Available online at JDR Website: http://journal.unublitar.ac.id/jdr

Journal Of Development Research, 4 (1), May 2020, Page 1-6

DOI: https://doi.org/10.28926/jdr.v4i1.93

Tomatoes and Chilies Type Classifications by using Machine Learning Methods

Irzal Ahmad Sabilla[(1)], Chastine Fatichah[(2)]

Informatics Department, Institut Teknologi Sepuluh Nopember, Indonesia

E-mail: [(2)]chastine@cs.its.ac.id

Received: 14 January 2020 ; Revised: 21 January 2020 ; Accepted: 23 January 2020

Abstract

Vegetables are ingredients for flavoring, such as tomatoes and chilies. Both of these ingredients are processed to accompany the people's staple food in the form of sauce and seasoning. In supermarkets, these vegetables can be found easily, but many people do not understand how to choose the type and quality of chilies and tomatoes. This study discusses the classification of types of cayenne, curly, green, red chilies, and tomatoes with good and bad conditions using machine learning and contrast enhancement techniques. The machine learning methods used are Support Vector Machine (SVM), K -Nearest Neighbor (K-NN), Linear Discriminant Analysis (LDA), and Random Forest (RF). The results of testing the best method are measured based on the value of accuracy. In addition to the accuracy of this study, it also measures the speed of computation so that the methods used are efficient. From the experimental results, we obtained that Random Forest has the highest accuracy than that of other methods, that is 85.21%. Nevertheless, Random Forest has the longest computational time, that is 1092.93 s. The best computational time duration is obtained by KNN, which is 17.7488 s.

Keywords: classification, image processing, machine learning

Introduction

Fruit is one of the complementary foods that are consumed daily. Fruit has many benefits for the body because it contains vitamins and fiber. There are several types of fruit that are easily recognized. This is due to prominent physical differences, such as size and color.

In fact, some people still have difficulty determining the maturity and freshness of the fruit. In addition, sellers often commit fraud to buyers by exchanging ripe fruit for less good fruit. Thus, we need specific technique to classify the maturity and freshness of fruit automatically to prevent fraud by the seller.

Fruit classification is a technique to differentiate fruit types based on physical characteristics from the skin to the shape of the fruit. Previous research (Hossain, 2019) made a fruit harvesting robot to make it easier for farmers to harvest tomatoes. This study aims to ensure the tomatoes harvested have a uniform quality in the processing of sauce. However, the accuracy produced is quite low and can only distinguish tomatoes.

The other previous research (Pavithra, 2015) has focused on differentiating tomato species by their level of freshness. The method proposed in the study is the K-Nearest Neighbor (K -NN) Algorithm and Support Vector Machine (SVM). However, this research has a weakness in terms of accuracy that is not up to 90% and the dataset used only distinguishes rotten variations or not.

Machine learning is not only used to classify tomatoes (Sabilla, 2019). The previous study tried to compare several machine learning algorithms with a dataset in the form of seven types of bananas and three categorical levels of maturity. Accuracy results obtained on average get a value of 96%. Preprocessing techniques are also used to improve time efficiency and accuracy. However, the dataset used does not rep-

resent the whole banana. Because the dataset used is single.

This paper focuses on classifying tomatoes and chilies from their level of maturity and species using machine learning. The machine learning algorithms used are K-Nearest Neighbor (K-NN), Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), Random Forest (RF), and Decision Tree (DT). This research also proposes to reduce the size of the image so that the duration of computing time can run faster.

Materials and Methods

Machine learning is an artificial intelligence that is able to learn the character of data (Sunaryono, 2019). In this study, machine learning is used to study data with images. Each image is converted in pixels to produce a color intensity value for each pixel. The color values will be entered into machine learning to be studied and labeled. Thus, each class has different intensity characteristics. This paper uses KNN, SVM, LDA, RF and DT algorithms for machine learning. Each algorithm has a different way to separate each data, but all of them have the same goal of comparing values to be grouped into a class.

K-Nearest Neighbour (K-NN)

K-Nearest Neighbour (K-NN) (Connell, 1996) is an algorithm by comparing each neighbor's value and finding the closest value of the whole. This algorithm is quite lazy because it has to compare all available data, so it takes a long computational duration. However, the advantage of K-NN is the value of data precision because all data plays a role in parameters. The K-NN algorithm can be seen in Figure 1.

