International Journal of Advanced Smart Convergence Vol.11 No.2 22-29 (2022)
http://dx.doi.org/10.7236/IJASC.2022.11.2.22

IJASC 22-2-4

Fundamental Function Design of Real-Time Unmanned Monitoring System Applying YOLOv5s on NVIDIA TX2™ AI Edge Computing Platform

SI HYUN LEE

Professor, Department of Information and Communication, Dong Seoul University, Seoul, Korea
lsh4185@du.ac.kr

Abstract

In this paper, for the purpose of designing an real-time unmanned monitoring system, the YOLOv5s (small) object detection model was applied on the NVIDIA TX2™ AI (Artificial Intelligence) edge computing platform in order to design the fundamental function of an unmanned monitoring system that can detect objects in real time. YOLOv5s was applied to the our real-time unmanned monitoring system based on the performance evaluation of object detection algorithms (for example, R-CNN, SSD, RetinaNet, and YOLOv5). In addition, the performance of the four YOLOv5 models (small, medium, large, and xlarge) was compared and evaluated. Furthermore, based on these results, the YOLOv5s model suitable for the design purpose of this paper was ported to the NVIDIA TX2™ AI edge computing system and it was confirmed that it operates normally. The real-time unmanned monitoring system designed as a result of the research can be applied to various application fields such as an security or monitoring system. Future research is to apply NMS (Non-Maximum Suppression) modification, model reconstruction, and parallel processing programming techniques using CUDA (Compute Unified Device Architecture) for the improvement of object detection speed and performance.

Keywords: *AI, Object Detection, NMS, YOLOv5s*

1. Introduction

With the recent development of the 4th industry, SoC (System-on-a-Chip), computer hardware performance, and cloud computing technology has been rapidly developed. Therefore, the developed artificial intelligence (AI) technology and systems were applied in various application fields such as autonomous vehicles, weapon systems, robot, home appliances, and medical devices. In various fields where artificial intelligence is applied, computer vision includes semantic segmentation, classification + localization, object detection, and instance segmentation.

In this paper, the purpose of this study is to design a real-time unmanned security system on NVIDIA TX2™ AI edge computing platform[1]. The objection detection model applied to the real-time unmanned monitoring system, the YOLOv5 model, a one-stage object detection model, was applied based on the results of previous studies[2],[3],[4]. In addition, in this study, the performance of four YOLOv5 models (small, medium, large,

Manuscript Received: March. 10, 2022 / Revised: March. 13, 2022 / Accepted: March. 15, 2022
Corresponding Author: lsh4185@du.ac.kr
Tel: +82-31-720-2087, Fax: 82-31-720-2286
Professor, Department of Information and Communication, Dong Seoul University, Seoul, Korea

and xlarge) was evaluated, and based on this, the YOLOv5s model was applied to the system without modification. In future research, several research directions were suggested to improve its performance. (See Fig. 2). The research results can be directly applied to various application fields that detect and recognize objects in real time.

The structure of this paper is as follows. Following the introduction of Section 1, Section 2 analyzes the related works on the real-time object detection algorithm, Section 3 defines the design requirements of the real-time unmanned monitoring system, and Section 4 Performance evaluation and results shows 4 models that is, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. The performance of the models is evaluated and ported to a hardware system based on the evaluation and results. In the conclusion of Section 5, the research results and future research directions are explained.

2. Related Works

Early object detection research has been studied in a way based on SIFT (Scale Invariant Feature Transform), SURF (Speeded-Up Robust Features), HOG (Histogram of Oriented Gradients), etc. that finds objects by designing and detecting features. After that, in the Deformable Part-based Model (DPM), the object recognition performance was improved by dividing the object into several parts to compose feature information, and combining it with a machine learning method such as SVM (Support Vector Machine).

At the ImageNet 2012 competition, the convolutional neural network (CNN) showed better performance than the existing method, and the method of recognizing objects using deep learning began to receive a lot of attention. ZFNet, GoogLeNet, VGG, ResNet, DenseNet, etc. have been studied to construct a deeper network for the purpose of improving the recognition rate of CNNs. Meanwhile, in addition to the study of recognizing objects using CNNs in images, object detection methods for various applications such as autonomous vehicles, robots, and drones have begun to be studied.

