Presentation link: https://www.youtube.com/watch?v=aPQASAkchTo

Chest X-ray project

Abstract:

We present two algorithms to assist chest X-ray image diagnosis, using ChestX-ray14 dataset that contains about 114,000 chest X-ray medical images with 14 diseases. First, we build a simplified VGG Neural Network to classify 14 pathologies, using a multiple-label binary classification approach. We achieve about 87% accuracy on the test data. Second, we customize YOLO V3 algorithm to combine classification and bounding box prediction to detect and localize lung disease on the X-ray images. We could detect lung disease with up to 0.3 mAP, which is on par with current research work on X-ray images. We analyze the results and discuss potential improvements for future work.

1. Introduction and background:

X-ray has been widely used to produce diagnostic images for organs, tissues, and bones to help identify abnormalities or diseases, with relatively low cost comparing to other image techniques like computed tomography (CT). However, current diagnostics using X-ray images in medical practices still highly depend on the expert of radiologists. In addition, the diagnostic process may take very long time due to the availability of radiologists.

In recent years, Deep Learning has been widely used in many science and engineering areas. Convolutional Neural Network (CNN) has been shown by Setio et al. in LUNA16 [6] to be the primary method for image classification and detection. Yao et al. [2] worked on the ChestX-ray14 dataset and implemented a two-stage end-to-end neural network, which achieved better performance than the results by Wang et al. [1]. Rajpurkar et al. [3] applied a 121-layer pre-trained CNN on the same dataset to achieve better performance than average performance of 4 expert radiologists. However, neither work directly addressed the more challenging localization issue. Rajpurkar et al. [3] extracted heatmap from the network to show the problematic area, which didn't leverage the given Bounding Box information. In recent object detection and localization work, You Only Look Once (YOLO) by Redmond et al. [7][9] offered one of the start-of-the-art solutions using a different approach: combining the classification with bounding box regression into a single CNN model. The results were much better in the accuracy and runtime performance than previous work on object detection and localization.

In this project, we apply recent Deep Learning/Computer Vision technology to to assist and expedite disease diagnostics. The goal is to classify different diseases as well as detect and localize those diseases through X-ray images, with reasonable accuracy. We work on the largest publicly available X-ray dataset: ChestX-ray [1] and focus on 14 diseases. The dataset is extracted from a clinic database from NIH Health Clinical Center, consisting of more than 112k frontal-view images from about 30k different patients [1]. The dataset is labeled with 14 common diseases, including Atelectasis, Consolidation, Infiltration, Pneumothorax, Edema, Emphysema, Fibrosis, Effusion, Pneumonia, Pleural_thickening, Cardiomegaly, Nodule and Mass and Hernia.

2. Problem formulation:

In this project, we have two problems to solve, using the ChestX-ray14 dataset. The first one is how to identify the 14 different lung diseases and the second is how to find out the potential disease area given an X-ray image. These two tasks naturally lend themselves to two classical Machine Learning problems: multiple-label classification and object detection.

2.1 Multiple-label classification.

The dataset contains the clinical chest X-ray images with labels, which indicate whether a patient has one of the 14 labeled diseases or no findings. This is a known supervised multiple-label binary classification problem. For a single example in the training set, we define a multiple-label binary cross entropy loss $L(X, y)$ as

$$L(X, y) = \sum_{i=1}^{14} [-y_i \log p(Y_i = 1|X) - (1 - y_i) \log p\,(Y_i = 0|X)]$$

where $p(Y_i = 1|X)$ is the predicted probability that the image is classified as disease i.
The optimization goal is to minimize total loss for the best classification results.

