Application of MobileNet V2 Architecture for Chinese Chess Classification Using DNN

Ryan Satria Wijaya[1], Irvant Glory Afryanto Tamba[2]

{ ryan@polibatam.ac.id[1] }

Politeknik Negeri Batam, Batam, Indonesia[1], Politeknik Negeri Batam, Batam, Indonesia[2]

Abstract. Currently, there are many methods available in the use of Deep Neural Networks (DNNs), but most of them really need devices that support high-level GPUs to get maximum results, and in this study aims to get good results by using devices that do not require the use of GPUs. Therefore, in this study, MobileNetV2 is used as a model used for DNN where this model is very easy to use without having to have a GPU. And the study has results that show that the DNN model using MobileNetV2 is able to distinguish objects with an average accuracy of 98.30%, precision of 98.70%, and recall of 98.90%. These findings have the potential to have a significant impact on various fields and industries, or on the game of Chinese chess, because the dataset used in this study is Chinese chess.

Keywords: Deep Neural Networks (DNN), MobileNetV2, Object Recognition.

1 Introduction

Object recognition is one of the main challenges in the field of computer vision, the need for object recognition models is becoming mainstream, especially for devices with low computing levels [1]. Deep Neural Networks (DNNs) have shown outstanding performance in object recognition [2]. However, DNNs often require large computing resources, which makes them less ideal for resource-constrained devices such as devices that do not have GPUs. Therefore, the main goal of this study is to develop a Deep Neural Network (DNN)-based recognition model that can recognize objects. This research leverages Tensorflow 2, a software that has been improved in performance and scalability [3], as well as the use of Roboflow as a data labeling medium, which has tools to allow users to quickly train computer vision models without the need to write complex code [4], as well as the application of the MobileNetV2 architecture when trained with Tensorflow 2 and data prepared using Roboflow can generate object recognition models.

According to a study conducted by Tej Bahadur Shahi et al [5], the researchers compared several models, such as DenseNet-121, NASNetMobile, VGG-16, MobileNetV1, InceptionV3, and MobileNetV2, for the classification of fruit images. Of the six models, MobileNetV2 showed the best performance with a stable classification accuracy of 95.75% in Dataset 1, 96.74% in Dataset 2, and 96.23% in Dataset 3. These results show that MobileNet V2 is not only efficient in terms of speed and resource usage, but also excels at delivering consistent and accurate results on various image classification datasets.

However, in this study, the dataset used was Chinese chess, because this study was interested in classifying characters in Chinese chess. According to research conducted by Dedy Arisandi et al., who explained that Chinese chess has quite complex contours, strokes, and character patterns, so that Chinese chess characters are quite difficult for new learners to recognize, so that in the research he conducted using the Backpropagation and Direction Feature Extraction (DFE) methods, he can achieve 98% accuracy with various data augmentations, one of which is rotation up to 60° [6]. Then in the research conducted by Yang Lei et al., who explained that in the study using MobileNet embedded in the Xavis platform which is an X-ray Automatic Vision System, which is combined with a robot arm to realize the recognition of Chinese chess, from this research the results were obtained that MobileNet has good chess classification capabilities [7]. Therefore, this research contributes to the development of an object recognition system using DNN generated using TensorFlow 2 and MobileNetV2, which is designed to be able to recognize objects with a high level of accuracy without having to have a GPU, as well as utilizing Roboflow as a platform for dataset labeling, and the research also succeeds in optimizing system performance without sacrificing accuracy.

2 Methods

The purpose of this study is to develop a DNN-based object recognition model to recognize objects by using TensorFlow 2 technology for model creation and training, Roboflow for dataset management, and MobileNetV2 as a model architecture. In this method, there will be several steps, ranging from data collection training, image classification, model training, system model, to method evaluation. And in this study, there is already a dataset that has 8600 images, with Chinese chess objects.

2.1 Training Data Collection

In this study, the process of collecting datasets was carried out by taking image samples on each type of character in Chinese chess. The goal is to group and identify predetermined objects. This process uses Machine Learning (ML) algorithms to analyze visual features in images and determine the most appropriate class category [8]. Image classification has various important roles in this research [9], [10], namely in the recognition of shapes in each type of character in Chinese chess. A dataset is a grouped image, as shown in Figure 1. The dataset in this study has 8600 images, each containing 1075 images which are then labeled according to what has been determined. To help computers recognize data in the Deep Learning process [11], [12], an annotation process is required. This process is the labeling of information on the dataset to be used. Once the annotation process is complete, the data will be used to train the model using the MobileNetV2 architecture. Thus, the computer can learn and understand the data to perform the requested task.

