Acta Scientiarum
http://periodicos.uem.br/ojs/acta
ISSN on-line: 1807-8672
Doi: 10.4025/actascitechnol.v47i1.70512

Convolutional neural network applied to the classification of bird species

Claudio José Biazus[1*], Vagner Cavarzere[2] and Glauco Vieira Miranda[3]

[1]Ciência da Computação, Universidade Tecnológica Federal do Paraná, Prolongamento da Rua Cerejeira, s/n, São Luiz, 85892-000, Santa Helena, Paraná, Brazil. [2]Departamento de Biodiversidade e Bioestatística, Universidade Estadual Paulista, Botucatu, São Paulo, Brazil. [3]Universidade Tecnológica Federal do Paraná, Santa Helena, Paraná, Brazil. *Author for correspondence. E-mail: cbiazus@utfpr.edu.br

ABSTRACT. The objective of this work is to design a classification system using a Convolutional Neural Network (CNN) applied to the classification of bird species. The classes of species used in this classifier are *Ardea alba*, *Butorides striata* and *Dendrocygna viduata*. The dataset used is composed of 6,000 samples of color images, being divided into two sets, one for training and the other for testing. CNN's architecture consists of 5 layers of Convolutional and 5 layers of MaxPooling interspersed respectively, in addition to a Flatten layer and a Fully Connected layer. The results obtained by the successful classifier system can be visualized through the confusion matrix, for the three species. Likewise, the cross-validation performance measure for the classifier system corresponds to an average accuracy value of approximately 94% of the test images. It conclude that the classifier system behaved appropriately.

Keywords: architecture; dataset; machine learning.

Received on November **24**, 2023.
Accepted on March 15, 2024.

Introduction

Although DNA tests have some revelations, in many cases, they have confirmed scientists' suspicion; that is, there are between 25 and 30 orders of birds, containing approximately 148 families (Hoyo, 2020). Most of our current knowledge regarding the classification process of birds is from deoxyribonucleic acid (DNA) studies, which can be used to determine the relationships among them with much greater precision than the observation of their morphological characteristics (Schmidt, Hoban, Hunter, Paz-Vinas, Garroway, 2023; Brusatte, O'Conner, & Jarvis, 2015).

The classification of bird species is considered a difficult problem that limits visual abilities both for humans and for classifier systems (Alswaitti et al., 2022; Zhang et al., 2022). Based on their similarities, they can be classified into several categories. The order with the highest species richness is Passeriformes, which includes more than 5,000 species, that is, more than half of all known species (Brusatte, O'Conner, & Jarvis, 2015).

In recent years, the use of machine learning with deep learning has become popular in several areas of pattern recognition, as we can observe in published works (e.g., in (Bengio, Lecun, & Hinton, 2021; Bharadiya, 2023; Toofanee et al., 2023; Li, Hao, & Lei, 2016; LeCun, Bengio, & Hinton, 2015). Various applications in computer science intertwine with computational vision systems and pattern recognition (Weiss, Khoshgoftaar, & Wang, 2016). These applications include object tracking (Ciaparrone et al., 2020), pose estimation (Chen, Feng, & Wu, 2023), text detection and recognition (Cong & Zhou, 2023) and recognition of actions (Krichen, 2023). Additionally, there is a recognition system for bird species with different classes and hundreds or thousands of images (Qiu et al., 2022; Wang et al., 2023).

In this category of systems, some applications may present similarities with other classifier systems. For example, both have a dataset comprising image data, which are pre-classified into classes through the attribution of their labels and are used for both training and testing processes and validation processes (Hinton, 2022; Agarwal et al., 2021; Tavares, 2022). Similarly, all images undergo segmentation and feature extraction (Gonzales & Woods, 2018; Fieguth, 2022). Thus, despite these differences, many similarities can be used in different classification algorithms, such as color, appearance, and shape (Salehin & Kang, 2023; Singh, Goyal, & Chandel, 2022).

A collection of a set of data or images is necessary (Bharadiya, 2023) as they can be used for the training process and can contribute to the construction of the classification model (Cong & Zhou, 2023). Thus, Cong and Zhou, (2023) present a CNN model that consists of 8 layers for a classification system of 30 different fruit classes. However, the classification difficulty arises because discriminative features are located not only in the foreground object but also in important parts of an object (e.g., the head, beak, or background) (Yuxiang, Quanzhi, Xidao, Jie, & Jiahui, 2023). This becomes an interesting topic to investigate, with broad and several important applications (Kiranyaz et al., 2021).

