ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

NORTH SULAWESI SINGLE LOCAL FRUIT DETECTION USING EFFICIENT ATTENTION MODULE BASED ON DEEP LEARNING ARCHITECTURE

Vecky C. Poekoel[1], Muhamad Dwisnanto Putro[2], Jane Litouw[3], Rivaldo Karel[4], Pinrolinvic D. K. Manembu[5], Abdul Haris Junus Ontowirjo[6], Feisy D. Kambey[7], Reynold F. Robot[8]

[1-8]Department of Electrical Engineering, Engineering Faculty, Sam Ratulangi University

email: vecky.poekoel@unsrat.ac.id[1], dwisnantoputro@unsrat.ac.id[2], janelitouw@unsrat.ac.id[3], rivaldokarel023@student.unsrat.ac.id[4], pmanembu@unsrat.ac.id[5], aharisjo@unsrat.ac.id[6], feisykambey@unsrat.ac.id[7], reynold.robot@unsrat.ac.id[8]

Abstract

A Local fruit detection system is an agricultural vision field that can be implemented to increase the profit of a commodity. Besides that, North Sulawesi has a variety of local fruits which are widely used by people in their area and have a high selling value. The sorting system is an essential process of agricultural robots to sequentially separate fruit one by one. This automation process requires an accurate vision system to detect and separate fruit precisely and precisely. In addition, the implementation of a practical application demands a method to be able to work in real-time on low-cost devices. This work aims to design a local single fruit detection system for Sulawesi North by applying deep learning architecture to produce high performance. The architecture is designed to consist of an effective backbone for rapidly separating the distinctive features, an efficient attention module to improve feature extraction performance, and a classifier module employed to estimate the probabilities of each local fruit category. As a result, the designed model produces an accuracy value of 99,27% and 99,57% on the Fruits-360 and the local datasets, respectively. It outperforms other light architectures. In addition, deep learning models are designed to produce higher efficiency values than other competitors and can operate quickly at 100,488 Frames per Second.

Keywords : local fruits, detection system, convolutional neural network, efficient architecture, attention module

Received: 05-12-2022 | Revised: 23-04-2023 | Accepted: 11-05-2023
DOI: https://doi.org/10.23887/janapati.v12i2.54754

INTRODUCTION

North Sulawesi has abundant natural resources, one of which is the plantation sector which has high-selling commodities that can be exported abroad. North Sulawesi fruit commodities are diverse such as bananas, mangoes, duku, langsat, durian, salak and so on. Based on data from the Central Bureau of Statistics of North Sulawesi Province, in 2021 the most fruit production in quintals is as follows: banana (1,019,222), mango (235,567), papaya (230,103), durian (182,330), duku/langsat (38,773), salak (30,443) and many other fruits [1]. Some fruits increased in the number of products but some decreased significantly compared to previous years. This increase in productivity needs to be aligned with selling value which will positively impact farmers and plantation business actors. The implementation of supporting technologies such as artificial intelligence with the application of computer vision for the introduction of various local fruits, along with the ability to detect the ability to sort correctly, makes this plantation system more advanced and can be better competitive. In the future, this plantation system will be useful in learning and carrying out independent tasks such as harvesting, sorting for packing, and weighting the selling value of plantation products that can not only be consumed locally but also for export.

These processes require effective technology and can operate automatically such as robot technology. Agricultural robots offer a structured process to help increase production value in various sectors. Vision on a robot is an important element in detecting, recognizing, and understanding surrounding objects. Along with the development of robot technology, there is a

technology that can process raw data for decision making, in this case, needed in fruit image recognition for further analysis and execution in its application. Meanwhile, the number of uses of artificial intelligence in the industrial world has increased by 270% over the past four years. Deep Learning is a branch of Machine Learning (part of artificial intelligence) that models data with a high level of abstraction using many layers of neurons with complex structures or non-linear transformations [2]. Due to the increasing amount of data and computational capabilities, the structure of deep learning has attracted attention for development and applications in various fields for image processing, speech recognition, and object detection [3], [4], [5], [6], [7].

Performance and speed requirements in real-world applications and industrial automation demand a sensing method to operate quickly on cheaper devices. On the other hand, Convolutional Neural Network (CNN) architecture, which is a robust deep learning method, tends to rely on expensive video graphics tools. So that the need for an efficient model is very large when related to economic factors. In addition, the accuracy of the method is also the most important thing. Therefore, an efficient and high-performance model was developed to detect and recognize local fruit of North Sulawesi which can be reliably implemented in a low-cost device.

