ILKOM Jurnal Ilmiah Vol. 13, No. 3, December 2021, pp. 259-265
Accredited 2nd by RISTEKBRIN No. 200/M/KPT/2020; E-ISSN 2548-7779 | P-ISSN 2087-1716

Classification of cendrawasih birds using convolutional neural network (CNN) keras recognition

Warnia Nengsih[a,1*], Ardiyanto[a,2], Ayu Putri Lestari[a,3]
[a] *Politeknik Caltex Riau, Jl. Umban Sari No. 1 Rumbai Pekanbaru Riau, 28265, Indonesia*
[1] *warnia@pcr.ac.id;* [2]*ardiyanto@gmail.com;* [3] *ayulestari@gmail.com*
** Corresponding author*

Article history: Received June 08, 2021; Revised October 01, 2021; Accepted October 01, 2021; Available online December 07, 2021

Abstract

Classification is part of predictive modeling and supervised learning. This method is used to determine the data class based on the previous value. In solving certain cases, there are various classification methods with varying degrees of accuracy. Convolutional Neural Network (CNN) is part of the Multilayer Perceptron (MLP) for processing two-dimensional data. CNN is also part of the Deep Neural Network and is applied to image objects. Some sources state that the classification process using images is not appropriate to be implemented in MLP. This will result in the accuracy of the method in handling certain cases. This study uses cendrawasih bird as object in the classification process to determine the accuracy of the keras recognition method. From the results of this study, a training model was conducted using 10 epochs with an accuracy and loss value of 0.0850 and 2.5658 respectively. These results indicate that MLP can successfully complete the classification process using images. These results indicate that MLP can successfully complete the classification process using images.

Keywords: Tensor; CNN Keras; Classification

Introduction

Convolutional Neural Network (CNN) is part of the Deep Neural Network and can be implemented for image objects. In addition, CNN is part of the development of Multilayer Perceptron (MLP) [3]. Classification of image objects will map data into certain classes [4]. This study applied keras recognition in object classification process with both accuracy and loss values as output.

The data used in this study is a dataset of cendrawasih bird images. There are 13 types of birds of paradise that will be used, namely Astrapia, Blue Cendrawasih, Red Cendrawasih, Panji Cendrawasih, Bald Cendrawasih, Golden Cendrawasih, Belah Rotan Cendrawasih, Matikawat Cendrawasih, Manucodia, Kerah Cendrawasih, Raja Cendrawasih, Toowa Cemerlang, Bidadari Halmahera. The dataset was obtained from google with a total of 578 images.

Several studies have been conducted in relation to this research. Firstly, a study on the comparison of several CNN models using the object of paintings [6]. The number of datasets used was 30000 of public dataset. This study resulted in an accuracy of 98%. Secondly, research on the detection of traffic signs using the CNN method with convolution number of 3 layers. It is not clear how many datasets involved, but they were obtained through geolocation. The level of accuracy resulted from this study is 90% [7]. Moreover, research used a spatial dimension hyperspectral dataset in the form of image classification using the Support Vector Machine (SVM). This study resulted in an accuracy rate of 80% [8].

Some references state that the classification process using images is not appropriate to be implemented in Multi-Layer Perceptron (MLP). MLP does not store spatial data on each image object used [9], so it will impact the accuracy of the method in handling certain cases. Therefore, this study is conducted to prove this argument. In this study, the object classification process uses keras recognition to determine the accuracy value of this method by using the object of cendrawasih bird.

Method

The data used in this study is a dataset of cendrawasih bird images. There are 13 types of cendrawasih bird that will be used in this study. For the purposes of data visualization and manipulation, several libraries were required including Numpy, Pandas, Matplotlib and Seaborn.

For model selection purposes, several libraries were used. Train test_split from the sklearn library was used for training and testing distribution; Kfold from the sklearn library was to evaluate the performance of the model or algorithm; accuracy_score, precision_score, recall_score, confusion_matrix, roc_curve, roc_auc_score from the sklearn

library was to obtain GridSearchCV accuracy from the library; Sklearn was to search for parameters in a given grid; LabelEncoder from sklearn library was for changing Image Data Generator data labels. Keras. preprocessing.image for data processing to be used. There are severa; sub-libraries in the CNN library that were used namely Dropout, Flatten, Activation from Keras Conv2D, MaxPooling2D, BatchNormalization from Keras, Tensorflow.

