Jurnal Ilmiah KURSOR
Menuju Solusi Teknologi Informasi
Vol. 12, No. 2, Desember 2023

ISSN 0216 – 0544
e-ISSN 2301– 6914

DEEP LEARNING ARCHITECTURE BASED ON CONVOLUTIONAL NEURAL NETWORK (CNN) ON ANIMAL IMAGE CLASSIFICATION

[a]Fawaidul Badri, [b]M. Taqijuddin Alawiy, [c]Eko Mulyanto Yuniarno

[a,b]Electrical Engineering, University of Islam Malang, Malang.
[c]Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya
E-mail: fawaidulbadri@unisma.ac.id, e-mail: taqijuddin.alawiy@unisma.ac.id.
ekomulyanto@ee.its.ac.id

Abstract

In current technological developments, Deep Learning is one of the most popular types of research currently, especially in the fields of machine learning and computer vision. Deep Learning has excellent capabilities for solving classic problems in the field of computer vision, one of which is the case of object classification in image. One of the deep learning methods that is often used in image processing is Convolution Neural Network (CNN). In this research, 3000 data were used, divided into 1500 training data and 1500 testing data and divided into two classes, namely the dog class and the cat class. This research has several stages, namely preprocessing by resizing the data to make the data size uniform, then the next stage uses a CNN architecture which consists of convolution layers, as well as fully connected layers, and the determination and loss function of the CNN. Implementation of this method uses Google Colab (Tensorflow and Keras) with the Python programming language. In the testing process with CNN, the average accuracy obtained from 4 scenarios using the CNN method gave effective results and produced quite good accuracy values with an average accuracy and loss value of 99.99% accuracy. And the average loss result is 4.

Key words: CNN, Deep Learning, Keras, Konvolusi, Phyton, Tensorflow.

INTRODUCTION

Techniques for classifying objects in images in general are the main problem in machine learning and computer vision for which a solution has long been sought. The process of classifying objects manually will be very difficult, especially if the amount of data is quite large and therefore it would be very helpful if the classicization process was used automatically with computer vision.

The development of computing technology is currently experiencing very rapid development, especially in the field of artificial intelligence, which allows computers to retrieve information from an image object for the purposes of object recognition or object classification automatically.

Image classification is an important task in the field of image processing and pattern recognition. In recent years, the use of Convolutional Neural Network (CNN) methods has brought revolutionary changes in image classification capabilities. CNN has proven itself as one of the best approaches to tackle the complex challenges of image classification.

Traditional classification methods often rely on features extracted manually from images, such as texture extraction or shape extraction. However, the use of this method has several disadvantages, including limitations in extracting complex features and non-linear relationships between these features. This is what drives the emergence of CNN as a more effective solution [1][2].

Deep Learning is a method that is widely used in digital image processing. One method that is often used in Deep Learning is Convolution Neural Network (CNN). CNN is a development of Multi Layer Perceptron (MLP). The CNN method has very high accuracy capabilities in object recognition or classification, because CNN tries to imitate the image recognition system in the human visual cortex, so that it is able to process the captured image information [3].

There were several algorithms used in image classification before the popular CNN method. The image features are then entered into a classification algorithm such as SVM (Support vector machines), in this algorithm uses image pixel level values as input feature vectors [4][5].

The CNN model follows a hierarchical path where this model builds a network or neutrons, like a funnel, and finally provides layers that are all connected to the neutrons so that all input that enters the CNN network architecture undergoes a training process to calculate the weights which will then be searched for the correct weights. the highest, the longer the training process on CNN will provide higher experience, so it will produce high weights to be matched with testing data, then the next process is the output where the entire training and testing process will show the results of processing with the CNN architecture[6][7] .

Deep Convolutional Neural Networks for Multi-Label Brain Tumor Classification Using MRI Images" by Mahmudul Hasan, Mohammed Algarni, and et al. (2019): This research uses CNN for brain tumor classification based on MRI images. They developed a CNN model that can classify images MRI into several labels that reflect the type and characteristics of brain tumors. This helps in better diagnosis and treatment planning for patients with brain tumors [8][9].

