Accurate fashion and accessories detection for mobile application based on deep learning

Yamin Thwe, Nipat Jongsawat, Anucha Tungkasthan
Faculty of Science and Technology, Rajamangala University of Technology Thanyaburi, Pathum Thani, Thailand

Article Info

Article history:

Received Jun 24, 2022
Revised Oct 26, 2022
Accepted Nov 6, 2022

Keywords:

E-commerce
Machine learning
Mobile application
Product categorization
YOLOv4

ABSTRACT

Detection and classification have an essential role in the world of e-commerce applications. The recommendation method that is commonly used is based on information text attached to a product. This results in several recommendation errors caused by invalid text information. In this study, we propose the development of a fashion category (FC-YOLOv4) model in providing category recommendations to sellers based on fashion accessory images. The resulting model was then compared to YOLOv3 and YOLOv4 on mobile devices. The dataset we use is a collection of 13,689, which consists of five fashion categories and five accessories' categories. Accuracy and speed analysis were performed by looking at mean average precision (mAP) values, intersection over union (IoU), model size, loading time, average RAM usage, and maximum RAM usage. From the experimental results, an increase in mAP was obtained by 99.84% and an IoU of 88.49 when compared to YOLOv3 and YOLOv4. Based on these results, it can be seen that the models we propose can accurately identify fashion and accessories categories. The main advantage of this paper lies in i) providing a model with a high level of accuracy and ii) the experimental results presented on a smartphone.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Anucha Tungkasthan
Faculty of Science and Technology, Rajamangala University of Technology Thanyaburi
Nakhon Nayok Road, Khlong Hok, Khlong Luang District, Pathum Thani, Thailand 12110
Email: anucha_t@rmutt.ac.th

1. INTRODUCTION

Globally, the volume of online store sales is increasing significantly. Between 2018 and 2021, the transaction volume of online product sales will reach hundreds of billions of dollars [1], [2]. Fashion products rank number one among these online retail sales in all categories. Customer convenience is a major priority when it comes to online shopping; therefore, product recommendations are required so that consumers can simply filter information from a huge number of online stores and select the appropriate product. To get the right recommendation, the data entered by the seller must be accurate. Mistakes often occur when choosing a fashion product category that does not match the product image [3]. This will significantly impact customer satisfaction on several e-commerce platforms, such as Shopee. Therefore, it is necessary to increase the accuracy of category recommendations so that sellers can place their products in the correct category.

Researchers' interest in the problem of categorizing fashion and accessories is quite significant, especially in the field of computer vision. This can be seen from several previous studies that have been carried out. Many studies use clothing categories as their main features, such as fashion style recognition [4], clothes detection [5]–[8], job recognition [9], clothing recommendation [10], and clothing-based identification of people used in a video. In general, the clothing categorization method used in [11]–[13] is visual feature extraction, in addition to [14] using histograms of oriented gradients (HOG) and color histograms from an image. After feature extraction is complete, classification is carried out by recognizing clothing patterns based on previous features. However, some of these studies have limited performance due to hand-crafted features (traditional).

The deep convolutional network has made significant progress in recent years. Several applications of deep convolutional networks have achieved satisfactory performance in various fields, including object detection [15], [16] image classification [15], [17], [18], task recognition, and others [19]–[23]. Deep learning has the power to learn image representation automatically and effectively. Several deep learning methods utilize the deep convolutional neural network for the clothing category, which in parallel studies the feature representation and classification of clothing from an image.

A deep learning architecture for apparel classification is suggested [24] with fewer clean label-to-noisy label ratios than millions of noisy labels. In 2017, a deep learning system with the capacity to transfer learning models from apparel image datasets from several sources, including retail and street websites, resulting in multilabel attribute identification, was introduced [25]. The fashion sub-categories and attributes prediction (FSCAP) deep learning model [3], [26] is proposed by combining YOLO and DeepSORT architectures for the detection and tracking of people, then faster-region-based convolutional neural networks (RCNN) architecture for subcategory classification, and Custom-EfficientNet-B3 architecture designed for attribute prediction proposed a new approach by adapting EfficientDet. This study aims to detect multi-clothing and fashion landmark estimation with fast and accurate detection results. Lee and Lin [27] proposed a two-phase fashion clothing detection method named YOLOv4-TPD, with the detection target categories divided into jackets, tops, pants, skirts, and bags. Fashion category (FC-YOLOv4) was employed in our earlier study [28] to recognize multiclass fashion products in a semi-supervised labeled image. Based on a number of the conducted research, the use of fashion detection on smartphone devices has not been taken into consideration. The contributions of our study are that i) we added the number of datasets used in previous studies regarding FC-YOLOv4 [27], ii) we deploy FC-YOLOv4, YOLOv4, and YOLOv3 for mobile applications, iii) we use image datasets with various resolutions, sizes, and positions to train the FC-YOLOv4 model, and iv) we compared the deployment of FC-YOLOv4 with YOLOv4 and YOLOv3 in terms of detection speed and precision.

Dataset image additions were made in the initial five categories (fashion products: pants, dress, hoodie, jacket, and skirt), then added five additional categories (accessories products: bracelet, earring, neckless, ring, and belt). This dataset was taken from Shopee and Google with various details. The model deployment uses the TensorFlow lite library to create a pre-trained object identification model compatible with Android devices. The model deployment was evaluated by comparing FC-YOLOv4 with YOLOv4 and YOLOv3, and some experiments are presented and discussed.

