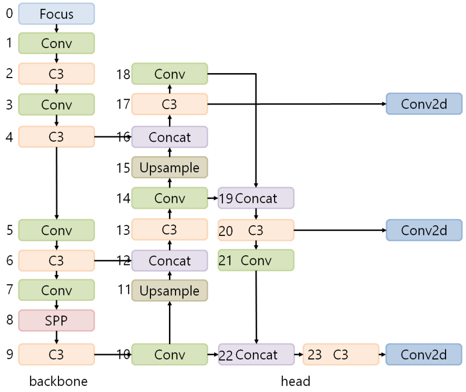

Fig. 3. Architecture of the model [11]

Also, it is state-of-the art model in terms of robustness, ease of use and performance [12].

2) Experiments setup: All the experiments were conducted using Pyhton as a programming language and PyTorch as a deep learning backend. The training of the model was performed on NVIDIA Tesla V100 with 32 Gb RAM.

As to the training procedure, several hyperparameters were defined. The medium model (yolov5m) was trained with the image size 1280 and batch size 16. Regarding the number of epochs, the default value was chosen as 50 epochs. In the process of conducting the experiments we decided to double the number of epochs, yet then we came to conclusion that the results are significantly less satisfactory than for 50 epochs. Because of that, only the results of the experiments with 50 epochs are used as the main ones in this research work.

IV. Results

A. Metrics

Performance of our model can be assessed using several different metrics. The first one that is common in object detection tasks is mean Average Precision (mAP). It computes the average precision value for recall value over 0 to 1. Precision shows how accurate model is in predictions. Recall depicts how well model finds all the positives.

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

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

$$TP = True\, positive, TN = True\, negative$$

$$FP = False\, positive, FN = False\, negative$$

Further, the effectiveness of predicted bounding box can be computed by using the Intersection over union (IoU):

$$IoU = \frac{area\, of\, overlap}{area\, of\, union}$$

Now, the mAP can be calculated for different IoU ranges. The most common one are 0.5 and 0.5:0.95.

The next metric that can show the performance of the model is the precision-recall graph. Precision and recall was explained previously and here they are just plotted over each other. This metric shows us how good model is performing for different recall values.

The last but not least metric is confusion matrix. This metric shows the percentage distribution over predicted and ground truth labels for different classes. If the model performs very well the main diagonal will have high numbers. Because our methods and model differs from the original paper, confusion matrix is the only way to somehow compare the results between models.

B. Two Classes

The first experiment that was conducted included only two classes, normal and infected (pneumonia) cases. Thus, coronavirus and pneumonia are united in one class and model tries to find if it is X-Ray of normal or infected human.

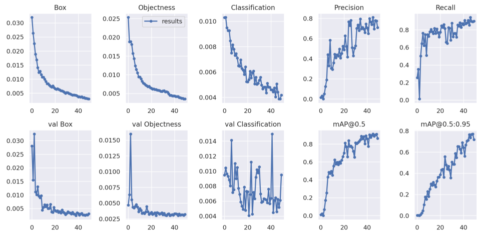

Fig. 4. Results of the experiment with 50 epochs

The training graphs above for 50 epochs suggest that there is no over fitting as train and validation loss curves are close to each other. Also looking to the loss it can be said that model converged. The overall results are great, both mAP values are high meaning model draw bounding boxes well.

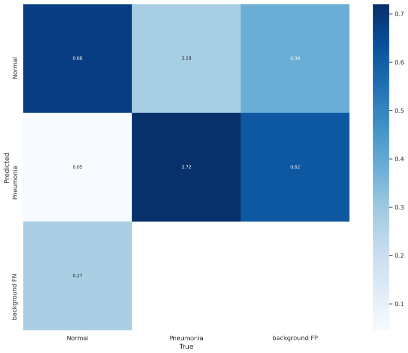

Fig. 5. Confusion matrix for the experiment with 50 epochs

The confusion matrix above shows how well model classifies the images. The main diagonal has the highest values meaning model generally works well. However, sometimes gives false predictions on pneumonia. The same can be said about normal cases but there rate is lower. Overall, it successfully learned to find X-Ray images and classify them.