2.2 Lung disease detection and localization

Besides disease classification, it is also beneficial to show where a disease is located in order to better assist clinic diagnosis. In this project, we combined classification and localization into one single model to predict disease bounding boxes whenever an image is classified as to contain lung diseases. To achieve this, we used YOLO algorithm and defined the total loss as sum of the localization loss and the classification loss as followings:

$$\lambda_{\mathbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} L_{ij}^{\text{obj}} \left[\left(x_i - \hat{x}_i \right)^2 + \left(y_i - \hat{y}_i \right)^2 \right]$$

$$+\lambda_{\mathbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} L_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right]$$

$$+\sum_{i=0}^{S^2} L_i^{\text{obj}} \sum_{c \in \text{classes}} \left(p_i(c) - \hat{p}_i(c) \right)^2$$

where λ_{coord} is the localization penalty, s is grid number and B is number of anchor boxes. L_{ij}^{obj} = 1 if the j bounding box in cell i is responsible for detecting the object, otherwise = 0. x_i, y_i, w_i, h_i is the ground truth bounding box center x, y coordinates, width and height, while $\hat{x}_i, \hat{y}_i, \hat{w}_i, \hat{h}_i$ is the output bounding box by the network. $p_i(c)$ and $\hat{p}_i(c)$ is the bounding box confident score for ground truth and predicted box respectively.

3. Approach and implementation

3.1 Multi-label classification

3.1.1 Data Analysis and Preprocessing

ChestX-ray14 dataset released by Wang et al. (2017) contains 112,120 frontal-view X-ray images of 30,805 unique patients. Wang et al. (2017) annotate each image with up to 14 different thoracic pathology labels using automatic extraction methods on radiology reports. The original dataset was split into two parts: train/validation set and test set. Train/validation set has about 80k images, which we use to train our models.

We analyzed the 14 diseases distribution in the original 80k images and found that the 14 diseases are highly unbalanced, as shown in Table1. The most count is 13782 from “Infiltration” while the least count is 141 from “Hernia”. Although the unbalanced distribution is due to the natural popularity of different disease, it is extremely difficult for Machine Learning algorithm to learn useful

information from them. Since we couldn't gain more data, we have to try to select training images as balanced as possible.

	Nodule	Effusion	Mass	Consolidatio	Edema	Pneumonia	Fibrosis	Atelectasis	Pleural_Thick	Infiltration	Emphysema	Cardiomegaly	Pneumothorax	Hernia	SUM
Original count	4708	8659	4034	2852	1378	876	1251	8280	2242	13782	1423	1707	2637	141	53970
Original ratio	8.7%	16.0%	7.5%	5.3%	2.6%	1.6%	2.3%	15.3%	4.2%	25.5%	2.6%	3.2%	4.9%	0.3%	100.0%
Reduced training set	2000	2433	2000	2118	1378	876	1251	2000	2037	2520	1423	1707	2105	141	23989
Reduced set ratio	8.3%	10.1%	8.3%	8.8%	5.7%	3.7%	5.2%	8.3%	8.5%	10.5%	5.9%	7.1%	8.8%	0.6%	100.0%

Table1: 14 diseases distribution on Original training set vs Reduced set

We used a "heuristic" approach to reduce the train/validation set and make it more balanced. We kept all images that contains diseases less than 2000 and gradually added other diseases not far beyond 2000. We couldn't make perfect balance because many images contain more than 1 diseases. As a result, the reduced train/validation set contains around 20k images, from which 16,000 were randomly selected into training set and the rest 4,000 images were randomly selected into validation set. There is no overlap between the sets.

Before inputting the images into the network, we downscaled the images to 128×128 and scaled the raw pixel intensities to the range [0, 1].

3.1.2 Structure of CNN

The structure of SmallerVGGNet we used is summarized by Table1. The network contains 5 convolutional layers, with each convolutional layer followed by batch normalization, max pooling and dropout layer. Finally, it ends with two fully connected layers with a dropout layer. We choose this network because it is a simplified VGG16 network. It shows good balance of accuracy and simplicity.

