Image Classification using YOLO5

Anonymous

ABSTRACT

Image classification is one of the most common use cases of Computer Vision. In our project, we have used YOLO5, a pre-trained model. We have conducted tests using Failure Dataset and improved the efficiency using augmentation. We have evaluated and presented the results based on our experiments and concluded that after augmentation of images results in better image classification. In this report, we have also presented a few other observations based on our evaluation.

1. INTRODUCTION

Today, the world is full of images. Thus, image classification has a wide range of application such as analysing a crime scene, traffic control and ticketing system, etc. [6] Artificial Neural Networks have proven to be the most efficient for image classification. In this project, we performed image classification using pre-existing trained model called - YOLO (You Only Look Once). [3] In addition to thousands of training hours, YOLO utilizes anonymised datasets to develop its algorithm and continuously evolve and currently it is in version 5. [5]

In the second, we analyzed the performance by taking a single image from each video. In this report, we give the related work in section 2. In section 3, the method which we opted for is described. In order to test this model, we conducted experiments using the Failure Dataset provided by the course. For our testing purposes, we used two different approaches. First, we took all the frames as images from a complete video and analyzed the model's performance. We describe our experiments and present the results in section 4. Finally, the conclusion of the report is explained along with the novelty in our work in section 5.

2. RELATED WORK

YOLO is a widely used detection model which evolved over the years. Wang et al. [7] examined the performance of Convolutional Neural Networks with a Cross Stage Partial Network backbone. The research demonstrated positive results of a reduction in around 80% bottlenecks using YOLOv3. This is further employed in both YOLOv4 and YOLOv5, i.e these versions use CSP Bottleneck to calculate image features. Multiple research is done on top of YOLO, like face mask-wearing detection Yu et al. [8], weed detection by Ying et al. [1]., etc.

Other famous object detection models - SPP-net, Single Shot Detector, etc. Spatial Pyramid Pooling (SPP-net) [2] was developed by Kaiming et al. which produced fixed-length representations no matter how small or big the images are. Kaiming et al. developed Single Shot Detector(SSD) [4] and it used a single deep neural network, hence the name. In this model, the output space of bounding boxes is approximated by a set of default boxes for different aspect ratios. Further, it combines predictions from multiple feature maps to handle inputs of different sizes.

3. METHOD

YOLO does object detection, object localization, and image classification. As the name suggests in this model, you only look once at the image to detect it, and thus it is extremely fast. It is a state-of-art object detection system developed by J.Redmon et al. It integrates separate components of object detection into one neural network and it uses features from the whole image for predicting the objects' bounding boxes. Images are split into cells and each cell predicts 5 bounding boxes. Non-max suppression is applied on these bounding boxes, to remove the ones with low probability and create only one with the highest shared area.

In the initial model i.e in version 1, it has 24 convolutional layers and 2 fully connected layers, takes an input size of 448 x 448 x 3, and used ReLU and Linear Func (final layer) as activation functions. Since then, it was improved over time with 5 versions. In YOLOv5, the network is divided into three main pieces - Backbone, Neck, and Head. Backbone is a convolutional neural network, it creates image features. The neck contains series of layers that combine features and forward them for prediction. Head takes the features from the neck and performs the class and box prediction steps. For training, the different augmentations like - scaling, color space, and mosaic augmentations are performed on the data to improve the model's performance. In Mosaic data augmentation four different training images are combined into one in certain ratios. This helps locate images in different portions of the frame and identify smaller-scale images.

4. EXPERIMENTS AND EVALUATION

In our experiments, we have used the pre-trained model YOLOv5 and performed various tests on it and observed the classification accuracy. To evaluate our experiments, we are using accuracy, precision and recall. As we are doing multi-class classification, we are calculating the average accuracies for each class and then taking an average of them to get the accuracy of the model. In calculating these evaluations, we need True Positive(TP), True Negative(TN), False Positive and False Negatives. If a object is present and the model