Based on the above review, the details of the contribution of this work can be summarized as follows:

1. A new and fast single-fruit detection system is proposed to detect and recognize local fruits of North Sulawesi.
2. The new deep learning architecture focuses on the efficiency of data processing without compromising performance. Efficient attention modules are introduced to capture relevant and essential features through spatial dimension representation and feature map channels used to improve the performance of classification systems.
3. The performance of the classification system achieves high accuracy and efficiency. Implementing detection systems on robotic processing devices such as Jetson Nano obtains a speed of 44 frames per second.

RELATED WORK

Studies related to fruit classification systems have been conducted using artificial neural networks [8], [9], [10], [11]. Kabir et al [12] applied a deep neural network with gradual input to design a SpinalNet architecture. The architecture designed is able to adapt to traditional learning and transfer learning approaches. This method tested on the fruit dataset (Fruits-360) and obtained high performance with parameters of 125M. On the other hand, the Cascade-ANFIS (Adaptive Network-based Fuzzy Inference System) method has been used to predict 131 pieces by applying various descriptors to identify distinctive features [13]. This method has been verified in the process of evaluating iterations and confusion matrix that obtains high accuracy with low computation in operation. The next efficient model was designed by Pande et al [14]. This work established an inexpensive classification system for classifying fruits using deep CNNs. InceptionV3 is used as a feature extractor that applies transfer learning methods to improve its accuracy. Modern methods apply deep learning to extract features by using the Convolutional Neural Network method. Some work has shown high performance results when assigned to classify images using this method [15], [16], [17]. Even popular architectures have been designed to overcome saturation problems in data training processes containing complex instances [18]. These jobs implement weighted kernels that are used to find distinctive features of data learning-based spatial operations. These weights go through an update process in the backpropagation to drive the predicted output according to the target. On the other hand, efficient and light architectures have emerged [19], [20], [21], [22], [23], [24], [25] to predict the class of an image focusing on usage low computing power. Judging from the accuracy, these efficient architectures still produce low accuracy when compared to architectures that use deep layers.

Several previous works have introduced the attention module to improve the quality of the feature map of the external backbone module [26], [27]. This method represents a global feature encapsulating useful information in a channel-based vector. Vector weights are used to update input features through the learning process. On the other hand, CBAM [28] combines spatial and global context representations to enhance important features. Different things are done by coordinate attention modules that apply pooling operations on different axes to obtain a representation of varying spatial features. The attention module has been shown to improve the accuracy and precision of object classification and detection systems by strengthening important specific features and reducing trivial features.

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

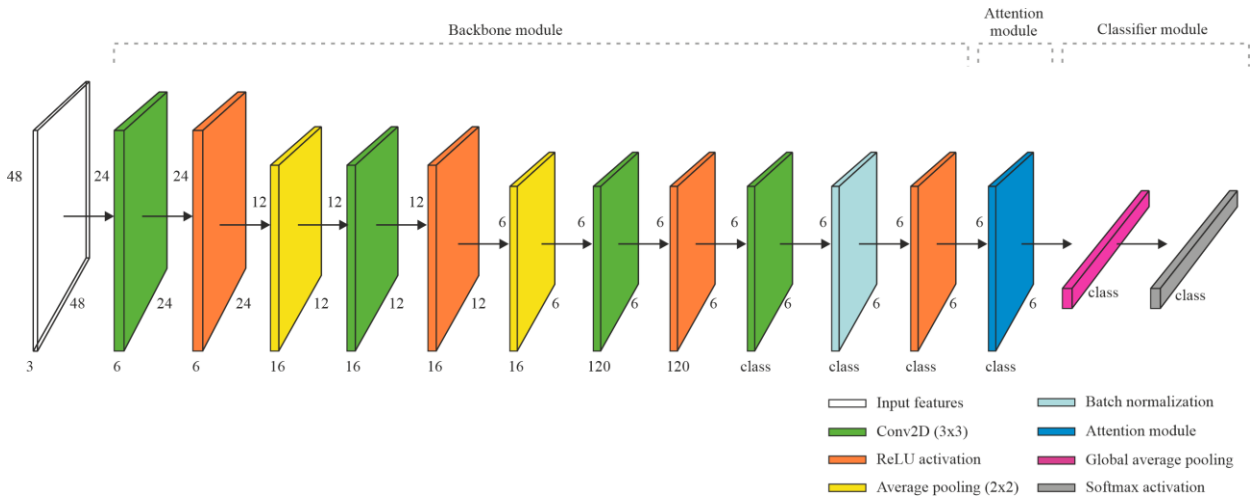

Figure 1. The Overall Architecture of North Sulawesi's Local Fruit Classification System