This study addresses the process of classifying bird species using a convolutional neural network (CNN). This classification system involves classifying three species of birds (*Ardea alba*, *Butorides striata*, and *Dendrocygna viduata*) that are genetically distinct enough to be considered undoubtedly species, based on their morphology and genetics. The methods used for the classification process were obtained by extracting characteristics from the colors, shapes, and textures of the images.

Recently, numerous studies related to the identification and classification of bird species have been published. Various methods are used for the identification and classification of bird species. One method is the extraction of features from a dataset for the species identification or classification stage. These features can be extracted from audio data or images (Stowell, Wood, Pamuła, Stylianou, & Glotin, 2021; Jakaria & Pardede, 2022). Thus, only a few studies published in the literature have used images to extract species identification or classification. Similarly, there are a growing number of studies using the most varied CNN techniques (Bharadiya, 2023; Zhao, Feng, Wu, & Yan, 2017).

Niemi and Tanttu (2018) presented an automatic bird identification system that uses deep learning. The purpose of this system is to detect birds during flight, but it also provides parameters for bird identification. Manna et al., (2023) presented a system for detecting moving birds in agricultural environments using image processing and neural networks. The method applied comprises subtracting one image from another. Thus, moving objects were detected. Similarly, Huang and Basanta (2019) presented a mobile application for retrieving and recognizing bird images using a deep learning approach. In contrast, Das and Kumar (2018) presented a bird species classification system using transfer learning with training in several stages.

In this article, we present a brief introduction, which describes the state of the art, the Materials and Methods, with the classification system model, the description of the dataset and the proposed architecture, the results and discussion, and finally the conclusion.

Material and methods

Image features

For each of the highlighted classes, 400 color images were acquired, containing the format (120, 180, 3), that is, “lines, columns, and channels”, respectively. A data augmentation process was applied to this set of image samples to expand their size. Data augmentation is considered an easy and common method in CNN applications, applied to reduce overfitting in image data, and at the same time, artificially enlarge the dataset, using transformations that preserve its characteristics and labels, without causing damage, to the results obtained by Toofanee et al. (2023) and Maharana, Mondal, & Nemade, (2022). This procedure was performed due to the difficulty encountered in collecting a large number of images of the same species. Thus, four new images were generated from an original image. New images were generated randomly using one of the following transformation techniques: rotation, translation, zoom, resizing, displacement, and filling. Thus, it was possible to obtain a dataset with 2,000 samples of each species, totaling 6,000 samples. Of which 1,200 are original images and another 4,800 are artificially generated. These images were used for both the training and testing processes and for the validation process. Here it is important to highlight that the separation of images from the training and test sets occurred on request and before the data augmentation procedure. To make sure that it cannot happen that the same image belongs to both sets.

When choosing the 400 samples of the original images of each species, a pre-classification process was applied. Thus, we chose only images that presented a certain standard of quality and visibility. That is, for classes *Ardea alba* and *Butorides striata*, we chose images that presented only a single bird in an image, and for class *Dendrocygna viduata*, up to two birds in the same image, which is justified by the behavior of this species living in pairs most of the time. Moreover, birds are as visible as possible in the image, which can be in different environments in nature. Regarding the choice of images, we can emphasize that they rest on branches of plants, partially hidden by the bushes and leaves, in flight, or even feeding in the surrounding environment. Furthermore, for each of the specific classes, we can find significant similarities between the images, that is, very similar appearance, colors, and appearance.

Dataset description

Many datasets contain a wide variety of images with different classes, including (Alswaitti et al., 2022; Smelyakov, Honchar, Bohomolov, & Chupryna, 2022; Yu, Liu, & Wang, 2023; Jange, Kattimani, & Patil, 2022; Lin et al., 2022). Thus, given the variety of classes in the available datasets, it is much easier to limit our dataset to the classes that we intend to classify. Therefore, the best solution was to build a dataset. However, we noticed that computer vision algorithms still have difficulties in the classification process, especially when the appearance of the classes is not well-defined or there is a composition of images (Jange, Kattimani, & Patil, 2022).

The training set comprises 4,800 samples, that is, 80% of the dataset images. The set of test samples comprised 1,200 images and was used to test the validity of the CNN after training. This corresponds to 20% of the images in the dataset. Also, the set of test images is new to the network. The choice of which images are part of which set, training, or testing is performed randomly. Here, note that a label is associated with each image that corresponds to the class where the image belongs. It is also important to note that each image belongs to only a single class among the available species. Table 1 presents an example of a dataset containing samples of the images used.

