



Implementation of Image Segmentation for Pakcoy Plant Using Tensorflow Framework

Erika Lety Istikhomah Puspita Sari*¹, Mas Aly Afandi²,
Eka Setia Nugraha³, I Ketut Agung Enriko⁴

¹ Indonesia Telecommunication and Digital Research Institute, ^{2,3,4} Institut Teknologi Telkom Purwokerto
erkalety@gmail.com

Abstract—Pakcoy is a type of vegetable that is very difficult to distinguish to determine the maturity level of the greenish color intensity for determination of harvest readiness in the Pakcoy plant. There needs to be a system reliable and smart to help farmers in the harvest process. On research This camera is used to take images of the Pakcoy plant. This research aims to test the accuracy of the detection object with two categories namely Pakcoy plants "Ready to Harvest" and "Not Ready to Harvest" using the Tensorflow Lite framework with the EfficientDet Lite 2 architectural model, Pakcoy plant detection was tested via an Android device and website online real time to assess the performance of the detection model. The test results show the average detection accuracy of Pakcoy Ready to Harvest vegetables touched the number 98.02% and 99.00% for detection of Pakcoy vegetables not ready for harvest, this indicates that the detection model is working reasonably well with the device android as well as website.

Keywords—Harvest Readiness Detector, Pakcoy, Tensorflow Lite, EfficientDet Lite 2, Deep Learning

I. INTRODUCTION

Indonesia is an agrarian country with part of the population working in the agriculture and plantation sector. The agricultural sector is a sector which is strategic and plays an important role in the national economy and community survival, especially in terms of contribution to GDP, provider of employment and food supply in the country. Until now and perhaps for the next few decades or a few hundred years, agriculture will still be a source of food for the population [1]. Indonesia has started emphasizing the use of technology in the field agriculture to improve the quality and quantity of agricultural commodities. Smart farming is also called digital farming, agriculture 4.0 or Data-driven agriculture is emerging as a revolutionary paradigm shift of the sector agriculture, which aims to optimize farm management and increase in agricultural efficiency [2]. Smart farming is different from precision farming ancestry, both in terms of logic and scope.

Smart agriculture 4.0 applies soil and weather detector technology, so that it can monitor land conditions in real time, and is expected to be able to provide more measurable and precise agricultural data. In addition with the digitalization of agriculture can help simplify and expedite the entire agricultural process from production to marketing. One of technology that can be applied to increase the quality and quantity is factory plant. Factory plant is an automation concept so that agriculture can run without human intervention. Robotics and intelligence technology artificial development capable of driving increased efficiency in technology factory plants [1].

Image processing is a branch of technology in a factory plant determinant of height increase. The process of sorting between plants that are ready for harvested and crops that are not yet ready for harvest are a particular focus on discussion of image processing. A good image processing system is able to 2 improve the accuracy of artificial intelligence in determining harvest readiness. One focus of image processing to improve this is image segmentation. Image segmentation has the meaning of dividing something image into homogeneous areas based on similarity criteria between a pixel and other pixels, then the result of the process This segmentation will be used for further high-level processing which can be carried out on an image, for example the image classification process and process object identification [2].

Pakcoy (*Brassica rapa* L.) is a very popular type of leaf vegetable plant important in Indonesia, because it has a high economic value. Plenty of pakcoy It is in great demand as a vegetable because of its high nutritional content and delicious taste. The nutritional content contained in 100 g. Pakcoy has 2.39 mg of protein, 0.39 mg fat, 4.09 mg carbohydrates, 220 mg calcium, phosphorus, 38 mg iron and vitamin C 102 mg. With these ingredients, Pakcoy is efficacious for preventing cancer, cataracts, stroke, congenital defects, hypertension and heart disease [12].

Pakcoy plant cultivation can be done organically or inorganically, but to get high quality Pakcoy plants, of course organic cultivation must be done immediately, given the more residue resulting from the use of inorganic fertilizers and inorganic pesticides. Thing Such can affect the yield of crop production and speed up physical and biological damage to the soil. Pakcoy cultivation organically with using vermicompost can be used as a good alternative considering the potential of earthworm manure as organic material has nutrient content quite high, namely: C 20.20%, N 1.58%, C/N 13, P 70.30 mg/100g, K 21.80 mg/ 100g, Ca 34.99 mg/100g, Mg 21.43 mg/100g [12].

In this research, a system for detecting the readiness of harvesting vegetables was created Pakcoy based on the color image of Pakcoy vegetables using deep learning. The training process for the Pakcoy vegetable harvest readiness detection model was carried out using the Tensorflow Lite framework. The architectural model used in this research is EfficientDet Lite2. By implementing deep learning 3 in the detection of vegetable harvest readiness, the process of detecting Pakcoy harvest readiness will be faster and more efficient.

II. RELATED WORKS

A. Development of Agricultural Technology in Indonesia

In the midst of rapid technological developments, PT Bank Negara Indonesia Tbk (BBNI) encourages agriculture to implement smart farming, through Movement Welcoming Agriculture 4.0. This is expected to prepare farmers for planting period, with the help of the use of digital technology. Through smart farming, farmers' cultivation process is more effective and will produce more production increase, and ultimately increase the welfare of society. Smart farming 4.0 applies soil and weather sensor technology, so it can monitor land conditions in real time, and are expected to be able to provide data agriculture that is more measurable and precise [6].