Layer (type)	Output Shape	Param #
Conv2D_1	(None,128, 128, 32)	320
activation_1 (Relu)	(None, 128, 128, 32)	0
batch_normalization_1	(None, 128, 128, 32)	128
max_pooling2D_1	(None, 42, 42, 32)	0
Dropout_1 (Dropout=0.25)	(None, 42, 42, 32)	0
Conv2D_2	(None, 42, 42, 64)	18496
activation_2 (Relu)	(None, 42, 42, 64)	0
batch_normalization_2	(None, 42, 42, 64)	256
Conv2D_3	(None, 42, 42, 64)	36928
activation_3 (Relu)	(None, 42, 42, 64)	0
batch_normalization_3	(None, 42, 42, 64)	256
max_pooling2D_2	(None, 21, 21, 64)	0
Dropout_2 (Dropout=0.25)	(None, 21, 21, 64)	0
Conv2D_4	(None, 21, 21, 128)	73856
activation_4 (Relu)	(None, 21, 21, 128)	0
batch_normalization_4	(None, 21, 21, 128)	512
Conv2D_5	(None, 21, 21, 128)	147584
activation_5 (Relu)	(None, 21, 21, 128)	0
batch_normalization_5	(None, 21, 21, 128)	512
max_pooling2D_3	(None, 10, 10, 128)	0
Dropout_3 (Dropout=0.25)	(None, 10, 10, 128)	0
flatten_1	(None, 12800)	0
dense_1	(None, 1024)	13108224
activation_6 (Relu)	(None, 1024)	0
batch_normalization_6	(None, 1024)	4096
Dropout_4 (Dropout=0.25)	(None, 1024)	0
dense_2	(None, 14)	14350
activation_7 (Sigmoid)	(None, 14)	0

Table2: SmallerVGGNet structure

3.2 Lung disease detection and localization

3.2.1 YOLO algorithm

YOLO uses a single neural network to predict bounding boxes and class probabilities directly. It has excellent runtime performance, because it makes predictions with a single network evaluation. YOLO's main idea is to break down an image into grid cells. At each cell, it uses a fixed number of anchor boxes to detect target object. In this project, we divide each image into 13 x 13 grid cells as shown in Figure 1, where blue bounding boxes are ground truth labels and the red boxes are anchor boxes.

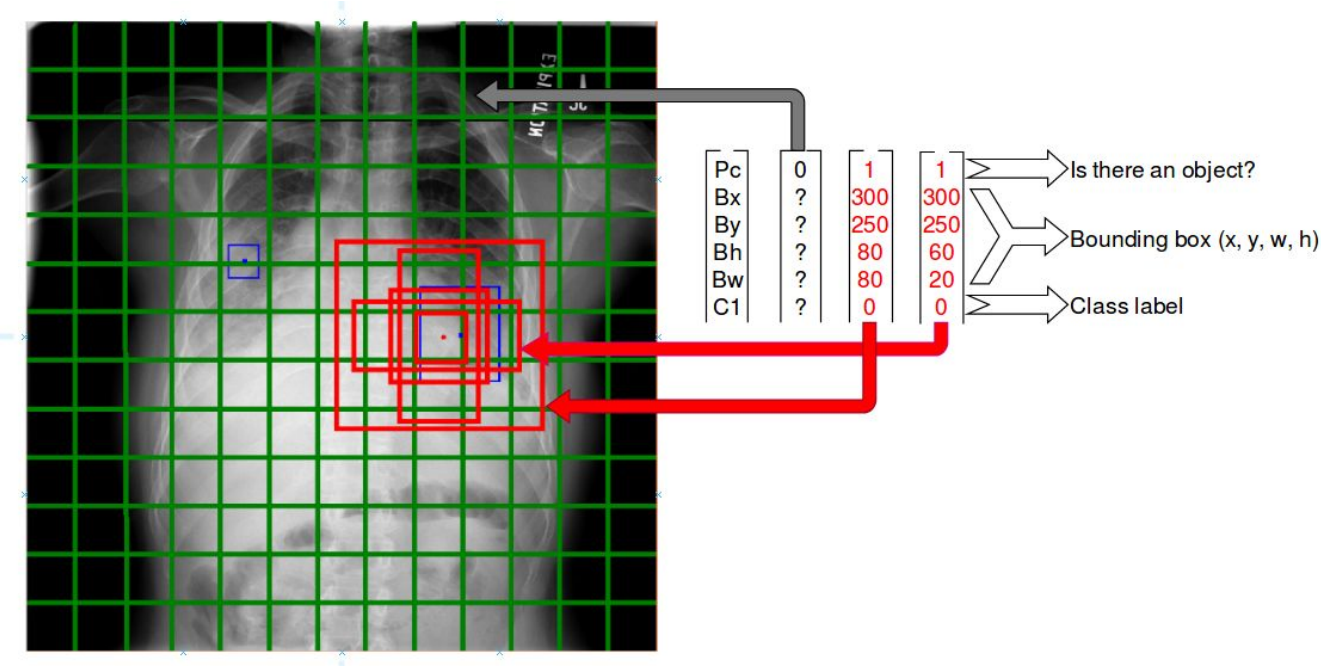


Figure 1: YOLO grids with anchor boxes (red) and ground truth bounding boxes (blue)

