Detecting Abnormalities in Athletes' Chest X-rays

Chaoran Huang
Donald Bren School of Information and Computer Science
University of California, Irvine
Irvine, CA, United States
chaorah1@uci.edu

Youling Yu
Department of Control Science and Engineering (College of Electronics and Information Engineering)
Tongji University
Shanghai, China
yuyouling@tongji.edu.cn

*Xia Jun
Shanghai University of Sport
Shanghai, China

***Abstract*—Diagnosing abnormalities in chest X-rays is a challenging problem that is important for human health. The interpretation of the results of radiological examination is not a simple classification problem, and the diagnostic report involves numerous parameters. Chest diseases have a significant impact on the lives and careers of athletes. Doctors also sometimes have trouble in correctly distinguishing chest abnormalities from X-rays due to the complexity of the case or their own carelessness. This study designs a software to help doctors diagnose chest abnormalities in X-rays through neural networks.**

***Index Terms*—Athlete, Health, YOLO, Faster R-CNN, chest X-ray, abnormalities, diagnosis**

I. Introduction

Cardiac problems can significantly impact the performance and lives of athletes. It is necessary for an athlete to have regular health checks, which incur a significant number of medical resources. A study has claimed that "the level of error for clinically significant or major error in radiology is in the range 2%–20% and varies depending on the radiological investigation." [1] if misdiagnosis happen to athletes, it will cause a devastating impact to athletes' career life. During exploratory data analysis, our team discovered that different radiologists tend to arrive at different diagnoses of the same chest X-ray image. Computer-aided systems of detection and diagnosis are important for radiologists to ensure consistent and correct diagnosis, and to reduce the expended medical resources. In this paper, we develop a system based on a neural network model that provides diagnostic suggestions intended to detect and locate abnormalities in chest X-ray images. This that can help doctors avoid incorrect diagnoses of chest diseases and find potential problems.

A. Background and Related Work

Machine learning (ML) and Artificial Intelligence (AI) are widely used in many fields and are thought to offer promise for use in medicine and radiology. In particular in case of primary care and suspicious illnesses, doctors can diagnose marginal cases, which are rare and difficult to diagnose, with higher confidence because of the large amounts of data and the tools afforded by these technologies. To efficiently ensure the health of athletes, we designed some models in past work to assist with diagnosis during health checks for athletes. We used a high-quality dataset from the Kaggle competition, "VinBigData Chest X-ray Abnormalities" [2]. It contained carefully annotated data on many cases. We believe this is a good dataset that can be used for athletes as well.

Table 1: Data collection and preliminary analysis

	Data Statistics
Number of Images	15,000
Size of All DICOM Images	160 GB
Number of Ground Truths, Including No Finding	67,913
Number of Ground Truths, Excluding No Finding	36,096

According to the description provided by the Kaggle competition, "the dataset used in this competition was created by assembling de-identified chest X-ray studies provided by two hospitals in Vietnam: Hospital 108, and the Hanoi Medical University Hospital." The data were in DICOM format, which is a standard file type used in medical communication and management. All DICOM files had been annotated by multiple radiologists and labeled with 15 classes, including 14 abnormality labels and the "no finding" label. (Figure 1) The "training.csv" data (shown in the Figure 2; training.csv file label example) describes an object through the image name, class label, and coordinates of the bounding box. As one image had been annotated by multiple radiologists, several bounding boxes were assigned to the same abnormality.