The implementation of smart farming 4.0 is not just about implementation technology alone, but the main key is measurable data. This data that later it will be read and used as a parameter for agricultural actors to get references and recommendations in agricultural practices, so expected to be more efficient and effective. Types of Smart Agricultural Technology (Smart farming technology / SFT) is divided into three main categories which, like mentioned above, includes precision farming cyclic systems:

1. Data acquisition technologies: this category includes all surveying, mapping, navigation, and sensing.
2. Technology analysis and evaluation of data: these technologies range from models simple computer-based decisions to complex farm management complex and information systems include many different variables.
3. Precision application technologies: this category contains all application technologies, with focus on variable rate application and guidance technology

There is a range of technologies that can be classified under each category Smart farming technology. Data Acquisition Technology Smart farming technology to record and map the fields and characteristics of the crops divided into category below:

1. Global navigation satellite system technology (actually this technology records actual position which can be used for different purposes like guides, mapping, etc.)
2. Mapping technology
3. Acquisition of environmental property data
4. Engine and its properties Global navigation satellite system (GNSS) technology

B. Image

Maturity of plants is usually determined by several parameters, including parameters of size, weight, color characteristics, fragrance of these plants and others. Parameters of plant maturity in terms of the color of the plant's skin itself is one of the important factors in the identification of maturity plant. In general, the classification of plant maturity is done by: manual, namely using the sense of sight to distinguish the ripeness However, this method has many weaknesses such as assessment by humans who are subjective and inconsistent [2].

Technological developments allow the identification of a maturity plants using image processing methods. Image is a representation (image), likeness, or imitation of an object. The

image is divided into 2, namely the image that are analog and there are images that are digital. Analog image is an image continuous, such as images on television monitors, X-ray photos, and others. Meanwhile, digital images are images that can be processed by a computer [7]. Image as output from a data recording system can be optical in the form of photos, analog in the form of video signals such as images on television monitors, and which can be directly stored on a magnetic tape.

C. Segmentation

Image segmentation is a process by which digital images are partitioned into various subgroups (pixels) called image objects, which can be reduced image complexity, so that it can analyze images more simply [9]. The process of image segmentation is experimental, subjective, and depends on goal to be achieved. Segmentation in Image Processing is being used 11 in the medical industry for efficient and faster diagnosis, detecting disease, tumors, and cell and tissue patterns from the resulting medical images from radiography, MRI, endoscopy, thermography, ultrasonography, etc. Satellite images are processed to identify various patterns, objects, contours geographic, land information, which can later be used for agriculture, mining, geo-sensing. Image segmentation has very wide application areas major in robotics, such as RPA, self-driving cars. Implementation image segmentation in python, matlab and other languages is widely used extensively for the process, the Image Segmentation technique:

1. Threshold Method
2. Edge Based Segmentation
3. Region Based Segmentation
4. Clustering Based Segmentation
5. Watershed Based Method
6. Artificial Neural Network Based Segmentation

Image processing in general has been implemented in various ways programming languages either java, matplotlib, C++.

By the nature of modularity, implementation and versatile use in data science, machine learning, and deep learning stacks, Python also has a powerful library to implement various techniques. used in Image Segmentation. Python libraries like scikit image, OpenCV, Mahotas, Pillow, matplotlib, SimpleITK etc. Famously used to implement general image processing and segmentation image in particular.

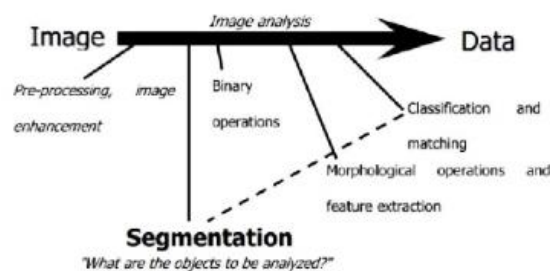


Fig. 1. Illustration of Image Segmentation Analysis

D. Confusion Matrix

Maturity of plants is usually determined by several parameters, including parameters of size, weight, color characteristics, fragrance of these plants and others. Parameters of plant maturity in terms of the color of the plant's skin itself is one of the important factors in the identification of maturity plant. In general, the classification of plant maturity is done by: manual, namely using the sense of sight to distinguish the ripeness. However, this method has many weaknesses such as assessment by humans who are subjective and inconsistent [2].

E. Maintaining the Integrity of the Specifications

Maturity of plants is usually determined by several parameters, including parameters of size, weight, color characteristics, fragrance of these plants and others. Parameters of plant maturity in terms of the color of the plant's skin itself is one of the important factors in the identification of maturity plant. In general, the classification of plant maturity is done by: manual, namely using the sense of sight to distinguish the ripeness. However, this method has many weaknesses such as assessment by humans who are subjective and inconsistent [2].

III. RESEARCH METHOD

A. Research Method

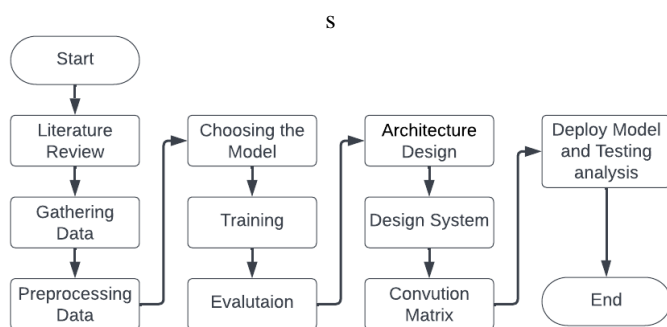


Fig. 2. Reseach Flow



Fig. 3. Seven steps of machine learning

In designing a detection system for the readiness of the Pakcoy plant to harvest Python and Java programming languages are used, data is trained using Tensorflow Lite framework with the EfficientDet lite2 model architecture. The design model used in this study was run on google colaboratory interpreter, roboflow as a database in data processing augmentation, android visual studio as a deployment to the application. All applications used in this design is based on an open-source web application can be used for codes, equations, visualizations and narrative text. This research implemented according to the 7 steps of Machine Learning flow shown in Figure 3.