$Classify\ (\alpha, \beta, \gamma)$
for $i = 1$ *to* j **do**
  Compute distance $d(\beta_i, \gamma)$
  $d(\beta_i, \gamma) = \sqrt{\sum_{j=1}^{k}(x_j - y_j)^2}$
end for
Compute the set of I containing indices for k smallest $d(\beta_i, \gamma)$
return *label for* $\{\beta_i$ *where* $i \in I\}$

Figure 1. K-NN Algorithm

where

α is class labels,

β is training data, and

γ is unknown samples

Support Vector Machine (SVM)

Support Vector Machine (SVM) is an algorithm by determining classes based on two straight or curved lines. But image classification would be better using linear lines. SVM determines lines by giving a random line and calculating the distance between the two different classes. If it is still far away, this line will shift. The value of the shift is symbolized by C and the maximum distance of each class is shifted using γ. In this study using the shift 0.0001, 0.001 and 0.01 seen the best conditions using the parameters grid (Sabilla, 2019). The computation time duration in SVM depends on the initial random line value. If the distance to the class is right then the SVM line does not need to look for the line position again. However, it can also have a long-lasting effect if the initial value of a random line position is far from the class value. Figure 2 shows the SVM algorithm.

$Model\ C <= 0.0001, 0.001, 1$
$Model\ \gamma = 0.0001, 0.001, 1$
$Kernel\ [Linear\ SVM]$
For all $\{xi, yi\}, \{xj, yj\}$ **do**
  $Model\ SVM.paramgrid(C, \gamma)$
end for

Figure 2. SVM Algorithm

where

C is the value of the shift, and

γ is the maximum distance of each class (grid).

Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) is a machine learning technique separating data classes by analyzing each data characteristic (Sunaryono, 2019). When the data has labels, the LDA will form a linear line to separate each class. This line was created from the results of data analysis conducted previously. LDA is often closely related to the PCA algorithm because it separates classes based on the results of data

analysis. Equation 1 is the formula for LDA.

$$P(x|C_i) = \frac{1}{(2\pi)^{\pi/2}|\sum|^{1/2}} e^{-1/2(x-\mu_i)'\Sigma(x-\mu_i)} \quad (1)$$

where

x is input feature,

C is class, and

μ_i is mean.

Random Forest (RF)

Random Forest (RF) is a machine learning technique by spreading data like trees (Chen, 2020). After being distributed or branched out until the maximum data is used up, it will be voted according to the class of data characteristics. To select data or vote by decision tree. Random forest has the advantage of comparing incomplete data because of its mechanism for extracting features. But the drawback of Random Forest is that it has a lot of computational parameters and all data is used as parameters even though the data has the potential to be noise, so it takes a long time to compute.

Decision Tree (DT)

Decision Tree (DT) is machine learning using a chain tree mechanism (Grana, 2012). This machine learning is an algorithm that is easy to understand. How it works DT by taking the middle or random values as the parent of the tree head. In this study the diagram tree works multi by dividing two sides. If the class data is in the final position, the computation will be unified. Each data will compare with the chain above until back to the parent. The advantage of DT is that of KNN, which is comparing in detail on each data. The computing time of DT is faster than that of KNN. This is because the computation process is multi or depends on the number of branches. However, DT has the disadvantage that the initial value of the tree is far from the intended class, giving rise to an unstable accuracy value. Figure 3 shows the DT algorithm.

Results and Discussions

Software Scenario

Image processing is done using python 3.7 software with the help of spyder tools. In the library, this research utilizes two open source libraries, namely OpenCV as an image process, Scikit-learn to model machine learning (Varoquaux, 2015).

```
GenerateDT(Sample ε, Feature σ)
if condition(ε, σ) is true
    then
    leaf = createNode()
    Leaf.label = Classify(ε)
end if
Return leaf
Root = createNode()
Root.condition = find the best split of ε and σ
W = {ω|ω is possible outcome of root.condition}
For each value ω ∈ W:
        ε_ω := {ε'| root.codition(ε') = ω and ε' ∈ ε};
        Child = GenerateDT(ε_ω, σ)
        Add child node
end for
Return root
```

Figure 3. Decision Tree Algorithm

Data Collection

Data is collected with the help of a 5 MP resolution camera with an average height of 30 cm. Each class of types of chilies and tomatoes has 17 pictures with different variations. Data collected was 16 classes consisting of red chili, cayenne pepper, and curly chili. For tomato data consists of fruit tomatoes, cherry tomatoes, and green tomatoes. Each class has two types of conditions, namely in a mature condition and a bad condition. For example data can be seen in Figure 4. Each class has 17 photographic images so that the total picture in this study is 272 images.

Testing Scenario

The trial scenario is carried out with several mechanisms, namely preprocessing and classification. Preprocessing aims to make it easier for classifiers to get high accuracy values and speed up the duration of computing time. Figure 5 is a research workflow.

