UDC 004.89, 004.912

K. Khotin, V. Shymkovych, P. Kravets, A. Novatsky, L. Shymkovych

CONVOLUTIONAL NEURAL NETWORK FOR DOG BREED RECOGNITION SYSTEM

Abstract: In this article a dataset with data augmentation for neural network training and convolutional neural network model for dog breed recognition system has been developed. Neural network model architecture using transfer learning to improve classification results was developed. Neural network based on the MobileNetV3-Large architecture. The structure of the dataset has analyzed and decision to use different methods for normalizes data. A large dataset containing 70 distinct categories of dog breeds was collected and balanced through the use of data augmentation techniques. Data augmentation enabled the reduction of the disparity between the minimum and maximum number of instances by eliminating redundant images and adding essential ones. The developed model was tested and the results were demonstrated. The final accuracy of the model is 96%. The result model implement in dog breed recognition system, which is based on mobile platform. The implemented application produces functionality to interact with the resulting model such as real-time process of identifying a dog's breed or from device's gallery. Further improvement the performance of the model classification quality can be achieved by expending the initial dataset or by applying other optimization methods and adjust the learning rate.

Keywords: convolutional neural networks transfer learning, dog breed recognition, data augmentation, Python, tensorflow, keras.

Introduction

Modern technologies based on artificial neural networks have a powerful potential to alter our perception and interaction with an integral part of most people's lives. Neural networks can be used in robotics, drone control, vehicle control, pattern recognition, analysis and decision-making in the Internet of things systems, spacecraft control, military equipment and other different applications in modern technology [1].

In computer vision, a series of exemplary advances have been made in several areas involving image classification, semantic segmentation, object detection, and image super-resolution reconstruction with the rapid development of deep convolutional neural network (CNN) [2]. The CNN has superior features for autonomous learning and expression, and feature extraction from original input data can be realized by means of training CNN models that match practical applications [2-10].

Currently, there are approximately 500 diverse breeds of dogs worldwide, distinguished by their phenotype and genotype [11]. People are accustomed to perceiving pets as family members and caring for their needs. Developing a model based on convolutional neural networks (CNNs) for implementation in a dog breed recognition system will contribute to enhancing dog care by enabling rapid and accurate breed identification. The dog breed recognition system can be valuable in grooming services for estimate time for different procedures [12]. One more way using system is identify dog breeds in animal-shelters. It can be increase general information about dogs and encourage more people to adopt dog from shelters.

The relevance of this development is determined by the active advancement and popularity of convolutional neural networks for achieving high-precision object recognition tasks in images [13-16]. This class of neural networks uses convolutional operations and filters, thereby simplifying the model structure without compromising its effectiveness. Transfer learning is based on reuse the model and acquired knowledge instead of rebuild model, when data will change. Thereby implementing transfer learning with general model, time of specific model development decrease and its performance is improved.

In the paper, we have describe using CNN models with transfer learning base on pre-trained architecture MobileNetV3-Large in the problem of multi-class image classification.

Development of model’s dataset

The process of arranging data collection is an essential component of training the CNN model.

The article applied dog pictures from a dataset that was generated by an actual user on the Kaggle platform [17]. Kaggle is an online platform that facilitates collaboration among experts in machine learning and hosts data science competitions. The platform offers the functionality to publish and utilize datasets, generate neural network models, and engage with other users.

The dataset is called "70 Dog Breeds-Image Data Set" and consists of 70 categories of unique dog breeds [18]. Relevant data for each breed was sourced from the Internet and verified. In addition, the images underwent a duplicate check, resulting in the removal of any duplicates to enhance the dataset's quality.

This collection comprises 9346 pictures in .jpg format, which are categorized into training, validation, and test sets at the dataset level. All images are resized to a uniform dimension of 224×224 pixels and consist of three channels: red (R), green (G), blue (B).

Regrettably, this dataset has imbalanced classes, as depicted in Figure 1.The yellow dashed line shows the mean value of the number of instances for the classes, and the orange line represents the maximum deviation of the data.