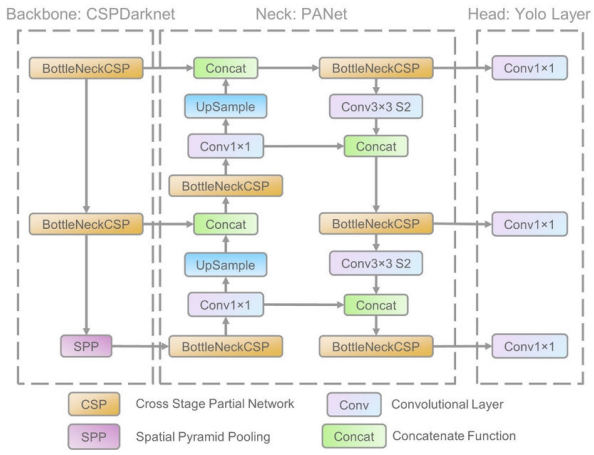


Figure 1: Network Architcture of YOLOv5

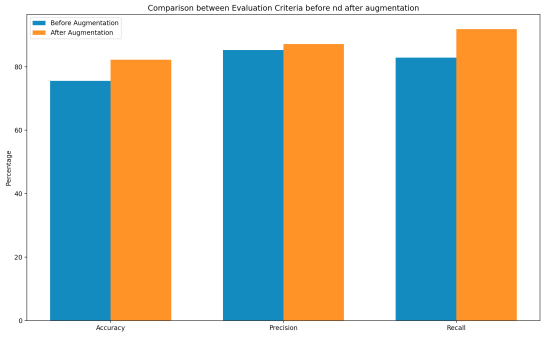


Figure 2: Accuracy before and after augmentation

is detecting it, we are considering it as a True Positive and if the object is not present and the model is still detecting it, we considering it as False Positive. If the object is not present and the model is not detecting it, then it is a True Negative and if the object does not exist and model is still detecting it then it is a False Negative. In addition, we also performed our experiments on augmented data and observed the results.

4.1 Approach 1

In this approach, 25 images have been extracted from 25 random videos of Failure Dataset. Most of these images are clear whereas some are blurred. These images are mostly centered around humans of ages ranging from infants to old people. However, there are about 20% of the images that do not consist any person. After running the basic object detection, we observed that in some cases even minute details have been used to detect the object correctly. The vice-versa is also true. Some very evident objects have been detected incorrectly. Some examples can be seen in Figure 4

Overall Accuracy has been calculated using the criteria - Accuracy, Precision and Recall. The detect.py file of this YOLO model has been run using the –augment parameter to analyse the object detection accuracy after augmentation of these images. For the person class alone, which constitutes the majority of the object detection, the accuracy obtained before augmentation was about 75% while after augmentation, the accuracy was 82%. Thus, we observed an increase in accuracy by approximately 7%. Precision is defined as the ratio of correctly predicted positive observation to the total number of positive observations. For the person class, the precision before augmentation was 85.29% while after augmentation it was 87.17%. Thus, the precision of the model has also improved. Recall is the sensitivity of the model. In our case, it is the ratio of the correctly classified objects with all the objects classified correctly and not detected at all. The recall before augmentation was close to 83% while after augmentation, it was 93%. Figure **??** shows a bar graph clearly showing the increase in accuracy after augmentation. Figure 3 shows the number of objects detected correctly for each image before and after augmentation.

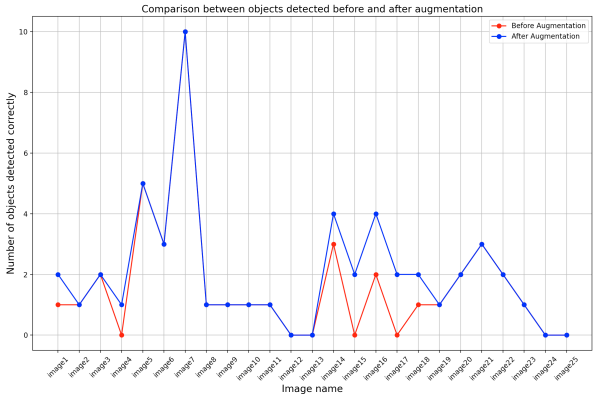
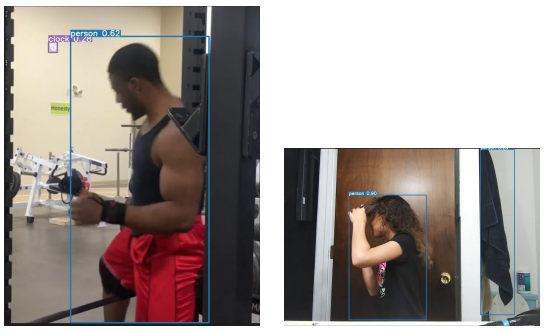