Deep learning-based object detection techniques are to solve the problem of predicting the location of an object and the problem of identifying the detected object. Object detection can be divided into one-stage and two-stage based on object detection algorithms according to the method of performing two problems in this process. One-stage object detection algorithm is an algorithm that can process two processes at the same time, and two-stage object detection algorithm processes two processes sequentially. In general, the one-stage method has a disadvantage of faster processing speed but lower accuracy compared to two-stage method. The one-stage-based object detection algorithm includes YOLO (You Only Look Once), SDD (Single Shot MultiBox Detector), RetiaNet, ExtremeNet, CenterNet, etc. The two-stage-based object detection algorithm is R-CNN, Fast R- CNN, Faster R-CNN, Mask R-CNN, etc.

R-CNN is an algorithm for finding and classifying a region containing an object by a deep learning regression method using a CNN structure[5]. Fast R-CNN[6] was developed to improve the slow detection speed, which was a disadvantage of R-CNN, but it had the disadvantage that deep learning could not be used to find the candidate region of an object. By solving this problem, Faster R-CNN[7] could improve not only the detection speed but also implement object recognition using only deep learning. In addition, R-FCN [8], which supplemented the shortcomings of Faster R-CNN, has been developed. Although the object recognition speed was greatly improved in these related studies, there was a limit to its application to applications requiring fast processing (robots, autonomous driving, etc.).

So, in YOLO, to solve the speed problem, we proposed a method of composing all processes of object recognition into one deep learning network. YOLO has been developed into YOLOv1, YOLOv2, YOLOv3, YOLOv4, and YOLOv5 since its development in 2015.

On the other hand, methods with fast detection speed and operability even in mobile devices such as SSD

[9] announced at ECCV in 2016 have been proposed. In addition, other one-stage detectors include RetinaNet [10] in 2018, and CernterNet[11] in 2019. Also, CornerNet and ExtremeNet were published around the same time. YOLOv5 was introduced with four models: YOLOv5s (small), YOLOv5m (medium), YOLOv5l (large), and YOLOv5x (xlarge). The basic structure of the four models of YOLOv5[12],[13],[14] is the same architecture for both the backbone and the head. Fig. 1 is the structure of YOLOv5s model used in this study[15], [16], [17].