The diversity of the images collected, as well as their appearances, can be observed. In this dataset, the position of the birds for each set of classes alternates, as well as the landscape where they are inserted, and these birds can be in the most diverse environments.

Table 1. Example of the dataset containing some of the image samples of the three species used.

Species	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5	Sample N
Ardea alba						
Butorides striata						
Dendrocygna viduata						

CNN architecture model

The random way to insert the images in this network follows the principle that the image keeps the label corresponding to its class. To ensure the success of the training and testing processes, certain configuration parameters should be adjusted. These are based on the effectiveness presented in the literature (Bai, 2022; Hinton, 2022; Salehin & Kang, 2023). Therefore, the following parameters are highlighted: batch_size, num_classes, nb_epoch, img_rows, img_cols, img_channels, nb_filters, nb_pool, and nb_conv. Specifically, batch_size represents the batch size used, which is 32; num_classes implies the number of classes used in the classification process of the three species, for this work three classes are used, and represents the number of neurons in the softmax layer used in classification; nb_epoch corresponds to the number of epochs that the network will be trained; for this network, we used 30 epochs; img_rows, img_cols, and img_channels represent the image size as well as the number of channels used; for this specific case, there are three channels because we are using color images, that is, the format (120, 180, 3); nb_filters correspond to the filter size for the first layer of the CNN network; in this case, a value of 512 is used, which is considered large so that it is possible to acquire the largest number of features in the first layer; and nb_pool and nb_conv represent the size of the mask to be used. This set of parameters was used to obtain a better adjustment of the network. Several CNN network architecture configurations are found in the literature, so an adjustment is necessary to adapt to the problem in question. Figure 1 shows the design of the layered architecture of the developed CNN network.

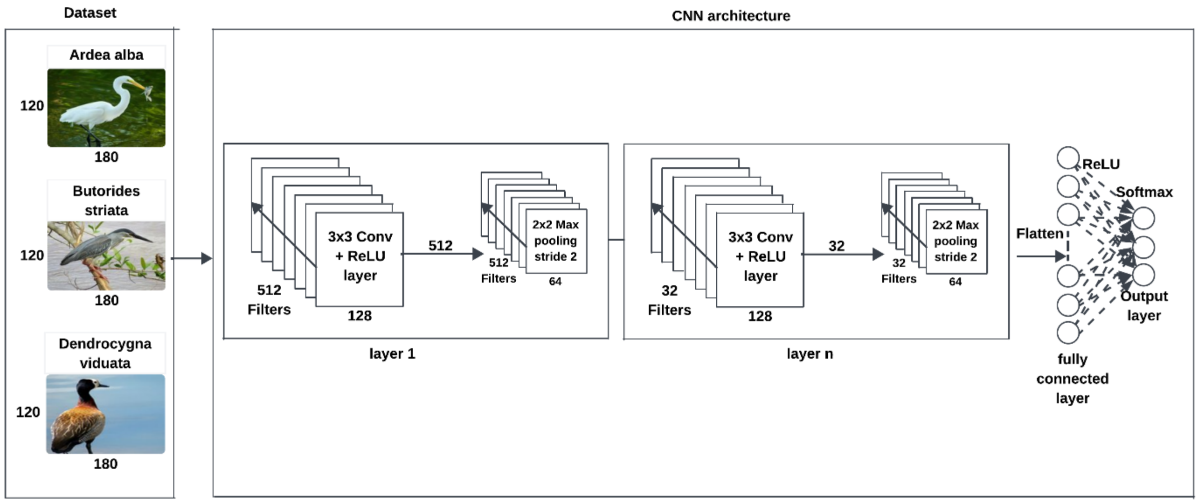

Figure 1. Layered architecture design of the CNN network.

This architecture is composed of several layers, as shown in Figure 1, and only a few layers can be observed. Among these, it is possible to highlight the input layer of the images, first layer of the network, layer before the flattened layer, fully connected rectified linear unit (ReLU) layer, and output layer. Thus, we have an input image column in the CNN that represents the process of inserting the set of images. Subsequently, we present the main characteristics of the layers and their structures; that is, how our CNN architecture is designed.

First, we add a Convolutional2D layer to process the input images. The first argument passed to the Conv2D function layer is the number of output channels. For this case, we define 512 output channels that correspond to the number of output filters. The next entry corresponds to the kernel size; for this case, we define a sliding window of size (3, 3). The activation function for this layer is the ReLU with the input image parameters (Bai, 2022).

