International Journal of Advanced Culture Technology Vol.9 No.3 298-304 (2021)
DOI https://doi.org/10.17703/IJACT.2021.9.3.298

IJACT 21-9-39

A study on object distance measurement using OpenCV-based YOLOv5

[1]Hyun-Tae Kim, [2]Sang-Hyun Lee

[1]student., Computer Engineering, Honam University, Korea
[2]Associate Professor., Department of Computer Engineering, Honam University, Korea
gusxo0560@naver.com, leesang64@honam.ac.kr

Abstract

Currently, to prevent the spread of COVID-19 virus infection, gathering of more than 5 people in the same space is prohibited. The purpose of this paper is to measure the distance between objects using the Yolov5 model for processing real-time images with OpenCV in order to restrict the distance between several people in the same space. Also, Utilize Euclidean distance calculation method in DeepSORT and OpenCV to minimize occlusion. In this paper, to detect the distance between people, using the open-source COCO dataset is used for learning. The technique used here is using the YoloV5 model to measure the distance, utilizing DeepSORT and Euclidean techniques to minimize occlusion, and the method of expressing through visualization with OpenCV to measure the distance between objects is used. Because of this paper, the proposed distance measurement method showed good results for an image with perspective taken from a higher position than the object in order to calculate the distance between objects by calculating the y-axis of the image.

Keywords: *COCO dataset, DeepSORT, OpenCV, Yolov5 model, COVID-19 virus, Euclidean Distance*

1. INTRODUCTION

From the outbreak of the COVID-19 virus until June 2, 2021, it has become a global problem. COVID-19 is a type of respiratory syndrome caused by a viral infection, the total number of confirmed cases as of June 2, 2021 is 171 million and the total number of deaths is 3.57 million, with a fatality rate of 2.08% [1]. It shows information and symptoms of the COVID-19 virus, and COVID-19 has a lower fatality rate than SARS (Severe Acute Respiratory Syndrome) and MERS (Middle East Respiratory Syndrome), which are the same virus family, but it is characterized by strong transmission power as it allows asymptomatic infection [2][3].

In South Korea, a patient with the COVID-19 virus was first reported on January 20, 2020. It seemed to spread slowly after the first case was reported, but as the number of confirmed cases surged after a number of cluster outbreaks occurred from mid-February, around February and March 2020, the number of confirmed cases appeared to be the second highest in the world after China, the source of the infectious disease [4]. To prevent the spread of the COVID-19 virus, the South Korean government has begun to impose a large-scale diagnostic test at a rapid pace and isolate confirmed or suspected cases. In order to contain the further spread, all citizens are urged to participate in high-intensity social distancing, and as of May 2021, the Republic of Korea is implementing systematic standards for social distancing and quarantine measures [5].

In addition to social distancing, the ban on gatherings of more than five people is being implemented as a countermeasure [6]. Here, rather than "social distancing, the effect on the ban on gathering is rather large [7]. However, even if the ban on gatherings of more than five people is enforced, there are cases of gathering of more than 5 people in the same space.

Manuscript received: July 31, 2021 / revised: August 25, 2021 / accepted: August 30, 2021
Corresponding Author: leesang64@honam.ac.kr
Tel: +82-62-940-5285, Fax: +82-62-940-5285
Associate Professor, Dept. of Computer Engineering, Honam University, Korea

Therefore, in this paper, the Yolov5 model was used to measure the distance between objects and process real-time images to solve the problem of a group of five or more people in the same space. Also, utilize Euclidean distance calculation method in DeepSORT and OpenCV to minimize occlusion.

2. RELATED RESEARCH

2.1 Yolo5 Model

Yolov5 has three layers, Backbone, Neck, and Head. Backbone extracts feature maps from images and uses CSP-Darknet. In addition, Yolov5's backbone is divided into four types: Yolov5s, Yolov5m, Yolov5l, and Yolov5x. The criterion for dividing this is the difference between depth multiple (model depth multiple) and width multiple (layer width multiple). Yolo5Vs is the fastest, but with less accuracy, and Yolov5x is the slowest, but with improved accuracy.