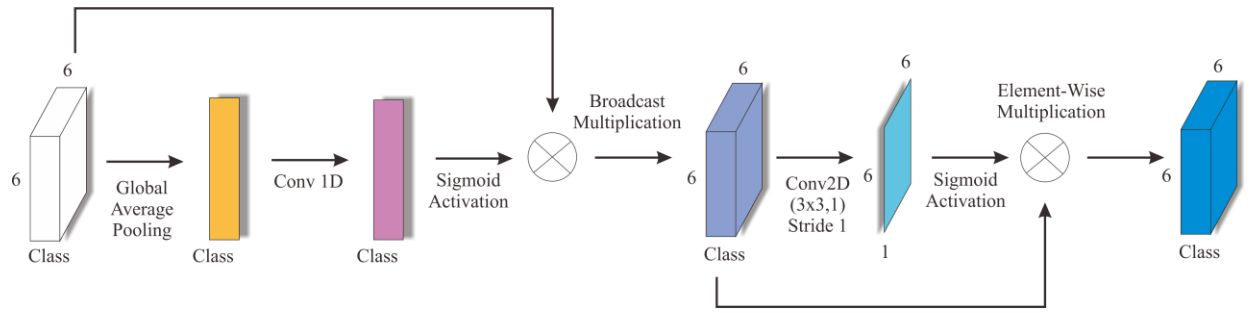

Figure 2. Efficient Attention Module

PROPOSED METHOD

Robotic applications require a visual method to operate quickly. Therefore, the proposed deep learning architecture focuses on improving accuracy and contributes to the model's efficiency. Figure 1 shows the architecture of the fruit classification system consisting of a backbone, attention module, and classifier.

1. Backbone module

The feature extraction process is the most important part of an image classification system to obtain specific features. The proposed backbone module implements a light CNN architecture without generating excessive computational complexity. This design employs small operation and canal lining but can capture the characteristics of each local fruit. The operating model applies 2D convolution operations to the input feature map employing trained weighted filters as shown in equation (1):

$$y_i = \sum_{i=0} W * x_i + b \qquad (1)$$

where y_i is the output of convolution operations, W is filter weight matrix, x_i is input feature map, and b is bias. The 48 × 48 × 3 image input is convoluted with 3 × 3 convolution sequentially to produce a 6 × 6 feature map with the number of channels adjusted to the number of fruit classes. This weight filter has the ability to optimally distribute features. Some works apply this filter and achieve high performance on their own jobs [17], [18]. In the feature map reduction stage, this design implements average pooling which extracts information by determining the average value of a running window. This efficient method was applied twice to shrink the size of the feature map to eight times smaller than the input image. This reduction approach helps to suppress network computing. On the other hand, convolution operations provide a possible selection of varied features that are used as important information for prediction decision-making.

2. Attention module based on channel and spatial

The shallow backbone module causes the output of the feature map on this network to have poor quality in separating specific features from trivial features. Therefore, previous studies have applied an attention module to improve the accuracy of a vision system [26], [27], [28]. The attention module can capture elements and components that have important information from fruit while enhancing these features. In addition, it also reduces features that are not useful for final predictions. This capability will automatically affect the performance of the classification system positively. The proposed Attention module combines the sequential channel and spatial representation approaches, as presented in Figure 2. This proposal overcomes the weakness of Efficient Channel Attention [29], which can only capture channel-based contextual information. At the start of the process, the feature map input is summarized by applying Global Average Pooling (GAP) to obtain the averaged features based on the channel size. Next, a one-dimensional weight filter is applied to extract the essential channel features. Sigmoid activation is applied to form a weight vector representing each channel's intensity. This vector is used to update the input features. Attention module based on the channel is described in equation (2):