Second, we add a MaxPooling2D layer, she define the size of the pooling in the coordinates (x y), in this case, size (2, 2), which corresponds to the step size. We add another Convolutional2D layer, a MaxPooling2D layer, and a number of output channels. For this case, we have 512 output channels that correspond to the number of output filters. We add two more Convolutional2D layers, a MaxPooling2D layer, and a number of output channels. For this case, we have 128 output channels that correspond to the number of output filters. We also added another Convolutional2D layer, a MaxPooling2D layer, and a number of output channels. For this case, we have 64 output channels that correspond to the number of output filters. Finally, we add a Convolutional2D layer, a MaxPooling2D layer, and a number of output channels. For this case, we have 32 output channels that correspond to the number of output filters.

After building the network with convolutional layers and Maxpooling2D layers, we intend to apply a Flatten, or rather flatten the output of these features so that it serves as an input for the fully connected layer. The next line represents a fully connected layer using a dense layer, in the Keras tool. First, we specify the size of our architecture, so we specify 128 nodes, each of which is activated by the ReLU. The ReLU function is defined in Equation (1) (Bai, 2022; Agarap, 2018).

$$R(x) = \max(0, x) \quad (1)$$

We also added a dropout layer; for this case, the layer used a dropout of 0.25, which corresponds to dropping 25% of the neuron weights. Dropout is a regularization technique used to reduce overfitting in neural networks, avoiding complex co-adaptations in training data (Liu et al., 2023). The term “dropout” refers to dropping hidden units in a neural network (Salehin & Kang, 2023). Finally, our classification process takes place through the softmax layer or output layer, which corresponds to the size of our classes, in this specific case, the three classes.

Training and evaluating the CNN network

The CNN network was designed using TensorFlow, Keras, and Python programming language. However, the loss function “loss” was not specified; that is, to specify which compiler structure the CNN network should use. Thus, let us compile the model developed using selected loss function, optimizer and metrics, to classify the three classes.

The too provides many optimizers, thus, we use for the Loss function "keras.losses.categorical_crossentropy" and the Adam optimizer "keras.optimizers.Adadelta". Finally, we can specify the "metrics" that are calculated when the function is executed in the model.

The model training process was initiated at this stage. This can be performed once or several times. First, we must pass the training and test sets; for this specific case, we pass the sets X_train for training and Y_test for testing. The next argument is the batch size. The next command line corresponds to the number of training epochs. The verbose flag was set to a value of 1, which specified whether you wanted the information to be printed on the console during the training process.

Finally, we pass the validation and test data to the fit function; thus, the application has knowledge of what data can be used to test the metric. This can be performed more than once when the evaluate function is run on the model. Thus, once the model has undergone training, it is possible to evaluate its results.

These results are presented through the loss function, that is, the mean squared error (MSE), which determines the mean squared difference between the predicted and actual values. The loss function is defined in Equation (2).

$$\text{loss} = \frac{1}{N}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (2)$$

where y_i is the actual value and $\hat{y}_i$ is the predicted value.

The sets of samples inserted for both the training and testing processes are balanced; that is, the classes have the same proportion distribution. This implies that all classes have the same number of samples, which are randomly distributed in a similar manner. Thus, the work performed by the classifier system includes acquiring knowledge of the source or training dataset and applying this knowledge learned in the target or test dataset so that a new image not yet classified can be predicted.

Results and discussion

We present the results obtained through the CNN implementation. To better exemplify the results obtained, Figure 2 shows the loss curve for both the training and validation processes as well as the number of epochs used.

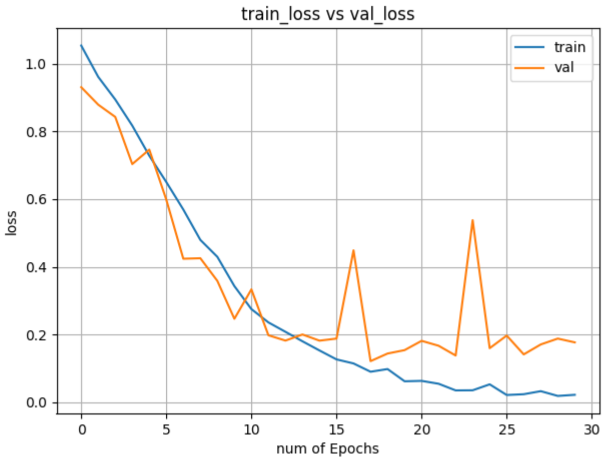

Figure 2. Plot of the curve of loss, train, val, and the number of epochs.