Fig. 1. Building Datasets

2.2 Image Classification

In this study, image classification is a very important part of image processing [13]. Therefore, classification has an important role in finding similar images in a dataset, or in new data. Figure 2 is the process of the image classification system used in this study. The process of an image classification system involves several steps, from taking pictures to visualizing the results. This includes capturing images, then classifying them using the MobileNetV2 model, then predicting the outcome, and visualizing the outcome.

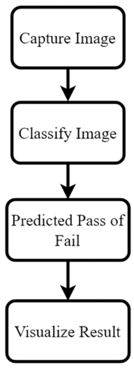

Fig. 2. Image Classification System Process

2.3 Model Training

In this study, TensorFlow was used to process MobileNetV2 [14]. It then retrains the model with new data, resulting in a classifier with fast calculations and good accuracy. TensorFlow provides many models that have been trained with Common Object in Context (COCO) datasets. The model has its own speed and accuracy level, according to the model architecture [15]. MobileNetV2 uses a fundamental convolution block known as "Inverted Residual", which consists of a cascade of layers to improve computational efficiency and parameter optimization in the network [16]. As depicted in Figure 3, the MobileNetV2 architecture starts with an initial convolution that accepts inputs measuring 224x224x3, using depthwise separable convolution, global average pooling, and fully connected layers for final classification.

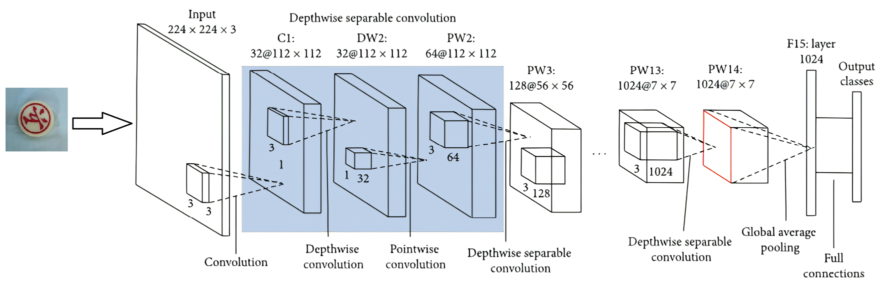

Fig. 3. MobileNetV2 Architecture

This research is continued by adopting the MobileNetV2 architecture to train the model. The training process is conducted on the IntelNuc platform. In the training configuration, a total of 20 iterations are used. Each iteration involves 128 images being processed in a batch to update the model's weights [17]. The images used in the training are 224x224 pixels in size [18]. In addition, in the training process, augmentation was added where the image can experience a maximum rotation of up to 15 degrees, a maximum hue of up to 15 degrees, saturation of up to 25%, brightness of up to 15%, and exposure of up to 10%, which can be seen in figure 4. This step helps increase the variety of data used in training. To ensure the accuracy of the model, the validation data is separated by 20% of the total available data. This allows the model to evaluate data that has never been seen before.

Fig. 4. Augmentation Dataset

2.4 System Model

To be able to perform high accuracy and fast detection, researchers took several steps. First, prepare the dataset that has been grouped, after that set up the data generator to train, and in the third step this is the important point, namely building the MobileNetV2 model which is the most important architecture in this research. Next is to train the data, after the train process on the data is completed, it is necessary to store it on the model, which can be observed in the system flowchart of the Chinese chess object classification system in figure 5.

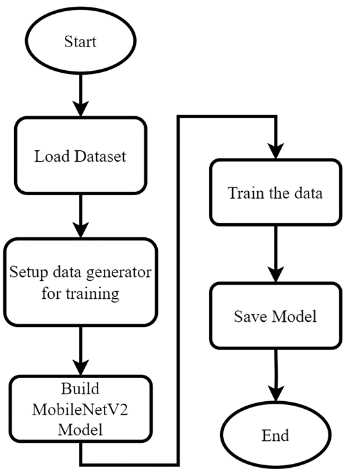

Fig. 5. Flowchart of the Chinese Chess Object Classification System

2.5 Method Evaluation

In the evaluation stage, this study aims to validate the resilience of the image classification model. Model resilience analysis is important to ensure that the model can continue to perform well in a variety of real-world situations [19]. In this study, the evaluation method used includes several scenarios such as correct classification, one point missing, two points missing, three points missing, and the order of exchanged points [20].