B. Gathering and Preparing Data

This stage is the data collection stage, this stage is very important because the quality of the data will directly affect the model object detection to be created. In this study there are 325 image data (real time) consists of ready-to-harvest Pak-coy vegetables and un-prepared Pak-coy vegetables ready to harvest obtained from photographing directly.

This stage is the data preparation stage, after the image data is processed collect it then do the labeling on the image using the same tools exist on the roboflow website and then export the dataset in the form of an image which has been labeled earlier by applying data preprocessing and augmentations by dividing the dataset by 60% for training data, 20% for data validation and 20% for test data.

C. Choosing a model

The next step is the selection of the model, this study uses EfficientDet Lite 2 model architecture, this model is a derivative of the model EfficientDet which consistently achieves better accuracy and efficiency from previous models (Tan et al., 2020, p.10788). EfficientDet has eight models namely D0-D7 which are the accuracy and time complexity of the model increases with model size [27]. Model size comparison can be seen in Figure 4.

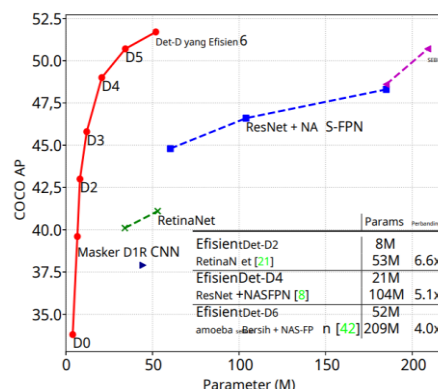


Fig. 4. Model Size Comparison[27]

Furthermore, Figure 4 shows an overview of the EfficientDet model architecture.

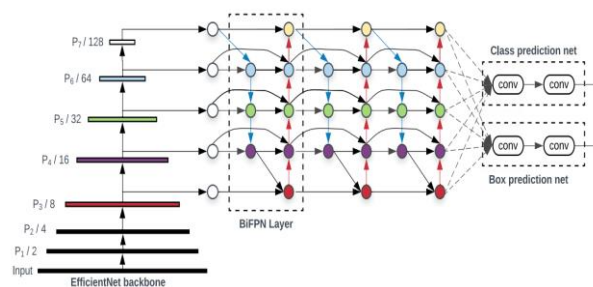


Fig. 5. EfficientDet Model Architecture [27]

The EfficientDet model uses the EfficientNet model as the network backbone, BiFPN as feature network, and class prediction network or shared box. The BiFPN layer and the class/box mesh layer were repeated several times based on different resource constraints [27].

The EfficientDet model uses the EfficientNet model as the network backbone, BiFPN as feature network, and class prediction network or shared box. The BiFPN layer and the

class or box mesh layer are repeated several times based on different resource limits [27]

D. Training

Then enter the training process, training is conducted in Colab Notebooks using the Tensorflow Lite framework, the batch size is set to 8 and uses 100 epochs for the training process, from this configuration the processing per epoch takes an average of 24 seconds, so the entire model training process takes approximately 40 minutes.

E. Evaluation

After the training is complete, then an evaluation is carried out using the data pre-split validation to see if the model has good, at this evaluation stage a safe object detection model will be tested 25 using image data that has never been seen before, at this stage we can see the performance model before being directly involved in the real world.

F. Hiperparameter Tuning

Hyperparameter tuning is done to further improve accuracy of the model after training by setting some parameters, on This research did not reset the parameters because of the accuracy obtained after the training felt good enough.

G. Prediction

After going through various processes, this stage is the core of all the process through which the value of machine learning is realized for detects Pakcoy plants ready for harvest and Pakcoy not ready to harvest, the model that have finished training then deploy to android mobile devices for free actually see the accuracy of the model's predictions when applied to devices and agar can be used by the community.

H. Roboflow

The database used for image processing communication with The smartphone on this system is Roboflow. The concept used in the database it is Real-time. The Roboflow platform makes it easy to create systems this final project in terms of data storage. Use this Roboflow too later it can be integrated directly with TF Lite App Inventor.

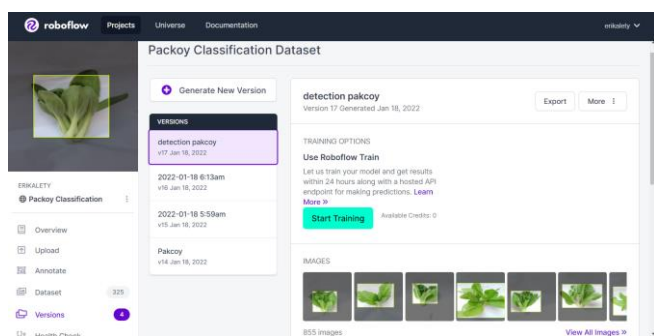


Fig. 6. Roboflow Project Creation View

This is the initial view of Roboflow. To carry out the process Database creation can be done by clicking start or if Roboflow is already logged in, you can click on the project section.

I. Design Architecture

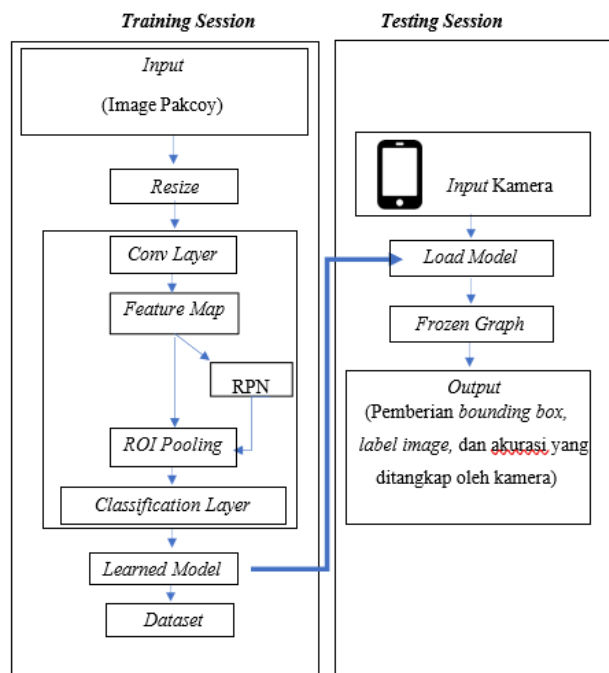


Fig. 7. General System Architecture

General Architectural Information (Figure 6) is:

1. Training Sessions

- Inputs: At this stage the input is given an image in the form of a Pakcoy plant.
- Resize: In the early stages of the preprocessing process, namely resizing for resize the image by reducing the size of the image automatically horizontally and or vertically, the image will be resized so that the file size does not change over 200 KB and image dimensions not larger than 720 x 1280.
- Convolutional Layer: The layer consisting of neurons arranged in such a way that forms a two-dimensional filter with length and height (pixels).
- Featured Maps: Feature Map is a map created by the convolutional layer which contains information about the vector representation of the image captured, the convolution layer will generate two feature maps the same, the first feature map will be processed at RPN to produce region proposals and other feature maps will be sent directly to pooling layer.
- RPN (Region Proposal Network): RPN is a module that works to process features map that has been made on the convolution layer to predict the part that is considered as an object and perform bounding predictions box of the object,
- Pooling ROI: ROI is the layer responsible for equalizing the size of the feature map and proposal region that has been processed by the RPN as well as sending feature map information and proposals for classification on the classification layer.
- Classification Layer: 28 The classification layer functions to group objects that have been detected on

the RPN and labeled the object as well as provide a bounding box on the object.

- h) Learned Model: Learned model is a model that has been learned by the system.
- i) Datasets: Collection of other learned models not used in testing which contains dataset weight information.

2. Testing Sessions

- a) Inputs: At this stage, the system is given input data in the form of an image of the Pakcoy plant.
- b) Load Models: Load model is the stage where the system will reload the model that has been saved during the training session.
- c) Frozen Graph: At this stage the input data received through the camera will be processed on the graph stored in the frozen model to do identification and giving of bounding box based on the weight that has been stored in the model that has been trained.
- d) Outputs: The output is the result of the identification that has been carried out, where Classified objects are given bounding boxes and labels.

J. System Design

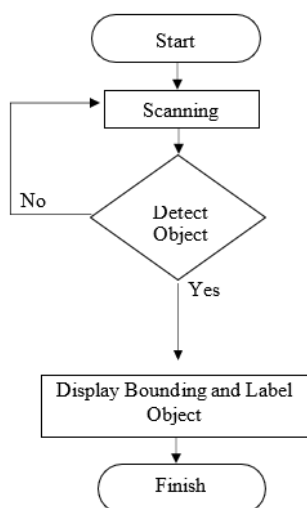


Fig. 8. System Flowchart

The flowchart in Figure 3.8 explains in detail the object detection process. First must execute the program, then the program will do initialization or checking of supporting device components 32 system and program code. If there is an error, the program will be terminated. but if no errors are detected then the program will activate the device camera. The camera device that has been activated then takes video in real time and then the system reads every frame on the video. The frame is then processed using the SSD method, if there is a human object, then the detection results will be shown to the display and then stored into the storage media. If not there detected human object, then the program will process the next frame.

K. Confusion Matrix

Confusion matrix is a measurement that can be used for calculate the performance or level of truth in the classification process. Confusion table matrix is shown in Table I below.

TABLE I. TABLE OF CONFUSION MATRIX

N = 200		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	True (1)	TP	FP
	False (0)	FN	TN

Information:

1. TP (True Positive) is the amount of data that has predictions ripe then read by the system as ripe.
2. FP (False Positive) is the amount of data that has predictions mature then read by the system as immature.
3. FN (False negative) is the amount of data that has predictions immature then read by the system as immature.
4. TN (True Negative) is the amount of data that has predictions immature then read by the system as ripe.

IV. RESULT AND ANALYSIS

A. Research parameter

At this stage, the process of detecting Pakcoy plant objects uses Deep Learning that will be implemented into the system. Object detection system built using the Python programming language. Image classification crops are divided into 2 categories, namely "Ready to Harvest" and "Not Ready to Harvest".

B. Preprocessing Result

At this stage labeling of each object will be trained on system, labeling is done using labelImg which will be where each object is in the image will be given a bounding box and also each class object. This labeling process generates an xml file for each object on labeled image, an example of an xml file for labeling can be seen in Figure 11.

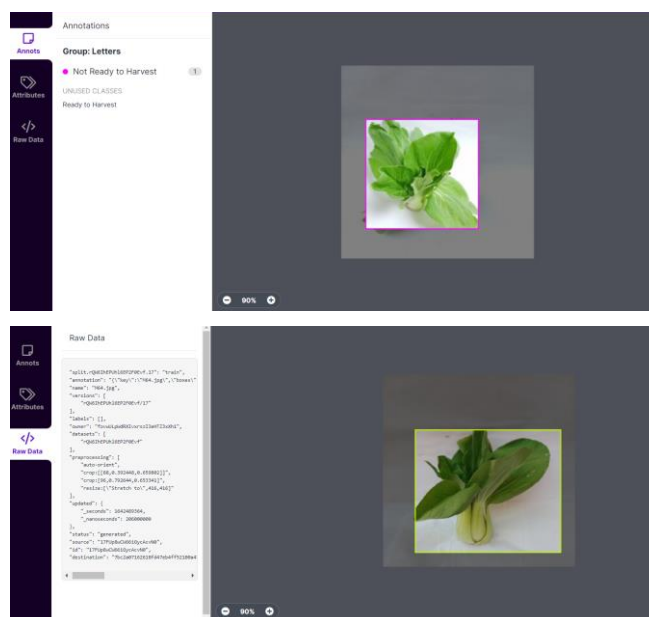


Fig. 9. Labeling Image

Figure 9 is an example of an xml file created in the process labeling where this file contains information such as the width, length, and class of the object that we label, each object has different information, if there are two objects in one image then there will likely be two objects with different information to be put in the xml file.