Neck performs refinement and reconfiguration of the feature map with the PANet layer. Therefore, it plays a role of connecting the backbone and the head.

Head is the part that finds the position of an object based on the extracted feature map. Anchor Box (Default Box) is initially set, and the final Bounding Box is created using it. Create a Bounding Box of three scales in the same way as in Yolov3. In addition, since three anchor boxes are used in each scale, there are nine anchor boxes.

2.2 DeepSORT

Simple Online and Real-time Tracking (SORT) uses a Kalman Filter to show the correlation of frame-by-frame data, and for correlation measurement, it is possible to use a Hungarian algorithm effective for tracking multiple objects [8]. Track estimator, a component of SORT algorithm, uses Kalman Filter to predict the position of the bounding box of an object. Here, the Kalman Filter operates recursively to predict a new value between the old value and the measured value [9].

To make predictions using the Kalman Filter, calculate based on the previous object velocity. It is also a location matrix that uses the difference of intersection over union (IoU) between the detected and predicted bounding boxes [10]. The steps of the Kalman Filter processing process consist of three steps: prediction, actual measurement, and correction [11]. The first step is a prediction step, which is generated by including some noise as an estimate of the current state variable. The second step is actually measuring the variable for the current state. Finally, the third is the calibration step, which reflects the difference between the predicted state and the actual measured value based on the estimated state and updates the current state. Using a weighted average, more weighted.

DeepSORT can be a more reliable matrix by using matching cascade instead of the conventional sort method [12]. DeepSORT connects new detection and new prediction when given a new bounding box tracked through Kalman Filter using the assignment problem. In this case, after quantifying the association with the Mahalanobis distance [13], use the quantified association using the Hungarian Algorithm to connect the independently performed results [14]. Here, Hungarian algorithm is performed by reflecting additional value of Kalman Filter and deep leaning feature. H. W. Kuhn published the Hungarian algorithm in 1955. It was created by the achievements of Hungarian mathematicians D. Kőnig and J. Egerváry, and is an algorithm to find the optimal solution [15].

Cosine Distance means the degree of similarity between vectors measured using the cosine value of the angle between two vectors in the inner product space. When an angle is 0°, the cosine is one, and the cosine of all other angles is less than one. Therefore, it is used for determining the similarity of the direction rather than the magnitude of the vector, and the cosine distance can be expressed as mxn (m=n) as the Euclidean distance.

In this study, the two-dimensional Euclidean formula is used to represent real-time processing of two-dimensional images.

3. DESIGN OF SYSTEM

3.1 COCO dataset

In COCO 2017 (train image, value image) two datasets, the labeling images and labeling excluding persons are arranged, the train image used for train is 64115, and the value image used for test has 2693 datasets. Train images consist of a minimum of 15 * 30 and a maximum of 640 * 640. The reason for the training by dividing it into train and value is used to prevent overfitting.

3.2 YoloV5 Model

The structure of Yolov5 is shown in Figure 3, and the explanation of the layers used for each learning in the order of Focus, Conv (Convolution), C3, SPP, Upsamepling, Concat, and Detect is shown in Figure 1 below.