2. METHOD

2.1. Images dataset collection and annotation

The collection of datasets is based on several characteristics of fashion imagery and accessories found in e-commerce. A fashion image and accessories can consist of several categories depending on the point of view, the apparel that the model uses, and the purpose of the way the picture is taken. Some of the images obtained have characteristics such as folded, hung, and stacked clothes, rings resembling earrings, bracelets similar to necklaces, and several different image positions. In addition to these few things, background, partial occlusion, and light levels can affect the detection of fashion and category accessories. For earing, the picture is detached from the ear and used as the background, but neck and clothes serve as the background for the neckless. The bracelet, ring, and belt categories, with the wrist, fingers, and pants as backgrounds, respectively.

Based on the characteristics of the existing image, we collected datasets from several online stores on the e-commerce platform related to fashion and accessories. Over the past few years, there have been a number of datasets relating to fashion and accessories [3], [4], [6], [15], [19], [20]. From some of these datasets, there is not much imagery related to the classification of fashion and accessories that are based on e-commerce platforms. Shopee Thailand's e-commerce is becoming the main place for image capture based on research [28]. Shopee data scraper is used to perform image extraction from the Shopee web platform.

In this study, we added five additional categories to the Shopee image dataset: bracelet, ring, necklace, and earrings. We focused on these classes in order to better identify the tiniest objects. Various images of fashion products and accessories were collected from seller stalls on the Shopee web platform Figure 1. In addition, we also collect imagery from the Google Image web platform based on ten categories. The approaches of image data augmentation (brightening, rotation, and mosaic) were utilized to obtain a greater variety of images and improved model precision. Brightening is performed to train the model on objects with varying levels of brightness. The mosaic technique is used to integrate four distinct photos into a single photograph so that the model can recognize tiny things, while the rotation technique is used to improve the illumination of the object pattern during the final step of augmentation. Table 1 displays the number of photographs in each category as well as the augmented images.

The collected images are then sorted and annotated in the PASCAL VOC format manually using LabelImg. The format is readable by YOLO, an XML file containing information about the class, coordinates, height, and width of the bounding boxes. These two sets of images were then made into one Shopee Thailand image dataset [28] which was published on the IEEE Data Port.

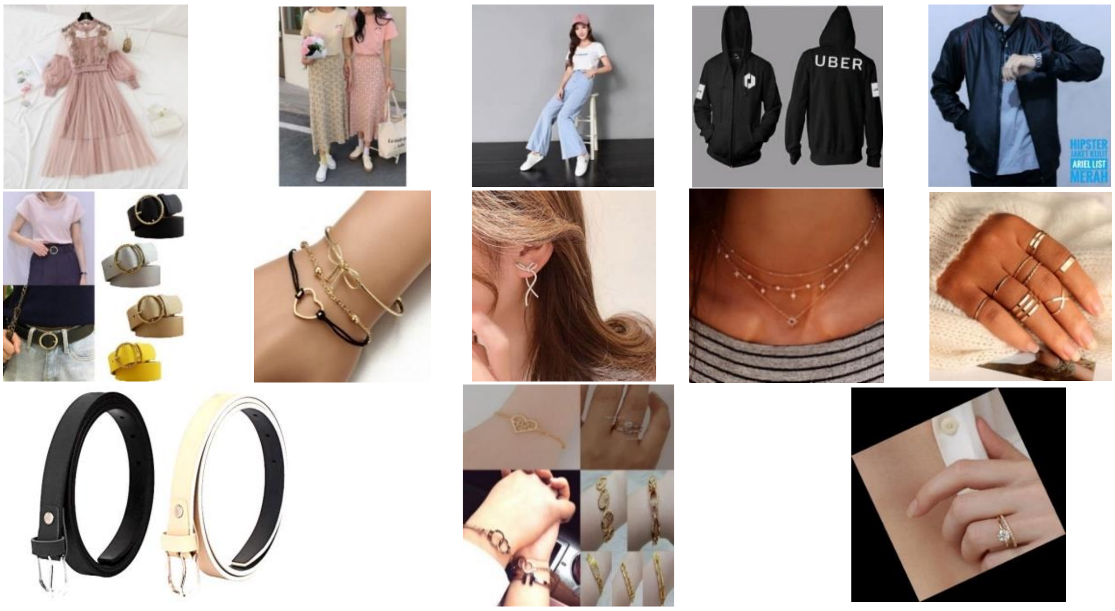


Figure 1. Shopee image dataset

Table 1. Distribution of new Shopee image dataset

Categories	Number of images
Dress	699
Belt	343
Pants	741
Bracelet	452
Earrings	159
Jacket	684
Necklace	219
Hoodie	1059
Ring	483
Skirt	933
Brightening	4968
Mosaic	1293
Rotation	1656
Total	13689