Deep Learning for Plant Disease Detection and Diagnosis" by Saleem et al. (2019): This research applies CNN for plant disease detection and diagnosis using leaf images. They developed a CNN model that can recognize disease symptoms on plant leaves with a high level of accuracy. This helps farmers and agricultural experts in early detection of plant diseases and proper management[10].

In research conducted by Rulisiana Widodo et al, they carried out lung cancer image classification segmentation techniques using the threshold method, the threshold method used was Otsu with an accuracy of 99% and the lowest accuracy was 96% [20]. Another research conducted by I Putu Bayu Wira Brata et al, carried out drum tone classification using the Neural Network Backpropagation method, the accuracy results were carried out by adding normalization with a backpropagation architecture with a learning rate value of 0.9 and a hidden layer value of 10 and an epoch value of 2000 resulting in an accuracy of 60.92 [21] . In research conducted by Iman Fahruzi, sleep disturbance detection was carried out using the data method (ECG) for feature extraction using CNN. The average system accuracy results are 83.03% [22].

In 1989, Yann LeCun et al succeeded in classifying zip code images using a special case of Feed Forward Neural Network called Convolution Neural Network (CNN) [7]. Due to hardware limitations, Deep Learning was not developed further until 2009 where Jurgen developed a new method known as Recurrent Neural Network (RNN) which obtained significant results in handwriting recognition [11]. With the development of Graphical Processing Unit (GPU) hardware computing, the development of CNN models has become very rapid, in 2012, CNN methods can perform image recognition with accuracy that rivals humans on certain datasets [12][13]. Deep learning architectural models are one of the popular topics in machine learning, because they have high intelligence in modeling very large and complex data, especially in image and sound processing. [14][15].

Research conducted by S. Divya Meena carried out animal breed classification techniques using the MP-CNN method. The data used in this research was 35,992 into 27 different animal classes. The results of this research achieved 99.95% accuracy[23]. Another research was conducted by Peiyi Zeng. Carrying out animal classification techniques using the CNN method used data from 800 of 200 animal images. The results of this research show an average accuracy value of 96.67%[24]. Another research conducted by Weitao Xu carried out multiview Wireless Acoustic Sensor Network (WASN) animal classification using the CNN data method using animal sounds. The results of this research were 84.4% [25].

The deep learning architectural model provides high accuracy results in image recognition using the CNN method. This is because the CNN model duplicates the visual cortex network system in humans [16] o it is able to extract very accurate information features. [17]. However, CNN also has a weakness, namely the long computational process of model training.

In this research, the classification technique uses deep learning with a CNN architectural model automatically on digital images. Automatic CNN model by setting multiparameter values in the CNN architecture. With the CNN architecture model, it is hoped that image classification can be obtained with good accuracy.

MATERIAL AND METHODS

The system design in this research consists of 3 stages, the first is research data, the second is data preprocessing, the third is the CNN classification analysis method shown in Figure 1.

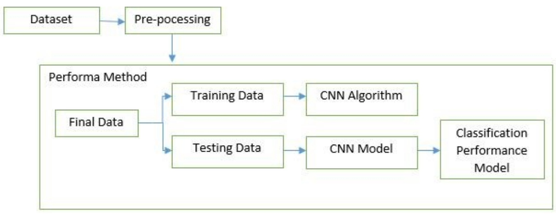

Fig 1. Design system.

Dataset

Fig 2. These are normal images of cats and dogs used in research. Training images and testing images are public images obtained on Kaggle.

The data used in this research is cat and dog data obtained from public Kaggle data. The amount of data used in this research was 3000 data. Which consists of 1500 for training data and 1500 for testing data. Each training image and test image has a size of 227 x 227 pixels before the rezising process is carried out. The original training image and testing image can be seen in Figure 2.

Pre-Processing

The pre-processing stage is the initial stage of preparation in data processing, this stage has a very important role in improving the quality of image data. At this pre-processing stage, the image size is uniform so that the input image information is clear. In this preprocessing we use the rezesing process. to equalize the data size.

Resizing

From all the raw data in this research, a resizing process will then be carried out. The resizing process aims to ensure that the image size matches what the system requires. In general, the larger the image size, the greater the system accuracy results. because more and more information will be extracted, the training process will take a very long time. then the data needs to be resized to be smaller. In this research, the raw image in this research has a size of 227 x 227 after going through the resizing process and the image changes to 150 x 150.