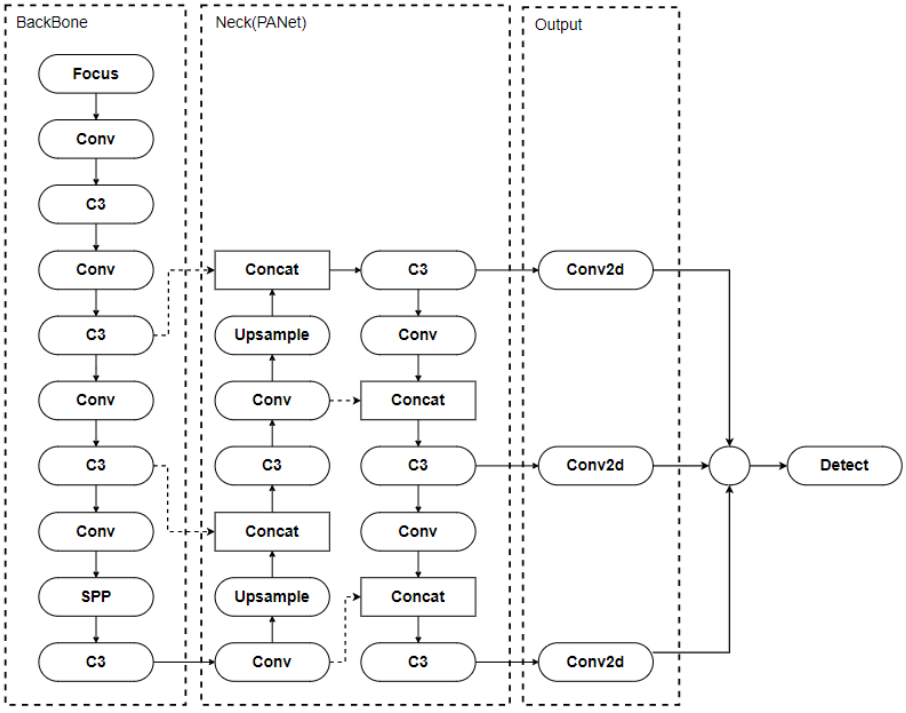


Figure 1. Layer structure in Yolov5

As for the explanation of Figure 1, first, focus is a layer that divides the image and divides the divided images into layers. Conv goes through Conv2d, calculates the value of batchNorm2d + LeakyRELU, and exports it. C3 converts the input data, calculates it at the bottleneck layer, adds the conv value from the initial input and the bottleneck layer calculation value in Concat, then converges and outputs it. The bottleneck proceeds with Conv (1, 1) on the input value and outputs the calculated value of Conv (3, 1). After the Conv operation, SPP exports three MaxPooling values 5*5, 9*9, 13*13, combines the three MaxPooling values in Concat with the Conv value from the existing input value, and sends them out after Conv.

Upsampling is a basic library function of pytorch, and the function doubles the number of each array of feature maps in structure values. Concat plays the role of merging input layers. Since detect currently has three Conv2d values as a result, combine the three and output them.

3.3 Distance measurement process between objects on an image

In order to detect the distance between people, real-time image or video has two-dimensional coordinates and perspective, so there is a difference in the size of an object box under the influence of the y-axis compares the calculated value with the value calculated from the Euclidean distance to calculate the degree of closeness between objects. The criterion for determining that the distance between objects is close indicates the distance between objects when it is greater than 0 and less than the value obtained by dividing the y-axis between objects by 2 and multiplying by 0.25.

Figure 2 shows the flow chart for outputting the distance between objects in this paper, and it explains in order from video or real-time data transmission to the process of drawing a box.

