

Birds Images Prediction with Watson Visual Recognition Services from IBM-Cloud and Conventional Neural Network

Fatima-Zahra Elbouni^{*1}, Aziza EL Ouazizi^{1,2}

¹ Laboratory of Engineering Sciences (LSI), Sidi Mohamed Ben Abdallah University, Taza, 1223, Morocco

² Laboratory of Artificial Intelligence, Data Sciences and Emergent Systems (LIASSE), Sidi Mohamed Ben Abdallah University, Fez, 30013, Morocco

ARTICLE INFO

Article history:

Received: 21 August, 2022

Accepted: 30 November, 2022

Online: 20 December, 2022

Keywords:

Bird classification

Gradient descent

Deep learning

Android app

Convolution

ABSTRACT

Bird watchers and people obsessed with raising and taming birds make a kind of motivation about our subject. It consists of the creation of an Android application called "Birds Images Predictor" which helps users to recognize nearly 210 endemic bird species in the world. The proposed solution compares the performance of the python script, which realizes a convolutional neural network (CNN), and the performance of the cloud-bound mobile application using IBM's visual recognition service to choose the platform one. android form. In the first solution we presented an architecture of a CNN model to predict bird class. While the other solution, which shows its effectiveness, is based on IBM's visual recognition service, we connect the IBM project that contains the training images with our Android Studio project using an API key, and the IBM process classifies the image captured or downloaded from the application and returns the prediction result which indicates the type of bird. Our study highlights three major advantages of the solution using IBM's visual recognition service compared to that of CNN, the first appears in the number of images used in training which is higher compared to the other and the strong distinction between bird types where images and bird positions come together in color. The second advantage is to create a trained model saved on the cloud in order to use it with each prediction the most difficult thing to do locally due to the low processing capacity of smartphones. The last advantage is reflected in the correct prediction with a certainty of 99% unlike the other solution due to the instability of the CNN model.

1 Introduction

This article is an extension of the work originally "Bird Image Recognition and Classification Using IBMCloud's Watson Visual Recognition Services and Conventional Neural Network (CNN)" presented in the conference "International Conference on Electrical Engineering, Communication and Computing (ICECCE) 2021"[1].

Birds are among the living organisms that play an important role in the life cycle of the planet. They have become a center of interest for many people such as ornithologists who are interested in researching different types of birds, curious and passionate people who wish to live closer to them, without forgetting the sanctuaries, associations or large bird stores that are interested in breeding or protecting birds from extinction. These people often need to know certain characteristics of birds such as their diet, family (or gender) and housing in order to provide them with a suitable and comfortable lifestyle. To make it easier for these people, we had the idea to

develop an intelligent bird recognition system.

Science has progressively evolved through several chronological stages. The first phase was based on an experimental paradigm, from the 17th century, which took into account the experiment, the observation of phenomena and the use of this experimentation [2]. The second phase was theoretical, it uses an idea or a hypothesis to explain an observed phenomenon, it is a way of synthesizing the acquired knowledge and inventing new experiments. In fact, there was a dynamic relationship between experimental and theoretical science since one completes the other [3]. Around the year 1995 another paradigm appeared, it is essentially interested in the production, processing and use of information after storage, it is the science of information. This latter one has been developed in an exponential way stimulating the birth of the 4th paradigm of the data oriented science which leads to significant advances in the statistical techniques and machine learning [4].

Machine learning is one of the disciplines of artificial intel-

*Corresponding Author: Fatima-Zahra ELBOUNI, Fez-Morocco, +212-616338783 fatimazahra.elbouni@usmba.ac.ma

ligence [5], this notion of intelligence defines two fundamental approaches. The first is the connectionist approach, considers intelligence as a result of all minimalist functions, which are called neurons, by connecting several neurons, which can solve more complex cases and have good results. When we talk about the connectionist approach, we say that it is inductive, we have an input and an output and from the observation we try to understand what it is in order to define one or more rules. On the other hand, there is the symbolic movement of things which affirms that intelligence is only symbols interpreted at a high level. For example when we say "Every Cauchy sequence is a convergent sequence" it means that "If U is a Cauchy sequence, then U is a convergent sequence". This approach is deductive, it deduces the result by applying one or more inference rules on the input, which leads to NP-complete problems due to the exponential complexity of the rule application algorithm.

To achieve our solution, a comparable supervised machine learning model using a convolutional neural network (CNN) [6] has been proposed. The first step is to perform a convolution [7] of raster or tensor image representation of our bird database in order to extract the main features. The second step is to perform the [7] flattening operation on the result of the convolutional layer to have a data format that can be injected into an artificial neural network (ANN). The third step is to train the trained model by back-propagating each ANN layer that are fully connected [8]. In the end, we get a well-trained model that is able to predict the type of a given bird from its image. All the steps mentioned above will be detailed in the third section.