For each grid cell, we used the following label for training: $Y = [p_c, b_x, b_y, b_w, b_h, c_i]$, where $P_c = 1$ indicate whether a cell contains a target object; b_x, b_y, b_w, b_h is bounding box center x, y coordinates, width and height; c_i is the class label (0 or 1), where $i \in 0, 1, ..., k-1$ for k classes.

In addition, we used 5 anchor boxes for each cell, to allow the model to specialize better for target objects of different sizes. Given all training images with ground truth bounding boxes, we train a KNN (k nearest neighbor) algorithm with K = 5 to produce 5 different anchor boxes, as shown in Figure 1. As a result, the output label of each grid is extended to contain information for 5 anchor boxes:

$Y = [p_{c0}, b_{x0}, b_{y0}, b_{w0}, b_{h0}, c_{i0}, ..., p_{c4}, b_{x4}, b_{y4}, b_{w4}, b_{h4}, c_{i4}]$

At last, we used intersection over union(IOU = 0.3) and Non-max Suppression(NMS) to clean up multiple boxes that are predicted within a single cell.

3.2.2 Data Analysis and Preprocessing

The training data for YOLO are images with class labels and ground truth bounding boxes. Unfortunately, ChestX-ray14 only comes with very limited (about 1000) ground truth bounding boxes

with some diseases. Distribution of these bounding boxes with disease labels are shown in Table3.

	Nodule	Effusion	Mass	Consolidatio	Edema	Pneumonia	Fibrosis	Atelectasis	Pleural_Thick	Infiltration	Emphysema	Cardiomegaly	Pneumothorax	Hernia	SUM
Bounding box count	79	153	85	0	0	120	0	180	0	123	0	146	98	0	984
Bounding box percentage	8.0%	15.5%	8.6%	0.0%	0.0%	12.2%	0.0%	18.3%	0.0%	12.5%	0.0%	14.8%	10.0%	0.0%	100.0%

Table3: Dataset bounding box label distribution

Out of the 14 diseases, only 8 diseases have labeled bounding boxes. We had to treat the 14 diseases as a single type of lung disease in order to provide enough training samples for detection and localization algorithm. Therefore, we trained a YOLO model to detect whether a given test image has disease and to locate where the disease is in the image.

Since we are lack of training data, we intentionally created more variants randomly from available training data through image augmentation techniques like shifting, rotating, resizing, scaling and mirroring, with bounding boxes adjusted accordingly. These augmentation will provide us up to 5 times more training images.

4. Experimental evaluation and results:

4.1 Multi-label classification

Keras with TensorFlow as backend is used to carry out this classification. We use Adam as optimizer, 32 as batch size and 200 as number of epochs with early stop to train the CNN model. Figure 2 shows the loss and accuracy for training and validation by number of epoch and there is no sign of overfitting. The accuracy of train and validation can both reach about 0.9. Then we applied the trained model to test set and the accuracy is 0.877.