Performace Method

CNN is an artificial neural network method used in the field of recognition or computer vision. The structure of the CNN model imitates the way human nerve cells in each neuron cell are connected to each other to process input. In the CNN artificial neural network there is a convolution operation as a filter or kernel to extract input features to produce model weights that will be processed in the CNN model.

The development of computer vision technology in the 1980s, research conducted by Fukushima, has the unique property of not being affected by changes in rotation and position for pattern recognition.

Some of the most popular research in this field is the development of the LeNet-5 method carried out by LeCunn[18]. in 1997. It was one of the first CNNs used in banks to read checks in real-time, the LeNet-5 was able to read more than one million checks. Even though there are other algorithms such as Support Vector Machines whose accuracy is close to LeNet-5, many opinions say that CNN's calculation speed is exponentially faster than other algorithms.

Computer vision research experienced quite rapid development, in 2010, to support research in the field of recognition and computer vision, ImageNet was introduced. ImageNet is a fairly large image database and holds open competitions every year to promote research results. In 2012, the winner of the ImageNet competition was Alex Krizhevsky. Alexnet is a CNN model[18][19]. similar to LeNet-5 which has significant capabilities in several ways, thus becoming the beginning of developments in the field of artificial intelligence.

In the CNN computing process using GPU. The platform using Nvidia CUDA provides much faster performance compared to using the CPU model. Some of the main component parts in CNN are:

1. Convolution Layer
2. Pooling Layer
3. Fully Connected Layer

In the CNN method there are several convolution layers to extract features, namely the pooling layer and map layer to avoid overfitting in the image training process. In the last network layer a fully connected layer predicts the classification of the image using soft-max classification or using the regression classification shown in Figure 3.

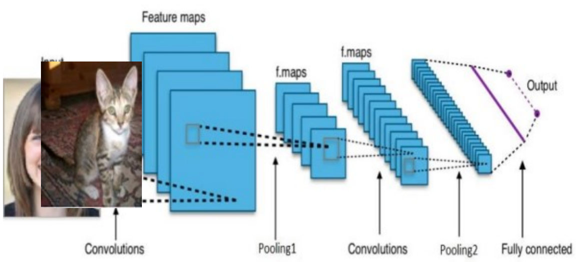

Fig 3. Structure of CNN.

In Figure 3 The CNN model used consists of four convolution layers with a filter size of 34×34, the activation function used is reLu 64, and 2 pooling layers with a size of 17×17. with 2 class output.

Convolution Layer

The convolution layer is the most important layer in CNN. The convolution parameters in this layer consist of a collection of kernels or filters that can be studied or studied. The convolution layer processes the input image by extracting local features of the image by shifting the input kernel and calculating each multiplication index coordinates on the image to produce a two-dimensional activation map feature of the kernel for the process stages can be seen in Figure 4.

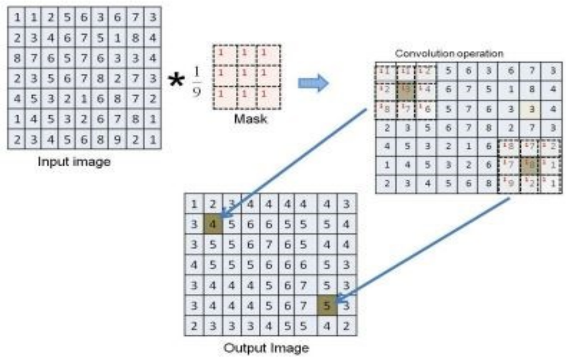

Fig 4. Image convolution operations.