As one of the developments of Multi Layer Perceptron (MLP), CNN is an architecture that can be trained and consists of several stages. The input and output of each stage is composed of several arrays (feature map). Each stage consists of three layers, namely convolution layer, activation and pooling layer. Convolution layer has output that can be used as feature map [15]. The formula (1) is used for this layer.

$$s(t) = (x*t)(t) = \sum_{\alpha=-\infty} x(\alpha) * w(t-\alpha) \quad (1)$$

Where:
S(t) = Convolution operation result function
X = Input
W = weight (kernel)

However, if the convolution operation has more than one dimension, the formula (2) dan (3).

$$s(i,j) = (K*I)(i,j) = \sum\infty \sum\infty I(i-m, j-n)K(m,n) \quad (2)$$

$$s(i,j) = (K*I)(i,j) = \sum\infty \sum\infty I(i+m, j+n)K(m,n) \quad (3)$$

Furthermore, the process of calculating neurons for each activation applies the following formula (4).

$$(W - F + 2P)/(S + 1) \quad (4)$$

Where:
W = Image volume size
F = Filter size
P = Padding value used
S = Stride

In general, an overview of the convolutional layer shown in **Figure 1**.

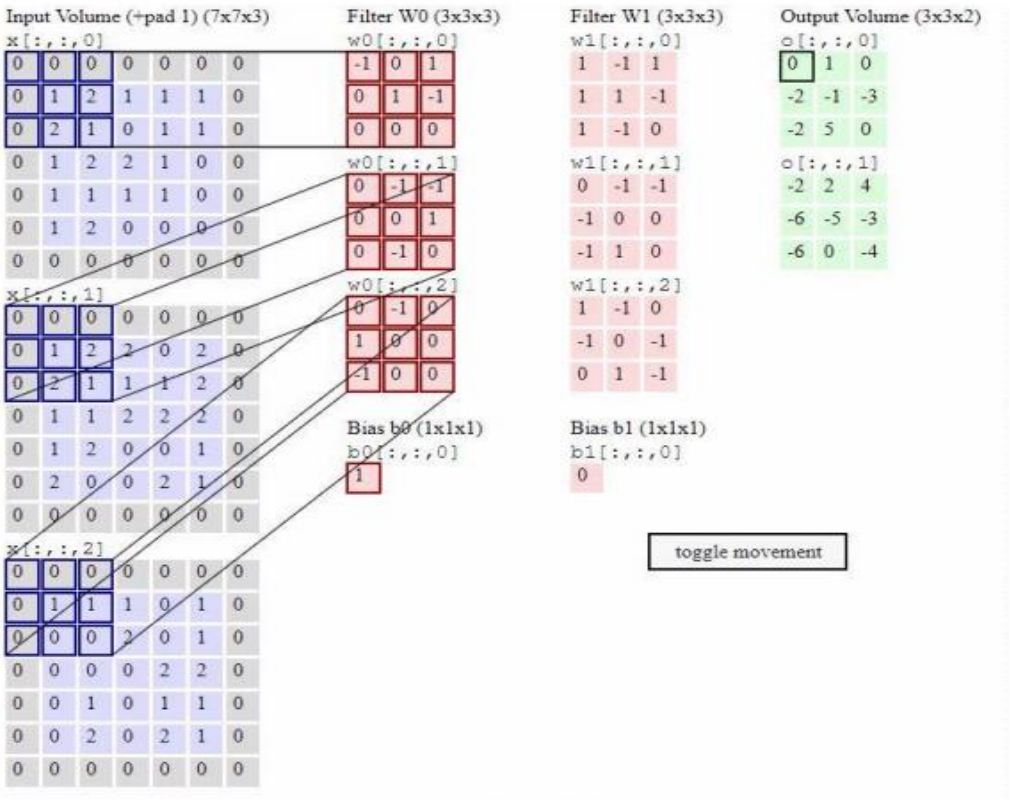

Figure 1. Convolution Layer [10]

Figure 2 is a block diagram that the design of the CNN structure that will be used.