According to the graph, it is evident that the categories "Shih-Tzu" and "Lhasa" have the largest number of specimens, namely 198 and 187, respectively. However, the categories such as "American Hairless" and "Yorkie" contained only 60 and 78 specimens, respectively. In the future, the imbalance of data could potentially reduce the accuracy of predictions. Reducing this imbalance is crucial for enhancing the performance of the neural network model trained on this dataset.

Image augmentation is a viable approach for solving the issue of imbalanced data. This technique enables the augmentation of the current dataset by generating new images with modifications based on the existing examples. Data augmentation involves using modified techniques such as adjusting brightness and contrast, rotating, flipping horizontally and vertically, adding noise, and so on [19].

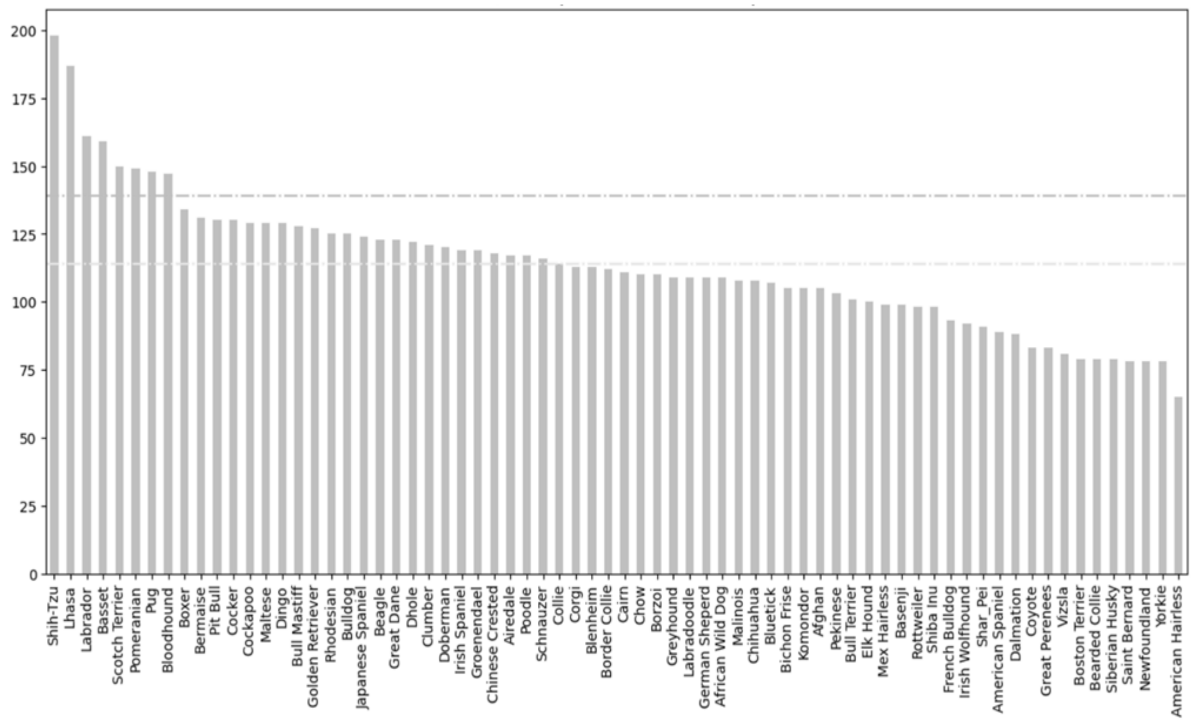

Figure 1. The distribution of images by categories

The beginning analysis of the data set indicated that the distribution of class labels is significantly imbalanced. We determined which categories should include images and which should randomly remove images by computing the mean and the maximum deviation. The research project employed a variety of image transformation techniques, including randomly selecting a delta value to adjust brightness, randomly flipping the image both vertically and horizontally, setting lower and upper limits for the contrast ratio to produce random changes in contrast, and arbitrary modifying the image quality. The variation in image quality is caused by the specification of the minimum and maximum encoding quality values of the jpeg format within the range of [0, 100], with the condition that the minimum value is less than the maximum value. We offer images that closely resemble real-life examples for the model to learn from. After the data augmentation process, a repeated analysis is necessary to validate the distribution of class labels. The results are illustrated in Fig. 2.