2.2. FC-YOLOv4 architecture and deployment

YOLOv4 is the fundamental deep-learning algorithm employed in this study to create FC-YOLOv4. The three major components of YOLOv4 are the head, neck, and backbone. The backbone comprises a convolution layer, a normalization layer, and an activation function and is responsible for feature extraction. CSPDarknet53 [29] was utilized in this investigation to get a greater frame rate [30] than ResNet-50 [31], VGG16 [32], CSPREsNeXt50 [29], and EfficientNet [33]. In the neck section, spatial pyramid pooling (SPP) blocks [34] and PANet [35] are utilized in conjunction. The YOLOv4 single-stage object detector is utilized to produce predictions in the Head portion. Figure 2 depicts the overall network architecture of the proposed FC-YOLOv4 model. Additionally, the activation function portion was modified for improved detection precision. FC-YOLOv4 substitutes rectified linear unit (ReLU) with the Mish technique on the backbone in order to boost the network model's depth. The LeakyReLU activation function is still employed on the head and neck. As shown in (1) and (2) depict the calculation of the Mish and LeakyReLU activation functions.

$$y_{mish} = x\ tanh(ln(1 + e^{x})) \quad (1)$$

$$y_{Leaky\ Relu} = \begin{cases} x, & if\ x\ \geq 0 \\ \lambda x, & if\ x < 0 \end{cases} \quad (2)$$

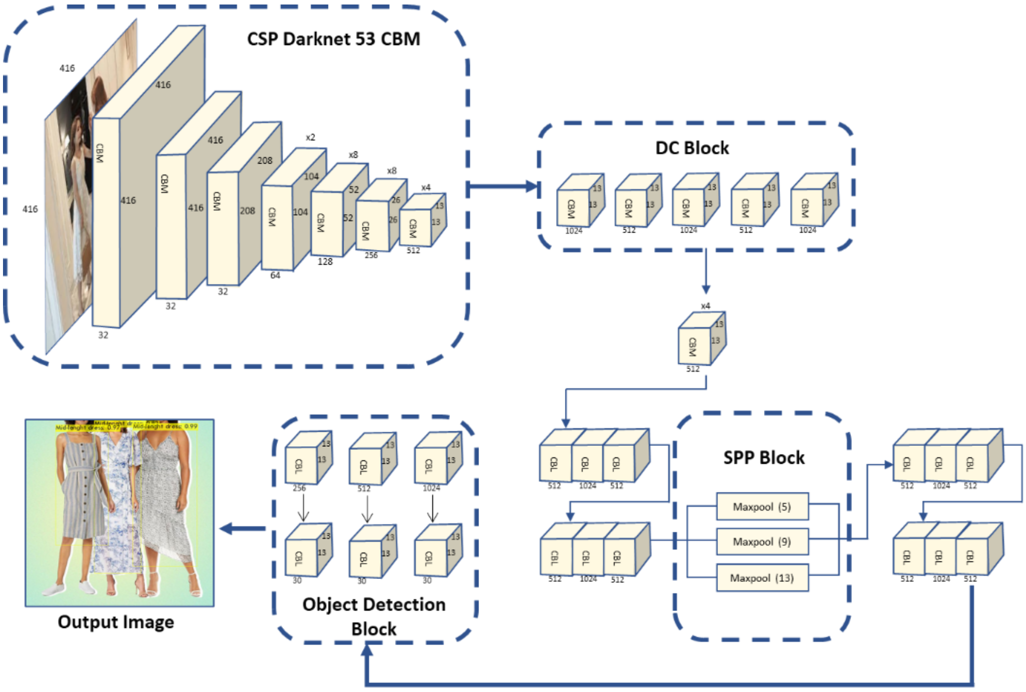


Figure 2. FC-YOLOv4 network architecture

2.3. TensorFlow lite for real-time scenario

In general, gathered datasets are annotated after collection. It is necessary to annotate images with labels or image markings so that machines can better learn and identify items [36]. The annotated data is then subjected to image augmentation and pre-processing in order to provide a greater range of training arrays. In addition, YOLOv3, YOLOv4, and FC-YOLOv4 were used to train the results of the augmentation and pre-processing. Each model must be trained for approximately 5 hours on Google Colab, which requires 20,000 epochs (Tesla T4).

The model built by these three approaches has a large weight size to be used in mobile apps; hence, the TensorFlow-Lite mobile library is employed. This library will drastically reduce the model's weight, allowing it to be deployed into a mobile application. The converter and inverter are the two modules that make up TensorFlow-Lite. The converter serves as a performance optimization to maintain its efficiency, while the inverter serves as a model optimization when operating on a mobile device. YOLOv3 Lite, YOLOv4 Lite, and FC-YOLOv4 Lite are then incorporated into a mobile application and compared using seven parameters: i) normal model size, ii) lite model size, iii) application size, iv) loading time, v) accuracy, vi) speed, vii) average and maximum RAM usage. Experimental setup and procedure can be seen in Figure 3. In this study, mobile applications are deployed using Android Studio.