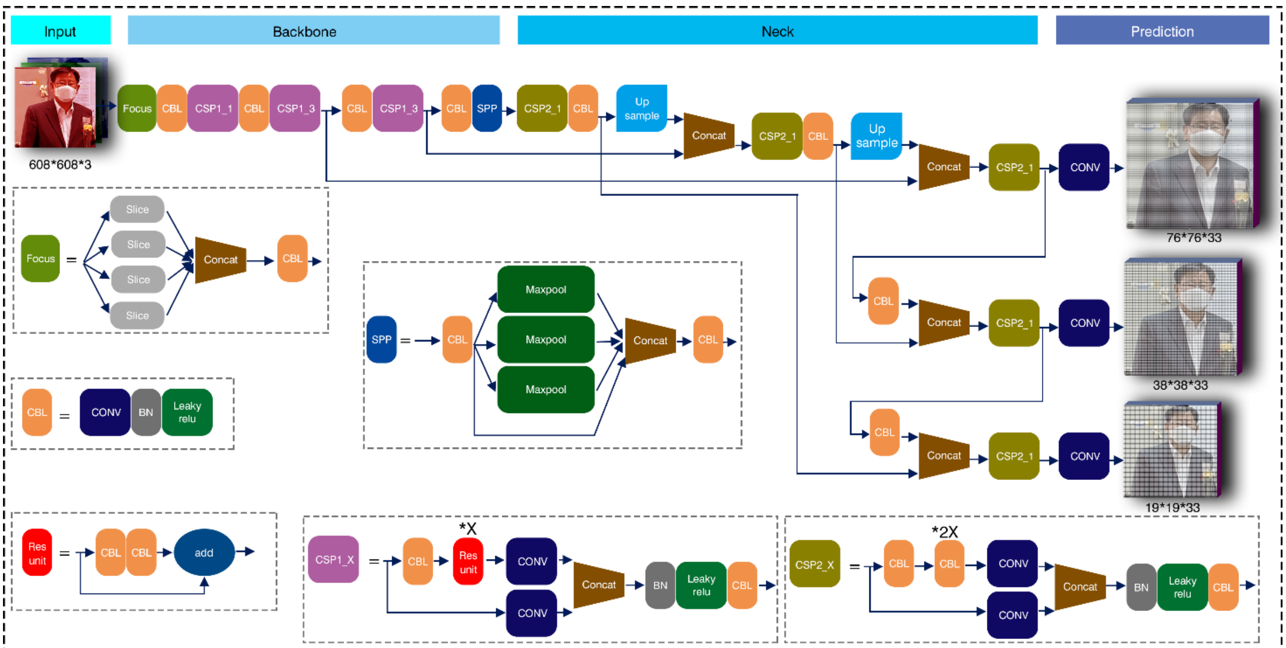


Figure 1. The structure of YOLOv5s model used in this study.

3. Real-time Unmanned Monitoring System Design

3.1 Requirements of Real-time Unmanned Monitoring System

The requirements for the entire system to be developed in this study are as follows.

·The detection of objects should be processed in real time.

·The developed system should be able to be implemented in a small and low-cost type.

·The system must be able to distinguish the type of object accurately.

·It should be possible to use an external recording device that makes it easy to add and remove objects and store the results.

·The system design should be designed considering scalability for future system performance improvement.

·It should be designed for the places that don’t need many types of objects and can be used in a designated range of places (e.g., building entrances, apartment entrances, etc.).

·This study is to implement fundamental functions (step 1 out of 3 steps in Fig. 2) in the final development system.

3.2 Design of Real-time Unmanned Monitoring System

Step-1 design in Fig. 2 is the contents to be developed in this study. The overall structure of the real-time unmanned monitoring system is Fig. 2. In the structure (Fig. 2), the following is designed.

·Object detection model performance test

·Performance evaluation of 4 models (s, m, l, x) of YOLOv5

·Porting in NVIDIA TX2^{TM} AI edge computing platform

In the step-2 design of Fig.2, the following items are designed based on the research results in the overall system structure (Fig. 1).

·NMS (Non Maximum Suppression) model reconstruction

·Reconstructing the model to improve the performance of the YOLOv5s model

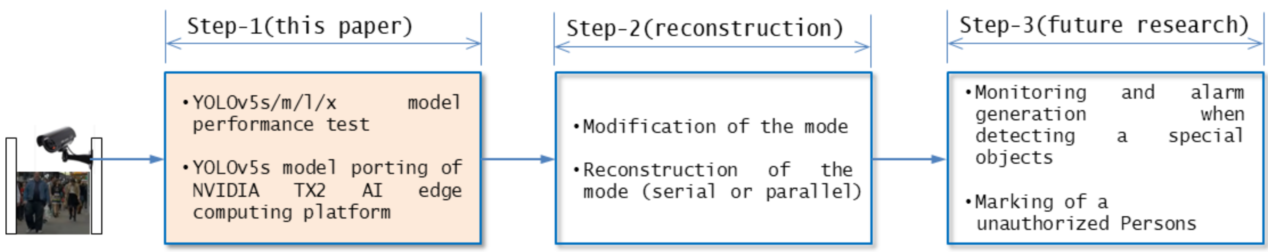


Figure 2. The overall structure of the real-time unmanned monitoring system in this research using NVIDIA TX2^{TM} AI edge computing platform

4. Performance Evaluation and Results

4.1 Performance Evaluation Environment of Object Detection [18],[19],[20]

In machine learning (ML) application, speed and accuracy of an object detection is very important. For example, the detection speed and accuracy of moving object detection are very important in self-driving system application. The speed of an object detection or recognition should satisfy the driving speed of the vehicle to meet the real-time respond of the vehicle to emergencies on the road. Fig. 3 is confusion matrix used to drive Eq. 1, Eq. 2, and Eq. 3.