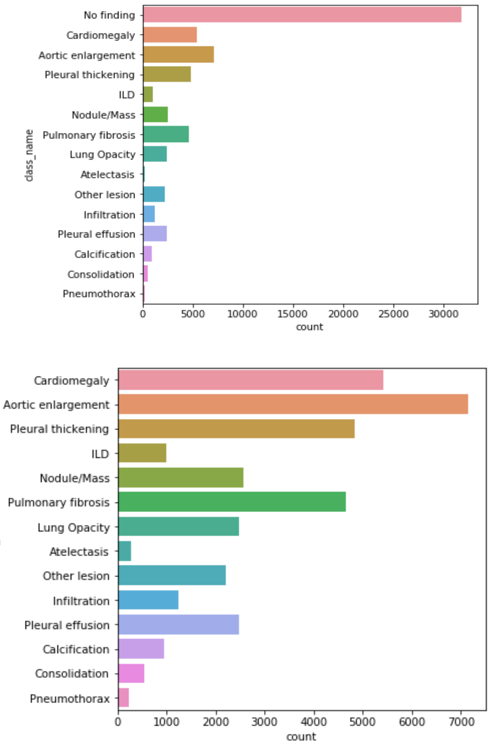
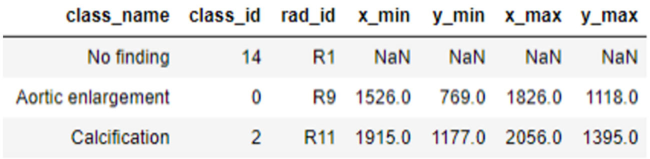


Figure 1. Overview of the dataset

Sponsored by Scientific Research Program of Shanghai Science and Technology Commission (No. 19080503000), National Key Research and Development Project of China (No. 2018YFF0300505)



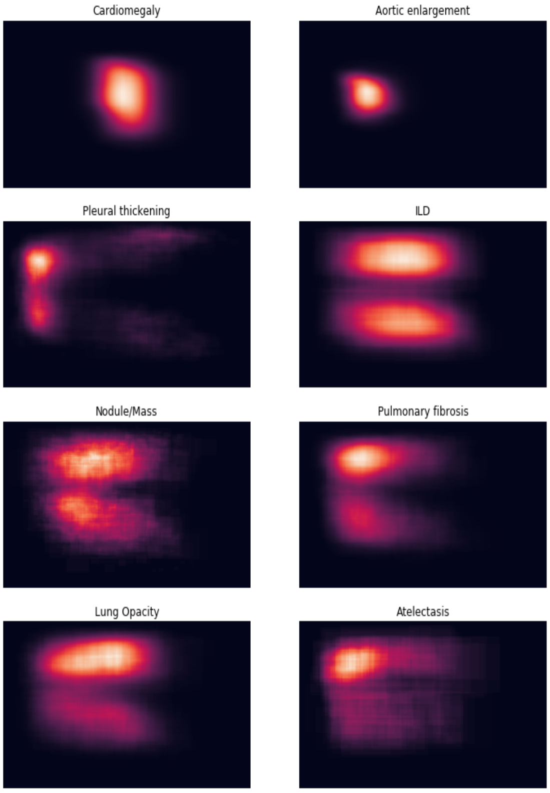
	class_name	class_id	rad_id	x_min	y_min	x_max	y_max
	No finding	14	R1	NaN	NaN	NaN	NaN
	Aortic enlargement	0	R9	1526.0	769.0	1826.0	1118.0
	Calcification	2	R11	1915.0	1177.0	2056.0	1395.0

Figure 2. "Training.csv" labels example

B. Data Processing and Augmentation

During our exploration of the dataset, we discovered that a large number of images labeled with “no finding.” When they were removed, the dataset shrank to 4,394 images. The “no finding” images contained little information and thus were not used in the training process. In addition, most mainstream neural networks do not accept DICOM files as input, and we thus converted them into a NumPy array using the pydicom class. To fit the data with the model and analyze all their parts at the same weight, padding and resize operations were applied to the images. The padding operation converted non-square images to square ones by introducing new pixels to their edges. These images were later resized to 640 × 640.

To reduce overfitting and improve the robustness of our model, some data augmentations were used. After a few runs, we flipped the data with a probability of 0.5, a scale by an additional 0.5 image size, and sheer with 1 degree. We also used translation in our augmentation. In Figure 3, the heatmap generated by ground truth showed that most abnormalities were in fixed positions. Therefore, we used a small value of 0.1 for image resizing for translation. This improved the capability of the model to detect objects in the dataset.