The next stage of the input feature operation will produce a new matrix model from the image convolution process using nonlinear activation as shown in equation (1) to be processed into a nonlinear model.

$$x_f^{(l)}\ f(\sum_{s,s} x^{(l-1)} * wf^{(l)} + bf^{(l)}) \quad (1)$$

The f value is the result of the non-linear activation function, $bf^{(l)}$)image matrix features, $x^{(l-1)}$output results on the previous network layer,* Process Convolution Operations, and $wf^{(l)}$convolution filter results on image size s × s.

to find the gradient value in the Loos function with weight values (w) and matrix feature values (b) from each of these layers can be calculated with the equation:

$$\nabla w_f^{(l)} = \sum_{s,s} \nabla xf^{(l+1)}\ l)_{s,s}\ x_{s,s}^{(l)} * w_f^{(l)} \quad (2)$$

$$\nabla bf^{(l)} = \sum_{s,s}(\nabla_{xf}^{(l+1)}\ l_{s,s})\ (x_{s,s}^{(l)} * w_f^{(l)}) \quad (3)$$

All images have the same weight (kernel) for each feature map. the ability to select weight features to reduce the number of parameters to detect the same features, given the input parameters. The parameters of each convolution layer select the number of features in the input image so as to produce the desired output such as filter size, number of filters that can be trained.

From nonlinear activation functions, such as sigmoid, ReLu. a faster activation function is ReLu. image convolution process, there is a kernel or filter size and feature map (W x W) with the input image at kernel size (H x H) with size (F x F) with result value (S):

$$w = \left[\frac{W-F}{s}\right] + 1 \qquad (4)$$

Pooling Layer

A pooling layer is a layer consisting of filters with a certain size and step that shift over the entire feature map. The pooling layer will be added after the convolutional layer. This addition is used to order layers within a CNN that may be repeated one or more times in a particular model. In general, filters in the pooling layer use 2×2 filters applied with a stride of 2, which then operates on each slice of the input. Pooling operations are defined, learned aren't they. Pooling partitions the contribution to isolate areas of size (R × R) to create one result from every area. There are two poolings: max pooling to calculate the maximum value for each patch in the feature map and average pooling to calculate the average value for each patch on the feature map and the result size will be acquired from the contribution of size (W × W):

$$Polling = [\frac{W}{R}] \qquad (5)$$

worldwide data about area, and where in the picture something is going on by applying most extreme pooling in the CNN. In any case, we hold data about regardless of whether the main highlights show up in the picture. The most extreme worth of non-covering blocks of the past component map (l−1) is calculated during the forward phase as follows:

$$x^{(l)} = max_{R,R}\left(x^{(l-1)}\right)_{R,R} \qquad (6)$$

In the max pooling process there are no limits in the learning process, therefore, the level of slope in the next layer greatly influences processing to the next network, the feature with the largest value will be used as input to the next layer. Figure 5. Max Poolingfunctions.

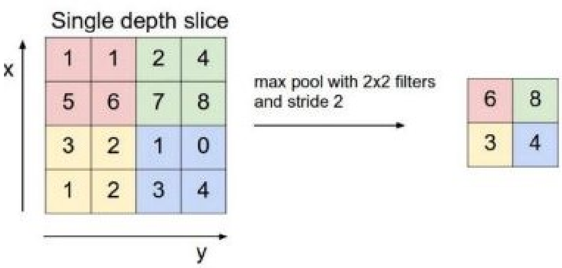

Fig 5. Max poolingfunction.

Fully Connected Layer

A CNN closes with at least one completely associated layers that associate each neuron in one layer to each neuron in another layer and has a solitary number of hyper-neuron boundaries. The objective of a completely associated layer is to remove worldwide elements from the information, and the output is calculated by the equation:

$$x^1 = f\left(w^{(l)}\right)T.x^{(l-1)} + b^{(l)} \qquad (7)$$

Where (l), (l), and (l) are the inputs, weights, and biases of the current layer (l), x (l-1) is the output of the previous layer, is the dot product, and f is the non activation function linear. The welding layer is a classifier layer such as soft-max classifier and regression classifier.

Activation Function

CNN is a type of artificial neural network method used for recognition or computer vision in image processing. The enactment capability is a non-direct capability that permits an ANN to change input information into higher aspects so basic hyperlane removes can be conveyed which permits order. In CNN there is an enactment capability utilized, in particular the sigmoid capability. The sigmoid function transforms the range of values [0,1] with the distribution form of the sigmoid function form:

$$\sigma = \frac{1}{(1+e^{-x})} \qquad (8)$$

RESULT AND DISCUSSION

The order consequences of each picture went into the CNN model structure a probalistic class dissemination comprising of two class marks. In view of the CNN order, the grouping results with the most noteworthy likelihood esteem are taken. Taking the outcomes with the most elevated likelihood esteem, it is expected that the grouping results are the outcomes that best match the input image. The order results taken are still as names for specific classes. Marks then should be characterized to know the importance of each identified person for each article in the input image.