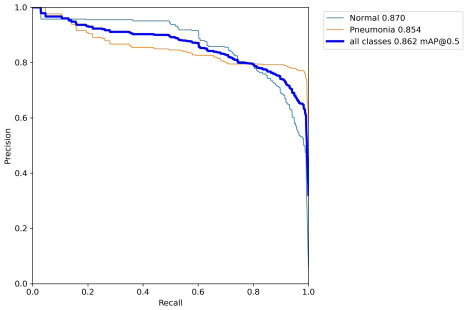

Fig. 6. PR curve for the experiment with 50 epochs

Lastly, Precision-Recall graph tells how model works for different recall values. For the most recall values accuracy is quit high and drops only for recall values close to 1. It is excellent result for object detection, but tells little about classification.

C. Three Classes

The dataset can also be divided into three classes: normal, pneumonia and coronavirus. Now model has to distinguish not only infected subject or not but also which types of infection it is. The task considered to be harder and it is the main goal of the original project.

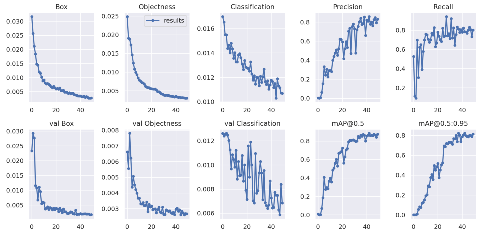

Fig. 7. Results of the experiment with 50 epochs

The training curves tells us that model is converged to some optimal state. There is no over fitting as train and validation graphs are close to each other. mAP values are as high as in two class approach, meaning model draw bounding boxes well.

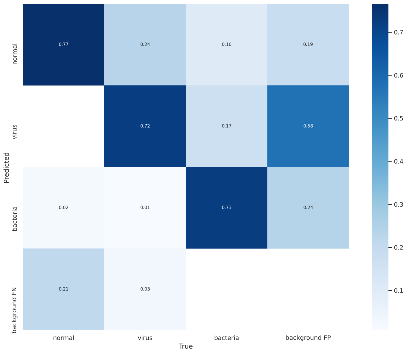

Fig. 8. Confusion matrix for the experiment with 50 epochs

Confusion matrix for this model depicts the accuracy for three classes. The main diagonal is darkest meaning model generally works satisfactory. However, there is some confusion, sometimes it can not detect viral images and rarely pneumonia images. Overall, accuracy for all classes is more than 70 percent and can be said to be quit high.

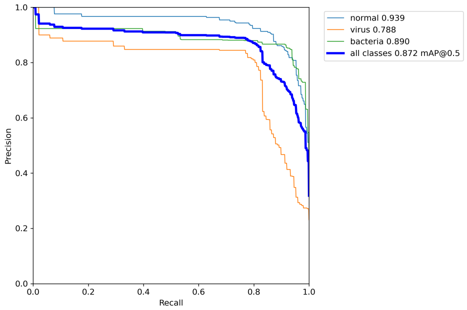

Fig. 9. PR curve for the experiment with 50 epochs

The precision recall graph is similar to the same graph for two class model. precision is high for most values of recall and only to values close to 1 precision drops. Also, it can be noticed that that virus class is lower then other meaning it is harder for model to find and classify the X-Ray images with virus.

Overall, both models show comparable results on object detection and classification. There is some confusion when model do not give any class to the image.

V. Discussion

In this paper we have carried out several experiments with YOLOv5 and CT and X-Ray images in order to contribute to the comparative study of deep learning architectures conducted by K. El Asnaoui and Y. Chawki.

By the comparison of the results for the experiments with two and three classes it can be seen that the best result is for the experiment with three classes. In this experiment the diagonal of the confusion matrix contains the best scores for the accuracy of classification, which are in the range from 0.73 to 0.77 for each class. Thus, the average accuracy score is 0.74. This means that we got the comparable results since our model performs better than VGG19 in the original paper [7]. As to the results for the experiment with 2 classes, the average accuracy score is 0.7, which is lower that the lowest accuracy of the models in the original paper.