2.4. Evaluation metrics

Several criteria, including precision (P), recall (R), F1 score, AP, mAP, and intersection over union (IoU), were utilized to compare the performance of the three models [37]. The accuracy of a set of object detection from the model is evaluated using mAP. The amount of overlap between the predicted bounding box and the underlying truth is measured using mAP, which ranges from 0 to 1, as opposed to IoU, which is employed while calculating mAP. As shown in (3) to (8), the mathematical representation of these has six measures.

$$P = \frac{TP}{FP+TP} \quad (3)$$

$$R = \frac{TP}{FN+TP} \tag{4}$$

$$F1\ score = \frac{2PR}{P+R} \tag{5}$$

$$AP = \int_0^1 p(r)dr \tag{6}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{7}$$

$$IoU = \frac{Area\ of\ overlap}{Area\ of\ union} \tag{8}$$

Nomenclature for evaluation metrics:

P	: precision is the level of accuracy between the user’s information request and the system’s response
TP	: true positive
FP	: false positive
R	: recall is the success rate of the system in retrieving information
FN	: false negative
p(r)	: plotting precision

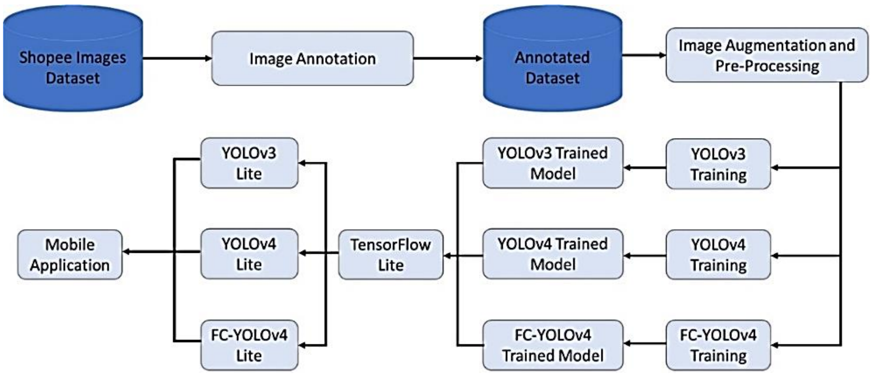


Figure 3. Experimental setup and procedure

3. RESULTS AND DISCUSSION

This section compares FC-YOLOv4 with YOLOv4 and YOLOv3 to evaluate its performance. Instead of YOLOv3 and YOLOv4, CSPDarknet53 is utilized as the backbone with an input image size of 416×416 pixels. In each iteration of the 16,000 and 18,000 training steps, the value 64 is used for the batch size. The number of classes multiplied by ten times 2,000 results in a maximum batch size of 20,000.

3.1. Evaluation of the FC-YOLOv4 model with traditional YOLOv4 and YOLOv3 models

Comparative tests between the suggested model and the conventional model are undertaken to evaluate the performance of the proposed system. Afterward, the proposed FC-YOLOv4 was compared to YOLOv3 and YOLOv4. On the dataset created as described in section 2.1, 20,000 epochs were utilized for training. The above-mentioned metrics (i.e., AP of each class, P, and R) are collected and accumulated at the conclusion of 20,000 epochs, and the outcomes are displayed in Table 2. In each class, the YOLOv3, YOLOv4, and FC-YOLOv4 models are compared using a summary of the results. It was observed that after 20,000 epochs, FC-YOLOv4's mAP% was better than YOLOv4 and YOLOv3. As seen in Table 3, the FC-YOLOv4 model yielded superior results for every statistic. In the validation set, the FC-YOLOv4 model attained the highest mAP percentage of 99.84%. Compared to the intended model, YOLOv3 reached 93.74%, and YOLOv4 achieved 99.81%, with margins of 6.1% and 0.3%. The P and R values for YOLOv3 were 0.96 and 0.74, while YOLOv4 and FC-YOLOv4 had ratings of 0.99 and 0.99, respectively. Therefore, FC-YOLOv4 and YOLOv4 performed better than YOLOv3 by 3.03% for P and by 25.25% for R. F1 scores were determined as 0.83 for YOLOv3, 0.99 for YOLOv4, and 0.99 for FC-YOLOv4. FC-YOLOv4 and YOLOv4, therefore, performed 16.16% better than YOLOv3. FC-YOLOv4 is also superior to YOLOv4 and YOLOv3 by 12.49% and 1.23%, respectively, in terms of IoU.

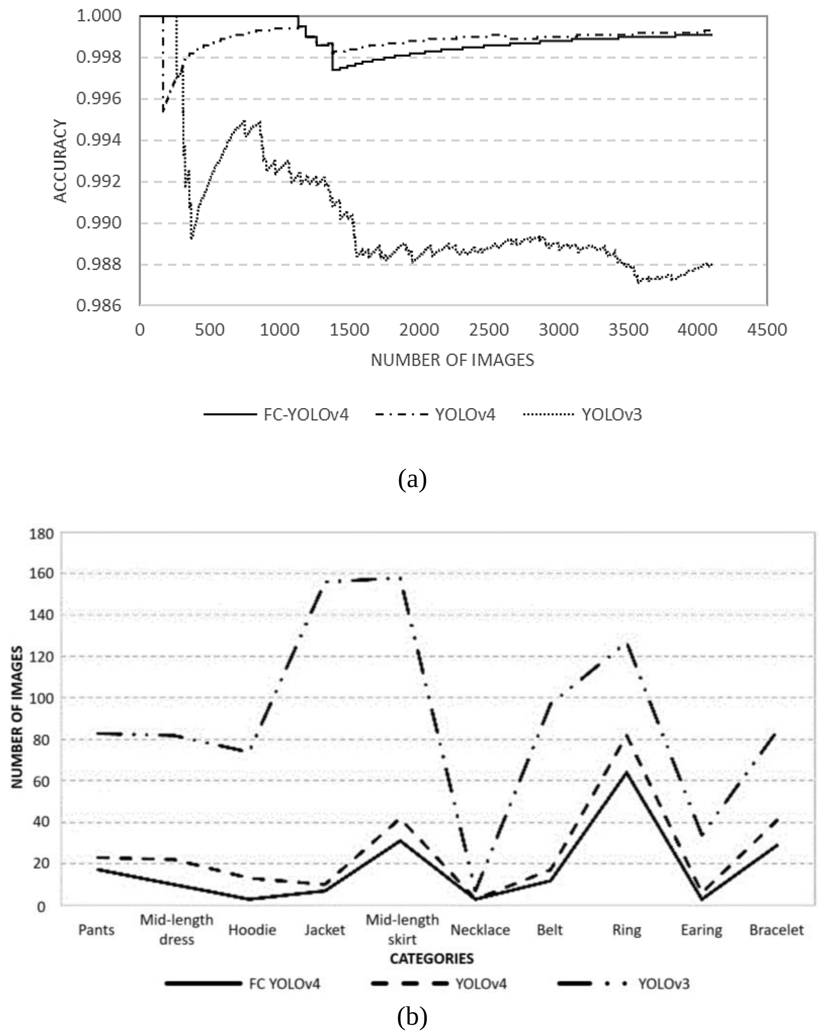
Table 2. Average precision for each category