Figure 3: comparison between objects detected before and after augmentation



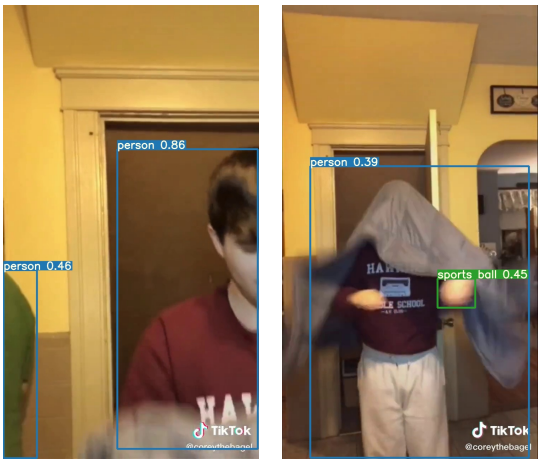
(a) Correct Detection (b) Incorrect Detection

Figure 4: Examples from Approach1 - using YOLOv5 Image Classification

4.2 Approach 2

In experiment 2, we have used video fixingrepairs345 from the dataset. It has 584 frames containing mainly two persons and a chair. In our experiment, the model performed very well in detecting these main objects, even when they are barely visible in the picture 5(a). In addition to these, the model detected some objects which are not present in the frames. In some images, the model predicted the elbow of a person as a sports ball 5(b), a frame in the background as a mobile phone, some small figure in the background as a person. Other than these incorrect detections, the model

gave very good accuracy on this test data. For the class person, the accuracy is 0.967 and for the class chair, the accuracy is 0.9811, totally giving an average accuracy of 0.974.



(a) Correct Detection (b) Incorrect Detection

Figure 5: Examples from Approach2 - using YOLOv5 Image Classification

5. CONCLUSION

The working of the YOLO model has been tested with random images from the failure dataset and with all the frames of a video from the dataset. The evaluations are calculated and plotted for both normal data and augmented data. Comparisions between correctly identified and incorrectly identified images are presented. From these evaluations and visualizations, it is evident that data augmentation sometimes helps to detect smaller and other objects correctly, and sometimes more objects are detected but some of them might not actually be in the image. There should be a balance in the data augmentation. Also, we have observed that when images are blurred, fewer objects are detected. This seems to be a limitation which needs to be looked into by modifying the training set and using blurred images as well in the training data. What is novel about our work is that we have used various approaches to test the model and come up with evaluation and limitations. We have also used the augmentation technique and showed that, augmentation does increase the accuracy of object detection.

5.1 Contributions:

We, as a group have divided most of the work equally amongst the two of us. Sri Rachana was responsible for Approach 1 while Amrutha Varshini was responsible for experimenting using Approach 2. Sri Rachana focused on evaluating using multiple criteria and presenting visualization while Amrutha focused on the literature review and also the model review. Thus, we were both on an even foot.

6. REFERENCES

[1] Weed detection in images of carrot fields based on improved yolo v4. *Traitement du Signal*, 38:341–348, 04 2021.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916, 2015.

[3] G. Jocher, A. Stoken, J. Borovec, NanoCode012, A. Chaurasia, TaoXie, L. Changyu, A. V, Laughing, tkianai, yxNONG, A. Hogan, lorenzomammana, AlexWang1900, J. Hajek, L. Diaconu, Marc, Y. Kwon, oleg, wanghaoyang0106, Y. Defretin, A. Lohia, ml5ah, B. Milanko, B. Fineran, D. Khromov, D. Yiwei, Doug, Durgesh, and F. Ingham. ultralytics/yolov5: v5.0 - YOLOv5-P6 1280 models, AWS, Supervise.ly and YouTube integrations, Apr. 2021.

[4] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[6] N. Sharma, V. Jain, and A. Mishra. An analysis of convolutional neural networks for image classification. *Procedia Computer Science*, 132:377–384, 2018. International Conference on Computational Intelligence and Data Science.

[7] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh. Cspnet: A new backbone that can enhance learning capability of cnn. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 390–391, 2020.

[8] J. Yu and W. Zhang. Face mask wearing detection algorithm based on improved yolo-v4. *Sensors*, 21:3263, 05 2021.