

A Study on Algorithm Selection and Comparison for Improving the Performance of an Artificial Intelligence Product Recognition Automatic Payment System

¹Heeyoung Kim, ²Dongmin Kim, ³Gihwan Ryu and ⁴Hotak Hong

¹*Doctoral Course, Department of Immersive Content Convergence,
General graduate school, Kwangwoon University, Korea*

²*CEO, JLK Inc., Korea*

³*Professor, Department of Tourism Industry, Graduate school of smart convergence,
Kwangwoon University, Korea*

⁴*Researcher, AI R&D Center, JLK Inc., Korea*

k3h3y3@kw.ac.kr, dmkim@jlkgroup.com, allryu@kw.ac.kr, hthong@jlkgroup.com

Abstract

This study is to select an optimal object detection algorithm for designing a self-checkout counter to improve the inconvenience of payment systems for products without existing barcodes. To this end, a performance comparison analysis of YOLO v2, Tiny YOLO v2, and the latest YOLO v5 among deep learning-based object detection algorithms was performed to derive results. In this paper, performance comparison was conducted by forming learning data as an example of 'donut' in a bakery store, and the performance result of YOLO v5 was the highest at 96.9% of mAP. Therefore, YOLO v5 was selected as the artificial intelligence object detection algorithm to be applied in this paper. As a result of performance analysis, when the optimal threshold was set for each donut, the precision and reproduction rate of all donuts exceeded 0.85, and the majority of donuts showed excellent recognition performance of 0.90 or more. We expect that the results of this paper will be helpful as the fundamental data for the development of an automatic payment system using AI self-service technology that is highly usable in the non-face-to-face era.

Keywords: *Self-Service Technology, Automatic payment system, Artificial Intelligence, Object detection, YOLO*

1. INTRODUCTION

With the development of information and communication technology, the food service industry and the distribution industry are changing rapidly, and as the consumption culture changes due to the new environment, self-checkout counters, a type of self-service technology, are also emerging with new concepts. In a situation where self-service technology is expanding due to the preference for contactless payments due to COVID-19, it is necessary to develop a payment system using AI (Artificial Intelligence) self-service technology that can improve manpower efficiency and reduce labor costs for small and medium-sized companies. In this study, in order to find an AI model suitable for an artificial intelligence-based product recognition automatic payment system, the algorithm is compared by selecting a YOLO [1] series specialized for real-time object detection.

Among the YOLO series, YOLO v2 [2], Tiny YOLO v2 [2], and YOLO v5 [3] are selected and compared.

Manuscript received: January 20, 2022 / revised: March 1, 2022 / accepted: March 8, 2022

Corresponding Author: hthong@jlkgroup.com
Tel: +82-70-4651-4051, Fax: +82-505-300-4051
Researcher, AI R&D Center, JLK Inc., Korea

Copyright©2022 by The International Promotion Agency of Culture Technology. This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/4.0>)

2. RELATED RESEARCH

2.1 Object detection

Object detection is a field of research that analyzes visual information within a given image with computer vision technology related to identifying objects in digital photos to locate objects in the image.

The object detection algorithm consists of two parts: backbone and head. Backbone is the part that transforms the input image into a feature map. Typical backbones include VGG16 [4], ResNet-50 [5], ResNeXt-101 [6], and Darknet53 [7] pre-trained with ImageNet dataset. Head is the part that performs the location of the feature map extracted from the backbone. Predict classes classifying what objects are for multiple objects in head and bounding boxes representing location information through boxes where the objects are located are performed. Head is largely divided into dense prediction and sparse prediction, which is directly related to whether it is a one-stage detector or a two-stage detector, a type of object detection. Two-stage detectors using head as sparse prediction include Faster R-CNN [8] and R-FCN [9]. It is characterized by the separation of the predict classes and the bounding box regression. One-stage detector, which uses head as dense prediction, typically includes RPN [8], YOLO, SSD [10], and RetinaNet [11]. Unlike the two-stage detector, the one-stage detector features a combination of predict classes and bounding box regression.