The testing process is carried out using a loss function. The function of the loss function is to measure how good the performance produced by the model is in making predictions about the target. The loss function used in this model is cross-entropy, Cross entropy is generally used in binary classification cases.

where the classification process uses binary classification with a target value range between [0,1] in the calculation process, the loss function looks for the maximum likelihood value. then the optimization function used in this model is the Adam algorithm with a learning rate of 0.0001. CNN built with TensorFlow and the phyton library.

The data used in this research is cat and dog data obtained from public Kaggle data https://www.kaggle.com/c/dogs-vs-cats/data. The amount of data used in this research was 3000 data. Which consists of 1500 for training data and 1500 for testing data. Each training image and testing image has a size of 150x150 pixels.

Table 1. CNN model architecture.

Layer Type	Size	Output
Input image	(1, 150, 150)	-
Convolution Layer + ReLU	(148, 148, 16) filter	448
conv2d_1 (Conv2D)	(72, 72, 32) filter	4640
Max-Pooling	(36, 36, 32)	0
Convolution Layer + ReLU	(34, 34, 64) filter	18496
Max-Pooling	(17, 17, 64) filter	0
Fully-Connected Layer + ReLU + Dropout Layer	9470464 neuron	512
SoftMax Layer	2 class	512

Table 1 shows the CNN architecture for each layer which is used to extract features in the input image. The input image used in this architecture is 150x150, then the next process is Convolution Layer + ReLU Convolutional Layer is one of the important layers in CNN which is responsible for extracting visual features from input data. This layer uses convolution operations, which involve applying small filters (kernels) to the input to generate new features. This filter is shifted repeatedly across inputs to obtain unique, more detailed information. Meanwhile, ReLU (Rectified Linear Unit) is an activation function that is commonly used after convolution layers in CNNs. This function provides a non-linear response to the input and helps introduce non-linearity into the model.

In the first scenario testing, the image size is 150x150 pixels, this image will be input as training data using the CNN method with pixel normalization preprocessing [0.1] from the original data range [0.255]. Then the training data was 2000 images and 1000 validation images using 20 epochs. The accuracy value of each epoch will be displayed in Table 1.

Table 2. Accuracy value of the first scenario.

Epoch	Loss	Accuracy
1	7680	54.65
2	6354	64.55
3	5590	71.95
4	5154	74.70
5	4722	76.90
6	3938	81.95
7	3285	86.00
8	2413	90.30
9	1672	93.25
10	1093	96.00
11	0930	96.80
12	0613	98.55
13	0991	97.80
14	0274	99.35
15	0427	98.60
16	0055	99.90
17	0799	98.75
18	0020	100.0
19	0570	98.55
20	0166	99.45

From Table 2, the iterations of each epoch have different loss values and different accuracy values. The loss value from the first iteration to the last iteration has a smaller loss value, this shows that the more learning processes in the CNN will produce better accuracy values.

Figure 6 shows a graph of the results of the loss values and accuracy values with an input image of 150x150 in the first scenario with an epoch value of 20.

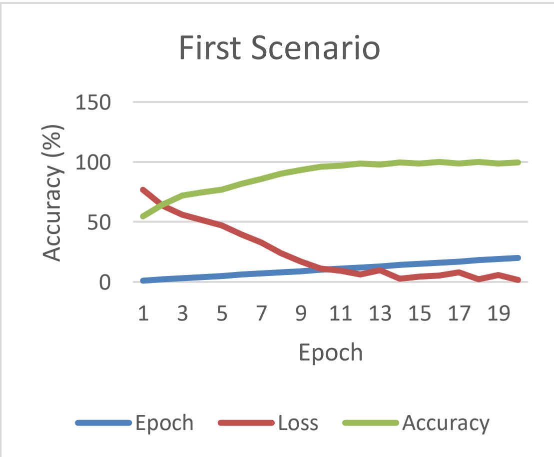

Fig 6. Accuracy results and loss values for the first scenario.