The diagram shows that the largest category has a maximum of 160 instances, which is lower than the initial dataset of 198 images. The minimum number of instances is 98, compared to the initial state of 68. The yellow dashed line represents the initial mean of the number of images in each class, and the orange line represents the maximum deviation.

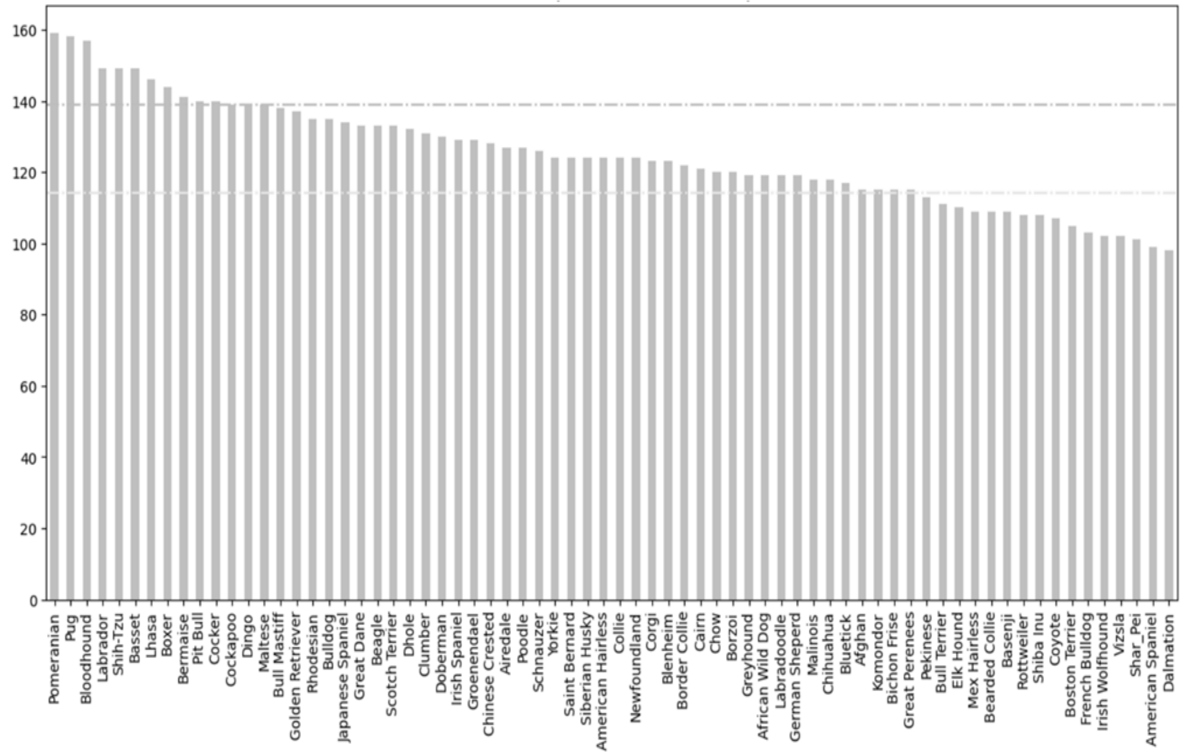

Figure 2. The distribution of images by categories

According to the diagram shown, two-thirds of the total amount of data is within the permissible value, which indicates an increase in the uniformity of the distribution of class labels for the existing data. As a result, we achieved data balance by eliminating redundant images and incorporating the transformed images. The data pre-processing is now finished. This process of data balancing and pre-processing is crucial for ensuring accurate and reliable results in future analyses or applications.

Development of CNN

Convolutional neural networks are the preferred choice for image classification tasks due to their proven high performance in various research papers [20-25]. The CNN typically contains five basic types of layers: input layers, convolutional layers, pooling layers, fully-connected layers and output layers [25]. We used pre-trained convolutional neural network architecture for the purpose of implementing transfer learning in our case.