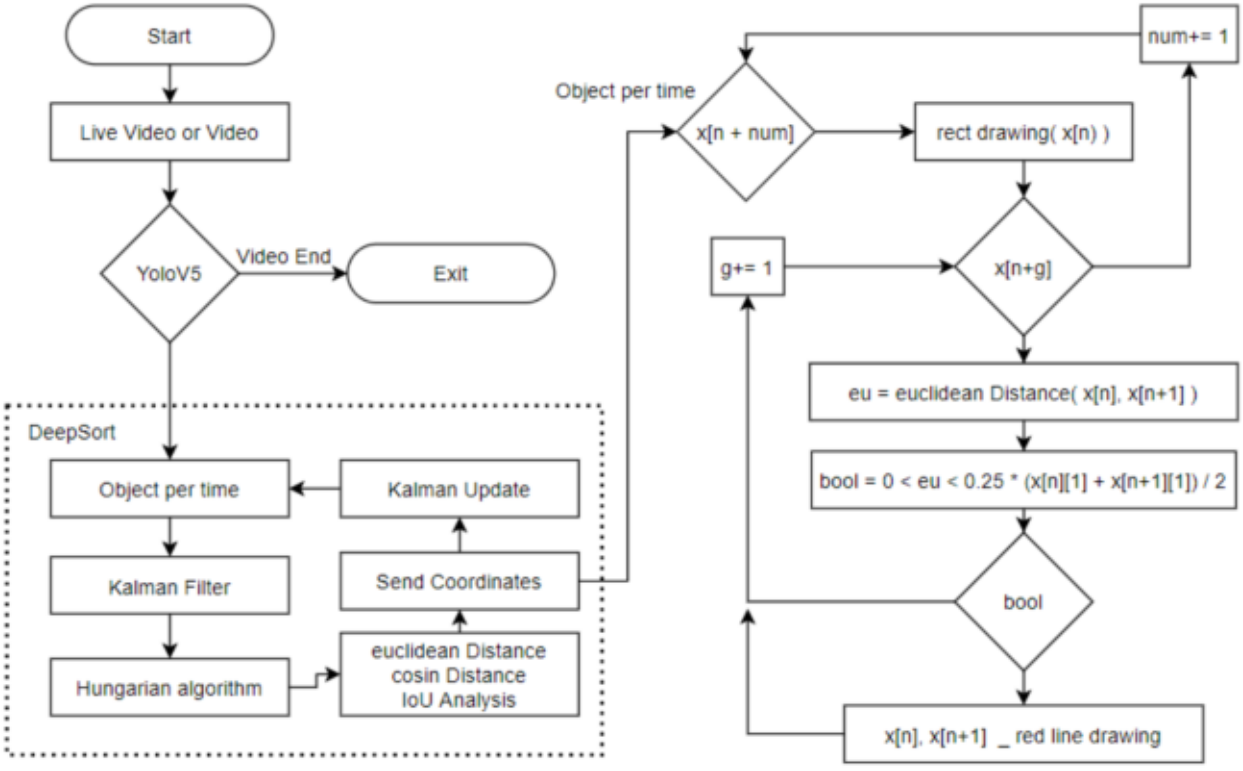


Figure 2. Distance measurement process between objects on an image

The progress of Figure 2 is as follows. First, it receives video or real-time image data. In YoloV5, the video is processed until the end of the program and the end of the program. DeepSORT starts by receiving the time object from YoloV5 as a Kalman Filter, and matches through the Hungarian algorithm. When comparing with predicted points through IoU analysis, vector angle and distance are calculated using cosine distance and Euclidean distance. After updating in Kalman Filter, it proceeds as object per time. DeepSORT operates recursively, transmits the coordinates and inserts the bounding box into the OpenCV image using the transmitted coordinates. To check the distance between objects during the insertion process, check the center point of the bounding box and calculate the Euclidean distance. The distance is judged from the bool value. If the distance is judged close as a compared value between objects, it is indicated by a line and repeated until the video of YoloV5 ends again.

4. IMPLEMENT AND RESULT

The learning environment of the Yolo model in this paper is a Windows environment, CPU: i9-9900K, GPU: RTX Quadro 6000, RAM: 128GB. The coding and test environment is CPU: i5-6600, GPU: Geforce. GTX 1660 super, RAM: 12GB, Torch 1.6.0, torchvision: 0.7.0, CUDA: 11.3 were implemented and tested. In the experimental results, the test was carried out with the model of Yolov5 used in the implementation of this

paper.

In this paper, the model for detecting people uses Yolov5s, and to detect people, only the person is left in the COCO dataset, and the remaining images are removed and labeled before training.

Table 1. Train / val_Loss

Step	Train/box_loss	Train/obj_loss	val/box_loss	val/obj_loss
1	0.07073	0.05230	0.06619	0.03773
50	0.05874	0.04842	0.05358	0.03619
100	0.05783	0.04795	0.05186	0.03336
250	0.05331	0.04560	0.04864	0.03136
300	0.05258	0.04536	0.04806	0.03093

For the loss rate of the progressed training, Table 1 describes the train loss. The box_loss of Step 1 at the beginning of learning is 0.07073, obj_loss is 0.0523, in step 50, box_loss is 0.05874, obj_loss is 0.04842, and in Step 100, box_loss is 0.05783 and obj_loss is 0.04795. The number of training times of Yolov5's model is 300, and the change value of the loss of box_loss and obj_loss in steps 250 and 300 is insignificant, so the training is terminated.