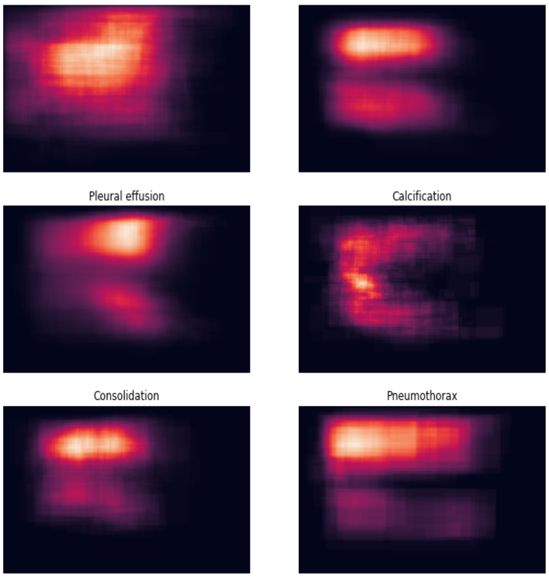


Figure 3. Heatmaps of all 14 abnormalities

We also separated the training data into training and validation sets with a ratio of 0.8 to 0.2 for a better result.

II. PROPOSED METHOD

We propose two models to solve the problem: one based on YOLOv5, and the other a PyTorch-based one-stage detector.

The network architecture of YOLOv5 is similar to YOLOv4 [3]. They both use CSPDarknet53 as the backbone for pretraining data, PANET, and SPP as the neck to collect feature maps and YOLOv3’s head to predict classes and bounding boxes shown in Figure 4. Also, the sub-model YOLOv5 s/m/l/x share the same architecture. The only difference is that the model uses *depth_multiple* and *width_multiple* to control its depth, has a different number of filters. In the subsequent training, different models were tested on YOLOv5.

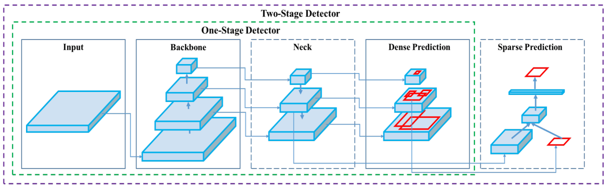
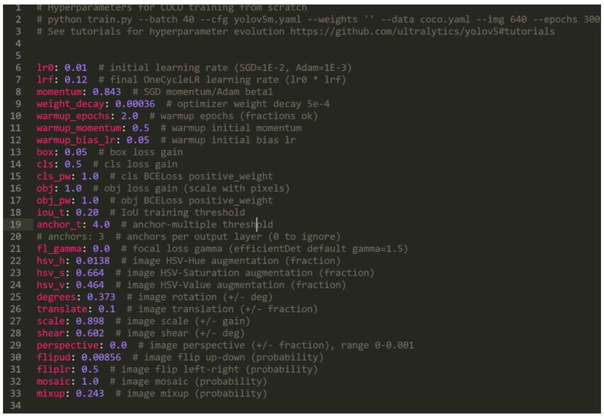