 ISSN 1560-8956

As mentioned before, it is recommended to use the method of transfer learning to speed up the learning process and solve the issue when there is a limited quantity of data available. Transfer learning is the concept that the knowledge gained from training on a large data set can be applied to a different data set. Transfer learning involves utilizing pre-trained models that have been trained on certain tasks, such as image classification on ImageNet [26].

After the weights of the base model are loaded, the final layers involved in making predictions are adjusted. So, amodel emerges that exhibits rapid learning and achieves a high level of prediction accuracy [15].

The MobileNetV3-Large model, one of the known architectures of convolutional neural networks, has been used as a pre-trained network. Compared to other models, this model requires fewer computing resources and less time for training due to its smaller number of parameters. It is specifically optimized for mobile devices and demonstrates sufficient accuracy in predicting results [27, 28].

The detailed structure of the final model is shown in Fig. 3.

We set the input image size to 224x224x3, initialize the model weights using the ImageNet dataset [25], signal the use of an averaging pooling layer for the convolutional layer, and specify that the output layer should not participate in the classification before loading the base model.

The customized model structure consists of an input layer that defines the input image's dimensions. The input images are then processed by resizing and scaling them. Subsequently, data augmentation techniques are applied to effectively expand the dataset. To generate various variations of images, the following methods are used: random horizontal and vertical orientation, random scaling, and rotation to a randomly selected angle. After image processing and at the output of the base model layers, two fully connected layers are implemented: one with dimensions of 512×1 and another with dimensions of 256×1. Both layers apply the Rectified Linear Unit (ReLU) activation function. Dropout layers are added between the classification layers to prevent the issue of overfitting. At each epoch, a random selection process with a probability of 0.3 selects neurons for training. Neurons that have received more training are assigned greater weight. The model uses a fully connected layer with 70 neurons and a softmax activation function at its output to classify images into 70 unique class labels.

The categorical cross-entropy loss function was chosen for this model due to the presence of 70 class labels in the dataset. This selection is suitable for addressing the multiple class classification problems, where a unitary code labels the data. Categorical cross-entropy is a method that operates on the model's output values, which indicate the likelihood of the input data belonging to various classes.

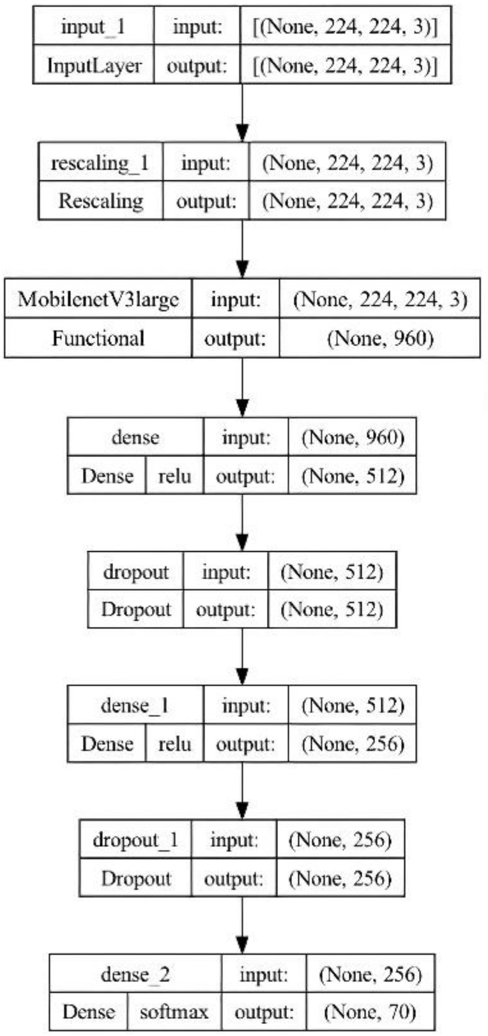

Figure 3. The structure of model

Description of application implementation

The dog breed recognition system consists of two components: a mobile application that uses a trained model to identify a dog's breed, and a server component that grants access to full details about the identified breed. Using the functionality of a mobile application to engage with the classifier is an effective approach, as demonstrated in works [29, 30].