The problem with this method is that the script takes a long time to train the model, updating the network weights each time with the gradient back-propagating algorithm that minimizes the error function, and carrying with small speed CPUs the problem becomes heavier. For this reason, we propose another method for prediction that uses IBM Cloud [9]. In fact, the visual recognition service of IBM Cloud forms the prediction classes once and saves them to exploit them when we need a prediction. This applied technique minimizes the prediction time: we don't need to train the model each time and also it gives the possibility to link the IBM Cloud project with an Android studio project in order to generate an APK installable on an Android OS smart phone, which makes the solution portable and usable.

There are other bird class prediction methods in the literature such as The "Application of Two-level Attention Models in Deep Convolutional Neural Network for Fine-grained Image Classification" [10]. This paper focuses on fine-grained classification by applying visual attention using a deep neural network, It incorporates three types of attention: bottom-up, top-down at the object level and top-down at the part level. All of these attentions are combined to form a domain-specific deep network and then used to enhance aspects of objects and their discriminative features. Bottom-up attention proposes candidate patches, top-down object-level attention selects patches relevant to a certain object, and top-down part-level attention locates discriminative parts. This pipeline provides significant improvements and achieves the best accuracy under the weakest supervision conditions compared to subsets of the ILSVRC2012 dataset and the CUB200 2011 dataset.

The article "Handcrafted features and late fusion with deep learning for bird sound classification" [11], makes a study on acoustic

features, visual features and deep learning for bird sound classification. In order to classify bird caller species a unified model built by combining convolution neural network layers, convolution layers derives the important features and reduces the dimensionality, then fully connected conventional layers for classification. Based on the experimental results on 14 bird species, this method indicates that the proposed deep learning can achieve an F1 score of 94.36 %, superior to the use of acoustic feature approach which gives a score of 88.97% and the use of visual feature approach which gives a score of 88.87%. To further improve the classification performance, they merged the three approaches: acoustic feature approach, visual feature approach and deep learning approach, to expect a final better score of 95.95%.

The authors of the paper titled "Image based Bird Species Identification using Convolutional Neural Network" [12] have developed a deep learning model to recognize 60 species of birds in order to give the possibility to birders to easily identify the type of birds. The method implements a model to extract information from bird images using the Convolutional Neural Network (CNN) algorithm. The data is gathered from Microsoft's Bing v7 image search API. The classification accuracy rate of CNN on the training set included 80% of data reached 93.19% and the accuracy rate on the test set that included on 20% of data reached just 84.91%. The experimental study was conducted on Windows 10 operating system in Atom Editor with TensorFlow library.

There are other applications on the Play Store such as "Merlin Bird ID" [13] whose purpose is to identify birds using the Photo ID pane which offers a short list of matches between photos taken or imported from photo gallery thanks to patterns trained on thousands of photos of each species. This application covers over 8000 species, has a code size of 25 MB and is trained on images downloaded from the eBird checklists and archived in the Macaulay library. This application can identify species that are currently included in the regional kits that contain the reindeer species in that region.

This article is organized as follows: in the second section we describe the approach adopted for the realization of the solution, in the third section we give a description of the method adopted during recognition, in the fourth section we will give the results obtained then we will devote the fifth section to the discussion and evaluation of the results and the last one to conclude the work and open another research track.

2 Materials and Method

2.1 IBM Watson:

Watson [14] is a cognitive artificial intelligence computer program developed by IBM. It has several sub-services, including Watson Visual Recognition Ref used primarily for visual content recognition and analysis using machine learning. The IBM platform offers choices for managing application interfaces, among which the Android Studio is chosen. After authentication on the IBM Ref Cloud, we create a project as follows:

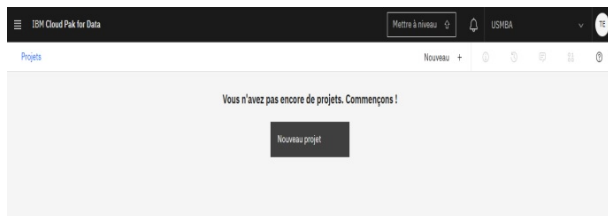


Figure 1: IBM Cloud interface to create new project

The project name and description (optional) must be defined.