In the second test, an image size of 150x150 pixels is carried out, this image will be input as training data using the CNN method with pixel normalization preprocessing [0.1] from the original data range [0.255]. Then the training data was 2000 images and 1000 validation images using 15 epochs. The accuracy value of each epoch will be displayed in Table 3.

Table 3. Accuracy values from the second scenario

Epoch	Loss	Accuracy
1	74.16	59.00
2	59.58	69.85
3	48.43	75.90
4	41.82	80.35
5	31.45	86.80
6	22.85	90.25
7	16.09	93.85
8	10.97	96.00
9	08.83	96.85
10	05.40	98.30
11	06.51	98.35
12	03.91	98.90
13	05.99	99.00
14	02.92	98.90
15	03.03	99.00

From Table 3, the iterations of each epoch have different loss values and different accuracy values. The loss value from the first iteration to the last iteration has a smaller loss value, this shows that the more learning processes in the CNN will produce better accuracy values.

Figure 7 shows a graph of the results of the loss values and accuracy values with an input image of 150x150 in the first scenario with an epoch value of 15.

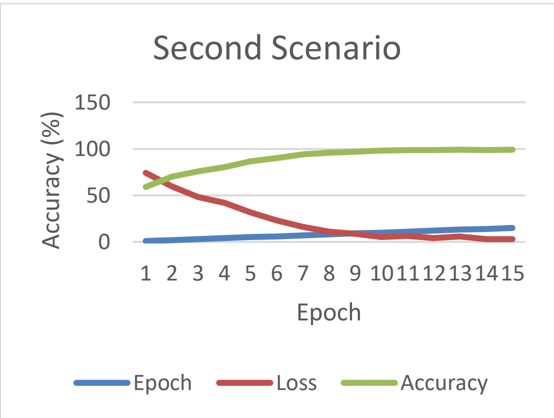

Fig 7. Accuracy results and loss values for the second scenario.

In the third test, an image size of 150x150 pixels is carried out, this image will be input as training data using the CNN method with pixel normalization preprocessing [0.1] from the original data range [0.255]. Then the training data was 2000 images and 1000 validation images using 10 epochs. The accuracy value of each epoch will be displayed in Table 4.

Table 4. Accuracy values from the third scenario.

Epoch	Loss	Accuracy
1	73.57	52.65
2	66.90	61.55
3	60.19	68.90
4	55.65	71.90
5	49.30	76.50
6	81.35	81.35
7	84.50	84.50
8	89.35	89.35
9	92.45	92.45
10	95.55	95.55

From Table 4, the iterations of each epoch have different loss values and different accuracy values. The loss value from the first iteration to the last iteration has a smaller loss value, this shows that the more learning processes in the CNN will produce better accuracy values.

Figure 8 shows a graph of the results of the loss values and accuracy values with an input image of 150x150 in the first scenario with an epoch value of 10.

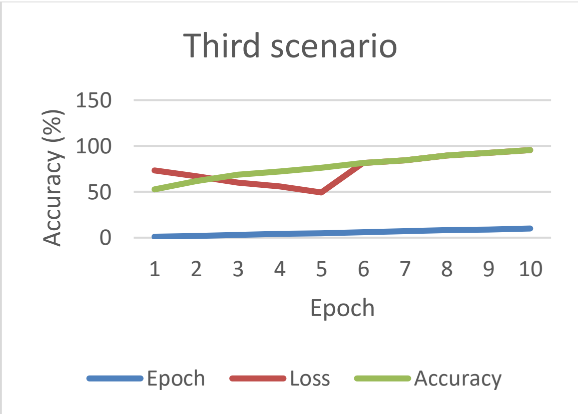

Fig. 8 Accuracy results and loss values for the third scenario.

In the fourth test, an image size of 150x150 pixels is carried out, this image will be input as training data using the CNN method with pixel normalization preprocessing [0.1] from the original data range [0.255]. Then the training data was 2000 images and 1000 validation images using 5 epochs. The accuracy value of each epoch will be displayed in Table 5.