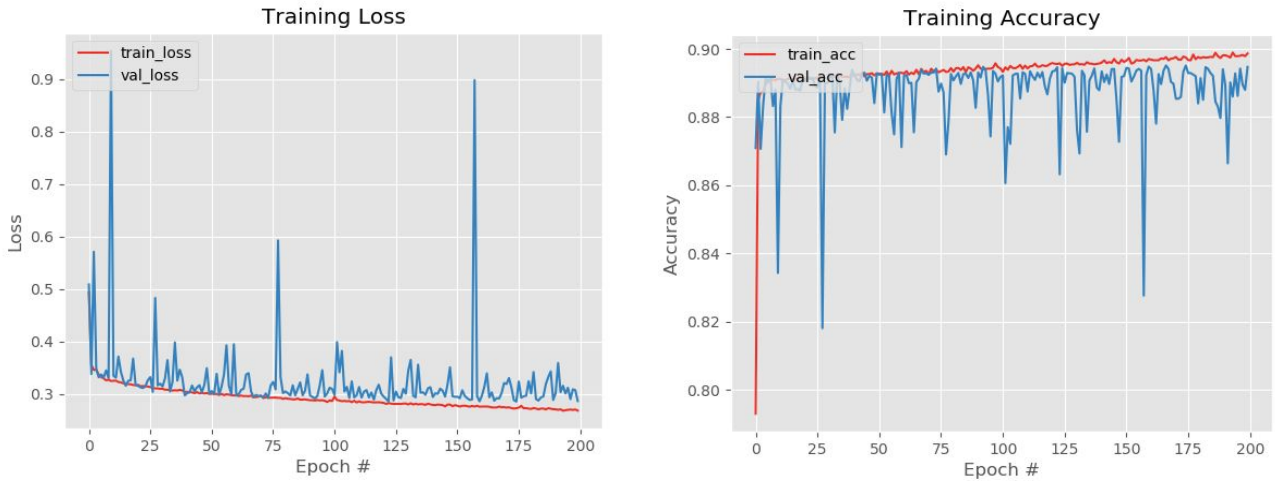
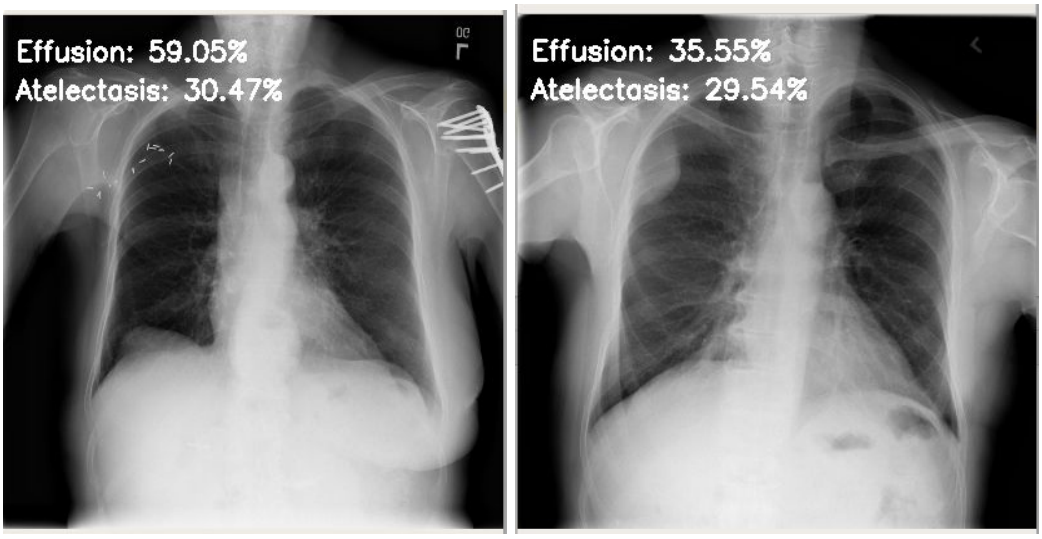


Figure 2: Training and validation loss and accuracy

Figure3 shows the predicted results of 2 sample images from test set, using our best model. We have successfully predicted the left image but failed on the right image.



ground truth: Effusion ground truth: Hernia

Figure 3: Sample classification results

Although we have shown good accuracy on our initial results, we found some test images are not correctly classified. Initial investigation shows that the dataset is very unbalanced on different diseases. Some types of diseases like Atelectasis and Effusion have several times more occurrences than other types of diseases, which cause the bias towards the popular disease type. pa

On the other hand, we have tried to improve our results by using few more complex models like VGG16, ResNet50 and InceptionV3. These models couldn't generalize well to produce better results, since they show sign of overfitting quickly during the training process. The main reason could be unbalanced training images, which still remains one of the biggest challenges with this dataset.

4.2 Lung disease detection and localization

We build a customized YOLO v3 model on Keras with TensorFlow backend, based on our dataset. YOLO v3 is a 106 layer fully convolutional underlying architecture. It makes detections at three different scales. We changed the input channel from original 3 to 1 since our dataset is on gray scale. We also run KNN to extract new anchor boxes, customized for our dataset. To speed up the training process, we applied Transfer Learning and used a pre-trained YOLO weight from [10].