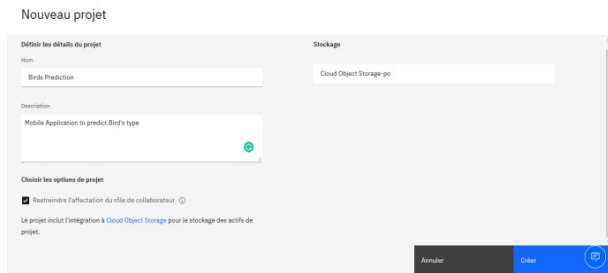


Figure 2: Project information

The next step is to choose the Visual Recognition subsystem of Watson :

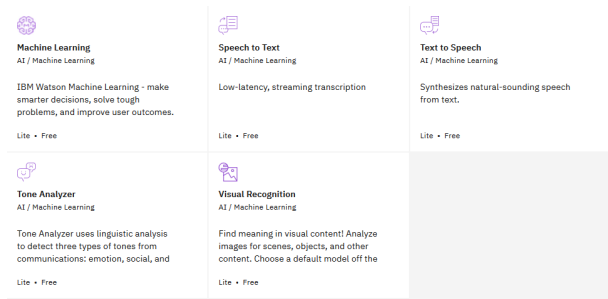


Figure 3: Creation of Visual Recognition Service

The visual recognition service helps to create a registered pattern of bird classes, from a database of images where each class must contain at least 10 different images to facilitate prediction.

On the Kaggle website [15], a dataset of training images has been uploaded which contains approximately 210 bird types, where each type contains more than 10 images.

2.2 Firebase

To store the different characteristics of each bird, including name, family, diet and housing in addition to other similar images; the external Google "Firebase" database [16] was used.

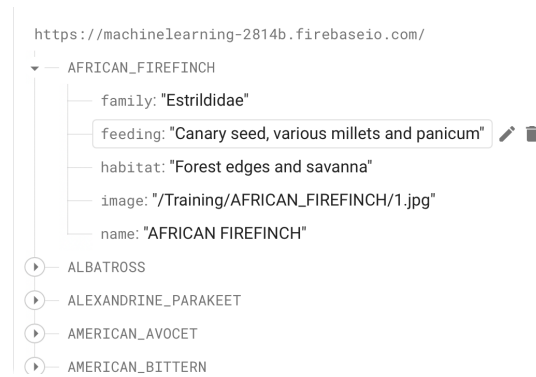


Figure 4: Realtime database of Firebase Storage

3 Recognition Method

Several innovations in various fields of science derive their operating principle from the behavior of the human body, for example fundamental computer science is inspired by the nervous system to develop models of artificial intelligence. The human brain consists of more than 80 billion living cells [17] called "neurons", interconnected with each other to transmit or process chemical or electrical signals. A biological neuron consists of the dendrites (inputs) which are branches that receive information from other neurons, a Soma or cell nucleus that processes the information received from the dendrites, an axon as a cable used by neurons to send information to the output and synapses (outputs) that connects the axon and the other neural dendrites.

By analogy, a formal model has been proposed to be integrated into so-called intelligent systems, which can react almost like humans; named artificial neuron or perceptrons [18]: it is a mathematical function that takes values into account to make a simple calculation and return one or more values as output. The calculation performed by a perceptron is broken down into two phases: a calculation by a linear function(1):

$$f(x_1, x_2, \dots, x_n) = w_1x_1 + w_2x_2 + \dots + w_nx_n \quad (1)$$

where the w_i are the weights and the x_i are the input values with $(1 \leq i \leq n)$, followed by an activation function [19] (or transfer function) φ which will decide whether the information will be transmitted to another neuron or not.

There are several activation functions that are chosen according to the type of problem to be solved, such as the Relu function defined by $f(x) = \max(0, x)$ and Softmax generally used to solve multi-classification with single-label problems. Note that we can add an additional weight b_j called bias [20] or error which measures the difference between the expected output value and the estimated output value y_j and the estimated value to have an "affine perceptron".

The juxtaposition of several neurons, grouped by layers, gives the notion of the artificial neural network [6]. Previously, the connectionist approach consists in finding a relation between the input and the output, more formally looking for a function F which connects the input to the output. In real life many types of relationships result in linear or quadratic functions, however there are more complex problems such as predicting the class of a bird which results

in a non-linear function. According to the universal approximation theorem [21], any continuous and bounded function can be approximated by a neural network with n inputs and y_i outputs, possessing a finite number of neurons in a single layer hidden with an activation function and a output whose activation function is linear.