From Table 5, the iterations of each epoch have different loss values and different accuracy values. The loss value from the first iteration to the last iteration has a smaller loss value, this shows that the more learning processes in the CNN will produce better accuracy values.

Table 5. Accuracy values from the fourth Scenario.

Epoch	Loss	Accuracy
1	72.53	51.00
2	68.01	58.55
3	60.78	67.80
4	56.29	70.10
5	51.92	73.60

In Figures 7, 8, 9 the green line is the number of epochs used in the training process, while the red line is the result of each loss in the training process, and the green line is the accuracy result of each scenario.

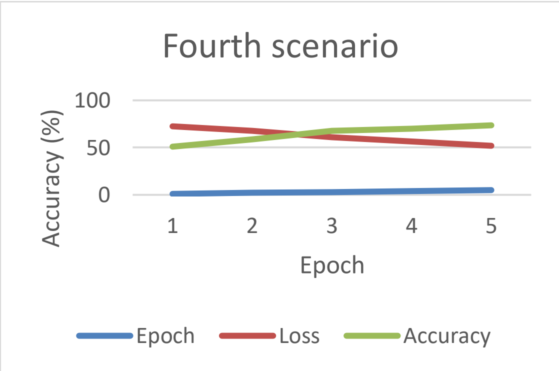

Fig. 9 Accuracy results and loss values for the fourth scenario

Figure 9 shows a graph of the results of the loss values and accuracy values with an input image of 150x150 in the first scenario with an epoch value of 5.

CONCLUSION

Experiments with 4 scenarios using the CNN method provide effective and powerful results in learning representative features automatically, dealing with variations in shift and scale, and creating hierarchical feature representations. This method is able to produce accurate classification results with high adaptive capacity so that the accuracy results from 4 experiments have an average accuracy of 99.99%. And the average loss result is 4.

Although it requires greater computational resources and longer training time, the use of CNNs in image classification provides significant advantages in the processing and analysis of structured data.

ACKNOWLEDMENT

The researcher would like to express his gratitude to the Islamic University of Malang, especially the Institute for Research and Community Service, which has provided moral and material support through the Islamic University of Malang (HI-ma) institutional grant for novice lecturers based on output, so that this research can be carried out well.

REFERENCES

[1] M. Abuzneid and A. Mahmood, “Improving Human Face Recognition using Deep Learning based Image Registration and Multi-Classifier Approaches,” Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl. AICCSA, vol. 2018-Novem, 2019.

[2] Y. Harjoseputro, “Convolutional Neural Network (Cnn) Untuk Pengklasifikasian Aksara Jawa,” Buana Inform., p. 23, 2018.

[3] Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen, “Medical image classification with convolutional neural network,” 2014 13th Int. Conf. Control Autom. Robot. Vision, ICARCV 2014, pp. 844–848, 2014.

[4] A. Muntasa, “Features fusion based on the fisherface and simplification of the laplacian smoothing transform,” Int. J. Electr. Eng. Informatics, vol. 9, no. 4, pp. 716–731, 2017.

[5] N. A. Azman, A. Mohamed, and A. M. Jamil, “Artificial Intelligence in Automated Bookkeeping: A Value-added Function for Small and Medium Enterprises,” Int. J. Informatics Vis., vol. 5, no. 3, pp. 224–230, 2021.

[6] I. A. Sabilla, “Arsitektur Convolutional Neural Network (Cnn) Untuk Klasifikasi Jenis Dan

Kesegaran Buah Pada Neraca Buah," Tesis, no. 201510370311144, pp. 1–119, 2020.

[7] S. R. Suartika E. P, I Wayan, Wijaya Arya Yudhi, "Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) Pada Caltech 101," J. Tek. ITS, vol. 5, no. 1, p. 76, 2016.

[8] X. Gu, Z. Shen, J. Xue, Y. Fan, and T. Ni, "Brain Tumor MR Image Classification Using Convolutional Dictionary Learning With Local Constraint," Front. Neurosci., vol. 15, no. May, pp. 1–12, 2021.