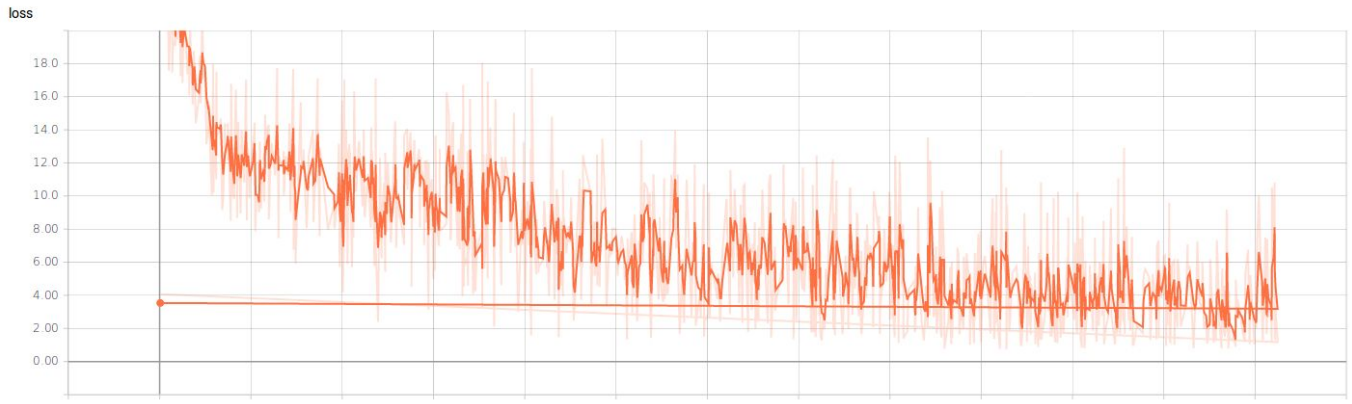


Figure4: YOLO v3 training loss

We trained our model for 200 epoches with early stop and learning rate decay. The training loss was monitored through TensorBoard as shown in Figure4. The training loss is consistently

reduced to less than 3 and stabilized there. Our learning rate was started with 1e-4, reduced to 1e-5 after 125 epoches and further decay to near zero. The learning rate decay can be seen in Figure5.

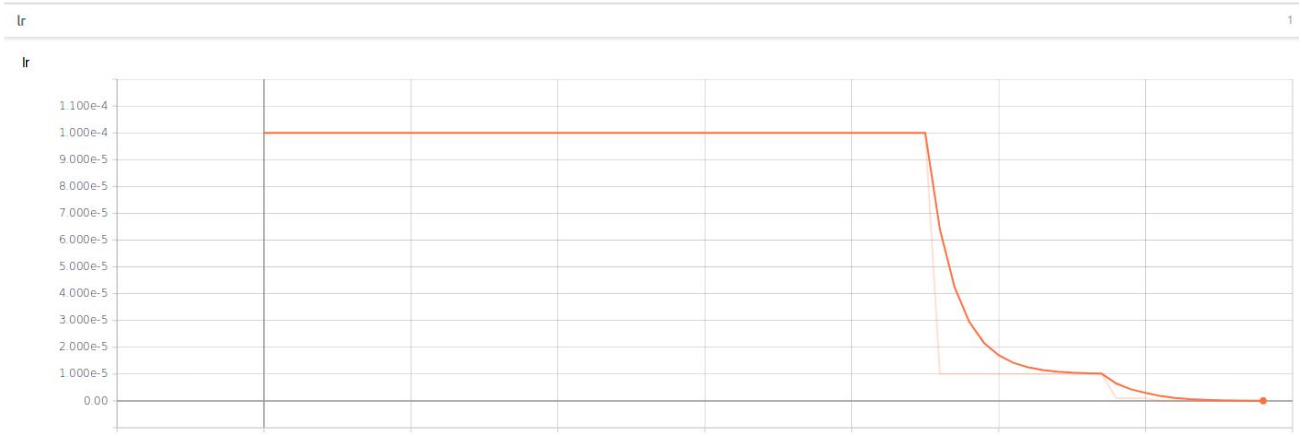


Figure5: YOLO v3 learning rate decay

We use mAP (mean accuracy precision) as our evaluation metric, which is the most popular metric used by most object detection research [7][8][9]. Our final model could achieve mAP () about 0.29, which is much lower than YOLO results on other dataset like COCO and VOC. Considering X-ray image has much lower resolution than other dataset, the result is still considered to be good. In addition, the limited training dataset also contributes to the lower mAP.