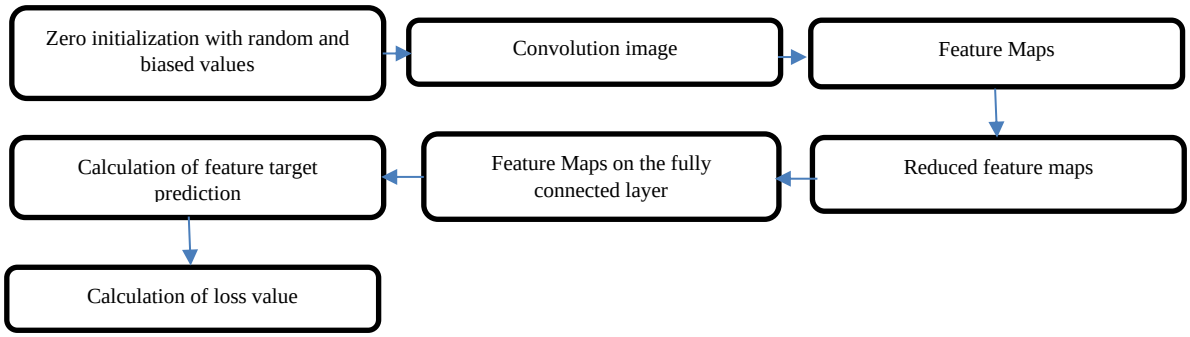

Figure 2. Convolutional Neural Network Structure

Results and Discussion

The dataset import process will be used in the kernel. The dataset used was 13 species of cendrawasih bird shown in **Figure 3**.

Figure 3. Dataset Categories

Based on the dataset categories presented in **Figure 3**, there were several stages to go through, namely creating an array to accommodate the images, creating a variable to call each folder directory for each type of data, and creating the assign_label() function which is used to adjust the label to the type of dataset. The next step is to create the make_train_data() function which is used to process the training data. This process will continue to repeat until all datasets have been processed. Finally, images from the dataset are displayed randomly.

The result of random image visualization is given in **Figure 4**.