Model	Iterations	Classes AP (%)									
		Pants	Mid-length dress	Hoodie	Jacket	Mid-length skirt	Necklace	Belt	Ring	Earing	Bracelet
YOLOv3	20000	96.80	89.84	92.59	92.97	88.61	98.26	96.83	93.64	92.01	95.82
YOLOv4		99.97	99.96	99.62	99.99	99.71	100.00	99.89	99.67	99.79	99.47
FC-YOLOv4		99.97	99.93	99.67	99.99	99.75	100.00	99.85	99.83	99.90	99.53

Table 3. Evaluation matrices

Model	Iterations	TP	FP	FN	P	R	F1-score	mAP (%)	IoU (%)
YOLOv3	20000	21649	902	7713	0.96	0.74	0.83	93.74	77.44
YOLOv4		29116	259	246	0.99	0.99	0.99	99.81	87.40
FC-YOLOv4		29093	179	269	0.99	0.99	0.99	99.84	88.49

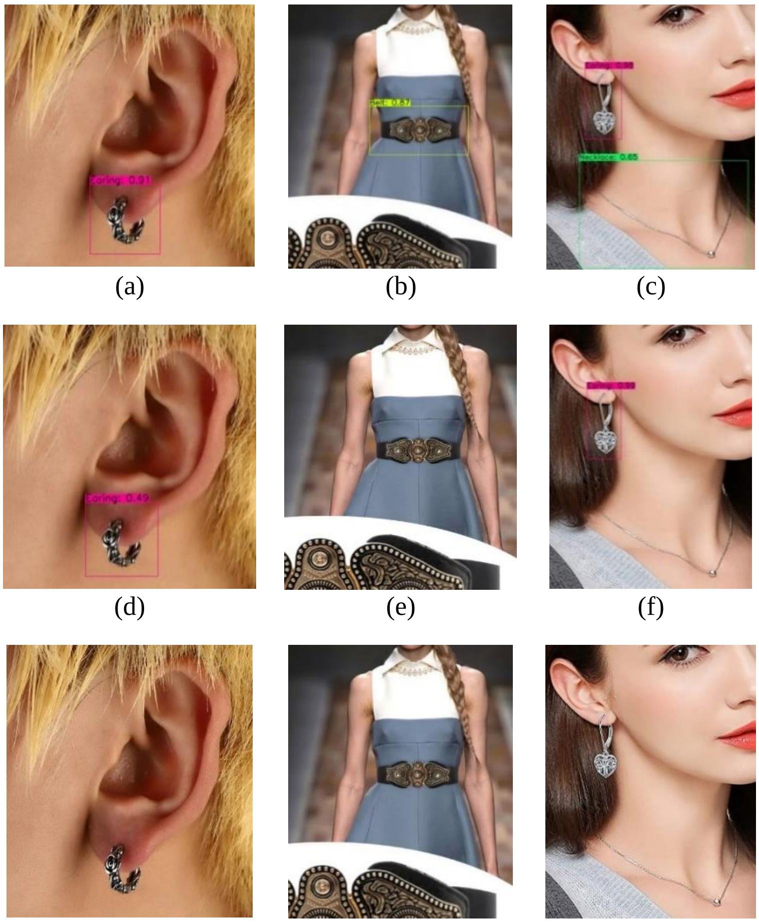
Figure 4(a) depicts the Recall comparison value in which YOLOv3 and YOLOv4 attain an accuracy value that is one less than that of FC-YOLOv4. Figure 4(a) can be used to establish that FC-YOLOv4 possesses a higher level of accuracy. Figure 4(b) also depicts the number of false positives in each category, where YOLOv3 has the greatest error rate, YOLOv4 has the second highest number of false positives after YOLOv3, and FC-YOLOv4 has the lowest error rate. As shown in Figures 5(a), (b), and (c), we represent several detection results where FC-YOLOv4 shows superiority in detecting single or multiple objects in one image. Figures 5(d), (e), and (f) represent the detection result of the YOLOv4 for accessory categories. The YOLOv3 model is unable to recognize the accessory categories depicted in Figures 5(g), (h), and (i).