To find the best function F (2) which connects the input to the output

$$F(X_i) \approx Y_{whith}(Y = (y_1, y_2, \dots, y_m)) \quad (2)$$

It suffices to configure the parameters (the weights $P = (w_1, w_2, \dots, w_n)$ and the bias b_j) of the network, so that the mide error is minimal. To precisely measure the performance of the approximation, we define a cost function [22]in our case Cross-entropy for categories(3) [23]:

$$E = -y_j \log(\tilde{y}_j) + (1 - y_j) \log(1 - \tilde{y}_j) \quad (3)$$

It sum of E_i the local errors, generally convex and we are looking for its minimum, so we are faced with an optimization problem that can be solved by the method of descent of the gradient [24], the latter assimilates the notion of gradient [24] which is a generalization of the derivation in the case of functions with several variables. The derivation studies the variations of a function and tries to find its extrema; similarly the gradient is a vector which gathers the partial derivations of a function with several variables and which indicates the direction of the greatest slope of the function from a point. The calculation of the gradient vector of the cost function requires the calculation of the partial derivatives of E from those of F by the formula (4):

$$gradf(x_1, \dots, x_n) = \begin{pmatrix} \frac{\partial f}{\partial x}(x_1, \dots, x_n) \\ \vdots \\ \frac{\partial f}{\partial x}(x_1, \dots, x_n) \end{pmatrix} \quad (4)$$

Back-propagation [8] is the application of gradient descent on an artificial neural network . Gradient descent is an optimization algorithm used to find the minimum of a function, based on the opposite vector [24] to the gradient vector. To find the minimum error, we start from the initial weights $W_0 = (w_0, w_1, \dots)$, the direction of the opposite vector is followed by a step δ , called the learning rate, and we start again from the new weights and stop at the end of a number of iterations fixed in advance or when the error tends to zero; according to the following recurrence formula (5)[8]:

$$P_{k+1} = P_k - \delta \times grad(E_{global}(P_k)) \quad (5)$$

There are three variants of the gradient descent, which is differentiated by the amount of data used in the calculate the gradient of the cost function. If we use the global error E_{global} , we say that there is a classical gradient descent. If we consider at each iteration a single gradient E_i instead of E_{global} , we say that there is a stochastic gradient descent, this technique reduces the amount of calculations, because the calculation of the gradient of E_{global} , involves the calculation of the gradient of each of the E_i , that is to say the partial derivation with respect to each of the weights w_j . If we divide the data into packets of size K . For each packet (called a "batch"), a gradient is calculated and an iteration is performed,

we say that there is a gradient descent in batches (or mini-batches). It is an intermediate method between the descent of the classical gradient (traversing all the data at each iteration) and the descent of the stochastic gradient (which uses only one data at each iteration), at the end of $\frac{N}{K}$ iterations, we have traversed the entire dataset: this is called an epoch.

However, these variations do not always lead to the minimum point, which requires another variation in the learning rate, where δ must be chosen neither too large nor too small: if δ is too large, the P_k points will oscillate around the minimum, but if δ is too small, the P_k points will approach the minimum only after a very long time. For example, it is possible to choose, during training, a fairly large δ_k , then smaller and smaller over the iterations; either by a linear, quadratic or exponential decrease.

The proposed model applies gradient descent Adam [25] stands for Adaptive Momentum Estimation , which is a stochastic gradient descent that combines between two methods Momentum and RMSprop. Momentum [25] is a method that helps to speed up stochastic gradient descent in the appropriate direction and dampen oscillations by adding fractions of previous gradients to the current gradient. RMSprop [25] is based on a decreasing variation of aggressive learning rate.

Algorithm 1: ADAM algorithm

Result: P_t

$m_0 \leftarrow 0$; *** Initialize the 1^{ere} moment vector

$v_0 \leftarrow 0$; *** Initialize the 2^{ere} moment vector

$t \leftarrow 0$; *** Initialize the time step

while P_t is not convergent **do**

$t \leftarrow t + 1$

$grad_t \leftarrow \nabla f(P_{t-1})$;*** The gradient of the error function at time step t

$m_t \leftarrow \beta_1 \times m_{t-1} + (1 - \beta_1) \times grad_t$; *** Update first moment biased estimate

$v_t \leftarrow \beta_2 \times v_{t-1} + (1 - \beta_2) \times grad_t^2$; *** Update biased raw second moment estimate)

$\hat{m}_t \leftarrow \frac{m_t}{(1-\beta_1^t)}$; *** First moment estimate corrected for computational bias

$\hat{v}_t \leftarrow \frac{v_t}{(1-\beta_2^t)}$; *** Second moment estimate corrected for computational bias

$P_t \leftarrow P_{t-1} - \alpha \times \frac{\hat{m}_t}{(\sqrt{\hat{v}_t} + \epsilon)}$; *** Update settings

end

Our study is focused on a convolutional neural network (CNN) , inspired by the operating principle of the visual lobe of vertebrates. It is a type of artificial neural network suitable for image recognition which is composed of two parts: a convolution part and a classification part corresponds to a classic multilayer model. Convolution [7] is a mathematical method that applies a convolutional product between an array (matrix representation of an image) and a pattern (filter or kernel), frequently used for image processing in order to bring a transformation or derive its main characteristics. This convolutional product [7] can be realized by a neural network with convolution neurons, in which also we apply the gradient descent algorithm to adjust their weights. The result of this convolution forms a feature map. The convolution part also consists of a pooling phase[7], the purpose of which is to reduce the dimensionality which aims to reduce the computational complexity while keeping

the important characteristics of the input image. There are two types of Pooling: max-Pooling and average-Pooling. The first takes the max of each of the sub-matrices (windows) while the second takes the average. The last phase of the convolution part is the flattening phase which converts the feature maps obtained after the Pooling operation into a column vector.