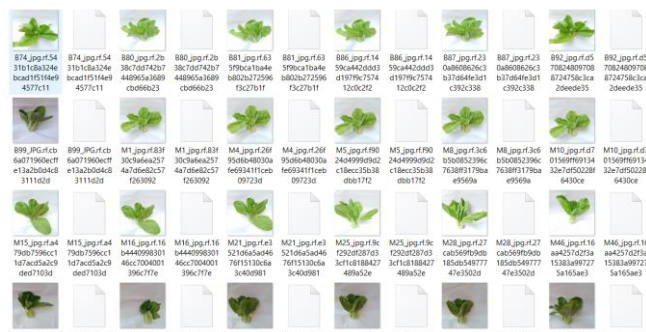


Fig. 10. Collection of images and XML files from Image

Figure 10 is a collection of images used for processing training along with the xml file of the labeling results. This file will later be created into a file "train.record" which will be entered into the system to be made system training data. The training process after each image is labeled and have an xml file then this xml file will be generated into two csv files namely "train_labels.csv" and "test_label.csv" which contain information from all images which will be used as training and validation data, an example of this csv file can be seen in Figure 11.

Filename	width	height	class	xmin	ymin	xmax	ymax
B95_jpg.rf.ec638dfdf7968f3c7cc0978a3a1b.jpg	416	416	Not Ready to Harvest	15	109	402	379
B29_jpg.rf.ecb25ba9455a7d93ce5ed423957aa3b.jpg	416	416	Not Ready to Harvest	137	148	364	320
M27_jpg.rf.ee0b01559556bc29f649a616ec49bf8d.jpg	416	416	Ready to Harvest	0	58	405	334
B90_jpg.rf.ed903a2eed35ef6772af7addebf799.jpg	416	416	Not Ready to Harvest	0	91	413	393
B90_jpg.rf.f0b97fd5570baa17763742ed70e148aa.jpg	416	416	Not Ready to Harvest	27	120	415	415
B99_jpg.rf.f10c015827ba8bf6c4e7baa3bc720.jpg	416	416	Not Ready to Harvest	18	54	415	340
B37_jpg.rf.e354e1309167a7b1a1a2719e690893.jpg	416	416	Not Ready to Harvest	49	139	289	298
B94_jpg.rf.ed502caf1a74b709ec406179541.jpg	416	416	Not Ready to Harvest	25	119	392	397
B60_jpg.rf.ee843139e1b1e68b8989407b648028.jpg	416	416	Not Ready to Harvest	17	121	362	346
M27_jpg.rf.eef922121a47401a04efcd9a55c35.jpg	416	416	Ready to Harvest	0	52	407	330
B91_jpg.rf.f2a5716230c7f71926aff636f99707.jpg	416	416	Not Ready to Harvest	15	54	415	363
B41_jpg.rf.f0e2ab2290b0610ee718a11c2288b2d0.jpg	416	416	Not Ready to Harvest	19	77	350	289
B74_jpg.rf.f0e2ab2290b0610ee718a11c2288b2d0.jpg	416	416	Not Ready to Harvest	1	107	415	386
B66_jpg.rf.f2aaaa6484a9ff4d7e710ec4ed7993.jpg	416	416	Not Ready to Harvest	16	97	398	323

Fig. 11. Labeling Image

C. Training and Learning Rate Variation

Learning rate in a model is used to calculate the correction value from the weight when training data is carried out. In script code, normal learning rate written with lr and followed by a value called a constant. In other words, this learning rate shows how many times k experiences change. These constants will be updated each time the batch is run. on analysis This learning rate variation aims to determine the accuracy and loss values obtained after training, and data validation. On learning variations This rate uses a learning rate of 0.01. Learning rate, is wrong one training parameter to calculate the weight correction value at runtime training. This learning rate value is in the range of zero (0) to (1). The more the greater the learning rate, the faster the training process will run.

The results of the accuracy in training and validation will be comparison and proves that the learning rate affects the magnitude of the accuracy value of a model. The following is a learning variation experiment rate performed on the image data of the Pakcoy plant to see the accuracy and the resulting loss as well as the graphic pattern.

Here is an experiment using a learning rate of 0.001 with epoch 50 produces a model that is appropriate fitting by looking at the accuracy and loss the function.

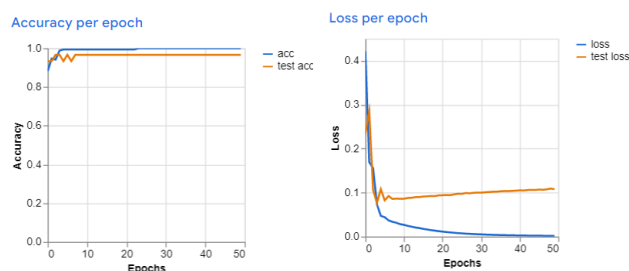


Fig. 12. display of variations in learning rate 0.001

The second experiment uses a learning rate of 0.1 with an epoch of 15 produce an overfitting model by looking at its accuracy and loss function.

The third experiment uses a learning rate of 0.01 with an epoch of 50 produce a good-fitting model by looking at the accuracy and loss function- 40 his. Shows the performance results of the model that has been trained. Model shows the results of high accuracy and low loss.