The object detection problem is a mixture issue of the localization of finding the location of an object and the classification of identifying the object [12]. One-stage detector is a way to do these two problems simultaneously, and two-stage detector is a way to do these two problems sequentially. One-stage detector is faster than two-stage detector because it simultaneously derives the results of localization and classification.

Therefore, this study compares algorithms by selecting one-stage detectors suitable for the condition that product recognition should be performed in real time and selecting YOLO series specialized in real-time object detection to find an artificial intelligence model suitable for deep learning-based product recognition automatic payment systems. In this paper, YOLO v2, Tiny YOLO v2, and YOLO v5 are selected and compared among the YOLO series.

2.2 YOLO v2

YOLO v2 improved accuracy and speed compared to YOLO v1, and the performance was improved by applying batch normalization to all convolution layers. In backbone, YOLO v1 also used backbone VGG16 for feature extraction purposes, but YOLO v2 improved its performance using its own Darknet-19 [13]. Darknet-19 starts max pooling very quickly and decreases the number of channels in the feature map by half through 1x1 convolution in the middle of convolution layer, reducing the computational volume compared to VGG16. In terms of processing speed, YOLO v2 drastically lowered the number of weight parameters by removing FC (Fully Connected) layers, enabling detection at a faster processing speed.

2.3 Tiny YOLO v2

Tiny YOLO v2 is a model designed for devices with low computing resources and low computing power, and has reduced the depth of the network to a 17-layer structure by removing certain layers from the 30-layer YOLO v2 model. To solve the gradient vanishing problem in YOLO v2, the output value of the 16th layer from the 26th layer was brought back to reduce the 23 conv layers to 9 conv layers by removing convolution. However, although the processing speed is improved by reducing the conv layer, there is a disadvantage in that the accuracy decelerates significantly.









2.4 YOLO v5

The YOLO v5 has a fast processing speed while maintaining performance similar to that of the YOLO v4. CSPNet-based backbone such as YOLO v4 was designed and bottleneck was applied to obtain faster processing speed. In addition, the performance was maintained while increasing the processing speed using various data augmentation techniques. There are four models of YOLO v5: YOLO v5s, YOLO v5m, YOLO v5l, and YOLO v5x. The model loading capacity is large and the processing speed is slow in the order of YOLO v5s, YOLO v5m, YOLO v5l, and YOLO v5x. This paper compared this paper with other algorithms by selecting a YOLO v5l model whose model loading capacity, performance, and processing speed are intermediate among the four models in light of limited resources due to the nature of the automatic payment system.

3. ALGORITHM EVALUATION RESULTS

In order to compare the three algorithms, a total of 100,000 test data were constructed and model evaluation was conducted. In order to build test data, the Logitech Brio 4k Pro Webcam camera was photographed with a donut in a tray at a height of 50 cm. Artificial intelligence learning was evaluated with eight classes targeting nine kinds of donuts. Table 1 shows the donuts and price information selected for target.

Table 1. Target donut image and price information

No	Class name	Image	Price
0	Cacao Frosted(CF)		1300 won
1	Boston Kreme(BK)		1300 won
2	Heart Donut(HD)		1900 won
3	Chewisty(CW)		1500 won
4	Old-Fashioned(OF)		1500 won
5	Café Mocha(CM)		1500 won
6	Bavarian Filled(BF)		1300 won
7	Strawberry Filled(SF)		1300 won

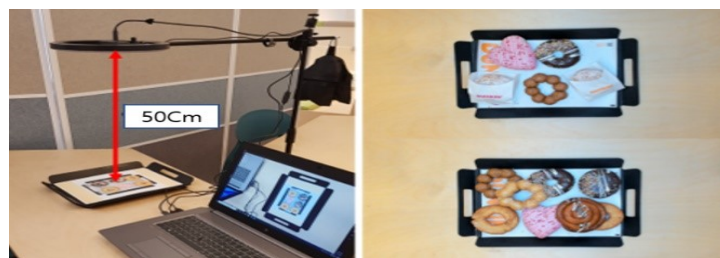


Figure 1. Test data shooting environment and sample of test data image

The left picture of Figure 1 is the test data photographing environment, and the right picture is the image photographed with test data. The test data was organized by difficulty level by adjusting the degree of donut overlap. Model evaluation was performed using the AP (Average Precision) used as an evaluation indicator in the field of object detection. The performance evaluation results of YOLO v2, Tiny YOLO v2, and YOLO v5l are as follows.