The dog breed classification model was developed using Python 3.9 programming language in the Jupyter Notebook environment. The analysis of the input images and the development of a neural network became easier by using the following libraries and frameworks: Matplotlib, TensorFlow, Keras, and coremltools.

The mobile application component is developed using the Swift programming language and uses frameworks such as SwiftUI, Vision, and Core ML. The mobile application component requests permission to use the built-in camera. After confirmation, there are two alternatives to identifying the dog's breed: either in real-time or by using the photos stored in the device's gallery.

In real-time recognition mode, a sequence of three operations occurs. During the cycle, the camera is directed towards a particular object, and the mobile application receives a continuous flow of images. At specific time intervals, the application applies a classifier model to these images in order to maximize the efficiency of the device's computing resources. The last step of the cycle is to display the predicted breed label on the main screen. Next, the user has the option to submit a request for full details regarding the most recent predicted class label.

An alternative is to select a recognition mode from a photo. The mobile application responds by presenting the user with the device's accessible gallery. The process of predicting the corresponding breed class label after choosing the desired image is similar to the previous mode. At the end of the operation, the mobile application sends a request to the server for detailed information according to the provided class label.

The server component is developed using Python programming language and uses CGI scripts. The component sends a request to a third-party server that stores data on the age, weight, behavior, and care of each breed, and then transfers this information to the mobile application. The mobile application contains extensive information and is capable of presenting it to the user.

Results of work

The learning outcomes of the developed CNN model with transfer learning technique are as follows:

1. Accuracy of image classification on the test set: 96%;
2. The number of epochs of learning the neural network – 21.

Consider an example of determining the breed of a dog on a sample of 12 images in Figure 4.

The correct prediction of the breed is marked in green, the wrong one in red.

The mobile application is running on iOS devices. Fig. 5 represents the main screen of the mobile application, called Doggy.

Furthermore, the real-time process of identifying a dog's breed occurs on this screen. The recognition process begins on the main screen. The user simply needs to aim the phone at the desired object in order to ascertain the breed of the dog.

The application logic is designed to avoid recognizing each image from the input stream. This method minimizes the strain on computational resources and eliminates the flickering of the dog breed's name.

Figure 4. The result of image label prediction

Figure 5. The main screen of Doggy

To learn more about the breed identified in the image, click on the text block. This will redirect them to a screen displaying the recognition result and a detailed description of the breed. This screen is shown in Fig. 6.

By clicking on the gallery icon and choosing a photo, the detailed information screen will be loaded, providing the sole means of determining the dog's breed.

The top part of the mobile screen displays the provided input image, the name of the requested breed, and the probability expressed in percentage format. The second component consists of a compilation of data obtained from the server, such as lifespan, weights, heights, grooming, barking, playfulness, drooling, etc.

Conclusion

In this paper, a convolutional neural network based on the MobileNetV3-Large architecture was developed to solve the problem of classification of dog’s breed. The transfer learning technique was applied to modify the model to the particular requirements of the task, in addition to speed up the learning process.

A large dataset containing 70 distinct categories of dog breeds was collected and balanced through the use of data augmentation techniques. Data augmentation enabled the reduction of the disparity between the minimum and maximum number of instances by eliminating redundant images and adding essential ones.

The resulting model demonstrated high accuracy in classifying. The implemented application produces functionality to interact with the resulting model such as real-time process of identifying a dog's breed or from device's gallery.

Further improvement of the classification quality of CNN can be done by collecting a larger dataset and researching ways to reduce the load on the limited computing resources of mobile devices.