$$C_{att} = x_i \odot \sigma(W_{1D} GAP(x_i)) \qquad (2)$$

where x_i is input of feature map, W_{1D} is the weight of the one-dimensional filter, σ is sigmoid activation, and $\odot$ *broadcast multiplication.* Furthermore, the output of the channel-based attention module is included in a spatial-based attention module. This is useful for capturing important elements along the dimensions of the feature map. The input feature applied convolution operations using 3 × 3 kernels to obtain a single-channel feature map. Then, Sigmoid activation produces a single feature map containing weights of feature intensity. This feature map is used to update the input feature so that this process can improve the quality of the feature map. This spatial-based attention module can be illustrated through equation (3):

$$S_{att} = C_{att} \otimes \sigma(W_{conv3x3} C_{att}) \qquad (3)$$

where $W_{conv3x3}$ is weight filter with a single channel, $\otimes$ multiplication operation for all elements, and σ is sigmoid activation. Combining this attention module can improve the quality of specific feature improvements that focus on strengthening important features and reducing unimportant features. Spatial plane and channel-based operations drive effective feature separation performance, directly impacting precise prediction accuracy. Meanwhile, the combination of this module does not significantly increase the number of parameters and computations. This supports the fruit detection system to keep working fast in real-time with high accuracy.

3. Classifier Module

The classifier module determines class predictions and fruit labels at the network's end. In addition, this module also generates a probability value for each fruit class which takes the maximum value and sets it as the predicted value. This forms a normalized probability weight using softmax activation, which obtains a multinomial distribution of fruit class estimates. The output of the attention module in the form of a tensor is first converted into a vector form. Instead of using the flatten method to convert the size, this time, Global Average Pooling is applied, which is more efficient for getting the average feature representation of each class channel. Furthermore, Fully Connected filters useful features in the final prediction before applying the softmax function. This classifier module is focused on efficiency, so it only implements light extraction operations without producing a lot of computational complexity and parameters.

4. Implementation Setup and Datasets

The training and testing of the classification model use several configurations to optimize the performance. This phase applies a batch size of 128 for 50 epochs. In addition, the Adam optimizer is used to optimize the learning process with a learning speed of 0.001. This speed will be updated automatically with a multiplier of 0.75 if the validation accuracy does not improve within ten epochs.

This study proposes a new local data set manually captured in a laboratory environment. The system uses 5,100 fruit images for training and 1,700 for testing. It consists of 17 local fruit classes of North Sulawesi, including apel Fuji, coklat, durian, jagung, jambu air, jeruk, jeruk nipis, kelapa, Langsat, Manggis, Nanas, Pala, Pepaya, Pisang, rambutan, salak, and tomat. It was picked from 10,200 images for each fruit, which were then manually selected to determine which images were chosen as the image dataset. Each fruit image was captured with a Full HD 1080 web camera with a distance from the camera to the fruit of 30 to 50 cm. In order to obtain the fruit size, shape, and texture variation, it provides three fruits for each fruit type with

manually rotated and random in capturing process. The augmentation techniques were applied by manipulating brightness, contrast, saturation, hue, rotation, and flip to enrich the knowledge of fruit features. Based on this process, it obtains 35,700 images for the training stage and 11,900 for the testing phase.

Table 1. Model Analysis of the Proposed Architecture

Model	Parameters	Accuracy (%)	GFLOPs	Model Speed (FPS)
LeNet	314,211	97.55	0.0063	131.553
LeNet-efficient	160,583	98.50	0.0012	131.347
LeNet-efficient-att	**161,768**	**99.27**	**0.0030**	100.488

Table 2. Comparison Results of The Proposed Attention Module with Other Attention Modules

Model	Parameters	Accuracy (%)
Backbone+SE	164,775	98.51
Backbone+CBAM	165,020	86.65
Backbone+CA	173,581	99.00
Backbone+ECA	160,589	98.83
***Backbone*+ECSA**	**161,768**	**99.27**

RESULTS AND DISCUSSION

This section presents the analysis of the proposed model and performance comparison with other architectures. The performance results are discussed in terms of accuracy, number of parameters, computational complexity, and model speed on low-cost devices, as shown in Table 1. The comparison of the proposed model with other models is shown in Table 2.