Figure 4. Architecture of different detectors[3]



```
# Hyperparameters for COCO training from scratch
# python train.py --batch 40 --cfg yolov5m.yaml --weights '' --data coco.yaml --img 640 --epochs 300
# See tutorials for hyperparameter evolution https://github.com/ultralytics/yolov5#tutorials


lr0: 0.01  # initial learning rate (SGD=1E-2, Adam=1E-3)
lrf: 0.12  # final OneCycleLR learning rate (lr0 * lrf)
momentum: 0.843  # SGD momentum/Adam beta1
weight_decay: 0.00036  # optimizer weight decay 5e-4
warmup_epochs: 2.0  # warmup epochs (fractions ok)
warmup_momentum: 0.5  # warmup initial momentum
warmup_bias_lr: 0.05  # warmup initial bias lr
box: 0.05  # box loss gain
cls: 0.5  # cls loss gain
cls_pw: 1.0  # cls BCELoss positive_weight
obj: 1.0  # obj loss gain (scale with pixels)
obj_pw: 1.0  # obj BCELoss positive_weight
iou_t: 0.20  # IoU training threshold
anchor_t: 4.0  # anchor-multiple threshold
# anchors: 3  # anchors per output layer (0 to ignore)
fl_gamma: 0.0  # focal loss gamma (efficientDet default gamma=1.5)
hsv_h: 0.0138  # image HSV-Hue augmentation (fraction)
hsv_s: 0.664  # image HSV-Saturation augmentation (fraction)
hsv_v: 0.464  # image HSV-Value augmentation (fraction)
degrees: 0.373  # image rotation (+/- deg)
translate: 0.1  # image translation (+/- fraction)
scale: 0.898  # image scale (+/- gain)
shear: 0.602  # image shear (+/- deg)
perspective: 0.0  # image perspective (+/- fraction), range 0-0.001
flipud: 0.00856  # image flip up-down (probability)
fliplr: 0.5  # image flip left-right (probability)
mosaic: 1.0  # image mosaic (probability)
mixup: 0.243  # image mixup (probability)

```

Figure 5. Hyperparameters of YOLOv5

There were three decisions to make when training YOLOv5, including those related to the optimizers and hyperparameters used. The two built-in optimizers in YOLOv5 are the SGD and ADAM. The SGD was selected because of its good performance in terms of optimizing models of large datasets. Moreover, YOLOv5 provides users with various hyperparameters. One of them is the learning rate. We determined the initial learning rate of all models to 0.01 because this is recommended by the developers of YOLOv5.

The second model we used was Faster R-CNN.

Faster R-CNN outperforms other models in the RCNN family in terms of efficiency. As shown in Figure 4, Faster R-CNN is different from YOLOv5 because it is a two-stage detector. Its model architecture consists of four building blocks. The first is a region proposal algorithm that generates the bounding boxes of objects in the image, the second one is a feature extraction algorithm, the third is a classification layer that predicts the classes to which the given object belongs, and the fourth is a regression layer that increases the precision of the coordinates of all bounding boxes.

Faster R-CNN can be considered a pipeline that uses a regional proposal network (RPN) to determine the region, and then uses a Fast R-CNN to identify these regions. The RPN improves the temporal efficiency of the model, such that the system uses only 10 ms to scan each image, and some layers of the network still be used for subsequent object detection.

We used anchors on the input image for each location in the output feature map using the backbone network. These anchors indicated possible objects of various sizes and aspect ratios at the given location.

III. Experiments And Results

A. Evaluation

We evaluated our model in terms of precision, recall, and mAP@0.5. The mAP is derived from the precision–recall graph, and the IoU threshold was set to 0.5.

Equation 1 mAP@0.5

$$mAP@0.5 = \frac{\sum_i^{\#\_of\_classes} \int {}_0^1 P_i(R_i)dR_i}{\#\_of\_classes} \mid IoU_{threshold} = 0.$$

Equation 2 Precision

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

Equation 3 Recall

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

Table 2. Evaluation

	Relevant	*Not Relevant*
Retrieved	True Positive	False Positive
Not Retrieved	False Negative	True Positive

If radiologists deploy an object detection network like the one proposed here, they should expect it to identify abnormalities in X-ray images. This coincides with the precision score, which describes the ability of a neural network to output as many correct bounding boxes as possible. Therefore, our goal was to generate models that output high precision scores. However, the model still requires a satisfactory recall score to output a minimal number of false-negative bounding boxes.

B. Results

YOLOv5: The developers of the PyTorch-based YOLOv5 released four major variants of their model. The table shows that as the size of the model increases, its performance improves. We augmented our images and chose the model the delivered the best performance. To select the model for the next stage of training, we selected YOLOv5xl because it had the highest PR and mAP scores.