(a)

(b)

Figure 4. Evaluation criteria results (a) recall and (b) number of false positive evaluation



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 5. Detection results of (a) earing FC-YOLOv4, (b) belt FC-YOLOv4, (c) multiple detection FC-YOLOv4, (d) earing YOLOv4, (e) belt YOLOv4, (f) multiple detection YOLOv4, (g) earing YOLOv3, (h) belt YOLOv3, and (i) multiple detection YOLOv3

3.2. Evaluation of the approach in mobile deployment

The TensorFlow Lite technique modifies the configuration files and weights of YOLOv3, YOLOv4, and FC-YOLOv4 to produce models with lower memory footprints. Table 4 compares the parameter of each model. YOLOv3 has the smallest model size, YOLOv4 the second biggest, and FC-YOLOv4 the largest. On the basis of the regular and lite sizes of the three models, one may deduce that the average size has decreased by 50%.

Table 4. Evaluation of mobile application deployment

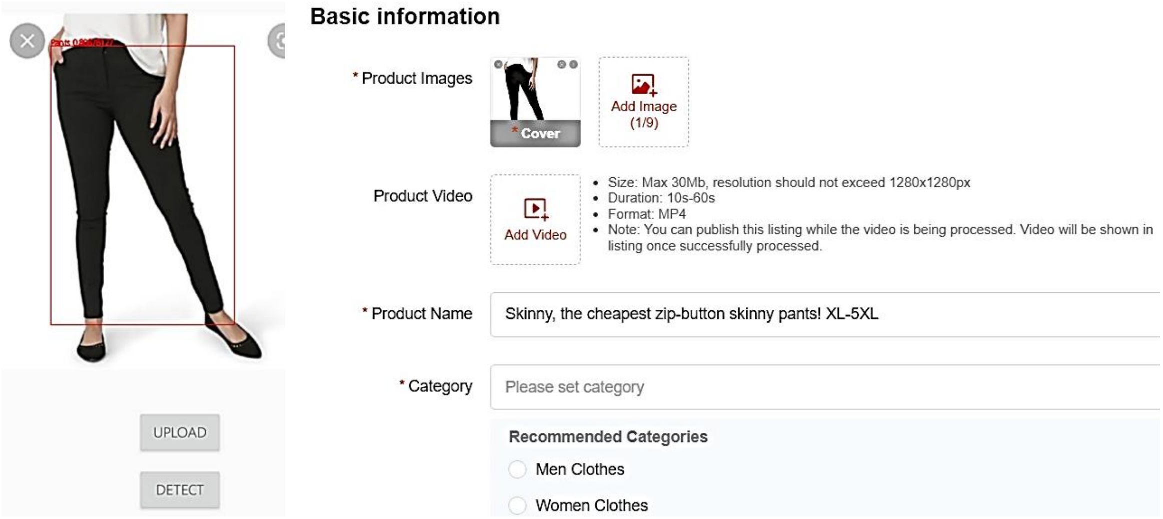
Parameters	YOLOv3 Lite	YOLOv4 Lite	FC-YOLOv4 Lite
Normal model size	235.1 MB	244.3 MB	302.4 MB
Lite model size	117 MB	122 MB	151 MB
Application size	138 MB	137 MB	168 MB
Loading time	25.268 ms	31.865 ms	34.06 ms
Average RAM usage	7.4 MB	7.2 MB	1.1 MB
Maximum RAM usage	769 MB	1.3 GB	567 MB

In this investigation, the smaller model will be loaded into memory more quickly. YOLOv3, with a model size of 117 MB, loaded in 25,268 milliseconds. YOLOv4, with a model size of 122 MB, loaded in 31,865 milliseconds. And FC-YOLOv4, with a model size of 151 MB, has a loading time of 34.06 milliseconds. This may be due to the smaller network size of YOLOv3 and YOLOv4 relative to FC-YOLOv4, which enables YOLOv3, and YOLOv4 to learn quickly while FC-YOLOv4 learns precisely.

Object recognition and image processing are examples of computationally complex jobs that can result in a higher average RAM consumption. This subsection examines the RAM used in the object recognition algorithm to recognize numerous images. According to Table 2, YOLOv3, YOLOv4, and FC-YOLOv4 use an average of 7.4 MB, 7.2 MB, and 1.1 MB of RAM, respectively. Comparatively, the greatest RAM consumption is 769 MB, 1.3 GB, and 576 MB. Therefore, we may infer that FC-YOLOv4 performed the best.

3.3. Evaluation of the approach in a real scenario

The final evaluation of this study compares the results of FC-YOLOv4 Lite detection on a mobile device to the Shopee web application. We collected some of the products with the wrong categories from the Shopee Thailand website. Figure 6(a) depicts one of the test results for a real-world scenario in which FC-YOLOv4 accurately recognizes the pants category while the Shopee recommendation system recognizes men's and women's clothing as shown in Figure 6(b).



Basic information

* Product Images

Product Video

- Size: Max 30Mb, resolution should not exceed 1280x1280px
- Duration: 10s-60s
- Format: MP4
- Note: You can publish this listing while the video is being processed. Video will be shown in listing once successfully processed.