REFERENCES

1. *Shymkovych V., Telenyk S., Kravets P.* (2021) Hardware implementation of radial-basis neural networks with Gaussian activation functions on FPGA. Neural Computing and Applications. vol. 33, no. 15, pp. 9467-9479. https://doi.org/10.1007/ s00521-021-05706-3
2. *Zhao, X., Wang, L., Zhang, Y., Han, X., Deveci, M., & Parmar, M.* (2024). A review of convolutional neural networks in computer vision. Artificial Intelligence Review, 57(4), 1-43. https://doi.org/10.1007/s10462-024-10721-6
3. *Singh, T., & Vishwakarma, D. K.* (2019). Video benchmarks of human action datasets: a review. Artificial Intelligence Review, 52, 1107-1154. https://doi.org/10.1007/s10462-018-9651-1
4. *Singh, T., & Vishwakarma, D. K.* (2021). A deeply coupled ConvNet for human activity recognition using dynamic and RGB images. Neural Computing and Applications, 33(1), 469-485. https://doi.org/10.1007/s00521-020-05018-y
5. *Bezliudnyi Y., Shymkovych V., Kravets P., Novatsky A., Shymkovych L.* Pro-russian propaganda recognition and analytics system based on text classification model and statistical data processing methods. Адаптивні системи автоматичного управління: міжвідомчий науково-технічний збірник. 2023. № 1 (42), с. 15-31. https://doi.org/10.20535/1560-8956.42.2023.278923

6. *Kobchenko, V.R., Shymkovysh, V.M., Kravets, P.I., Novatskyi, A.O., Shymkovysh, L.L., & Doroshenko, A.Y.* (2024). An intelligent chatbot for evaluating the emotional colouring of a message and responding accordingly. *PROBLEMS IN PROGRAMMING*, (1), 23-29. http://doi.org/10.15407/pp2024.01.23

7. *Ma, P., Li, C., Rahaman, M. M., Yao, Y., Zhang, J., Zou, S., & et al.* (2023). A state-of-the-art survey of object detection techniques in microorganism image analysis: from classical methods to deep learning approaches. Artificial Intelligence Review, 56(2), 1627-1698. https://doi.org/10.1007/s10462-022-10209-1

8. *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. https://doi.org/10.1109/TNNLS.2021.3084827

9. *Hu, K., Jin, J., Zheng, F., Weng, L., & Ding, Y.* (2023). Overview of behavior recognition based on deep learning. Artificial intelligence review, 56(3), 1833-1865. https://doi.org/10.1007/s10462-022-10210-8

10. *Kravets, P., Novatskyi, A., Shymkovych, V., Rudakova, A., Lebedenko, Y., Rudakova, H.* Neural Network Model for Laboratory Stand Control System Controller with Parallel Mechanisms. Lecture Notes on Data Engineering and Communications Technologies. Springer, Cham. 2023. Vol 181. pp. 47-58 https://doi.org/10.1007/978-3-031-36118-0_5

11. *Schoenebeck, J. J., & Ostrander, E. A.* (2014, October 11). Insights into Morphology and Disease from the Dog Genome Project. Annual Review of Cell and Developmental Biology, 30(1), 535–560. https://doi.org/10.1146/annurev-cellbio-100913-012927

12. *Ling Shao, Fan Zhu, & Xuelong Li.* (2015, May). Transfer Learning for Visual Categorization: A Survey. IEEE Transactions on Neural Networks and Learning Systems, 26(5), 1019–1034. https://doi.org/10.1109/tnnls.2014.2330900

13. *Ribani, R., & Marengoni, M.* (2019, October). A Survey of Transfer Learning for Convolutional Neural Networks. 2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T). https://doi.org/10.1109/sibgrapi-t.2019.00010

14. *Kaya, A., Keceli, A. S., Catal, C., Yalic, H. Y., Temucin, H., & Tekinerdogan, B.* (2019, March). Analysis of transfer learning for deep neural network based plant classification models. Computers and Electronics in Agriculture, 158, 20–29. https://doi.org/10.1016/j.compag.2019.01.041

15. *Thenmozhi, K., & Srinivasulu Reddy, U.* (2019, September). Crop pest classification based on deep convolutional neural network and transfer learning. Computers and Electronics in Agriculture, 164, 104906. https://doi.org/10.1016/j.compag.2019.104906