The Figure 3 presents the accuracy curve for both the training and validation processes. Evidently, the curves behaved appropriately. As the number of epochs increased, the network performance also improved. The accuracy is defined in Equation (3).

$$\text{Acc} = \frac{NP}{TNP} \quad (3)$$

where Acc is the accuracy of the classifier; NP, the number of correct predictions; and TNP, the total number of predictions.

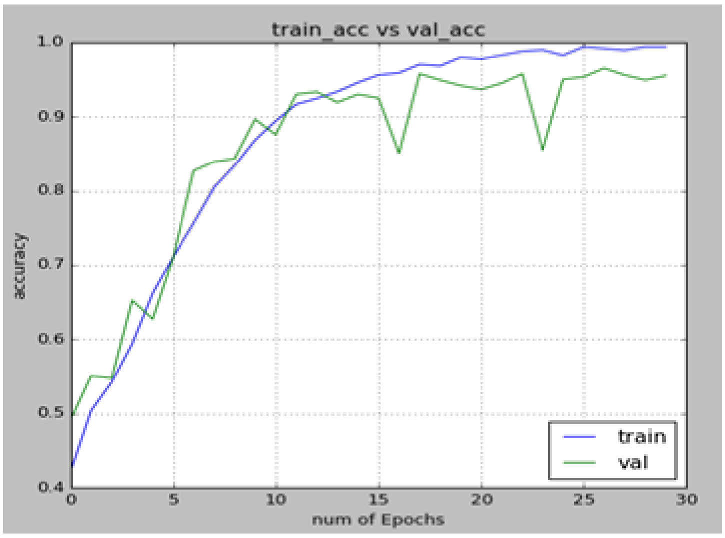

Figure 3. Plot of the accuracy curve and the number of epochs.

On the other hand, Table 2 presents the results obtained for Precision, Recall, F1-score, and Support for the confusion matrix of the three species.

Table 2. Precision, recall, F1-score and support.

	Precision	Recall	F1-score	Support
Class 0 (*A. alba*)	0.96	0.96	0.96	400
Class 1 (*B. striata*)	0.92	0.96	0.94	400
Class 2 (*D. viduata*)	0.98	0.94	0.96	400
Average / total	0.96	0.95	0.96	1.200

Precision - It corresponds to the correlation ratio of previous positive observations in relation to the total of the predicted positive observations. The question that should be answered by this metric is related to the total number of samples considered to belong to a certain class and how many samples were actually classified correctly. The higher the precision, the lower the false-positive rate.

Recall - "Sensitivity" is the proportion of correctly predicted positive observations to all the observations in the class. The question that the recall metric must answer concerns the total number of samples and how many were correctly labeled. For this metric, we must have correct values of above 50% so that it is considered adequate. As observed in Table 2, the average hit rate is 95%, which is considered optimal for the classifier.

F1-score – It represents the weighted average between the precision and recall. For this weighting process, samples representing false positives and negatives were considered. Intuitively, it is not as easy to understand as precision, but in general, the F1-score is much more useful than precision, especially if the classes do not have the same distribution, which is not the case in this application.

The results of the confusion matrix for the test set are listed in Table 3. The main diagonal elements represent the number of samples for which the predictive label is equal to the true label; that is, the samples were classified correctly. In contrast, the elements outside the main diagonal represent the samples incorrectly classified, that is, the errors made by the classification system. For example, for the case of class 0 "*A. alba*", the CNN network classified 383 samples correctly, and 17 "*A. alba*" incorrectly. Among the incorrectly classified samples, 2 are classified by the CNN network as belonging to class 1, and 15 belonging to class 2. This corresponds to 95.75% of the correct answers in this class. The class with the lowest hit rate was class 1 "*D. viduata*", with approximately 94.5% hit rate. Meanwhile, class 2 "*B. striata*" had the best hit rate of approximately 96.25%.

Table 3. Confusion matrix.

Class 0 (*A. alba*)	383	2	15
Class 1 (*B. striata*)	3	378	19
Class 2 (*D. viduata*)	11	4	385

To obtain a better estimate of our classifier, we applied a performance measure called cross-validation. Thus, a set of 10 tests was performed, as shown in Table 4, which presents the results obtained and the average of the cross-validation matrix (Koehrsen, 2018; Seraj et al., 2023).

Table 4. Comparison of the results obtained.