The result column vector will be the input of our artificial neural network which constitutes the second phase of the multilayer model, it is trained using the gradient back-propagation algorithm with two image databases of birds, one for training and another for testing.

A good learning model offers an optimal balance between bias and variance, it simultaneously minimizes these two error quantities, this problem is known as the bias–variance trade-off [20]. On the one hand, a high bias and a low variance cause underlearning[26]: lack of relevant relationships between input/output data; On the other hand, excessive data variance and low bias can lead to overfitting[26], where the model remains too specific to the training data and thus is unable to adapt to new data. To remedy this problem, we use the Dropout technique. Dropout [8] is a technique that temporarily deactivates certain neurons of a layer during training with a given probability, therefore it sets the weights to zero for both evaluation and backpropagation.

Here is a summary diagram of all the steps mentioned above of our adapted version of the RNC (go to the appendix section to see the source code in python):

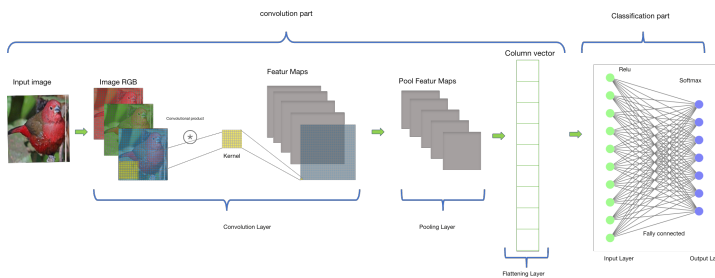


Figure 5: Descriptive schema of the applied convolutional neural network

The problem mentioned in this research is included in the family of image classification problems, so to clearly indicate to which class an image belongs, we need to minimize the gap (2) between each of the images of the input classes (training base) with the images of the same classes of the output (test base), this division of the images into two: test and training, is done to remedy the problem of bias–variance 3. First of all, each RGB image is considered as a matrix of pixels to which convolution 3 is applied to reduce its size and keep only its relevant characteristics in order to transform it into a column vector that we introduce in a Multilayer Perceptron. This operation decreases the computing power required to process the data. The column vector enters the artificial neuron network in which the backpropagation 3 is applied based on the ADAM 3 algorithm, at each iteration of the process and over a series of epochs, the model is able distinguish between dominant features and some low-level features in images and classify them using the Softmax 3 activation function. Here are the essential steps for creating the CNN model:

- Create an image generator for the training data;

- Create an image generator for the training data;
- Upload the training images;
- Create two convolutional layers;
- Create two Pooling layers;
- Create the Flattening Layer;
- Treated the over-learning problem;
- Added a Multilayer Perceptron with one hidden layer and an output layer with 128 neurons which allows the detection of 210 types of birds.

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 32)	896
conv2d_1 (Conv2D)	(None, 60, 60, 32)	9248
max_pooling2d (MaxPooling2D)	(None, 30, 30, 32)	0
max_pooling2d_1 (MaxPooling2D)	(None, 15, 15, 32)	0
flatten (Flatten)	(None, 7200)	0
dense (Dense)	(None, 128)	921728
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 210)	27090

Figure 6: Model architecture summary

4 Experimental Results and Analysis

This study is a comparison between the results of two methods of predicting the class of a bird, such as the python script that implements a convolutional neural network (CNN)[6] and the method of prediction by the IBM Cloud visual recognition service[9].

The first method of the CNN script was trained on 210 bird classes, while the second one offers the possibility to test up to 1010 classes. The latter minimises the error and prediction time, where the prediction reaches a percentage of servitude of 99% and a waiting time of less than 10 seconds. In addition, it offers the possibility to store the trained model in the cloud for later reuse or integration with a solution such as Java, Android Studio or Python. Thus, the process of learning bird images using the CNN method takes enough time for training despite being tested on a small dataset and gives a low prediction accuracy compared to the second method.

The proposed solution integrates the prediction method by IBM Cloud’s visual recognition service with an Android Studio project to generate an APK of an Android mobile application.