[9] E. Irmak, "Multi-Classification of Brain Tumor MRI Images Using Deep Convolutional Neural Network with Fully Optimized Framework," Iran. J. Sci. Technol. - Trans. Electr. Eng., vol. 45, no. 3, pp. 1015–1036, 2021

[10] M. Hammad Saleem, J. Potgieter, and K. Mahmood Arif, "Plant Disease Detection and Classification by Deep Learning," Nature, vol. 29, no. 7553, pp. 1–73, 2016.

[11] M. I. Mardiyah, "Implementasi Deep Learning untuk Image Classification Menggunakan Algoritma Convolutional Neural Network (CNN) Pada Citra Kebun dan Sawah," Univ. Islam Indones., no. June, 2020.

[12] A. Peryanto, A. Yudhana, and R. Umar, "Rancang Bangun Klasifikasi Citra Dengan Teknologi Deep Learning BerbPeryanto, A., Yudhana, A., & Umar, R. (2020). Rancang Bangun Klasifikasi Citra Dengan Teknologi Deep Learning Berbasis Metode Convolutional Neural Network. Format : Jurnal Ilmiah Teknik I," Format J. Ilm. Tek. Inform., vol. 8, no. 2, p. 138, 2020.

[13] B. Khagi and G. R. Kwon, "3D CNN based alzheimer's diseases classification using segmented grey matter extracted from whole-brain MRI," Int. J. Informatics Vis., vol. 5, no. 2, pp. 200–205, 2021.

[14] N. Khunafa Qudsi et al., "Identifikasi Citra Tulisan Tangan Digital Menggunakan Convolutional Neural Network (CNN)," Semin. Inform. Apl. Polinema , pp. 48–53, 2019.

[15] C. Yu, R. Han, M. Song, C. Liu, and C. I. Chang, "A simplified 2D-3D CNN architecture for hyperspectral image classification based on spatial-spectral fusion," IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., vol. 13, pp. 2485–2501, 2020.

16 A. Sharma and G. Phonsa, "Image Classification Using CNN," SSRN Electron. J., no. Icicc, pp. 1–5, 2021.

17 A. Peryanto, A. Yudhana, and R. Umar, "Klasifikasi Citra Menggunakan Convolutional Neural Network dan K Fold Cross Validation," J. Appl. Informatics Comput., vol. 4, no. 1, pp. 45–51, 2020.

[18] E. N. Arrofiqoh and H. Harintaka, "Implementasi Metode Convolutional Neural Network Untuk Klasifikasi Tanaman Pada Citra Resolusi Tinggi," Geomatika, vol. 24, no. 2, p. 61, 2018.

[19] A. A. Elsharif, I. M. Dheir, and S. S. Mettleq, Alaa Soliman Abu Abu-Naser, "Potato Classification Using Deep Learning Classification using Deep Learning," Int. J. Acad. Pedagog. Res., vol. 3, no. 12, pp. 1–8, 2019.

[20] R. Widodo *et al.*, "Segmentation of Lung Cancer Image Based on," *J. Ilm. KURSOR*, vol. 12, no. 1, pp. 41–48, 2023.

[21] I. P. B. Wira Brata and I. D. M. B. Atmaja Darmawan, "Neural Network Back-Propagation for Programmers," *J. Ilm. KURSOR*, vol. 11, no. 2, pp. 1–8, 2013.

[22] I. Fahruzi, "Sleep Disorder Identification From Single Lead Ecg By Improving Hyperparameters of 1D-Cnn," *J. Ilm. KURSOR*, vol. 11, no. 4, pp. 157–164, 2022.

[23] S. Divya Meena and L. Agilandeeswari, "An Efficient Framework for Animal Breeds Classification Using Semi-Supervised Learning and Multi-Part Convolutional Neural Network (MP-

CNN)," IEEE Access, vol. 7, pp. 151783–151802, 2019.

[24] P. Zeng, "Research on Similar Animal Classification Based on CNN Algorithm," J. Phys. Conf. Ser., vol. 2132, no. 1, 2021.

[25] W. Xu, X. Zhang, L. Yao, W. Xue, and B. Wei, "A multi-view CNN-based acoustic classification system for automatic animal species identification," Ad Hoc Networks, vol. 102, p. 102115, 2020.

.