1. Model Analysis

The proposed architecture focuses on practical application performance with respect to the accuracy, number of parameters, computational usage, and speed of the model implemented on an embedded system. LeNet inspires the developed backbone model [15], which has fewer parameters than other lightweight architectures, as shown in Table 3. The system was modified to improve accuracy and efficiency for practical application. The development results shown in Table show an increase in accuracy of 0.95% and a reduction in the number of parameters and GFLOPs by 154K and 0.006, respectively. Furthermore, an attention module with a combination of spatial and channel was added (LeNet-efficient-att) to improve the performance of the classification system. It boosts the accuracy by 0.77%. Although the speed result dropped by 30 FPS, it did not significantly increase the parameters and GFLOs. The results show that the proposed model can obtain high performance and efficiency.

The proposed attention module is compared with the popular attention module. The comparison results are shown in Table 2. This experiment places the same efficient backbone in all experiments. Table 2 shows that ECSA (Efficient Spatial and Channel Attention) is the proposed module that obtained higher accuracy than other competitors [26], [27], [28] with the same number of parameters as ECA (efficient channel attention) [29].

2. Performance Comparison with Other Models

The proposed model achieved a high accuracy of 99.27% on the Fruits-360 dataset. This result outperforms popular efficient models such as MobileNetV1, MobileNetV2, MobileNetV3, ShuffleNetV1, ShuffleNetV2, SqueezNetV1a, SqueezeNetV1b, SqueezeNetV1c, and GhostNet. Although the models [12] use ResNet50 and VGG19 to outperform the proposed model, they generate a large number of trained weights. The worked [9] also obtains high parameters compared to our proposed model. Even EfficientNet B0 achieves perfect accuracy. Nevertheless, it has 29 times heavier than the proposed model.

Furthermore, the performance comparison of the model was carried out on a local dataset with 17 fruit categories. The proposed model outperforms popular efficient and lightweight architectures. However, the

performance is still below ResNet34, which differs by 0.2%. This competitor does not generate a small number of training weights. This model produces a parameter of 21 M, which claims ResNet34 has a low-efficiency level. The heavy deep learning architecture slowly operates when implemented on low-cost devices.

Table 3. Comparison Results of The Proposed Model with Other Architectures on Fruits-360 Dataset

Model	Parameters	Accuracy (%)
LeNet	314,211	97.55
AlexNet	25,263,619	95.30
VGG13	35,115,715	95.30
ResNet18	11,255,055	98.60
ResNet34	21,374,351	98.77
MobileNetV1	3,051,151	98.80
MobileNetV2	2,150,951	98.33
MobileNetV3-Small	3,108,507	98.29
MobileNetV3-Large	5,286,683	98.35
ShuffleNetV1	1,045,115	98.94
ShuffleNetV2	4,153,015	98.77
SqueezeNetV1a	815,055	98.78
SqueezeNetV1b	815,055	87.36
SqueezeNetV1c	1,632,047	78.92
GhostNet	4,077,400	98.70
VGG16 [9]	40,425,411	99.31
ResNet50 [9]	23,622,545	99.83
VGG19 [12]	45,735,107	99.90
Wide ResNet-101[12]	127,024,552	99.94
Efficient B [12]	5,330,571	100.00
Proposed Model	**161,768**	**99.27**

Table 4. Comparison Results of The Proposed Model with Other Architectures on Local Fruit Dataset

Model	Parameters	Accuracy (%)
LeNet	304,521	96.76
AlexNet	24,796,561	97.19
ResNet18	11,196,117	99.49
ResNet34	21,315,413	99.77
MobileNetV1	2,933,845	99.37
MobileNetV2	2,004,461	99.40
MobileNetV3-Small	2,962,473	99.02
MobileNetV3-Large	5,140,649	99.14
ShuffleNetV1	979,337	99.26
ShuffleNetV2	4,036,165	99.56
SqueezeNetV1a	756,117	99.38
SqueezeNetV1b	756,117	93.92
SqueezeNetV1c	1,573,109	92.55
GhostNet	3,931,480	99.21
Proposed Model	**37,050**	**99.57**

Table 5. Efficiency Comparison of the Proposed Model with Other Architectures Tested on Jetson Nano

Model	Parameters	GFLOPs	Model speed (FPS)	Detection system speed (FPS)
LeNet	304,521	0.0063	131.553	48.543
ResNet18	11,196,117	0.1965	12.182	10.410
MobileNetV1	2,933,845	0.0757	29.653	20.948
MobileNetV2	2,004,461	0.0319	14.751	12.118
MobileNetV3-Small	2,962,473	0.0339	8.778	7.784
ShuffleNetV1	979,337	0.0154	18.067	14.327
ShuffleNetV2	4,036,165	0.0549	13.555	11.283
SqueezeNetV1a	756,117	0.0571	28.995	20.481
SqueezeNetV1b	756,117	0.0571	29.050	20.479
SqueezeNetV1c	1,573,109	0.1192	25.964	18.882
GhostNet	3,931,480	0.0218	10.728	9.205
Proposed Model	**37,050**	**0.0030**	100.488	43.691