Here is a table of comparison between CNN model and visual recognition service from IBM Cloud :

Table 1: comparative table between CNN and VRS-IBM

Comparison criterion	CNN	VR-IBM
Number of class	210	1010
Response time	2s	5s
Accuracy	22%	no access
Loss	64%	no access
Model storage	locally	on the cloud
Integration with mobile app	TensorFlow Lite	API

Images show the instability of the CNN model:

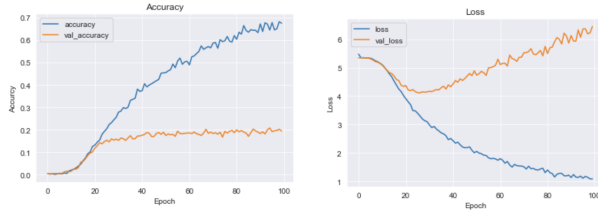


Figure 7: accuracy and loss per epoch graphs

the two graphs above indicate the variation of accuracy and loss over epochs. The first accuracy graph illustrates that the accuracy of the validation increased linearly with the accuracy, but from epoch 18 it fluctuates between values very close and away from the accuracy, this means that the model learns more information at each time; similarly for the loss graph, the validation loss shows that it decreased linearly, but after 18-19 epoch it started to increase. This means that the model tried to store the data. We also notice that the error rate is 64% and the accuracy rate is 22%, which means that the model requires a good adjustment of the hyperparameters to have good results.

The application represents in the first interface "figure 6" a list of bird images, when you press the button of the "eye" icon a detail containing the characteristics (name, family, housing, food) of this race, will be displayed in the form of a table "figure 7" as well as a small gallery of the same race to have more about this race.

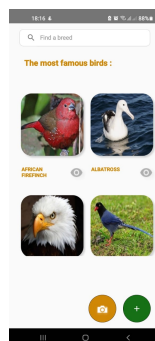


Figure 8: Home interface

Predicting the breed of a bird from its image is done by two options, the first allows you to choose an already captured image saved in the smartphone gallery as indicated by the plus "+" icon in "Figure 6", or by the second method which allows you to capture a new image instantly as illustrated by the camera icon in "Figure 6".

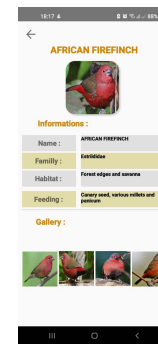


Figure 9: Detail interface

To detail the scenario, once the user uploads or captures an image of a bird, the image will be transmitted to the IBM Cloud Platform through the "ClassifyOptions" object, thereafter the visual recognition service will return the prediction result through the "ClassifiedImages" object in the form of the name of the class to which the bird belongs and the percentage of similarity. After retrieving the name of this bird a query will be sent to the FireBase database with the bird name as parameter in order to receive their characteristics and display them in a table as shown in "Figure 7".

5 Discussion

Our article presents a demonstration of three main results concerning the recognition problem: the first result is the prediction of the class of bird given in the image captured or chosen by the user, this result reflects the correct prediction with certainty 99% unlike the other solution due to CNN model instability. The second result is the presentation of major characteristics of the predicted bird such as its family, its habitat and its food with the display of a small gallery of bird images from the database used, This result illustrates the model's reliability in distinguishing between similar types in the face of the high number of images used in the training phase. The third result is the usability and easy portability of the solution on the phone in the form of an Android mobile application, the solution generates a trained model saved on the cloud in order to use it for each prediction the most difficult thing to be done locally due to the low processing capacity of smartphones. However, our study focuses on 210 classes of endemic bird species to make the exact classification in a targeted and efficient way. The proposed CNN model has been tested on training images, it returns the result with an accuracy of 22%. But due to the visual similarity between bird species, when using a new image, the model sometimes fails to distinguish between two types and ultimately leads to misclassification. On average, testing the data set gives a sensitivity of 64%. Therefore, the visual recognition service of the IBM Cloud method has been adopted to make the prediction of the class of bird species in order to have a simple, fast and efficient result. The latter applies a deep learning algorithm and gives a good numerical precision, which we bet on the success of this project, despite that it does not give access to visualize the model architecture and compare their metrics with that CNN model. have a simple, fast and effective result. The latter applies a deep learning algorithm and gives a good numerical precision, which we bet on the success of this project, de-

spite that it does not give access to visualize the model architecture and compare their metrics with that CNN model. have a simple, fast and effective result. The latter applies a deep learning algorithm and gives a good numerical precision, which we bet on the success of this project, despite that it does not give access to visualize the model architecture and compare their metrics with that CNN model.

In December 2021 the visual recognition service of Watson was obsolete [27], it is replaced by others like the machine learning service and to keep the usability of the application we proposed another solution consists in saving the trained model of the python script, after having improved the setting of the hyper-parameters, in the form of a file with extension "h5" then with this file we will generate another TensorFlow file [28] with extension "tfLite" [29] in order to integrate it with the Android studio project by replacing the part of prediction with the services by an image injection on the recorded model.