* Product Name: Skinny, the cheapest zip-button skinny pants! XL-5XL

* Category: Please set category

Recommended Categories

○ Men Clothes

○ Women Clothes

(a) (b)

Figure 6. Real scenario comparison of (a) FC-YOLOv4 mobile application and (b) Shopee web application

4. CONCLUSION

This article uses the FC-YOLOv4 Lite algorithm to classify fashion products. FC-YOLOv4 is a single-stage detection approach that excels in terms of efficiency, precision, and model size as the primary concern. In this study, Google Colab was employed as a development and experimentation environment (training, validation, and testing). Five categories of fashion and five categories of accessories are represented in a Google and Shopee-collected image database. This study focuses not only on model building and testing but also on the deployment of mobile applications. In addition, the deployed model is evaluated by comparing it to the official Shopee website. The experimental findings demonstrate that the FC-YOLOv4 single-stage object detection algorithm excels in numerous measures, including mAP, precision, recall, F1 score, model size, and RAM usage.

ACKNOWLEDGEMENTS

This work was fully funded by Rajamangala University of Technology Thanyaburi. We are appreciative to RMUTT for the participation, assistance, and computational resources utilized in our studies. Additionally, we appreciate the E-CUBE I scholarship program for the administration and financial support.

REFERENCES

[1] S. Sarbu, "The new economic warfare," *Annals-Series on Military Sciences*, vol. 10, no. 1, pp. 137–142, 2018.

[2] A. Salamzadeh, P. Ebrahimi, M. Soleimani, and M. Fekete-Farkas, "Grocery apps and consumer purchase Behavior: Application of gaussian mixture model and multi-layer perceptron algorithm," *Journal of Risk and Financial Management*, vol. 15, no. 10, Sep. 2022, doi: 10.3390/jrfm15100424.

[3] M. S. Amin, C. Wang, and S. Jabeen, “Fashion sub-categories and attributes prediction model using deep learning,” *The Visual Computer*, Jun. 2022, doi: 10.1007/s00371-022-02520-3.

[4] M. H. Kiapour, K. Yamaguchi, A. C. Berg, and T. L. Berg, “Hipster wars: Discovering elements of fashion styles,” in *Computer Vision-ECCV 2014*, 2014, pp. 472–488.

[5] W. Di, C. Wah, A. Bhardwaj, R. Piramuthu, and N. Sundaresan, “Style finder: Fine-grained clothing style detection and retrieval,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 2013, pp. 8–13, doi: 10.1109/CVPRW.2013.6.

[6] J. Huang, R. Feris, Q. Chen, and S. Yan, “Cross-domain image retrieval with a dual attribute-aware ranking network,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 1062–1070, doi: 10.1109/ICCV.2015.127.

[7] A. Veit, B. Kovacs, S. Bell, J. McAuley, K. Bala, and S. Belongie, “Learning visual clothing style with heterogeneous dyadic co-occurrences,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4642–4650.

[8] N. Ramzan *et al.*, “ACM international workshop on social and behavioral networked media access (SBNMA’11),” in *Proceedings of the 19th ACM international conference on Multimedia-MM ’11*, 2011, pp. 611–612, doi: 10.1145/2072298.2072390.

[9] M. Shao, L. Li, and Y. Fu, “What do you do? Occupation recognition in a photo via social context,” in *2013 IEEE International Conference on Computer Vision*, Dec. 2013, pp. 3631–3638, doi: 10.1109/ICCV.2013.451.

[10] E. Simo-Serra, S. Fidler, F. Moreno-Noguer, and R. Urtasun, “Neuroaesthetics in fashion: Modeling the perception of fashionability,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 869–877, doi: 10.1109/CVPR.2015.7298688.

[11] H. Chen, A. Gallagher, and B. Girod, “Describing clothing by semantic attributes,” in *Computer Vision-ECCV 2012*, 2012, pp. 609–623.

[12] K. Laenen, S. Zoghbi, and M.-F. Moens, “Cross-modal search for fashion attributes,” in *Proceedings of the KDD 2017 Workshop on Machine Learning Meets Fashion*, 2017, pp. 1–10.

[13] Z. Song, M. Wang, X.-S. Hua, and S. Yan, “Predicting occupation via human clothing and contexts,” in *2011 International Conference on Computer Vision*, Nov. 2011, pp. 1084–1091, doi: 10.1109/ICCV.2011.6126355.

[14] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, 2005, vol. 1, pp. 886–893, doi: 10.1109/CVPR.2005.177.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[16] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, “DeepFashion: Powering robust clothes recognition and retrieval with rich annotations,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 1096–1104, doi: 10.1109/CVPR.2016.124.

[17] C. Szegedy *et al.*, “Going deeper with convolutions,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.

[18] Y. Wei *et al.*, “Cross-modal retrieval with CNN visual features: A new baseline,” *IEEE Transactions on Cybernetics*, vol. 47, no. 2, pp. 1–12, 2016, doi: 10.1109/TCYB.2016.2519449.

[19] H. Lai, Y. Pan, Ye Liu, and S. Yan, “Simultaneous feature learning and hash coding with deep neural networks,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 3270–3278, doi: 10.1109/CVPR.2015.7298947.

[20] H. Lai, P. Yan, X. Shu, Y. Wei, and S. Yan, “Instance-aware hashing for multi-label image retrieval,” *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2469–2479, Jun. 2016, doi: 10.1109/TIP.2016.2545300.

[21] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “DeepFace: closing the gap to human-level performance in face verification,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1701–1708, doi: 10.1109/CVPR.2014.220.