4.1 Testing of Yolov5 Model

The model test of Yolov5 uses the 2017 Test image among the datasets provided by COCO to check the results. The result value was output in the form of Figure 3 model test, and the performance of Yolov5 is described in table 2.

Figure 3. Video Testing of Yolov5 Model

The test results of Yolov5 are shown in Table 2, the precision results are 61%, the recall value is 89.3%, the mAP is 61.2%, and the F1 score is 61%.

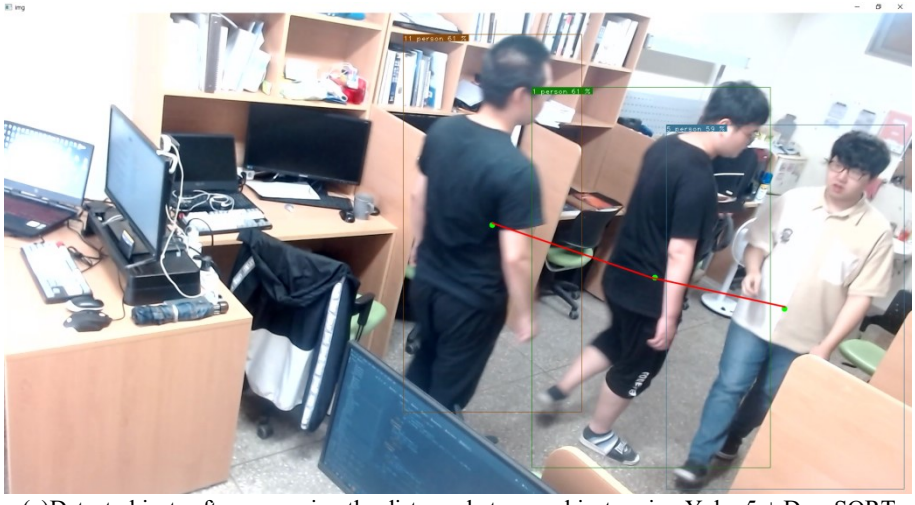
Table 2. Results of Yolov5 Model Test

Division	Precision/%	Recall/%	mAP/% @0.5	F1 score/%
Model	61.0%	89.3%	61.2%	61.0%

4.2 Result

The video used for the test in this paper is tested with an image of 1920 * 1080 resolution of 28.12 FPS. In addition, when using the Euclidean technique, perspective is applied on the image, and the effect of the Y-axis is large in a situation where a person is touching the floor, making it easier to track the distance between objects. The video tracking since then uses the video looking down from the height of the CCTV installation. The three test results in Figure 4 are explained, and in each image of (a), (b), and (c), people are referred to as A, B, and C from the left.

(a) Detect objects using Yolov5 (b) Detect objects using Yolov5+DeepSORT



(c)Detect objects after measuring the distance between objects using Yolov5 + DeepSORT

Figure 4. Real-time Testing of Yolov5 Model

In Figure 4 (a), when the detection result of the object using the Yolov5 model is confirmed, the phenomenon that the person C in the image blocks the front of the person B cannot detect the person. In the Yolov5 + DeepSORT method used in Figure 4(b), it can be seen that the object is tracked even in the presence of occlusion. In the image of Figure 4(c), it can be seen that the closer the distance between objects using OpenCV is, the closer they are connected with a line.

5. CONCLUSION

Currently, the world is working together to prevent the spread of the COVID-19 virus. Because of this series of processes, "If more than 5 people gather in the same space, they are easily infected with the Corona 19 virus", so gatherings are prohibited. The purpose of this paper is to "measure the distance between objects with real-time image" by "utilizing OpenCV and Yolov5 model" to "limit the distance between multiple people" in the same space to "concentrate people in a narrow space". to prevent.