Test	Precision	Recall	F1-score	Support
1	0.95	0.94	0.94	1.200
2	0.94	0.94	0.94	1.200
3	0.95	0.95	0.95	1.200
4	0.93	0.93	0.93	1.200
5	0.92	0.92	0.92	1.200
6	0.95	0.95	0.95	1.200
7	0.95	0.95	0.95	1.200
8	0.93	0.94	0.93	1.200
9	0.94	0.94	0.94	1.200
10	0.93	0.93	0.93	1.200
Average / total	0.94	0.94	0.94	1.200

For the proposed CNN network model, the results correspond to an average correctness value of approximately 95.6% of the test images (Table 3), which presents the result of the confusion matrix. However, through in-depth analysis, (i.e., with the execution of a larger set of tests presented in Table 4), we concluded that the average accuracy value was approximately 94%, which is considered good for this classifier. Note that the images used for test validation were not used in the training process because we had two separate sets of samples: one each for training and testing. As shown in the confusion matrix, *Dendrocygna viduata* presents the worst classification result. This result was expected because the images of this species display one or two birds in the same sample. We believe that, to improve the results of this class, it is necessary to increase the number of samples. As shown in Figure 2 and 3, the training curves behave properly, indicating that the training and validation processes are being carried out effectively. However, the validation curves for both graphs show some peaks, which are attributed to the change in the behavior of the characteristics of the samples.

The results obtained can be visualized through the confusion matrix, Table 3, for the three species. For class 0 "*A. alba*", 383 samples were correctly classified, that is, 95.75%, and 17 samples were classified incorrectly, that is, 4.25%. For class 1 "*B. striata*", 378 samples were correctly classified, that is, 94.5%, and 22 samples were classified incorrectly, that is, 5.5%. And finally class 2 "*D. viduata*", 385 samples were correctly classified, that is, 96.25% and 15 samples incorrectly, that is, 3.75%. If possible, conclude that the classifier system behaved appropriately.

There is some difficulty in finding similar published works that classify the same classes of species, and with the application of the same technologies. Santos, Souto, Ribeiro, de Lucena, and Guzzi (2020) presents a classification of wild bird species, and contains in their analysis the same three species addressed in this work. However, the work addresses the percentage of bird populations that live in the region, not the classification. Likewise, the work of (Dario, 2021; Kantek, de Melo, Miyazaki, Castilheiro, & dos Santos Filho, 2020).

Conclusion

Machine learning is a flexible solution for classifying the systems of different bird species. However, it has received little attention in the context of a CNN. The development of this technique is fundamental for the identification of bird species, particularly in Brazil, where almost 2,000 species have already been recorded. Consequently, currently, using data generated by citizen science starts to provide more evidence in scientific studies. Therefore, the correct identification of photographic records is vital for the reliability of the generated results.

References

Agarap, A. F. (2018). Deep learning using rectified linear units. *Computer Science, Neural and Evolutionary Computing*. DOI: https://doi.org/10.48550/arXiv.1803.08375

Agarwal, R., Melnick, L., Frosst, N., Zhang, X., Lengerich, B., Caruana, R., & Hinton, G. E. (2021). Neural additive models: Interpretable machine learning with neural nets. *Advances in Neural Information Processing Systems*, *34*, 4699-4711.

Alswaitti, M., Zihao, L., Alomoush, W., Alrosan, A., & Alissa, K. (2022). Effective classification of birds' species based on transfer learning. *International Journal of Electrical and Computer Engineering (IJECE)*, *12*(4), 4172-4184. DOI: http://doi.org/10.11591/ijece.v12i4.pp4172-4184

Bai, Y. (2022). RELU-function and derived function review. In *SHS Web of Conferences* (Vol. 144, p. 02006). EDP Sciences.

Bengio, Y., Lecun, Y., & Hinton, G. (2021). Deep learning for AI. *Communications of the ACM*, *64*(7), 58-65. DOI: https://doi.org/10.1145/3448250

Bharadiya, J. (2023). Convolutional neural networks for image classification. *International Journal of Innovative Science and Research Technology*, *8*(5), 673-677. DOI: http://dx.doi.org/10.5281/zenodo.7952031

Brusatte, S. L., O'Conner, J. K., & Jarvis, E. D. (2015). The Origin and Diversification of Birds. *Current Biology*, *25*(19). DOI: https://doi.org/10.1016/j.cub.2015.08.003

Chen, H., Feng, R., Wu, S., Xu, H., Zhou, F., & L., Zhenguang (2023). 2D Human pose estimation: a survey. *Multimedia Systems* 29, 3115–3138 (2023). https://doi.org/10.1007/s00530-022-01019-0