There are several reasons for such performance of the model. Firstly, YOLOv5's main functions are object detection combined with classification, but not solely classification. We suppose that this is the main reason why the results are not as high as expected. Secondly, we deem that possibly the preprocessing that we have done is not enough to produce outstanding results. The usage of more advanced preprocessing techniques might improve the performance.

It seems worth mentioning that the results that we obtained can be compared to the results of the original paper, yet our approaches for training the models differ significantly.

Even though the performance of the model is worse than of the models in the original paper, there is an advantage of using YOLOv5 for diagnosis process that is worth pointing out. Since YOLOv5 is an object detector, it can determine the state of the lungs in the X-Ray scans of larger scope, for example, of the whole body.

VI. Future perspectives

This reseach can be improved in future in several aspects. Firstly, the data preprocessing step can be enhanced. It is usual case for X-Ray images as new methods for preprocessing are developing every year. Data comes from different sources where different machines were used to take X-ray images and thus model is confused in some cases. Potentially, some domain adaptation technique might help there.

In the same way as preprocessing is improving with time, models are also developing. Bigger and more complicated model with better hyperparameters might achieve significantly beter results. YOLOv5 is rather fast then highly accurate,other models like FasterRCNN, PyramidBox etc. can be applied to this taask.

Lastly, the labels and bounding boxes can improved as well. Instead of the whole image as single box the exact places that are responsible for disease can be labeled. This can help doctors quickly find problematic areas.

VII. Conclusion

In conclusion, we have reached our goal of obtaining comparable results in classification of X-ray images into coronavirus infected and non-infected classes using YOLOv5 model. All the results are presented in section 4 and clarified in section 5. Although our average accuhttps://www.overleaf.com/project/619dd70f8f13e0dbf7c5bd56racy of classification is not as high as we expected, the research in the us of YOLOv5 can be extended for future works with modifications and corrections from section 6 of our paper.

References

[1] World Health Organization, “WHO COVID-19 dashboard,” covid19.who.int, 2021. https://covid19.who.int/.

[2] “WHO – COVID19 Vaccine Tracker,” covid19.trackvaccines.org. https://covid19.trackvaccines.org/agency/who/

[3] “COVID-19 Variants,” pandemic. https://pandemic.internationalsos.com/2019-ncov/covid-19-variants (accessed Nov. 24, 2021).

[4] P. M. Bossuyt, “Testing COVID-19 tests faces methodological challenges,” Journal of Clinical Epidemiology, Jul. 2020, doi: 10.1016/j.jclinepi.2020.06.037.

[5] K. Sharma, A. Kaur, and S. Gujral, “Brain Tumor Detection based on Machine Learning Algorithms,” International Journal of Computer Applications, vol. 103, no. 1, pp. 7–11, Oct. 2014, doi: 10.5120/18036-6883.

[6] “CoronaHack -Chest X-Ray-Dataset,” kaggle.com. https://www.kaggle.com/praveengovi/coronahack-chest-xraydataset.

[7] K. El Asnaoui and Y. Chawki, “Using X-ray images and deep learning for automated detection of coronavirus disease,” Journal of Biomolecular Structure and Dynamics, pp. 1–12, May 2020, doi: 10.1080/07391102.2020.1767212.

[8] “YOLOv5 model architecture overview · Issue 4518 · ultralytics/yolov5,” GitHub. https://github.com/ultralytics/yolov5/issues/4518 (accessed Nov. 24, 2021).

[9] K. El Asnaoui, Y. Chawki, and A. Idri, “Automated methods for detection and classification pneumonia based on X-ray images using Deep Learning,” Studies in Big Data, pp. 257–284, 2021.

[10] M. Ohanu, “YOLOv5 Tutorial,” Medium, Dec. 11, 2020. https://michaelohanu.medium.com/yolov5-tutorial-75207a19a3aa.

[11] “YOLOv5 model architecture overview · Issue 4518 · ultralytics/yolov5,” GitHub. https://github.com/ultralytics/yolov5/issues/4518.

[12] S. Gutta, “Object Detection Algorithm — YOLO v5 Architecture,” Analytics Vidhya, Aug. 04, 2021. https://medium.com/analytics-vidhya/object-detection-algorithm-yolo-v5-architecture-89e0a35472ef (accessed Nov. 25, 2021).