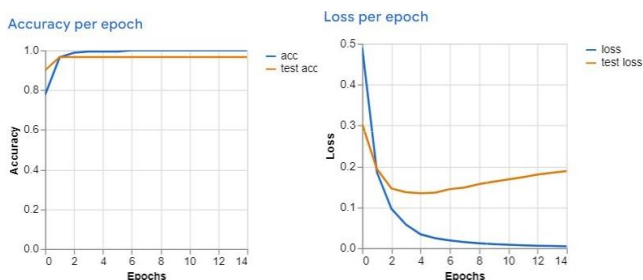


Fig. 13. display of variations in learning rate 0.01

D. Evaluation Model

In the modeling process, using the Teachable platform Machine from Google with a transfer learning algorithm so it can produce high accuracy and low loss even though using very little data. Teachable Machine uses Tensorflow.js, which is one of the popular libraries for machine learning in Javascript, to train and run an already created model in a web browser. Model uses a technique called transfer learning, and there is a pre trained neural network model.

More specifically, the image model learns from the MobileNet model that has been pre-trained, whereas voice models are built on Speech Recognition. MobileNet is an efficient network architecture and a set of two hyperparameters to build a very small low latency model that 41 can be easily adapted to the design requirements for mobile applications and embedded systems. MobileNet is trained using popular image datasets ImageNet. In Figure 13 is the performance result of the existing model trained. The model shows high accuracy and low loss. On Figure 14 is an illustration of the MobileNet-V2 architecture used in products to predict pakcoy readiness.

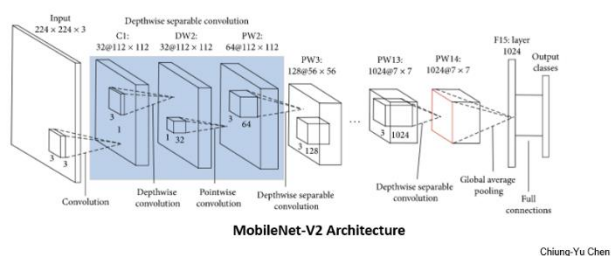


Fig. 14. MobileNet-V2 network architecture [24]

E. Model Deployment in Android Application platform

After going through the training process and parameter resetting, the detection model that produces the most accurate detection performance is taken, this model is then implemented in an Android application so that it can be tested on mobile devices to assess the performance of the model that has been trained directly. The following is a screenshot of the model that has been applied to the Android application.

The main page is the front view of the application, where the app will ask for access permission to use the camera, and once granted app access will open the camera and run object detection of objects captured by the camera.

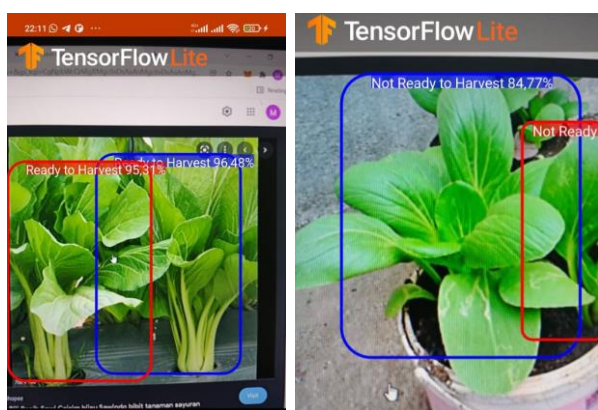


Fig. 15. display of website for harvest detection pakcoy

Show process is the page where the process is being executed or detected by the system, the object being detected can be seen in the left corner from android screen.

F. System Testing

At this stage, data and system testing will be carried out determine the ability of the system built in detecting readiness objects Pakcoy harvest. Data testing was carried out on 200 Pakcoy plants. An example of the results of training on object detection can be seen in the attachment. Tests were also carried out at various light intensities and also distances camera from the object that we can see in Figure 4.19. System testing done to find out whether the system built can run properly well and the detection results of the URL input provided by the user match with expectations. Testing is carried out in the training and testing section of the website. Test criteria as follows:

1. Users can input URLs.
2. The system can directly monitor content and pre-processing so as to obtain data to predict.

3. The system can perform feature extraction on the input data entered so that it can be predicted by the system.

4. The system can detect input data by using one Machine Learning models that have been trained and tested so that derive content detection results from the data entered.

5. The system can display the detection results according to the input user.

6. System testing is carried out with a Windows 11 operating system laptop with 4GB of RAM.

After going through the training process and resetting the parameters, then in take the detection model that produces the most accurate detection performance, This model is then implemented in an Android application along with the Facebook website can be tested on mobile devices to assess the performance of the model that has been in live training, Some screenshots of the model that has been in apply to android applications can be seen in Figures 19 and 20.

Here are the results of 200 experimental images. From the results of the data provide good result. The system is able to detect the readiness to harvest Pakcoy vegetables, either which are ready to harvest and those that are not ready to harvest with an accuracy level of 100% on the android system, while the accuracy value on the website is 1.00%. Score Accuracy was obtained for collecting the results from the detection data of 200 images worn.