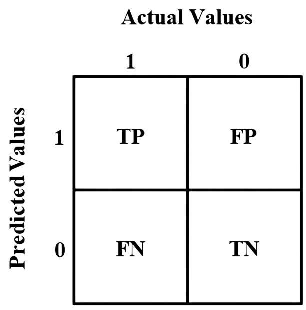

Fig. 6. Confusion Matrix

As seen in Figure 6. Confusion Matrix is used to obtain accuracy, recall, and precision values [21]. Here are the parameters used to decide these values:

1) True Positive (TP) is when the true value is expressed as Positive and predicted also as Positive.
2) True Negative (TN) is when the actual value is expressed as Negative, and the prediction is also as Negative.
3) False Positive (FP) is when the value is Negative but predicted as Positive.
4) False Negative (FN) is When the value is Positive but predicted as Negative.
5) Accuracy is a total calculation of how correct an object is across frames when undergoing a detection model. The accuracy calculation formula can be seen in equation (1).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

6) *Precision* is the total of all correct predictions versus all results predicted by the system. In this parameter precision will decide what number of objects are declared correct out of all the number of objects declared by the system. The precision calculation formula can be seen in equation (2).

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

7) Recall is the total correct estimate compared to all actual results. Recall becomes a determining parameter of the object that will be declared correct from the entire actual object. The recall calculation can be seen in equation (3).

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

The Loss function or cost function is used to measure the error between the predicted value and the actual value [22], [23]. Typically, the loss function is used as a learning criterion in optimization problems. In the context of DNN, diverse types of loss functions can be used to deal with regression and classification problems. The goal is to minimize prediction errors. One commonly used loss function in classification tasks, called cross entropy loss, evaluates the difference between the probability distribution predicted during training and the actual distribution [24]. This function compares the prediction probability with the actual output value (0 or 1) in each class and calculates the penalty based on the distance between them. This penalty is logarithmic, so this function assigns a lower value (0.1 or 0.2) for minor differences and a higher score (0.9 or 1.0) for larger differences [25].

3 Results

In this section, the researcher will discuss the results of his research. What will be discussed starts from the learning curve of the model training process. Then display and discuss the results of the confusion matrix, followed by displaying and also briefly discussing the output of the classification system carried out on random data.

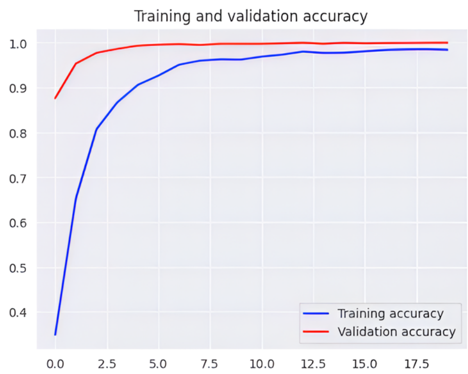

Fig. 7. Accuracy Data Training Graphs

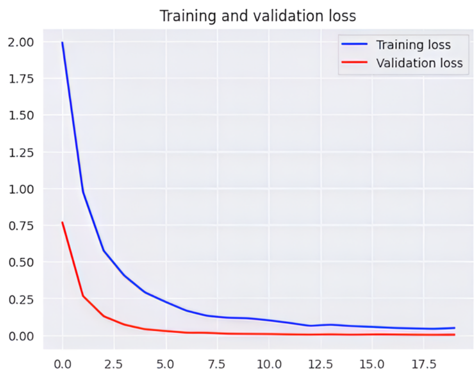

Fig. 8. Loss Data Training Graphs

Training graphs visualize the learning journey of the model from the beginning to the end of the training process. The main purpose of looking at the training graph is to identify, whether the model is overfitting or underfitting. The results of the data training test are shown in the figure above which is a graph at the time of data training. This graph is obtained when the model studies a dataset of images that have been grouped. And if based on the graph above, the results of the score are obtained by carrying out 20 iterations for the data that has been studied. Figure 7 shows that the model can achieve an accuracy value of 0.96, while the validation value is 1.00. And next in Figure 8 shows that the model can achieve a loss value of 0.09, while the validation value is 0.01. These results give an indication that the model is able to generalize well and can classify never-before-seen data with a high degree of accuracy.