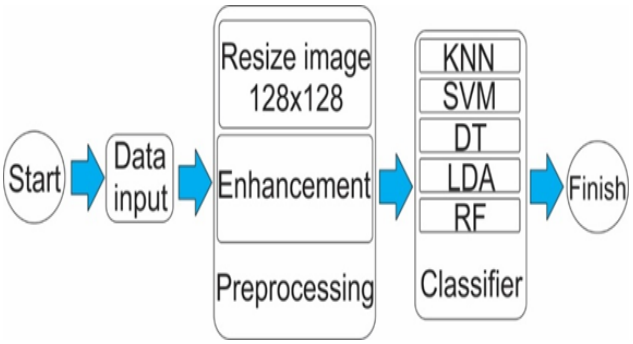

Figure 5. Testing Scenario

Figure 4. The Dataset (a) Green Chili, (b) Red Chili, (c) Cayenne Chili, (d) Curly Chili, (e) Red Tomatoes, dan (f) Green Tomatoes

Preprocessing

Before entering the image classification method, it will be preprocessed by increasing contrast and equalizing colors using Contrast Limited Adaptive Histogram Equalization (CLAHE). CLAHE aims to have images that have the same color and pattern values to increase the classification value. The results of the CLAHE method experiment can be seen in Figure 6. It is evident that preprocessing with CLAHE can improve image quality in terms of color spread. After CLAHE, the image will be resized to reduce the size of the image. This step aims at efficiency of computational time duration. The computational time duration after resizing the images reach 4.34 seconds, meanwhile the computational time duration before resizing

Table 1. Classification Results using the Machine Learning Algorithm

Explanation	SVM	KNN	LDA	RF	DT
Accuracy	81.03%	68.81%	80.45%	85.21%	59.34%
Time (s)	36.8 s	17.7488 s	31.2325 s	1092.93 s	39.2576 s

the images reach 34.2 seconds. In this study all images were resized to 128 x 128 pixels from 1280 x 720 pixels.

(a)

(b)

Figure 6. Image Comparison (a) before Preprocessed and (b) after Preprocessed by Increasing Contrast using Adaptive Histogram Equalization

Classification

The classification process is the process of determining the type and class of images. The data will be divided as much as 70% for training and 30% for testing. The data that has been divided will be entered into the matrix and flattened. The purpose of flatten in data is to speed up computing and balancing the behavior of each image. The classifications used include KNN with K = 3, SVM with C = 0.0001, Y = 0.0001, Random forest, Multi-Layer Perceptron, and LDA. The results of the classification experiment can be seen in Table 1.

Conclusion

It can be concluded that the accuracy of Random Forest has the highest value with an accuracy of 85.21%. However, the computational time duration in Random Forest is the longest compared to other classifiers. This is due to the fact that the Random Forest mechanism has branches up to the start-to-end value including noise data that is not used to contribute to the parameters. Random forest also compares each branch which results in a computational time duration. The best computational time duration is obtained by KNN, which is 17.7488 s.

For further research, data classes and preprocessing stages are added. Random forest can be an effective algorithm if the data classified is data that is free from noise. In general, the Principal Component Analysis (PCA) method is very suitable for selecting data. So that the branches in the random forest classification are not too many to reduce computational time and increase the value of accuracy.

References

M. S. Hossain, M. Al-Hammadi, and G. Muhammad, "Automatic Fruit Classification Using Deep Learning for Industrial Applications," *IEEE Trans. Ind. Informatics*, vol. 15, no. 2, pp. 1027–1034, Feb. 2019.

V. Pavithra, R. Pounroja, and B. S. Bama, "Machine vision based automatic sorting of cherry tomatoes," in *2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, 2015, pp. 271–275.

I. A. Sabilla, C. S. Wahyuni, C. Fatichah, and D. Herumurti, "Determining banana types and ripeness from image using machine learning methods," *Proceeding - 2019 Int. Conf. Artif. Intell. Inf. Technol. ICAIIT 2019*, pp. 407–412, 2019.

D. Sunaryono, J. Siswantoro, and R. Anggoro, "An android based course attendance system using face recognition," *J. King Saud*

Univ. - Comput. Inf. Sci., no. xxxx, pp. 1 –9, 2019.

R. M. B. J. H. Connell, N. H. R. Mohan, I. B. M. T. J. Watson, P. O. Box, and Y. Heights, "gievision : A Produce ecognition System," pp. 244–251, 1996.

L. Chen, W. Su, Y. Feng, M. Wu, J. She, and K. Hirota, "Two-layer fuzzy multiple random forest for speech emotion recognition in human-robot interaction," *Inf. Sci. (Ny).*, vol. 509, pp. 150–163, 2020.

C. Grana, M. Montangero, and D. Borghesani, "Optimal decision trees for local image processing algorithms," *Pattern Recognit. Lett.*, vol. 33, no. 16, pp. 2302–2310, 2012.

G. Varoquaux, L. Buitinck, G. Louppe, O. Grisel, F. Pedregosa, and A. Mueller, "Scikit-learn," *GetMobile Mob. Comput. Commun.*, vol. 19, no. 1, pp. 29–33, 2015.