		Predicted Condition		
		True	False	
Actual Condition	True	True Positive (TP)	False Positive (FP)	**precision**
	False	False Negative (FN)	True Negative (TN)	
		Recall		

Figure 3. Confusion matrix used to drive Equation (1), Equation (2), and Equation (3)

The accuracy includes the accuracy of class and location. Fig. 3 is confusion matrix used to drive Eq. 1, Eq. 2, and Eq. 3. Also, network performance is analyzed using the most common performance measures such as precision and recall, and $F_1 score$. They are respectively defined as Eq. 1, Eq. 2, and Eq. 3. Where, TP (True Positive) is the total amount of true-positive pixels, FP (False Positive) is the total amount of false-positive pixels, and FN (False Negative) is the total amount of false-negative pixels.

$$Precision = \frac{TP}{TP + FP} \qquad \text{Eq. 1}$$

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

Eq. 2

$$F_1 score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$$

Eq. 3

In this study, performance evaluation factors for object detection were AP, mAP, and Inference Time(ms). AP is Eq. 4, mAP is Eq. 5, and FPS is Eq. 6. Eq. 4 can be obtained by using the F1 Score using the harmonic average of Precision and Recall. The $F_1 score$ as a value between 0.0 and 1.0, and the higher the better. So, in this paper, we used the mean average precision (mAP) and the frames per second (FPS) as the performance evaluation indexes of the model.

$$AP = \int_0^1 P(R)dR$$

Eq. 4

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i$$ Eq. 5

$$FPS = \frac{N_{image}}{T_{total}}$$ Eq. 6

4.2 Object Detection Performance Evaluation

The performance evaluation of object detection is shown in Table 1. The models of YOLOv5 were used as the performance evaluation factors for object detection in this study. The images used in the experiment, the public dataset provided at https://public.roboflow.com/ was used [21]. The source of the YOLOv5 model used in this study was partially modified from the open source [22], and the four models used in the evaluation are s(small), m(medium), l(large), and x(xlarge) as in Table 1. Fig. 4 is a graph of learning results including box loss, object loss, class loss, precision, and recall according to the change of epochs. Also, in all graphs of Fig. 4, the x-axis is epochs.

Table 1. Object detection test results for 4 models of YOLOv5 for Pistol images

Model (YOLOv5)	mAP(0.5:0.95)	mAP(0.5)	Inference Time(ms)	FPS
s(small)	0.68	0.92	10.90	91.74
m(medium)	0.69	0.94	14.41	69.44
l(large)	0.71	0.95	16.45	60.97
x(xlarge)	0.72	0.96	35.29	28.40