Table 2. Average Precision by algorithm

Model	mAP	CF	BK	HD	CW	OF	CM	BF	SF
YOLO v2	78.09	84.64	85.00	98.12	65.47	73.12	96.63	48.74	72.97
Tiny YOLO v2	53.57	50.84	40.20	43.48	76.25	56.45	75.51	23.55	62.28
YOLO v5l	96.90	95.20	89.80	100	96.88	96.99	97.63	99.08	99.58

As shown in Table 2, YOLO v5l had the highest performance in all donuts, and mAP (mean Average Precision) showed the highest performance among the three algorithms with 96.9%.

4. COMPARISON OF ALGORITHM RESULTS

To compare algorithmic results, the evaluation was conducted using the same GPU as the same training data, and the results were compared and analyzed. The results of comparing the performance of each object are as follows.

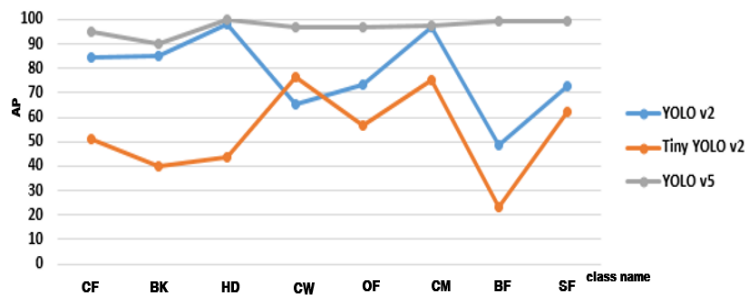


Figure 2. Comparison of performance by algorithm

As shown in Figure 2, the performance of YOLO v5 was the highest in all objects, and in the case of YOLO v2 and Tiny YOLO v2, the Tiny YOLO v2 models are lightweight, showing poor performance. HD (Heart Donut) and CM (Café Mocha), which show distinctly different characteristics from other donuts, show high performance, and in the case of YOLO v2, they show performance equivalent to YOLO v5. In the case of CF (Cacao Frosted) and BK (Boston Kreme), there is only one hole in the middle, which shows lower performance compared to the average, and the P-R (Precision-Recall) Curve is shown in Figure 3.

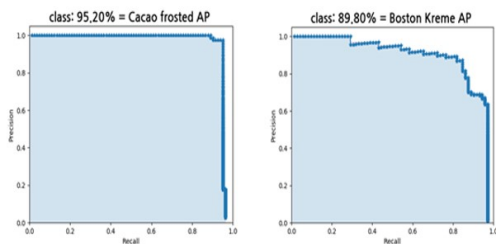


Figure 3. Cacao frosted and Boston Kreme P- R Curve



Figure 4. Example of detecting Cacao frosted in the lower left with Boston Kreme

When the model determines that it is Cacao Frosted, most of them are well matched, but the reason why recall fell a lot near 1 is that Cacao Frosted is detected as Boston Kreme, as shown in Figure 4. On the other hand, in the case of Boston Kreme, it can be seen that the ratio of objects (inferred boxes) that are not correct

answers increases near 1, resulting in a decrease in precision. In the case of Boston Kreme, even if the recall is slightly higher, it is easily detected as Cacao Frosted. Therefore, when applying an artificial intelligence-based product recognition automatic payment system, the threshold of the model can be adjusted and solved. The precision of donuts and the value of recall vary according to the threshold value, and over-detection and non-detection can be reduced by setting an appropriate threshold value.

Table 3 shows the optimal threshold value in consideration of precision and recall.

Table 3. Optimal threshold for each class

No	Class name	Threshold
0	Cacao Frosted(CF)	0.20
1	Boston Kreme(BK)	0.42
2	Heart Donut(HD)	0.42
3	Chewisty(CW)	0.72
4	Old-Fashioned(OF)	0.76
6	Café Mocha(CM)	0.60
7	Bavarian Filled(BF)	0.74
8	Strawberry Filled(SF)	0.76

The threshold value of each donut is set as shown in Table 3, and the results are confirmed as shown in Figure 5. Figure 5 is the result of detecting donuts before and after setting the optimal threshold.