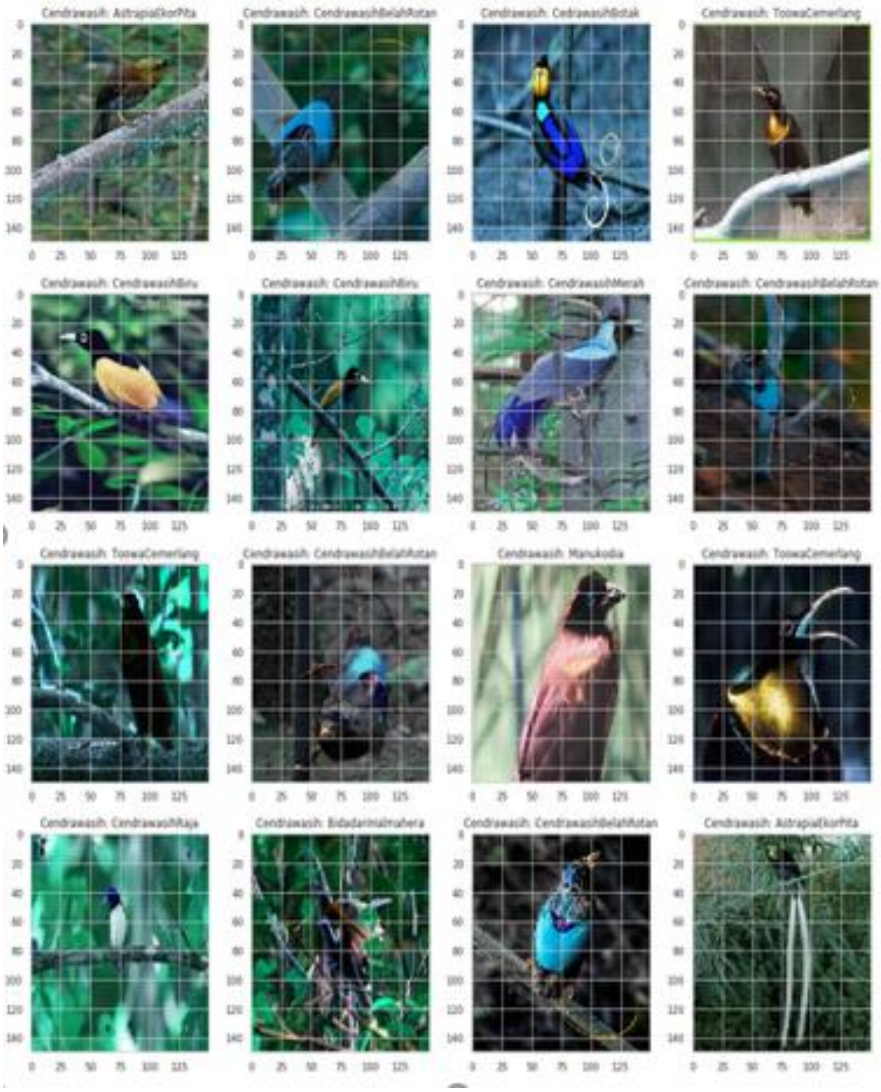

Figure 4. Random Visualization Result

This section aims to label each type of dataset with a number, such as astrapia = 0, Halmahera = 1, and so on until the entire dataset are labeled. Next is to divide the dataset into training and testing data. The test_size and the random state were set at 0.25 and 42 respectively.

To implement the convolutional neural network model, several things need to be considered, such as setting sequential, stride, flatten and dense to run the full connection neural network.

For the use of LR Annealer, the configuration setting for batch_size is the number of sample data propagated to the neutral network. Epoch is when the entire dataset has gone through the training process on the neural network until it is returned to the beginning for one round. Because one epoch is too large to be fed (feeding) into the computer, it is necessary to divide it into small units (batches). When compiling the keras and summary module, the default parameters for learning rate (Lr) >=0.

Parameter loss : used to determine loss function;
Parameter metrics : used to determine metric performance

Next is to display a summary of the Keras module as a whole. The next stage is making predictions for existing testing data where 50 epochs are used. In the model accuracy evaluation, checks will be made to display a visualization of the inaccuracy between training and testing data. The **figure 5** shows that the graph is inversely proportional which means that the analyzed dataset has little accuracy.

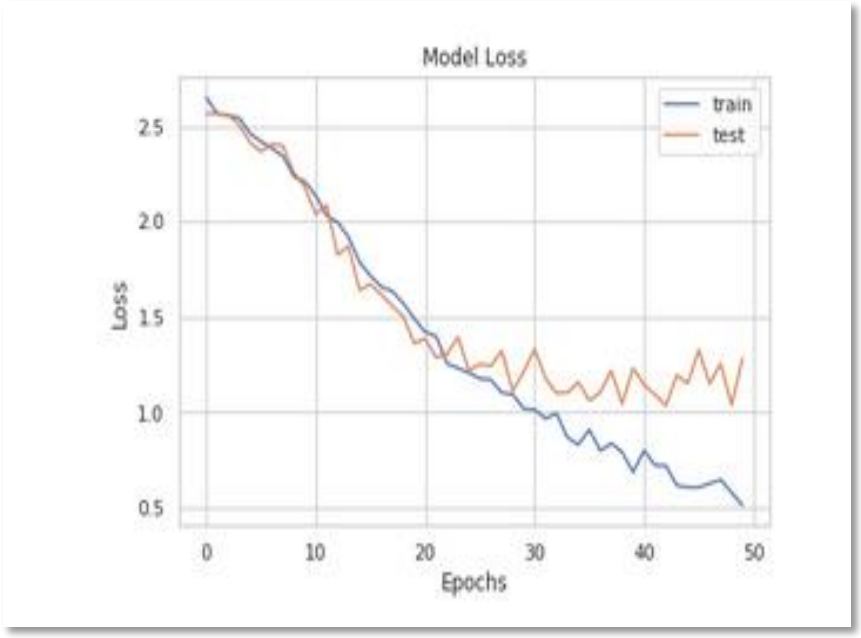

Figure 5. Loss Model Evaluation Chart

Furthermore, checks are carried out to display the visualization of the accuracy between the training and testing data. The **figure 6** shows a linear chart which means the dataset being analyzed has high accuracy.