TABLE II. PAKCOY VEGETABLE IMAGE EXPERIMENT TEST

No	Name	A	b	Prediction	No	Name	a	b	Prediction
1	Image-1	0.9 8	7 2	TN	51	Image-51	0.87 8	7 3	TN
2	Image-2	0.9 2	1 4	TN	52	Image-52	0.34 7	4 4	TN
3	Image-3	0.1 1	5 8	TN	53	Image-53	0.69 1	0 0	FN
4	Image-4	0.9 1	4 5	TN	54	Image-54	1.00 5	8 8	TN
5	Image-5	1.0 5	6 5	TN	55	Image-55	0.61 8	2 2	TN
6	Image-6	1.0 5	5 5	TN	56	Image-56	0.30 1	4 4	TN
7	Image-7	0.7 2	7 0	FN	57	Image-57	0.55 5	8 8	TN
8	Image-8	0.6 2	7 0	FN	58	Image-58	0.93 5	4 4	TN
9	Image-9	0.8 9	9 2	TN	59	Image-59	0.44 4	0 0	TN
10	Image-10	0.2 9	4 0	TN	60	Image-60	0.57 4	0 0	TN
11	Image-11	0.6 7	8 5	TN	61	Image-61	0.44 1	5 5	TN
12	Image-12	0.3 3	9 8	TN	62	Image-62	0.76 6	3 3	TN
13	Image-13	0.6 7	7 7	TN	63	Image-63	0.76 6	3 3	TN
14	Image-14	0.9 2	2 5	TN	64	Image-64	0.73 7	5 5	TN
15	Image-15	0.3 2	4 6	TN	65	Image-65	0.69 1	0 0	TN
16	Image-16	1.0 0	6 9	TN	66	Image-66	0.91 7	1 1	TN
17	Image-17	0.9 8	7 1	TN	67	Image-67	0.73 6	8 8	TN
18	Image-18	0.8 1	7 5	TN	68	Image-68	0.73 6	7 7	TN
19	Image-19	0.4 2	2 0	TN	69	Image-69	0.83 6	6 6	TN
20	Image-20	1.0 0	1 9	TN	70	Image-70	0.91 5	6 6	TN
21	Image-21	0.6 3	4 9	TN	71	Image-71	0.98 7	7 7	TN
22	Image-22	1.0 0	8 0	TN	72	Image-72	0.79 3	3 3	TN
23	Image-23	0.9 2	9 6	TN	73	Image-73	0.35 7	9 9	TN
24	Image-24	0.6 6	2 7	TN	74	Image-74	0.94 5	7 5	TN
25	Image-25	0.5 5	1 1	FN	75	Image-75	1.00 7	4 4	TN

26	Image-26	0.52	30	TN
27	Image-27	0.66	24	TN
28	Image-28	0.24	95	TN
29	Image-29	1.00	95	TN
30	Image-30	0.49	16	FN
31	Image-31	0.74	78	TN
32	Image-32	0.57	11	TN
33	Image-33	1.00	72	TN
34	Image-34	0.71	38	TN
35	Image-35	0.83	72	TN
36	Image-36	0.98	71	FN
37	Image-37	0.38	99	FN
38	Image-38	0.75	34	TN
39	Image-39	0.84	33	TN
40	Image-40	0.37	23	FN
41	Image-41	0.17	34	FN
42	Image-42	1.00	97	FN
43	Image-43	0.21	76	FN
44	Image-44	0.19	45	TN
45	Image-45	0.19	44	TN
46	Image-46	0.64	23	TN
47	Image-47	0.53	32	FN
48	Image-48	0.75	35	TN
49	Image-49	0.98	50	TN
50	Image-50	0.98	49	TN

76	Image-76	0.94	97	TN
77	Image-77	0.79	41	TN
78	Image-78	0.51	77	FN
79	Image-79	0.91	34	TN
80	Image-80	0.85	22	TN
81	Image-81	0.59	44	TN
82	Image-82	0.61	55	TN
83	Image-83	1.00	84	TN
84	Image-84	1.00	83	TN
85	Image-85	0.67	11	TN
86	Image-86	1.00	42	TN
87	Image-87	0.66	65	FN
88	Image-88	0.43	12	TN
89	Image-89	1.00	94	TN
90	Image-90	0.99	96	TN
91	Image-91	0.95	75	TN
92	Image-92	0.82	99	TN
93	Image-93	0.31	95	TN
94	Image-94	1.00	89	TN
95	Image-95	0.24	28	TN
96	Image-96	0.99	90	TN
97	Image-97	0.92	76	TN
98	Image-98	1.00	98	TN
99	Image-99	0.31	64	TN
100	Image-100	0.69	65	TN

V. CONCLUSION

After carrying out the stages in implementation and testing, then Some conclusions will be obtained in this study, among others that is The EfficientDet-Lite 2 model is capable of detecting plant objects Pakcoy is ready for harvest and not ready for harvest which has a resemblance to good accuracy of 97.98%. Based on the experimental results for the selection of parameters used, learning rate 0.001 is a good value in object detection at the training stage. In the process of testing the system can detect objects not only in the form of pictures of the Pakcoy plant but also the original Pakcoy plant captured from android camera as well as website features. Object detection failure in test data is caused by camera distance and light intensity so that the camera cannot capture objects directly either because the object looks small or it doesn't look clear so system cannot be recognized.

ACKNOWLEDGMENT (Heading 5)