Table 3. Results of YOLOv5

Best mAP@0.5 from 100 epochs				
	Precision	***Recall***	***mAP@.5***	***mAP@.5:.95***
YOLOv5s (Epoch 35)	.3917	.3418	.3173	.1353
YOLOv5m (Epoch 21)	.4268	.3477	.3212	.1316
YOLOv5l (Epoch 28)	.4120	.3502	.3256	.1423
YOLOv5x (Epoch 20)	.4215	.3594	.3324	.1461
YOLOv5x w/augmentation (Epoch 61)	.4661	.34	.3408	.1533

The results show that the augmented model incurred a smaller validation loss and converged much later than the model that had not been augmented. Although the augmented models required more epochs, it had higher precision and mAP scores at the expense of their recall scores. This coincides with our goal described earlier: that of a high precision and satisfactory recall.

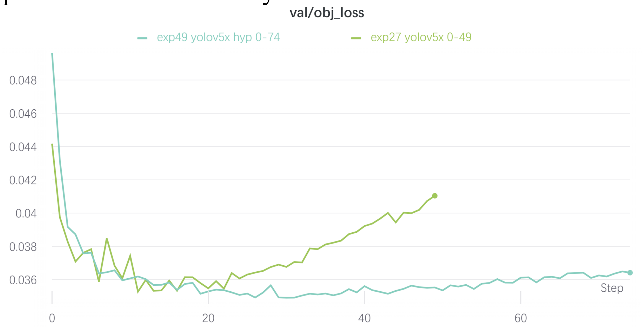


Figure 6. The loss of validation using YOLOv5x with and without augmentation

Data augmentation also increased the number of class labels with few images (atelectasis, pneumothorax), and thus considerably improves the mAP score for detection. The mAP and PR scores on every class are shown in Figure 4.

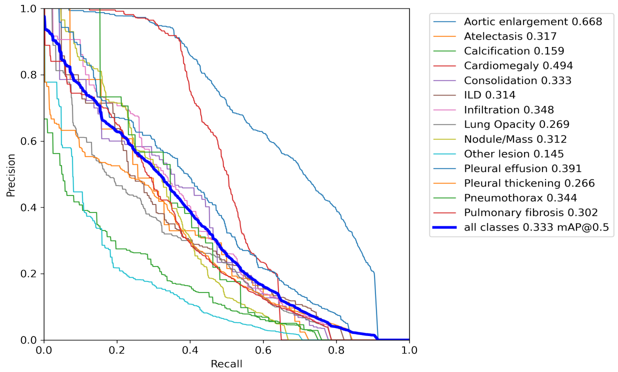


Figure 7. The precision–recall curve of YOLOv5x with augmentation

Faster R-CNN: In the context of detecting abnormalities, a false negative means that the model has failed to identify chest abnormalities, which is an unwanted outcome. We aim for more false positives than false negatives. The goal of our abnormality detection model is to increase the precision as much as possible, even at the expense of part of the recall score.

Table 4. Results of Faster R-CNN

Results of Faster R-CNN			
Precision (macro)	***Precision (micro)***	***Precision (weighted)***	***Recall***
.1580	.7763	.7956	.1548

C. Conclusions Regarding the Models

We systematically evaluated the performance of different models on the dataset. First, we assessed several models based on YOLOv5. There were five networks. The experiments showed that all models converged before 50 epochs, and featured overfitting. YOLOv5x took the longest to train but delivered the best performance. After applying data augmentation to YOLOv5x, the degree of overfitting was reduced. The models subsequently significantly improved in terms of the detection of abnormalities with few labels. Second, we evaluated the performance of Faster R-CNN. The trained model was characterized by a high precision score but a low recall score. According to these results, YOLOv5 had a higher recall score but a lower precision score on the given dataset while Faster R-CNN had a higher precision score but a lower recall score. Both models require more training to achieve higher accuracy, possibly through the evolution of hyperparameters. Although these models were unable to deliver highly precise diagnoses, they can still help physicians and technicians determine chest abnormalities in athletes.