3. Efficiency of The Detection System

In order to evaluate the model reliability and efficiency of the practical application, the trained model was installed on a detection system to predict the location of a single fruit in a frame. A contour detection process is employed to separate the fruit from the background at the beginning of the system. It applies the HSV (Hue, Saturation, and Value) color filtering process by setting the threshold. This process distinguishes between the object and the background based on color. Then contour detection [31] is applied to separate the object from the background. To enhance the ability, the model applies Canny edge detection [32], which aims to discern the shape feature from the background. The end of the sequential process provides the location and size information of the fruit with a bounding box. This box specifies the upper left and right coordinates obtained from the maximum and minimum locations of the fruit contour region.

The detection results of the proposed system are shown in Figure 3. The green bounding box indicates the location and size of the detected fruit. The fruit prediction label is located at the top right of the bounding box. Figure 3 shows that the designed detection system can accurately recognize local fruits of North Sulawesi while precisely recognizing different fruit locations in a frame. In addition, the designed detection system is able to recognize local fruits of North Sulawesi with different scales and sizes. These abilities describe the advantage of the proposed detection system. This image processing depends on the contour detection process at the beginning of the process. Therefore, the proposed classification model needs important regions as input from the network. If the detection phase mistakenly localizes the fruit object, it will negatively affect the final classification results.

On the other hand, this dependency also has a disadvantage in a multi-object case. Edge detection can only localize a single object. Varied and complex background features are also a weakness of the detection system. Therefore, the proposed detection has limitations that only identify a single object with minimal texture in the background.

The efficiency comparison with popular lightweight architectures is presented in Table 5. The obtained values show that the proposed model obtains the number of parameters by 37,050 and computational complexity by 0.0030, which is less than all competitors. The average speed of the proposed model compared to competing models on a video at 1,000 frames. The speed of the proposed model obtained 100,488 FPS, which is 31 FPS slower than the LeNet model. It also has the same impact on the detection speed integrated with the edge detection method. Thus, an industrial implementation does not ignore the performance of a robot vision system. Therefore, the proposed model has better reliability than its competitors, achieving high accuracy. These results also show that our model has high performance and efficiency in real-case scenarios for single fruit cases. In practice, the single fruit selection system can be implemented on an industrial robot equipped with a conveyor to sort fruits one by one according to their category.