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

REFERENCES

- [1] S. Nagano, S. Moriyuki, K. Wakamori, H. Mineno, and H. Fukuda, “Leafmovement-based growth prediction model using optical flow analysis and machine learning in plant factory,” *Front. Plant Sci.*, vol. 10, no. March, pp. 1–10, 2019, doi: 10.3389/fpls.2019.00227.
- [2] A. B. Prasetyo, Irwanto, M. Y. Sobari, and B. Anwar, “Implementasi Segmentasi Citra Dengan Metode Threshold Pada Pengolahan Citra Digital Tanaman Beracun Di Indonesia,” *J. Inform. Terap.*, vol. 4, no. 2, pp. 2–5, 2016, [Online]. Available: file:///C:/Users/hp/Downloads/implementasi%20segmentasi%20citra%20dengan%20metode%20threshold%20pada%20pengolahan%20citra%20digital%20tanaman%20beracun%20di%20indonesia%20(1).pdf.
- [3] A. Ramdan, B. Sugiarto, P. D. Rianto, E. Prakasa, and H. F. Pardede, “Support Vector Machine-based Detection of Pak Choy Leaves Conditions Using RGB and HIS Features,” 2018 Int. Conf. Comput. Control. Informatics its Appl. Recent Challenges Mach. Learn. Comput. Appl. IC3INA 2018 - Proceeding, no. November 2018, pp. 114–117, 2019, doi: 10.1109/IC3INA.2018.8629540.
- [4] M. Razaq, P. Zhang, H. L. Shen, and Salahuddin, “Influence of nitrogen and phosphorous on the growth and root morphology of Acer mono,” *PLoS One*, vol. 12, no. 2, pp. 1–13, 2017, doi: 10.1371/journal.pone.0171321.
- [5] D. Priadi and F. Nuro, “Seedling Production of Pak Choy (Brassica rapa L. var chinensis) using Organic and Inorganic Nutrients,” *Biosaintifika J. Biol. Biol. Educ.*, vol. 9, no. 2, 2017, doi: 10.15294/biosaintifika.v9i2.8537.
- [6] N. Ahmed, D. De, and I. Hussain, “Internet of Things (IoT) for Smart Precision Agriculture and Farming in Rural Areas,” *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4890–4899, 2018, doi: 10.1109/JIOT.2018.2879579.
- [7] F. T. Informasi, “Implementasi Segmentasi Citra Natural Thresholding Neural Network untuk Model Prediksi Implementation of Natural Image Segmentation Using Thresholding- Based Techniques Neural,” 2015.
- [8] H. Jeong, B. Jeong, M. Han, and D. Cho, “applied sciences Analysis of Fine Crack Images Using Image Processing Technique and High-Resolution 49 Camera,” 2021.
- [9] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, “Image Segmentation Using Deep Learning: A Survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–22, 2021, doi: 10.1109/TPAMI.2021.3059968.
- [10] A. Çuhadar, A. C. Downton, and M. Fleury, “Structured parallel design for embedded vision systems: A case study,” *Microprocess. Microsyst.*, vol. 21, no. 2, pp. 131–141, 1997, doi: 10.1016/S0141-9331(97)00012-4.

- [11] R. Udendhran, M. Balamurugan, A. Suresh, and R. Varatharajan, "Enhancing image processing architecture using deep learning for embedded vision systems," *Microprocess. Microsyst.*, vol. 76, p. 103094, 2020, doi: 10.1016/j.micpro.2020.103094.
- [12] H. Cho et al., "Development of raspberry Pi single-board computer architecture based ultra-compact optical coherence tomography," *Opt. Lasers Eng.*, vol. 148, no. January 2021, p. 106754, 2022, doi: 10.1016/j.optlaseng.2021.106754.
- [13] R. Chandana, S. a K. Jilani, and S. J. Hussain, "Smart Surveillance System using Thing Speak and Raspberry Pi," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 4, no. 7, pp. 214–218, 2015, doi: 10.17148/IJARCCCE.2015.4749.
- [14] G. Anand and A. K. Kumawat, "Object detection and position tracking in real time using Raspberry Pi," *Mater. Today Proc.*, vol. 47, no. xxxx, pp. 3221–3226, 2021, doi: 10.1016/j.matpr.2021.06.437.
- [15] Pramuda A. Cahyan, dkk., "Segmentasi Citra Digital Menggunakan Algoritma Watershed dan Lowpass Filter Sebagai Proses Awal", Brawijaya, Malang, 2013
- [16] M. Ahammad, "Machine Learning Model Accuracy and Loss." <https://www.kaggle.com/getting-started/186841>.
- [17] Stack Overflow, "How to interpret loss and accuracy for a machine learning model [closed]," 2020. <https://stackoverflow.com/questions/34518656/how-to-interpret-loss-and-accuracy-for-a-machine-learningmodel>. 50
- [18] F. Sulianta and A. Wahyu, "Pemanfaatan Teknologi Smart Farming & Intelligent Imaging untuk Meningkatkan Sikap dan Kesadaran Konservasi Area Perkebunan Kopi," *J. Inf. Technol.*, vol. 1, no. 1, pp. 31–38, 2019, doi: 10.47292/joint.v1i1.7.
- [19] X. Mai, H. Zhang and M. Q. -. Meng, "Faster R-CNN with Classifier Fusion for Small Fruit Detection," 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, 2018, pp. 7166-7172. doi: 10.1109/ICRA.2018.8461130
- [20] Android. (2019). Meet Android Studio. (Online) <https://developer.android.com/studio/intro> (8 Oktober 2019).
- [21] Taufiq, Imam. 2018. Deep Learning Untuk Deteksi Tanda Nomor Kendaraan Bermotor Menggunakan Algoritma Convolutional Neural Network Dengan Python Dan Tensorflow. Skripsi. Program Studi Sistem Informasi Sekolah Tinggi Manajemen Informatika dan Komputer AKAKOM.
- [22] Russa, Helder Filipe de Sausa. 2017. Computer Vision: Object Recognition With Deep Learning Applied to Fashion Items Detection in Images. Tesis. Faculdade de Economia Universidade Do Porto. [23] Jalled, Fares. 2016. Object Detection Using Image Processing. Diakses dari <https://arxiv.org/pdf/1611.07791.pdf>
- [23] McCarthy. 2007. Artificial Intelligence. Binus University.
- [24] Dobrev. 2004. Artificial Intelligence. Binus University.
- [25] Perkovic, Ljubomir. 2012. Introduction to Computing Using Python: An Application Development Focus.
- [26] Tan, M., Pang, R., & Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 10781-10790).

IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.

We suggest that you use a text box to insert a graphic (which is ideally a 300 dpi TIFF or EPS file, with all fonts embedded) because, in an MSW document, this method is somewhat more stable than directly inserting a picture.

To have non-visible rules on your frame, use the MSWord "Format" pull-down menu, select Text Box > Colors and Lines to choose No Fill and No Line.