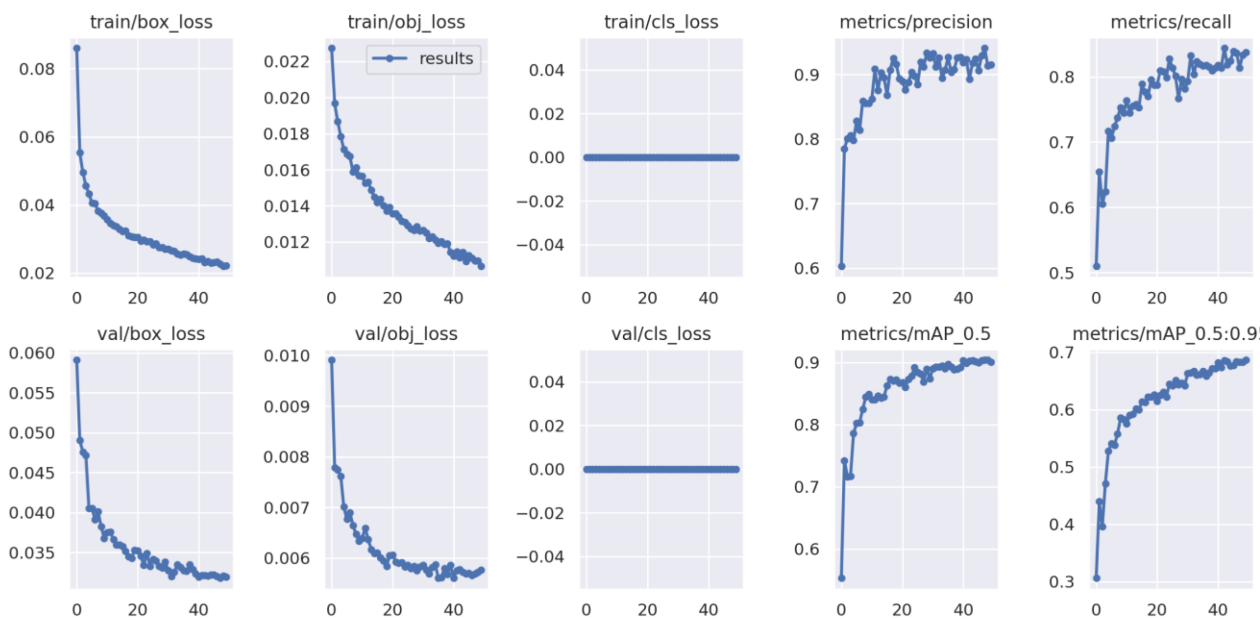


Figure 4. Learning results according to the change of epochs

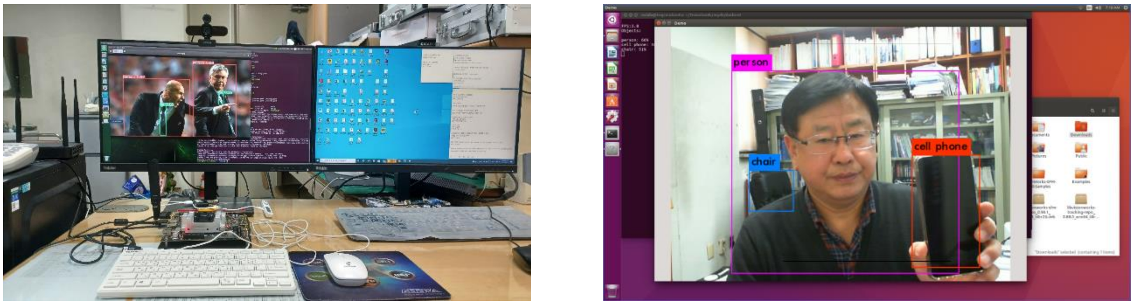
4.3 Designing a Real-Time Unmanned Monitoring System on NVIDIA TX2™ AI computing platform

In this paper, we designed a real-time unmanned monitoring system by applying YOLOv5s on NVIDIA Jetson TX2™ AI edge computing platform. Table 2 shows the specifications of the HW development environment used in the experiment. The network model used in the experiment was YOLOv5s in the DarkNet development environment, and a web camera mounted on USB in TX2 was used as the camera, and the COCO data set was used. The hardware specifications of the designed real-time unmanned monitoring system are shown in Table 2.

Table 2. Hardware specification of the real-time unmanned monitoring system for this paper

Item	Technical Specification
AI Performance	1.33 TFLOPS
GPU	Dual-core NVIDIA Denver 2 64-bit CPU and quad-core Arm Cortex-A57 MPCore processor complex
CPU	Dual-core NVIDIA Denver 2 64-bit CPU and quad-core Arm Cortex-A57 MPCore processor complex
OS	Ubuntu Linux
HW Platform	NVIDIA TX2™

Fig. 5(a) is a real-time unmanned monitoring system in the designed NVIDIA TX2™ AI Edge computing platform, and Fig. 5(b) shows the object detection results through an external camera in the system.



(a) (b)

Figure 5. Hardware development environment for our real-time unmanned monitoring system NVIDIA TX2^{TM} AI edge computing platform ((a)TX2 hardware development environment, (b)object detect result through camera)

4.4 Performance Evaluation and Review of Results

As a result of applying the YOLOv5s model to the hardware system used in this study, it was confirmed that there was no problem in object detection. However, the results showed some differences depending on the images used in the designed test. It is judged that this is the result of processing performance according to the number of objects and the movement of objects in the image. In addition, when the motion in the image is fast, the speed and result of detecting an object accurately are limited and depend on the performance of the system. In future research, it is judged that the method should improve the performance by using a high-performance GPU processor according to the application environment and the reconstruction of the model for performance improvement based on the YOLOv5s model.

5. Results

In this paper, the fundamental functions of a real-time unmanned monitoring system were designed by applying the YOLOv5s model on the NVIDIA TX2^{TM} AI edge computing platform. The designed system with YOLOv5s and NVIDIA TX2^{TM} AI edge computing platform was applied as the object detection model of the real-time unmanned monitoring system in specific applications. In addition, performance evaluation was performed to determine the object detection model, and YOLOv5s in this paper was determined based on these experiment results. We also evaluated the performance of four models of YOLOv5 and ported them to the NVIDIA TX2^{TM} AI edge computing platform.