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 12, Issue 2, July 2023

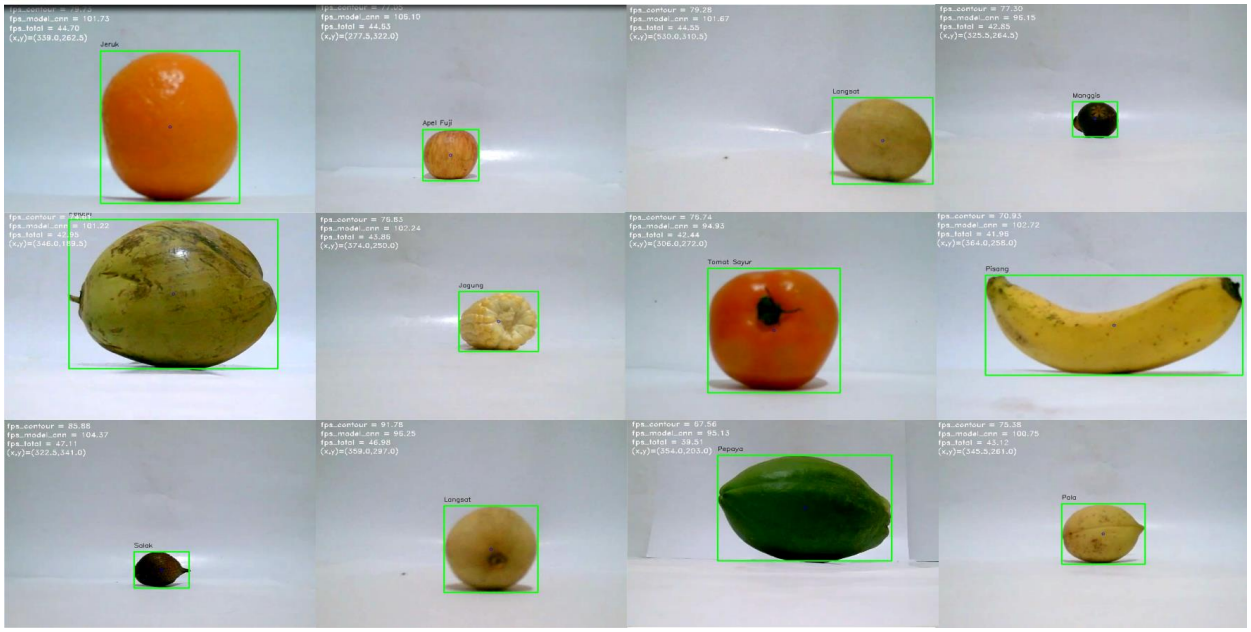

Figure 3. Single Local Fruit Detection Result

CONCLUSION

The single fruit detection system is proposed using an efficient attention module based on deep learning architecture. The architecture of the classification system employs a lightweight backbone and attention modules. The enhancement module based on spatial and channel can capture contextual information that can improve the prediction accuracy without significantly increasing the number of parameters and computation. On the other hand, this study also proposed a North Sulawesi local fruits dataset that consists of 17 categories. As a result, the model achieves high accuracy on both the local fruit and Fruits-360 dataset. This result outperforms the performance of popular mobile architectures. In addition, the proposed model obtains high efficiency and gains fewer parameters and computational than competitors. The classification is integrated with contour detection to build a single fruit detection system that obtained a real-time speed of 44 FPS on a Jetson Nano device. The overall system obtains satisfying performance when implemented in a real-world application. The edge detection could only localize a single object and depended on a non-varying background. Thus, further work is designing a fruit detection that can recognize multiple objects with complex backgrounds in a frame.

REFERENCES

[1] Utara, B. P. S. (2021). *Produksi Buah–Buahan dan Sayuran Tahunan Menurut Jenis Tanaman (Kuintal), 2019-2021*. https://sulut.bps.go.id/statictable/2022/06/24/200/produksi-buah-buahan-dan-sayuran-tahunan-menurut-jenis-tanaman-kuintal-2019-2021.html.

[2] Alem, A., & Kumar, S. (2022). Deep Learning Models Performance Evaluations for Remote Sensed Image Classification. *IEEE Access*, *10*, 111784–111793.

[3] Falahkhi, B., Achmal, E. F., Rizaldi, M., Rizki, R., & Yudistira, N. (2018). Comparison of AlexNet and ResNet Models in Flower Image Classification Utilizing Transfer Learning. *Jurnal Ilmu Komputer Dan Agri-Informatika*, *9*(Kew 2016), 70–78.

[4] Putro, M. D., Nguyen, D.-L., & Jo, K.-H. (2022). A Fast CPU Real-Time Facial Expression Detector Using Sequential Attention Network for Human–Robot Interaction. *IEEE Transactions on Industrial Informatics*, *18*(11), 7665–7674.

[5] Miranda, N. D., Novamizanti, L., & Rizal, S. (2020). Convolutional Neural Network Pada Klasifikasi Sidik Jari Menggunakan Resnet-50. *Jurnal Teknik Informatika (Jutif)*, *1*(2), 61–68.

[6] Hu, H.-C., Chang, S.-Y., Wang, C.-H., Li, K.-J., Cho, H.-Y., Chen, Y.-T., Lu, C.-J., Tsai, T.-P., & Lee, O. K.-S. (2021). Deep Learning Application for Vocal Fold Disease Prediction Through Voice Recognition: Preliminary Development Study. *Journal of Medical Internet Research*, *23*(6), e25247.

[7] Putro, M. D., Kurnianggoro, L., & Jo, K.-H. (2021). High Performance and Efficient Real-Time Face Detector on Central Processing Unit Based on Convolutional Neural Network. *IEEE Transactions on Industrial*

Informatics, *17*(7), 4449–4457.

[8] Yu, H., Xu, Z., Zheng, K., Hong, D., Yang, H., & Song, M. (2022). MSTNet: A Multilevel Spectral–Spatial Transformer Network for Hyperspectral Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 1–13.