[22] R. Xia, Y. Pan, H. Lai, C. Liu, and S. Yan, “Supervised hashing for image retrieval via image representation learning,” in *AAAI’14: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014, pp. 2156–2162.

[23] J. Yu, X. Yang, F. Gao, and D. Tao, “Deep multimodal distance metric learning using click constraints for image ranking,” *IEEE Transactions on Cybernetics*, vol. 47, no. 12, pp. 4014–4024, Dec. 2017, doi: 10.1109/TCYB.2016.2591583.

[24] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang, “Learning from massive noisy labeled data for image classification,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 2691–2699, doi: 10.1109/CVPR.2015.7298885.

[25] Q. Dong, S. Gong, and X. Zhu, “Multi-task curriculum transfer deep learning of clothing attributes,” in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Mar. 2017, pp. 520–529, doi: 10.1109/WACV.2017.64.

[26] H. J. Kim, D. H. Lee, A. Niaz, C. Y. Kim, A. A. Memon, and K. N. Choi, “Multiple-clothing detection and fashion landmark estimation using a single-stage detector,” *IEEE Access*, vol. 9, pp. 11694–11704, 2021, doi: 10.1109/ACCESS.2021.3051424.

[27] C.-H. Lee and C.-W. Lin, “A two-phase fashion apparel detection method based on YOLOv4,” *Applied Sciences*, vol. 11, no. 9, Apr. 2021, doi: 10.3390/app11093782.

[28] Y. Thwe, N. Jongsawat, and A. Tungkasthan, “A semi-supervised learning approach for automatic detection and fashion product category prediction with small training dataset using FC-YOLOv4,” *Applied Sciences*, vol. 12, no. 16, Aug. 2022, doi: 10.3390/app12168068.

[29] C.-Y. Wang, H.-Y. M. Liao, I.-H. Yeh, Y.-H. Wu, P.-Y. Chen, and J.-W. Hsieh, “CSPNet: A new backbone that can enhance learning capability of CNN,” *Prepr. arXiv1911.11929*, Nov. 2019.

[30] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “Yolov4: Optimal speed and accuracy of object detection,” *arXiv preprint arXiv:2004.10934*, 2020.

[31] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *Prepr. arXiv1512.03385*, Dec. 2015.

[32] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *Preprint arXiv:1409.1556*, Sep. 2014.

[33] M. Tan and Q. Le, “Efficientnet: rethinking model scaling for convolutional neural networks,” in *International conference on machine learning*, 2019, pp. 6105–6114.

[34] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” in *Computer Vision-ECCV 2014*, 2014, pp. 346–361.

[35] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, “Path aggregation network for instance segmentation,” *Prepr. arXiv1803.01534*, Mar. 2018.

[36] D. Komura and S. Ishikawa, “Machine learning approaches for pathologic diagnosis,” *Virchows Archiv*, vol. 475, no. 2, pp. 131–138, Aug. 2019, doi: 10.1007/s00428-019-02594-w.

[37] Y. Liu, L. Wang, L. Zhao, and Z. Yu, *Advances in natural computation, fuzzy systems and knowledge discovery*, vol. 1074. Cham: Springer International Publishing, 2020.

BIOGRAPHIES OF AUTHORS

Yamin Thwe was born in Yangon, Myanmar, in 1997. She received her B.E. degree in information technology from the Technological University, Hmawbi, Myanmar, in 2020. She worked as a software engineer for two years, creating solutions for numerous e-commerce platforms, government agencies, and non-governmental groups. In 2021, she enrolled as an E-CUBE I Scholarship student in the Faculty of Data and Information Science at Rajamangala University of Technology, Thanyaburi, Thailand. Machine learning, computer vision, and big data security are some of her current research interests. She can be contacted at yamin_t@mail.rmutt.ac.th.

Nipat Jongsawat was born in Bangkok, Thailand, in 1977. He received a B.S. degree in electrical engineering and M.S. in computer information systems from Assumption University in 1999 and 2002, respectively. He received a Ph.D. degree in Information Technology in Business from Siam University, Thailand, in 2011. He has been an assistant professor with the Mathematics and Computer Science Department, Faculty of Science and Technology, Rajamangala University of Technology Thanyaburi. He has been serving as the faculty's dean since 2018. He is the author of more than 40 articles and 3 book chapters. His research interests include artificial intelligence, collaborative computing, human-computer interaction, decision support systems, group decision support system, group decision-making, computer-supported collaborative learning, computer-supported cooperative work, and business data processing. He is an associate editor of the Journal Progress in Applied Science and Technology. He can be contacted at nipat_j@rmutt.ac.th.

Anucha Tungkasthan received a Ph.D. degree in information technology in Business (in cooperation with the University of Pittsburgh, USA) from Siam University, Bangkok, Thailand, in 2012 and an M.S. degree in computer education from King Mongkut's University of Technology North Bangkok, Thailand, in 2004. He is currently an assistant professor with the Department of Computer Technology, Faculty of Science and Technology, Rajamangala University of Technology Thanyaburi. His research interest includes object detection and tracking, real-time image processing, content-based image retrieval, machine learning, and deep learning. His awards and honors include The Certification of Merit for The World Congress on Engineering, 2010, and The Outstanding Thesis Award of the Association of Private Higher Education Institutions of Thailand. He can be contacted at anucha_t@rmutt.ac.th.