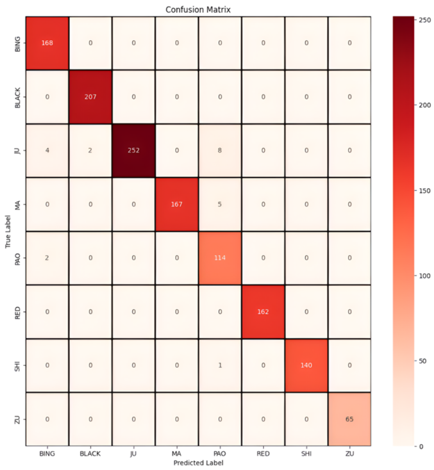

Fig. 9. Confusion Matrix Of The Classifier Under Test

Figure 9 shows the confusion matrix of the classifier tested. Based on the matrix, MobileNetV2 has a good level of classification, where there are not many errors and very few errors. From this confusion matrix, an in-depth evaluation can be carried out, starting from getting an accuracy value, which is how accurate the learning results are when evaluated with a confusion matrix, there is also a precision value, which is how precise the learning results are when matching the data taken and there is also a recall value, which is how high the success rate of the learning results is when a test is carried out to find information on the a system. However, it is also important to note that MobileNetV2 requires data augmentation to achieve high accuracy when used.

Fig. 10. Shows The Output Of The Classifier Under Test

In Figure 10, it can be seen that the classification obtained using MobileNetV2 is quite accurate and can be run even with a PC or laptop that does not have a GPU, as done by this study that uses IntelNuc as the medium to process and run the system. The average accuracy obtained with MobileNetV2 is 98.30% and for the average precision obtained with MobileNetV2 is 98.70%, and for the average recall obtained with MobileNetV2 is 98.90%.

4 Conclusion

This study has demonstrated the application of convolutional neural networks in object detection for the problem classification of Chinese chess objects in manipulation robots. Inspired by the Tensorflow Object Detection API, the classifier is trained from collected datasets labeled using Roboflow. The results of the experiment show that the model with MobileNetV2 as a feature extractor produces 98.30% accuracy, 98.70% precision, and 98.90% recall. And, in terms of size and speed, MobileNetV2 is a very easy-to-use architecture without having to have a GPU.

5 Footnotes and Acknowledgements

Footnotes. For future researchers, it is recommended to include other objects that are often seen in Chinese chess, and to collect more data on the objects to be used, as well as to perform enough data augmentation for classifiers, as it can further improve better results and can reduce system confusion.

Acknowledgements. The author would like to express his sincere gratitude to everyone who has provided support and assistance during this research process. The author also wants to acknowledge the contribution of Mr. Ryan Satria Wijaya, S.Tr.T., M.Tr.T. who has provided valuable guidance and advice during the research process. His knowledge and experience have been very helpful in shaping and improving this research. Last, but not least, the author would like to thank his comrades in arms for their cooperation and support. Without their help and dedication, this research would not have been possible. The author hopes that the results of this study can make a positive contribution to the scientific community and the wider community.

References

[1] Li, X., Wang, Z., Zhang, B., Sun, F., & Hu, X. (2023). Recognizing object by components with human prior knowledge enhances adversarial robustness of deep neural networks. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[2] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4510-4520).

[3] Leon, V., Mouselinos, S., Koliogeorgi, K., Xydis, S., Soudris, D., & Pekmestzi, K. (2020). A TensorFlow extension framework for optimized generation of hardware CNN inference engines. Technologies, 8(1), 6.

[4] Lin, Q., Ye, G., Wang, J., & Liu, H. (2022, January). Roboflow: a data-centric workflow management system for developing ai-enhanced robots. In Conference on Robot Learning (pp. 1789-1794). PMLR.

[5] Shahi, T. B., Sitaula, C., Neupane, A., & Guo, W. (2022). Fruit classification using attention-based MobileNetV2 for industrial applications. Plos one, 17(2), e0264586.