[9] Dandekar, M., Punn, N. S., Sonbhadra, S. K., Agarwal, S., & Kiran, R. U. (2021). Fruit classification using deep feature maps in the presence of deceptive similar classes. *Proceedings of the International Joint Conference on Neural Networks*, *2021-July*, 0–5.

[10] Himabindu, D. D., & Praveen Kumar, S. (2020). A comprehensive analytic scheme for classification of novel models. *Proceedings of the 3rd International Conference on Intelligent Sustainable Systems, ICISS 2020*, 564–569.

[11] Albardi, F., Kabir, H. M. Di., Bhuiyan, M. M. I., Kebria, P. M., Khosravi, A., & Nahavandi, S. (2021). A Comprehensive Study on Torchvision Pre-trained Models for Fine-grained Inter-species Classification. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, 2767–2774.

[12] Kabir, H. M. D., Abdar, M., Khosravi, A., Jalali, S. M. J., Atiya, A. F., Nahavandi, S., & Srinivasan, D. (2022). SpinalNet: Deep Neural Network With Gradual Input. *IEEE Transactions on Artificial Intelligence*.

[13] Rathnayake, N., Rathnayake, U., Dang, T. L., & Hoshino, Y. (2022). An Efficient Automatic Fruit-360 Image Identification and Recognition Using a Novel Modified Cascaded-ANFIS Algorithm. *Sensors*, *22*(12).

[14] Pande, A., Munot, M., Sreeemathy, R., & Bakare, R. V. (2019). An Efficient Approach to Fruit Classification and Grading using Deep Convolutional Neural Network. *2019 IEEE 5th International Conference for Convergence in Technology, I2CT 2019*, 2–8.

[15] Srivastava, H., & Sarawadekar, K. (2020). A Depthwise Separable Convolution Architecture for CNN Accelerator. *2020 IEEE Applied Signal Processing Conference (ASPCON)*, 1–5.

[16] Shadin, N. S., Sanjana, S., & Lisa, N. J. (2021). COVID-19 Diagnosis from Chest X-ray Images Using Convolutional Neural Network(CNN) and InceptionV3. *2021 International Conference on Information Technology (ICIT)*, 799–804.

[17] Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. 3rd International Conference on Learning Representations (ICLR 2015), 1–14.

[18] Zhuge, M., Fan, D.-P., Liu, N., Zhang, D., Xu, D., & Shao, L. (2023). Salient Object Detection via Integrity Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(3), 3738–3752.

[19] Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size*. 1–13.

[20] Xiong, B., Fan, S., He, X., Xu, T., & Chang, Y. (2022). Small Logarithmic Floating-Point Multiplier Based on FPGA and Its Application on MobileNet. *IEEE Transactions on Circuits and Systems II: Express Briefs*, *69*(12), 5119–5123.

[21] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 4510–4520.

[22] Andrew, H., Mark, S., Grace, C., Liang-Chieh, C., Bo, C., Mingxing, T., Weijun, W., Yukun, Z., Ruoming, P., & Vijay, V. (2019). Searching for mobilenetv3. *Proceedings of the IEEE International Conference on Computer Vision*, 1314–1324.

[23] Zhang, X., Zhou, X., Lin, M., & Sun, J. (2018). ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 6848–6856.

[24] Ferrari, V., Sminchisescu, C., Hebert, M., & Weiss, Y. (2018). ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11218*, vii–ix.

[25] Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., & Xu, C. (2020). GhostNet: More features from cheap operations. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1577–1586.

[26] Hu, J., Shen, L., Albanie, S., Sun, G., & Wu, E. (2020). *Squeeze-and-Excitation Networks*. *42*(8), 2011–2023.

[27] Hou, Q., Zhou, D., & Feng, J. (2021). Coordinate attention for efficient mobile network design. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 13708–13717.

[28] Woo, S., Park, J., Lee, J. Y., & Kweon, I. S. (2018). CBAM: Convolutional block attention module. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11211 LNCS*, 3–19.

[29] Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., & Hu, Q. (2020). ECA-Net: Efficient channel attention for deep convolutional neural networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 11531–11539.

[30] Mureşan, H., & Oltean, M. (2018). Fruit recognition from images using deep learning. *Acta Universitatis Sapientiae, Informatica*, *10*(1), 26–42.

[31] Suzuki, S., & be, K. (1985). Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, *30*(1), 32–46.

[32] Canny, J. (1986). A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *PAMI-8*(6), 679–698.