Ciaparrone, G., Sánchez, F. L., Tabik, S., Troiano, L., Tagliaferri, R., & Herrera, F. (2020). Deep learning in vídeo multi-object tracking: A survey. *Neurocomputing*, *381*, 61-88. DOI: https://doi.org/10.1016/j.neucom.2019.11.023

Cong, S., & Zhou, Y. (2023). A review of convolutional neural network architectures and their optimizations. *Artificial Intelligence Review*, *56*(3), 1905-1969. DOI: https://doi.org/10.1007/s10462-022-10213-5

Dario, F. R. (2021). Spatial distribution and trophic structure of bird's communities of Atlantic Forest fragments in Brazil. *World News of Natural Sciences*, *35*, 1-24.

Das, S. D., & Kumar, A. (2018). Bird species classification using transfer learning with multistage training. *Springer Nature Link*, 28-38.

Fieguth, P. (2022). *An introduction to pattern recognition and machine learning*. Springer Nature. DOI: https://doi.org/10.1007/978-3-030-95995-1

Gonzales, R. C., & Woods, R. E. (2018). Digital image processing (4th ed.). London, UK: Pearson 2018.

Hinton, G. (2022). How to represent part-whole hierarchies in a neural network. *Neural Computation*, 1-40.

Hoyo, D. J., (2020). *All the Birds of the World*. Barcelona, ES: Lynx Edicions.

Huang, Y. P., & Basanta, H. (2019). Bird image retrieval and recognition using a deep learning platform. *IEEE access*, *7*, 66980-66989. DOI: https://doi.org/10.1109/ACCESS.2019.2918274

Jakaria, A., & Pardede, H. F. (2022). Comparison of classification of birds using lightweight deep convolutional neural networks. *Jurnal Elektronika dan Telekomunikasi*, *22*(2), 87-94. DOI: http://dx.doi.org/10.55981/jet.503

Jange, A., Kattimani, D., & Patil, J. (2022). Bird species identifier using convolutional neural network. *Ijraset Journal For Research in Applied Science and Engineering Technology*, *10.*

Kantek, D. L. Z., de Melo, R. C., Miyazaki, S. S., Castilheiro, W. F. F., & dos Santos Filho, M. (2020). Aves Aquáticas da Estação Ecológica de Taiamã: Variação Sazonal da Estrutura da Comunidade e a Importância das Áreas Protegidas no Pantanal. *Biodiversidade Brasileira*, *10*(3), 24-40. DOI: https://doi.org/10.37002/biodiversidadebrasileira.v10i3.1513

Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., & Inman, D. J. (2021). 1D convolutional neural networks and applications: A survey. *Mechanical systems and signal processing*, *151*, 107398. DOI: https://doi.org/10.1016/j.ymssp.2020.107398

Koehrsen, W. (2018). Overfitting vs. underfitting: a complete example. *Towards Data Science*, *405*.

Krichen, M. (2023). Convolutional neural networks: A survey. *Computers*, *12*(8), 151. DOI: https://doi.org/10.3390/computers12080151

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436-444.

Li, Y., Hao, Z., & Lei, H. (2016). A review of convolutional neural networks [J]. *Computer applications*, *36*(9), 2508-2515.

Lin, C. W., Hong, S., Lin, M., Huang, X., & Liu, J. (2022). Bird posture recognition based on target keypoints estimation in dual-task convolutional neural networks. *Ecological Indicators*, *135*, 108506. DOI: https://doi.org/10.1016/j.ecolind.2021.108506

Liu, Z., Xu, Z., Jin, J., Shen, Z., & Darrell, T. (2023). Dropout Reduces Underfitting. *Proceedings of Machine Learning Research, 202*, 22233-22248.

Maharana, K., Mondal, S., & Nemade, B. (2022). A review: Data pre-processing and data augmentation techniques. *Global Transitions Proceedings*, *3*(1), 91-99. DOI: https://doi.org/10.1016/j.gltp.2022.04.020

Manna, A., Upasani, N., Jadhav, S., Mane, R., Chaudhari, R., & Chatre, V. (2023). Bird image classification using convolutional neural network transfer learning architectures. *International Journal of Advanced Computer Science and Applications*, *14*(3). DOI: https://doi.org/10.14569/IJACSA.2023.0140397

Niemi, J., & Tanttu, J. T. (2018). Deep learning case study for automatic bird identification. *Applied sciences*, *8*(11), 2089. DOI: https://doi.org/10.3390/app8112089