Figure 6. Accuracy Model Evaluation Chart

Prediction is conducted to obtain predict in the data testing. Next, the process will be carried out to classify the right and wrong. The following figure shows the correct or predictable image classification, by setting:

- Display 3 rows by 2 columns by using looping
- image/figure size of 15x15

The result of prediction shown in **Figure 7**.

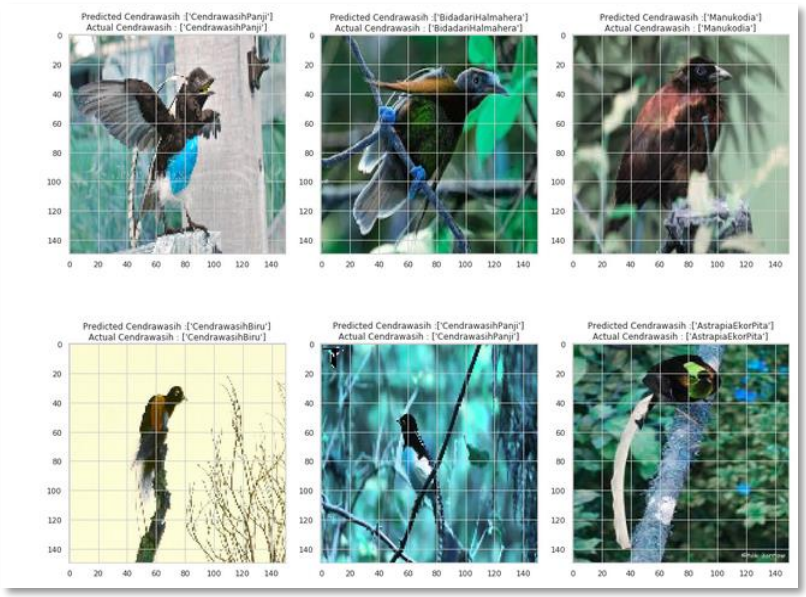

Figure 7. Correct Testing Data Prediction Results

The **figure 8** shows the classification of images that are incorrect or not as predicted.