[6] Arisandi, D., Rahmat, R. F., & Nababan, E. B. (2016, October). Chinese chess character recognition using Direction Feature Extraction and backpropagation. In 2016 International Conference on Data and Software Engineering (ICoDSE) (pp. 1-6). IEEE.

[7] Lei, Y., Wenlong, L., Sigan, P., & Jiu-Qiang, H. (2021, December). MobileNet based Chinese Chess recognition with Xavis Platform. In 2021 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE) (pp. 1-5). IEEE.

[8] Ren, J., & Huang, X. (2020, July). Defect detection using combined deep autoencoder and classifier for small sample size. In 2020 IEEE 6th International Conference on Control Science and Systems Engineering (ICCSSE) (pp. 32-35). IEEE.

[9] Rokhana, R., Herulambang, W., & Indraswari, R. (2021, September). Multi-class image classification based on mobilenetv2 for detecting the proper use of face mask. In 2021 International Electronics Symposium (IES) (pp. 636-641). IEEE.

[10] Yong, L., Ma, L., Sun, D., & Du, L. (2023). Application of MobileNetV2 to waste classification. Plos one, 18(3), e0282336.

[11] Almghraby, M., & Elnady, A. O. (2021). Face mask detection in real-time using MobileNetv2. International Journal of Engineering and Advanced Technology, 10(6), 104-108.

[12] Baumgartl, H., Sauter, D., Schenk, C., Atik, C., & Buettner, R. (2021, July). Vision-based hand gesture recognition for human-computer interaction using MobileNetV2. In 2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC) (pp. 1667-1674). IEEE.

[13] Dong, K., Zhou, C., Ruan, Y., & Li, Y. (2020, December). MobileNetV2 model for image classification. In 2020 2nd International Conference on Information Technology and Computer Application (ITCA) (pp. 476-480). IEEE.

[14] Gulzar, Y. (2023). Fruit image classification model based on MobileNetV2 with deep transfer learning technique. Sustainability, 15(3), 1906.

[15] Seetala, K., Birdsong, W., & Reddy, Y. B. (2019). Image classification using tensorflow. In 16th International Conference on Information Technology-New Generations (ITNG 2019) (pp. 485-488). Springer International Publishing.

[16] Zaki, S. Z. M., Zulkifley, M. A., Stofa, M. M., Kamari, N. A. M., & Mohamed, N. A. (2020). Classification of tomato leaf diseases using MobileNet v2. IAES International Journal of Artificial Intelligence, 9(2), 290.

[17] Akter, S., Shamrat, F. J. M., Chakraborty, S., Karim, A., & Azam, S. (2021). COVID-19 detection using deep learning algorithm on chest X-ray images. Biology, 10(11), 1174.

[18] Ragab, M., Alshehri, S., Azim, G. A., Aldawsari, H. M., Noor, A., Alyami, J., & Abdel-Khalek, S. (2022). COVID-19 identification system using transfer learning technique with mobile-NetV2 and chest X-ray images. Frontiers in Public Health, 10, 819156.

[19] Muntean, M., & Militaru, F. D. (2023, January). Metrics for evaluating classification algorithms. In Education, Research and Business Technologies: Proceedings of 21st International Conference on Informatics in Economy (IE 2022) (pp. 307-317). Singapore: Springer Nature Singapore.

[20] Kumar, D., & Kukreja, V. (2024). Image segmentation, classification, and recognition methods for wheat diseases: Two Decades' systematic literature review. Computers and Electronics in Agriculture, 221, 109005.

[21] Hussain, D., Ismail, M., Hussain, I., Alroobaea, R., Hussain, S., & Ullah, S. S. (2022). Face mask detection using deep convolutional neural network and MobileNetV2-based transfer learning. Wireless Communications and Mobile Computing, 2022(1), 1536318.

[22] Liu, J., & Wang, X. (2020). Early recognition of tomato gray leaf spot disease based on MobileNetv2-YOLOv3 model. Plant Methods, 16, 1-16.

[23] Souid, A., Sakli, N., & Sakli, H. (2021). Classification and predictions of lung diseases from chest x-rays using mobilenet v2. Applied Sciences, 11(6), 2751.

[24] Buiu, C., Dănăilă, V. R., & Răduţă, C. N. (2020). MobileNetV2 ensemble for cervical precancerous lesions classification. Processes, 8(5), 595.

[25] Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. IEEE transactions on neural networks and learning systems, 33(12), 6999-7019.