Qiu, Z., Zhu, X., Liao, C., Shi, D., Kuang, Y., Li, Y., & Zhang, Y. (2022). Detection of bird species related to transmission line faults based on lightweight convolutional neural network. *IET Generation, Transmission & Distribution*, *16*(5), 869-881. DOI: https://doi.org/10.1049/gtd2.12333

Salehin, I., & Kang, D. K. (2023). A review on dropout regularization approaches for deep neural networks within the scholarly domain. *Electronics*, *12*(14), 3106. DOI: https://doi.org/10.3390/electronics12143106

Santos, F. D. C. V., Souto, W. M. S., Ribeiro, A. S. N., de Lucena, R. F. P., & Guzzi, A. (2020). Traditional knowledge and perception of birds in the Parnaíba Delta environmental protection area, Northeast Brazil. *Acta Scientiarum. Biological Sciences*, *42*, 1-12. DOI: https://doi.org/10.4025/actascibiolsci.v42i1.47722

Schmidt, C., Hoban, S., Hunter, M., Paz-Vinas, I., & Garroway, C. J. (2023). Genetic diversity and IUCN Red List status. *Conservation Biology*, *37*(4), e14064. DOI: https://doi.org/10.1111/cobi.14064

Seraj, A., Mohammadi-Khanaposhtani, M., Daneshfar, R., Naseri, M., Esmaeili, M., Baghban, A., & Eslamian, S. (2023). Cross-validation. Handbook of Hydroinformatics, (89-105). DOI: https://doi.org/10.1016/B978-0-12-821285-1.00021-X

Singh, I., Goyal, G., & Chandel, A. (2022). AlexNet architecture based convolutional neural network for toxic comments classification. *Journal of King Saud University-Computer and Information Sciences*, *34*(9), 7547-7558. DOI: https://doi.org/10.1016/j.jksuci.2022.06.007

Smelyakov, K., Honchar, Y., Bohomolov, O., & Chupryna, A. (2022). Machine Learning Models Efficiency Analysis for Image Classification Problem. In *COLINS-2022: 6th International Conference on Computational Linguistics and Intelligent Systems* (p. 942-959). Gliwice, PO.

Stowell, D., Wood, M. D., Pamuła, H., Stylianou, Y., & Glotin, H. (2019). Automatic acoustic detection of birds through deep learning: the first bird audio detection challenge. *Methods in Ecology and Evolution*, *10*(3), 368-380. DOI: https://doi.org/10.1111/2041-210X.13103

Tavares, T. F. (2022). Open-set classification approaches to automatic bird song identification: towards non-invasive wildlife monitoring in Brazilian fauna. *IEEE Latin America Transactions*, *20*(11), 2388-2394. DOI: https://doi.org/10.1109/TLA.2022.9904764

Toofanee, M. S. A., Dowlut, S., Hamroun, M., Tamine, K., Petit, V., Duong, A. K., & Sauveron, D. (2023). Dfu-siam a novel diabetic foot ulcer classification with deep learning. *IEEE Access*, *11*, 98315-98332. DOI: https://doi.org/10.1109/ACESS.2023.3312531

Wang, K., Yang, F., Chen, Z., Chen, Y., & Zhang, Y. (2023). A fine-grained bird classification method based on attention and decoupled knowledge distillation. *Animals*, *13*(2), 264. DOI: https://doi.org/10.3390/ani13020264

Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). A survey of transfer learning. *Journal of Big data*, *3*(9), 1-40.

Yu, R., Liu, S., & Wang, X. (2023). Dataset distillation: A comprehensive review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. DOI: https://doi.org/10.48550/arXiv.2301.07014

Yuxiang, X., Quanzhi, G., Xidao, L., Jie, Y., & Jiahui, Z. (2023). A survey of fine-grained visual categorization based on deep learning. *Journal of Systems Engineering and Electronics*, 1-20. DOI: https://doi.org/10.23919/JSEE.2022.000155

Zhang, W., Li, D., Min, X., Zhai, G., Guo, G., Yang, X., & Ma, K. (2022). Perceptual Attacks of No-Reference Image Quality Models with Human-in-the-Loop. *Advances in Neural Information Processing Systems*, *35*, 2916-2929. DOI: https://doi.org/10.48550/arXiv.2210.00933

Zhao, B., Feng, J., Wu, X., & Yan, S. (2017). A survey on deep learning-based fine-grained object classification and semantic segmentation. *International Journal of Automation and Computing*, *14*(2), 119-135. DOI: https://doi.org/10.1007/s11633-017-1053-3