In this paper, YoloV5 was used to measure the distance between objects, and DeepSORT and Euclidean techniques were applied to minimize occlusion. In addition, a method was used to visualize and express the

distance measurement between objects with OpenCV.

The distance measurement method proposed in this paper represents the distance between objects by calculating the y-axis appearing in perspective. As the object gets further away, it uses the smaller size of the object box, calculates the distance between objects, and calculates the distance between the midpoints between object boxes using the Euclidean calculation method, and visualizes it according to the specified numerical value. Through this, the method of calculating the distance between objects shows good results in images such as CCTV where the shooting height and perspective are applied.

Through the method used in this study, it was confirmed that the object was tracked and visualized according to the distance between the objects. Based on these results, it can be seen that Kalman Filter and DeepSORT show good results in images to which perspective is applied, so they can be applied to various fields. Because distance problems in a wide space can be visualized and expressed more easily, it can be used for drones and robots, and good performance can be expected.

REFERENCES

[1] “CoronaBoard”, CoronaBoard, last modified, accessed June 02, 2021, https://coronaboard.com/.
[2] “Coronavirus Disease-19, Republic of Korea”, Ministry of Health and Welfare, last modified June 02, 2021, accessed Sep 06, 2021, http://ncov.mohw.go.kr/baroView.do
[3] “World Health Organization”, WHO, last modified Sep 06, 2021 accessed Sep 06, 2021, https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public
[4] S.C. Yoon, “Korea's response to COVID-19 using ICT and its achievements”, Korea International Cooperation Agency, pp.1- 64, 2020
[5] “Coronavirus Disease-19, Republic of Korea”, Ministry of Health and Welfare, last modified June 02, 2021, accessed Sep 06, 2021, http://ncov.mohw.go.kr/socdisBoardView.do?brdId=6&brdGubun=1
[6] D.T. Kim, “Characteristics of Korea's COVID-19 response system from the perspective of democratic governance”, Korea International Cooperation Agency, pp. 1-168, 2020.
[7] “Korea Institute of Science and Technology.”, KIST, last modified Sep 06, 2021 accessed Sep 06, 2021, https://www.kist.re.kr/kist_web/?sub_num=2935&state=view&idx=3229&sty=T&ste=%EC%BD%94%EB%A1%9C%EB%82%98&ord=10
[8] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, Ben Upcroft, “Simple Online and Realtime Tracking” IEEE International Conference on Image Processing (ICIP), pp.3464-3468, 2016. DOI: 10.1109/ICIP.2016.7533003
[9] “CODE PROJECT For those who code”, CODE PROJECT, last modified June 02, 2021, accessed June 02, 2021, https://www.codeproject.com/articles/865935/object-tracking-kalman-filter-with-ease
[10] Tuan Linh Dang, Gia Tuyen Nguyen, Thang Cao, “Object Tracking Using Improved Deep_Sort_YOLOv3 Architecture” ICIC Express Letters, Vol.14, No.10, pp.961-969, 2020. DOI:10.24507/icicel.14.10.961
[11] “KM-Hana Tistory”, KM-Hana, last modified, accessed June 02, 2021, https://kmhana.tistory.com/22
[12] N. Wojke, A. Bewley and D. Paulus, "Simple online and realtime tracking with a deep association metric," IEEE International Conference on Image Processing (ICIP), pp.3645-3649, 2017. DOI: 10.1109/ICIP.2017.8296962
[13] R. De Maesschalck, D. Jouan-Rimbaud, D.L. Massart, “The Mahalanobis distance” Chemometrics and Intelligent Laboratory Systems, Vol.50, No.1, pp.1-18, 2000, DOI: https://doi.org/10.1016/S0169-7439(99)00047-7
[14] “KM-Hana Tistory”, KM-Hana, last modified, accessed June 02, 2021, https://kmhana.tistory.com/23
[15] “Gazelle and Computer Science Tistory”, Gazelle and Computer Science, last modified, accessed June 02, 2021, https://gazelle-and-cs.tistory.com/29