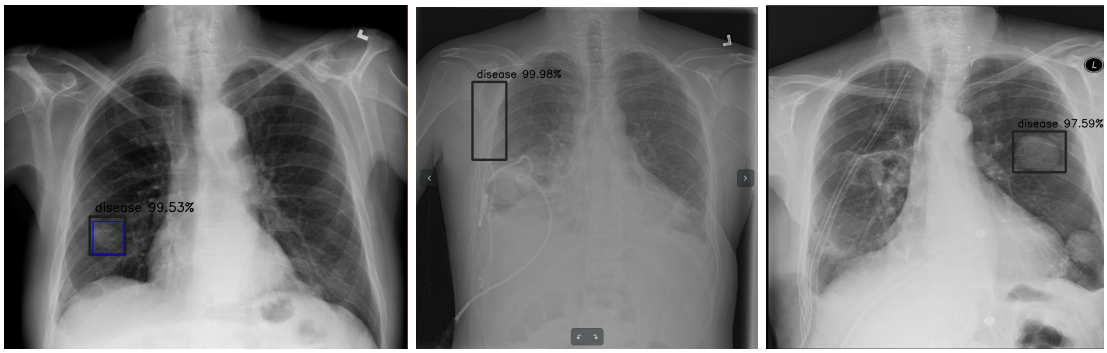


Figure 6: localization result examples

Figure 6 shows 3 good detection results from sample test images. In the left image, the blue box is the ground truth(GT) bounding box while the other bounding box is predicted by our model. It can be seen that the predicted bounding box is overlapped with GT boxes with excellent IOU > 0.9. All 3 results show confident score above 95% with good bounding boxes.

5. Conclusion

We implemented an algorithm to classify 14 diseases and customized YOLO v3 algorithm to detect and localize lung disease on X-ray images, using ChestX-ray14 dataset. The experiments produced reasonably good results on the two problems we were trying to solve. We also found some limitations on current dataset such as unbalanced data categories and limited availability of ground truth bounding boxes. In the future work, we will further improve our results by using more complex models with appropriate regularization to avoid overfitting. In addition, we should create more labeled bounding boxes to improve the detection results.

Reference

[1] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3462–3471. IEEE, 2017.

[2] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, M. P. Lungren, and A. Y. Ng. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. arXiv preprint arXiv:1711.05225, 2017.

[3] Yao, Li, Poblenz, Eric, Dagunts, Dmitry, Covington, Ben, Bernard, Devon, and Lyman, Kevin. Learning to diagnose from scratch by exploiting dependencies among labels. arXiv preprint arXiv:1710.10501, 2017.

[4] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg and F. Li. ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015

[5] Huang, Gao, Liu, Zhuang, Weinberger, Kilian Q, andvan der Maaten, Laurens. Densely connected convolutional networks. arXiv preprint arXiv:1608.06993, 2016

[6] Setio AAA, Traverso A, de Bel T, Berens MSN, van den Bogaard C, Cerello P, Chen H, Dou Q, Fantacci ME, Geurts B, van der Gugten R, Heng PA, Jansen B, de Kasten MMJ, Kotov V, Yu-Hung Lin J, Manders JTMC, So´nora-Mengana A, Carlos Garc´ıa-Naranjo J, Prokop M, Saletta M, Schaefer-Prokop CM, Scholten ETh, Scholten L, Snoeren M, Lopez Torres E, Vandemeulebroucke J, Walasek N, Zuidhof GCA, van Ginneken B, Jacobs C. Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the LUNA16 challenge. arXiv:1612.08012, 2016

[7] Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: CVPR, 2016

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. E. Reed. SSD: single shot multibox detector. CoRR, abs/1512.02325, 2015

[9] Redmon, J., Farhadi, A.YOLO9000: Better, Faster, Stronger. In: CVPR, 2017

[10] Redmon, J. https://pjreddie.com/darknet/yolov2, 2018