16. *Alzubaidi, L., Al-Shamma, O., Fadhel, M. A., Farhan, L., Zhang, J., & Duan, Y.* (2020, March 6). Optimizing the Performance of Breast Cancer Classification by Employing

the Same Domain Transfer Learning from Hybrid Deep Convolutional Neural Network Model. Electronics, 9(3), 445. https://doi.org/10.3390/electronics9030445

17. Kaggle: Your Machine Learning and Data Science Community. (n.d.). https://www.kaggle.com/

18. 70 Dog Breeds-Image Data Set. (2021, July 1). Kaggle. https://www.kaggle.com/datasets/gpiosenka/70-dog-breedsimage-data-set/data

19. *Yang, S., Guo, S., Zhao, J., & Shen, F.* (2024, April). Investigating the effectiveness of data augmentation from similarity and diversity: An empirical study. Pattern Recognition, 148, 110204. https://doi.org/10.1016/j.patcog.2023.110204

20. *Christopher, M. V., Wahid, A., Nabiilah, G. Z., & Rojali.* (2023). Comparing Age Estimation with CNN and EfficientNetV2B1. Procedia Computer Science, 227, 415–421. https://doi.org/10.1016/j.procs.2023.10.541

21. *Abu Jwade, S., Guzzomi, A., & Mian, A.* (2019, December). On farm automatic sheep breed classification using deep learning. Computers and Electronics in Agriculture, 167, 105055. https://doi.org/10.1016/j.compag.2019.105055

22. *Bezliudnyi, Y., Shymkovysh, V., & Doroshenko, A.* (2021). Convolutional neural network model and software for classification of typical pests. PROBLEMS IN PROGRAMMING, 4, 095–102. https://doi.org/10.15407/pp2021.04.095

23. *Bhatt, D., Patel, C., Talsania, H., Patel, J., Vaghela, R., Pandya, S., & et al* (2021). CNN variants for computer vision: History, architecture, application, challenges and future scope. Electronics, 10(20), 2470. https://doi.org/10.3390/electronics10202470

24. *Hryhorenko, Y., Shymkovysh, V., Kravets, P., Novatskyi, A., Shymkovysh, L., & Doroshenko, A.* (2023). A Convolutional Neural Network Model and Software Tool for Classifying the Presence of a Medical Mask on a Human Face. PROBLEMS IN PROGRAMMING, 2, 59–66. https://doi.org/10.15407/pp2023.02.059

25. *Guo, T., Dong, J., Li, H., & Gao, Y.* (2017, March). Simple convolutional neural network on image classification. 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA) https://doi.org/10.1109/icbda.2017.8078730

26. *Deng, J., Dong, W., Socher, R., Li, L. J., Kai Li, & Li Fei-Fei.* (2009, June). ImageNet: A large-scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition. https://doi.org/10.1109/cvpr.2009.5206848

27. *Wang, B., & Rezaei sofla, A.* (2023, November). Solution for sports image classification using modified MobileNetV3 optimized by modified battle royal optimization algorithm. Heliyon, 9(11), e21603. https://doi.org/10.1016/j.heliyon.2023.e21603

28. *Howard, Andrew & Zhu, Menglong & Chen, Bo & Kalenichenko, Dmitry & Wang, Weijun & Weyand, Tobias & Andreetto, Marco & Adam, Hartwig.* (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. https://doi.org/10.48550/arXiv.1704.04861

29. *Pannurat, N., Eiamsaard, K., Suthanma, C., & Banharnsakun, A.* (2023, September). Machine learning techniques for supporting dog grooming services. Results in Control and Optimization, 12, 100273. https://doi.org/10.1016/j.rico.2023.100273

30. *Ramadhan, A. T., & Setiawan, A.* (2023, June 28). Catbreedsnet: An Android Application for Cat Breed Classification Using Convolutional Neural Networks. Jurnal Online Informatika, 8(1), 52–60. https://doi.org/10.15575/join.v8i1.1007