The "Merlin Bird ID" app [13] is one of the powerful apps in this field of bird species identification, it is released on play store the first time in 2014; it is powered by Visipedia, relies on artificial vision and deep learning, its last update was published in May 2022. It offers other functionality more than the features offered in our image recognition solution. "Merlin Bird ID" contains several pane among them, image recognition photo id pane to identify birds from photos by calling sets of millions of photos contributed by bird watchers in eBird.org at 1 using other user-provided information such as date and location; it brings together regional kits for example from the United States, Mexico, Central America, from Europe, North Africa and Australia. . . The Sound Id component is dedicated to audio recognition, it identifies the species from the sound and contains an audio recording library by region. The start pane id who Merlin starts by asking you a few simple questions and then magically gives you a list of birds that best match your description After choosing one you can see more photos, listen to recordings sound and read tips for identification. APK size rating our application is small with 23.64 MB than Merlin of 227 MB, so our study is limited to 210 types compared to the other which has more than 40,000 photos of birds (males, females and juveniles). Finally, Merlin is developed for different languages such as English, Spanish, Portuguese, Hebrew, German, Simplified Chinese and Traditional Chinese unlike our solution which is limited to English language only. The remark that we have noticed as users of this application that always the download and installation of regional kits either in the Photo ID or Sound ID pane fails but is it because of weak connection or technical problem in the application.

The paper "A new mobile application for agricultural pest recognition using deep learning in a cloud computing system" [30] is one of the research topics that exists in the literature in the sense of classification using a mobile application, this paper makes a comparison between the performance of two models, including the convolutional neural network (CNN) with connection hopping and the convolutional neural network without connection hopping. This research focuses on the development of an application that recognises weeds in agricultural areas, such as fields, greenhouses and large farms. The mobile application was developed by Apache Cordova, using Flask Framework to handle the HTTP POST requests sent, the python code executed and hosted in the Cloud on the Python Anywhere environment and MySQL as the database to

store all the pest information. So the application is based on Faster R-CNN, it was the result of this study, it gave good results which reached a score of 98.9%, regarding automatic control of pests, about five popular agricultural pests, crops compared to machine learning and deep learning classifiers in previous studies. This study also shows a major advantage in being able to detect pests in similar and complex backgrounds to the images tested. Furthermore, the Faster R-CNN is suitable for real-time identification of agricultural pests in the field, without prior knowledge of the number of objects in the acquired images. Finally, the prediction of the categories and positions of the different agricultural pests are accurately identified.

There is also another achievement in the literature which is based on the visual recognition service of IBM Cloud: "Personal robotic assistants: a proposal based on the intelligent services of the IBM cloud and additive manufacturing" [31], it proposes a robotic assistant as a personalized version of the Otto robot, which relies on additive manufacturing and an intelligent chatbot using IBM Cloud services, which concerns speech and image recognition, in noisy environments. This project uses OpenCV and IBM Cloud's own service for visual recognition, in order to obtain better performance in image recognition both in virtual assistants and in the robotic assistant. So in this discussion we are interested in the image recognition part. The authors use techniques like Gaussian, Image Segmentation, Blur effects, and salt and pepper to add noise to the original images to determine if the machine learning model implemented in the robot (via IBM Watson Cloud) can identify the images. correctly based on 3 levels of acceptance that have only been applied to virtual assistants (telegram and messenger). The result of this article is a combination of precision obtained at the same time in all the assistants.

6 Conclusion

Several projects related to various fields of artificial intelligence exploit image recognition, which makes their success depends on finding the best solution to the recognition problem; our project is the study, realization and development of an Android mobile application named "Birds images predictor" for the classification and recognition of bird images using IBM's cloud visual recognition service which applies a machine learning algorithm more specifically a deep learning algorithm.

The application manages to identify about 210 types of birds from their images captured or chosen by the user, it can be improved by increasing the number of classes of birds, adding other additional options such as the geographical distribution on the map of each of these races. The architecture of the CNN model can also be improved by readjusting its hyperparameters to increase the prediction accuracy and to have a low error rate in order to compare it again with another solution in the same direction.

References

- [1] F. Z. ELBouni, T. Elhariri, C. Zouitni, I. BenBahya, H. Elaboui, A. ELouaazi, "Bird image recognition and classification using Watson visual recognition services from IBMCloud and Conventional Neural Network," the 3rd International Conference on Electrical, Communication and Computer Engineering, (6), 978-1-6654-3897-1, 2021.