IV. DISCUSSION

The ground-truth images shows that there were many small bounding boxes. However, due to restrictions on the hardware, we could afford only training models with images of size 640 × 640. If the hardware can be improved, we should be able to use images of size 1024 × 1024 to further train the model and examine the influence on the PR and mAP scores.

When examining the predictions of YOLOv5x, we discovered two intriguing scenarios. Some labels predicted by our model with high confidence had not been labeled by radiologists in the ground-truth images. For example, our model predicted "aortic enlargement" with a confidence level of 0.42 for one case. However, even though a similar pattern was labeled as "aortic enlargement" in the ground truth, this pattern did not appear in the ground-truth image. Physicians might have overlooked this abnormality when labeling the images. Moreover, the same abnormalities had been labeled multiple times (see Figure 6), possibly because 14 radiologists had participated in the labeling process. These labeled bounding boxes did not overlap correctly, and thus might have negatively affected the accuracy of the model. We eliminated some bounding boxes to ensure that each abnormality had only one bounding box. If most radiologists agree on a label, replacing the incorrect label for the given bounding box with the correct one might be helpful. Moreover, although neither of the proposed models achieved a sufficiently high accuracy to supplant expert human judgment, they can be used as a recommendation system. From the training/validation process, we collected confidence of each X-ray image. When a new X-ray image is detected by the system, the system will also provide several X-ray images with similarities so that doctors can compare their X-ray image with other X-ray images to assist their diagnosis.

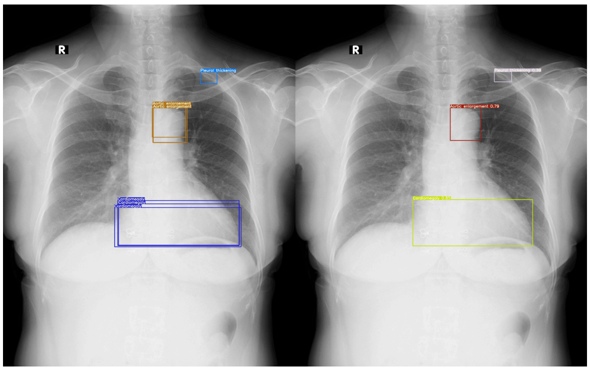


Figure 8. A sample of ground-truth predictions of YOLOv5x

V. CONCLUSION

The work here on detecting abnormalities in the chest X-rays of athletes can help them and healthcare professionals, such as pulmonologists and endocrinologists. Although our system is not yet mature, it has already demonstrated the promise of using neural network techniques to assist human experts in diagnosing abnormalities in chest X-rays. We evaluated the two proposed neural network models, and our results confirmed the feasibility of detecting chest abnormalities using them. We also proposed a diagnosis suggestion which is provide radiologists similar X-ray images to perfect the detection and decrease mis-diagnose possibilities. Therefore, it evidently shows that it will be an innovative system with plenty of room to improve and will be beneficial to athletes to keep them healthy and improve performance. The GitHub link for our models is https://github.com/ChaoRanHuang97/Athlete-X-ray-abnormality-dectection.

ACKNOWLEDGMENT

We thank Kaggle.com for providing its dataset, VinBigData Chest X-ray Abnormalities.

Data Availability Statement

The datasets analyzed for this study are available at https://www.kaggle.com/c/vinbigdata-chest-xray-abnormalities-detection.

References

[1] Goddard P, Leslie A, Jones A, Wakeley C, Kabala J. Error in Radiology. The British Journal of Radiology 2001; 74, 949-951.

[2] VinBigData Chest X-ray Abnormality Detection: Automatically localize and classify thoracic abnormalities from chest radiographs: https://www.kaggle.com/c/vinbigdata-chest-xray-abnormalities-detection

[3] Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao., "YOLOv4: Optimal speed and accuracy of object detection." Cornell Computer Science, 2004.