Our research results can be applied to systems that detect, recognize, and track objects in various application fields (autonomous vehicles, intelligent robots, home appliances, weapon systems, and medical devices, etc.).

Future research is to improve the performance of the YOLOv5s model. For this, model reconstruction (multi-stage serial connection or parallel connection) should be performed in parallel with the code modification of the YOLOv5s model.

Acknowledgement

This study was conducted by Dong Seoul University's intramural research grant in 2010. We would like to thank Dong Seoul University for its research support.

References

[1] https://nvidia.com
[2] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, You Only Look Once: Unified, Real-Time Object Detection, arXiv:1506.02640v5 [cs.CV] 9 May 2016.
[3] Qihai Cai, Shangping Zhong, Kaizhi Chen, Target Detection in Rural River Image Based on Yolo V5 Model with the Cross-Switch Unit, 2021 6th International Conference on Image, Vision and Computing (ICIVC) | 978-1-6654-4368-5/21/©2021 IEEE | DOI: 10.1109/ICIVC52351.2021.9527005
[4] Xingkui Zhu1, Shuchang Lyu1, Xu Wang 1 Qi Zhao1, TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios, ICCV 2021 open access.
[5] Ross. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation,” IEEE Conf. Comput. Vision Pattern Recogn., Columbus, OH, USA, June 23-28, 2014, pp. 580-587.
[6] R. Girshick, “Fast R-CNN,” IEEE Int. Conf. Comput. Vision, Santiago, Chile, Dec. 7-13, 2015, pp. 1440-1448.
[7] S. Ren et al., “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, 2017, pp. 1137-1149.
[8] J. Dai et al., “R-FCN: Object Detection via Region-based Fully Convolutional Networks,” Conf. Neural Inform. Process. Syst., Barcelona, Spain, Dec. 4-6, 2016, pp. 379-387.
[9] W. Liu et al., “SSD: Single Shot MultiBox Detector,” Eur. Conf. Comp. Vision, Amsterdam, Netherlands, Oct. 8-16, 2016, pp. 21-37.
[10] Tsung Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar, “Focal Loss for Dense Object Detection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 42, Issue 2, pp. 318-327, 2018.
[11] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian, “CenterNet: Keypoint Triplets for Object Detection”, arXiv:1904.08189v3 [cs.CV] 19 Apr 2019.
[12] Liu Ting, Zhou Baijun, Zhao Yongsheng, and Yan Shun, Ship Detection Algorithm based on Improved YOLOv5, IEEE Xplore, 2022, Fab. 24, 2022.
[13] Wei Ding, Li Zhang, Building Detection in Remote Sensing Image Based on Improved YOLOv5, IEEE Xplore, 2022, 2.
[14] Guihui Shi, et. el, Combined Channel and Spatial Attention for YOLOv5 during Target Detection, IEEE Xplore, 2022, 2.
[15] Yaqi Guan, et. el, Design and Implementation of Safety Helmet Detection System Based on YOLOv5, 2021 2nd Asia Conference on Computers and Communications (ACCC)
[16] Guanhao Yang, et. el, Garbage Classification System with YOLOv5 Based on Image Recognition, IEEE Xplore, 2022, 2.
[17] Pallavi Udatewar, et. el, Personal Protective Equipment Kit Detection using YOLOv5 and TensorFlow, IEEE Xplore, 2021, 2.
[18] Tian-Hao Wu1, et. el, Real-Time Vehicle and Distance Detection Based on Improved YOLOv5 Network, IEEE Xplore, 2022, 2.
[19] Zhiyue Zhang, et. el, Recognition of Casting Embossed Convex and Concave Characters Based on YOLOv5 for Different Distribution Conditions, IEEE, 2021, 2.
[20] WenZe Fan, et. el, Research on Abnormal Target Detection Method in Chest Radiograph Based on YOLOv5 Algorithm, IEEE, 2021, 2.
[21] https://public.roboflow.com/
[22] https://github.com/ultralytics/YOLOv5.git