- [2] P. Beck, A. Courdent, A. Feriati, P. Hyvon, "Sciences expérimentales et technologie," **79**, 2218945959, 9782218945953, 2012.
- [3] M. Michel, *A quoi sert l'histoire des sciences?*, 2006, ISBN: 978-2-7592-0082-5.
- [4] Y. Li, S. Abdallah, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *ELSEVIER*, **22**, 295–316, 2020.
- [5] B. Gilles, F. Jym, "Langage naturel et intelligence artificielle," *Hal-Archive*, **10**, 1638–1580, 1988.
- [6] A. Saad, A. M. Tareq, S. AL-ZAWI, "Understanding of a Convolutional Neural Network," *IEEE Access*, **6**, 978–1–5386–1949–0, 2017.
- [7] S. Dominik, M. Andreas, B. Sven, "Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition," *conference*, **10**, 164, 53117.
- [8] A. Bodin, F. Recher, "Algèbre–Convolution," in *Deepmath mathématique simple des réseaux de neurones (pas trop compliqués)*, Janvier-2021.
- [9] IBM, "IBM Cloud," <http://www.ibm.com/cloud>, 2022.
- [10] X. Tianjun, X. Yichong, Y. Kuiyuan, Z. Jiaying, P. Yuxin, Z. Zheng, "The Application of Two-level Attention Models in Deep Convolutional Neural Network for Fine-grained Image Classification," *IEEE Access*, **9**, 842–850, 2015.
- [11] X. Jie, Z. Mingying, "Handcrafted features and late fusion with deep learning for bird sound classification," *IEEE Access*, **8**, 75–81, 2019.
- [12] R. Satyam, G. Saiaditya, K. Sanu, S. Shidnal, "Image based Bird Species Identification using Convolutional Neural Network," 2278–0181, 2020.
- [13] C. University, "Merlin Bird ID," <https://merlin.allaboutbirds.org>, 2022.
- [14] IBM, "Watson Visual Recognition," <https://www.ibm.com/no-en/cloud/watson-visual-recognition/pricing>, 2022.
- [15] Machine, D. Learning, "Kaggle," <https://www.kaggle.com/>, 2022.
- [16] Google, "Firebase," <https://firebase.google.com>.
- [17] S. S. Mader, *Biologie humaine*, 22 janvier 2010, ISBN: 978-2-8041-2117-4.
- [18] C. Touzet, *Les réseaux de neurones artificiels, introduction au connexionnisme*, Ph.D. thesis, 2016.
- [19] E. Chigozie, W. Nwankpa, G. Ijomah Anthony, S. Marshall, "Activation Functions Comparison of Trends in Practice and Research for Deep Learning," **19**, 1811–03378, 2018.
- [20] j. Dong, J. Jaeseung, W. Sang, "Prefrontal solution to the bias-variance tradeoff during reinforcement learning," *Elsevier*, **37**, 37, 110185, 2021.
- [21] L. Andrea, S. Fabian, H. Johannes, N. Patrick, "Cybenko's Theorem and the capability of a neural network as function approximator," **32**, 2019.
- [22] S. Vézina, "fonction d'erreur," in *Le réseau de neurones*.
- [23] S. Anna, A. Bosman, Engelbrecht, M. Helbig, "Visualising basins of attraction for the cross-entropy and the squared error neural network loss functions," *Elsevier*, **24**, 113–136, 2020.
- [24] A. Frouvelle, "Méthodes de descente de gradient," in *Méthodes numériques optimisation*, 19 avril 2015".
- [25] P. K. Diederik, L. B. Jimmy, "ADAM: A method for stochastic optimization," *conference paper*, **15**, 2017.
- [26] D. Yehuda, M. Vidya, G. B. Richard, "A Farewell to the Bias-Variance Trade-off? An Overview of the Theory of Overparameterized Machine Learning," *Elsevier*, **48**, 2021.
- [27] IBM, "IBM Documentation," https://www.ibm.com/docs/en/app-connect/containers_cd?topic=SSTDS_contcd/com.ibm.ace.icp.doc/localconn_ibmwatsonvr.html, 2022.
- [28] "TensorFlow," <https://www.tensorflow.org/>, 2022.
- [29] "TensorFlow," <https://www.tensorflow.org/lite/guide?hl=fr>, 2022.
- [30] K. Mohamed Esmail, A. Fahad, S. Albusaymi, A. Sultan, "A new mobile application of agricultural pests recognition using deep learning in cloud computing system," *IEEE Access*, **10**, 66980–66989, 9 March 2021.
- [31] S. Amendan o Murriol, E. Duta n Go, Christian mez Lema-Condo, V. Robles-Bykbaev, "Personal robotic assistants: a proposal based on the intelligent services of the IBM cloud and additive manufacturing," *IEEE*, (6), 978–1–7281–9365–6, 2020.