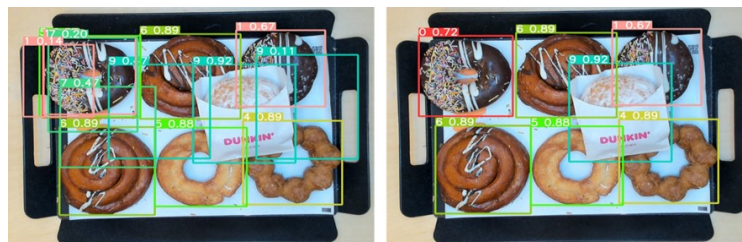


Figure 5. Result of detecting donuts before and after setting the optimal threshold

In Figure 5, the left figure is the result of detecting the donut before setting the threshold, and the right figure is the result of setting the optimal threshold for each donut and detecting the donut. From the right side figure, it can be noticed that the wrong detection has decreased a lot.

5. CONCLUSION

In this paper, performance comparison was conducted by forming learning data using 'donut' in the bakery store as an example, and the applied algorithm and performance analysis results are as follows. In order to find an artificial intelligence model suitable for the self-service technology-based product recognition automatic payment system, we compared three algorithms of YOLO v2, Tiny YOLO v2, and YOLO v5. YOLO v5 had the highest performance in all objects, and mAP showed the highest performance at 96.9%. However, YOLO v5 was also not perfect for all donuts, and there were cases where some incorrect inferences were made about classes that could be confusing visually. However, these problems are lower than the scores of well-recognized donuts, so when applying an artificial intelligence-based product recognition automatic payment system, it can be solved by adjusting the threshold of the artificial intelligence model. When the optimal threshold was set

for each donut, the precision and recall of all donuts exceeded 0.85, and the majority of donuts showed excellent recognition performance of 0.90 or more. In addition, as the amount of learning data is additionally accumulated, it is judged that the performance will become superior over time.

When the results of this paper are commercialized and released on the market, if applied to the calculation of products that are difficult to pay with barcodes, such as fresh food stores, bakery stores, and autonomous restaurants on highways, it will be helpful in reducing labor costs and increasing sales for small businesses. Therefore, it is necessary to integrate more optimization methods and state-of-the-art ideas in the field of computer vision to ensure that AI product recognition automatic payment systems perform best.

This paper is of incredible consequence in that it presented a competitive solution in the food service industry through comparison of the performance of artificial intelligence-based object detection algorithms.

REFERENCES

- [1] 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 (CVPR)*, pp. 779-788, 2016.
- [2] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger", In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7263-7271, December 2017.
- [3] D. Thuan, "Evolution of yolo algorithm and yolov5: the state-of- the-art object detection algorithm", Spring 2021.
- [4] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *arXiv preprint arXiv:1409.1556*, September 2014.
- [5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, 2016.
- [6] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks", In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1492-1500, 2017.
- [7] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement", arXiv preprint arXiv:1804.02767, April 2018.
- [8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks", *IEEE transactions on pattern analysis and machine intelligence*, Vol. 39, No. 6, pp. 1137-1149, June 2016. DOI: 10.1109/TPAMI.2016.2577031
- [9] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks", In *Advances in Neural Information Processing Systems (NIPS)*, pp. 379-387, 2016.
- [10] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. C. Berg, "SSD: Single shot multibox detector", In *European Conference on Computer Vision (ECCV)*, pp. 21-37, September 2016. DOI: 10.1007/978-3-319-46448-0_2
- [11] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection", In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2980-2988, October 2017.
- [12] Y. J. Kim, "Deep learning network architecture design for 3D sculpture identification", Ph.D. Thesis. University of Sangmyung, Seoul, 2020.
- [13] J. Redmon, "Darknet: Open Source Neural Networks in c", 2013. [Online]. Available: <https://pjreddie.com/darknet>