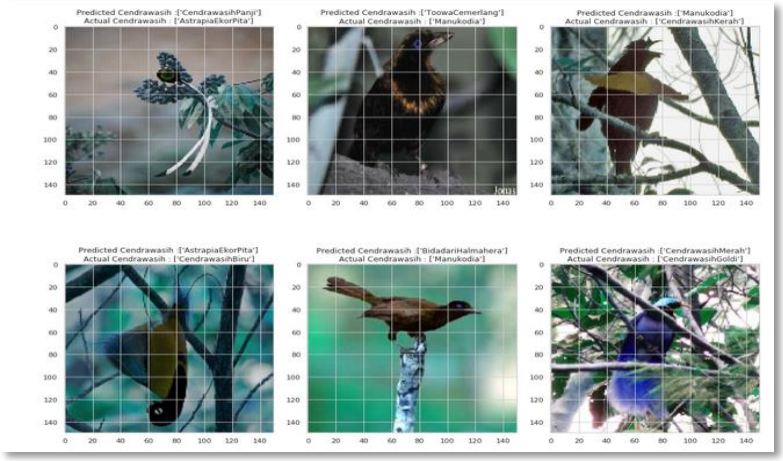

Figure 8. Correct Testing Data Prediction Results

In this section, a training model is carried out using 10 epochs. **Figure 9** shows that in the 10th iteration the accuracy is 0.0850 while the loss is 2.5658.

```
In[15]: model.fit(train,labels, epochs=10)

Epoch 1/10
953/953 [==============================] - 1s 543us/sample - loss: 4.4450 - acc: 0.0703
Epoch 2/10
953/953 [==============================] - 0s 285us/sample - loss: 2.7348 - acc: 0.0934
Epoch 3/10
953/953 [==============================] - 0s 290us/sample - loss: 2.5793 - acc: 0.0787
Epoch 4/10
953/953 [==============================] - 0s 295us/sample - loss: 2.5779 - acc: 0.0672
Epoch 5/10
953/953 [==============================] - 0s 301us/sample - loss: 2.5799 - acc: 0.0724
Epoch 6/10
953/953 [==============================] - 0s 287us/sample - loss: 2.5871 - acc: 0.0703
Epoch 7/10
953/953 [==============================] - 0s 288us/sample - loss: 2.5885 - acc: 0.0672
Epoch 8/10
953/953 [==============================] - 0s 293us/sample - loss: 2.5702 - acc: 0.0860
Epoch 9/10
953/953 [==============================] - 0s 287us/sample - loss: 2.5731 - acc: 0.0839
Epoch 10/10
953/953 [==============================] - 0s 285us/sample - loss: 2.5658 - acc: 0.0850
Out[15]:
<tensorflow.python.keras.callbacks.History at 0x7f1913c12978>
```

Figure 9. Training Model

Conclusion

Convolutional Neural Network (CNN) is part of the Multi Layer Perceptron (MLP) which is to process two-dimensional data. CNN is also part of the Deep Neural Network and is applied to image objects. Some references suggest that the classification process using images is not properly implemented in MLP leading to inaccuracy of the method in handling certain cases. In this study, the object classification process applied keras recognition to determine the accuracy value of the method using cendrawasih birds as object. From the results of this study, a training model was conducted using 10 epochs with an accuracy and loss value of 0.0850 and 2.5658 respectively. These results indicate that MLP can successfully complete the classification process using images.

References

[1] Bramer, M., Principles of Data Mining. London: Springer. 2007.
[2] Han, J., Kamber, M., & Pei, J., Data Mining Concepts and Techniques Third Edition. Waltham: Elsevier Inc. 2011.
[3] Joseph, R., S. Divvala, R. Girshick and F.-h. Ali.,. You only look once: Unified, Real-Time Object Detection, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, 779-788.
[4] Krizhevsky, A., A. Sutskever and G. E. Hikton., ImageNet Classification with Deep Convolutional Neural Networks. Adv. Neural Info. Proc. Syst. 25: 2012, 1097-1105.
[5] Lazaro, A., Bulial, J. L., & Amaliah, B., Deteksi Jenis Kendaraan di Jalan Menggunakan OpenCV. 6, 2017.
[6] Meilina, P., Penerapan Data Mining Dengan Metode Kalsifikasi Menggunakan Decision Tree dan Regresi, 2015.
[7] BKN. Statistik PNS per Desember 2018. Retrieved Juli Kamis, 2019 from Badan Kepegawaian Negara: http://www.bkn.go.id
[8] Saleh, A., Klasifikasi Gejala Depresi Pada Manusia dengan MetodeNaïve Bayes Menggunakan Java. Yogyakarta, 2015.
[9] Suartika, I. Y., Wijaya, A. Y., & Soelaiman, R., Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101. Jurnal Teknik ITS Vol. 5, No. 1, 2016.
[10] Szegedy, C., W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich., Going Deeper with Convolutions, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, 1-9.
[11] A.Coates, H.Lee, and A.Y. Ng. (2011). An Analysis of Single-Layer Network in Unsupervised Feature Learning.
[12] Alpaydin, E. (2009). Introduction to Machine Learning, Second Edition. London: MIT Press Budianita.
[13] E., Jasril. (2015). Implementasi Pengolahan Citra dan Klasifikasi K-Nearest Neighbor untuk Membangun Aplikasi Pembeda Daging Sapi dan Babi. Jurnal Sains, Teknologi dan Industri, 242- 247.
[14] Goodfellow, I., Bengio, Y, and Courville, A. (2016). Deep Learning (Adaptive Computation and Mechine Learning Series). The IMT Press.
[15] Hubel, D., and Wiesel, T. (1968). Receptive Fields and Functional architecture of monkey striate kortex. Journal of Physiology (London), 195, 215-243 92