Reptile/Amphibians Classification using Convolutional Neural Networks

Erika Fox
The Graduate School
Master in Interdisciplinary Data Science
Duke University
Durham, NC 27708
erika.fox@duke.edu

Peining Yang
The Graduate School
Master in Interdisciplinary Data Science
Duke University
Durham, NC 27708
peining.yang@duke.edu

Abstract

Inspired by the work of the Vermont Reptile and Amphibian Atlas, we experimented with different image processing techniques such as changing aperture, image contrast and image composition to determine the specifications optimized for classifying reptiles/amphibians using Convolutional Neural Networks (CNN). In addition, we also tested the potential of our model in classifying paired images, which were similar to the composition of the Atlas images. Our base model achieved a classification accuracy of 75.18%. Overall, image preprocessing techniques proved unsuccessful in improving classification accuracy using an XCeption CNN model.

1 Introduction

1.1 Forest Ecosystem Monitoring Cooperative (FEMC)

The Forest Ecosystem Monitoring Cooperative (FEMC) is an organization serving to understand and improve the forested ecosystem in the northeast temperate forest region. A part of FEMC is the Vermont Reptile and Amphibian Atlas, which has gathered and disseminated data on reptiles and amphibians in Vermont with the ultimate goal of informing individuals and organizations to better study and protect the wildlife habitat. James S. Andrews is the Principal Investigator of the Atlas and has been collecting information on reptiles and amphibians, more specifically frogs, salamanders, turtles, snakes and lizards, since the 1990s. We reached out to him in hopes of gaining access to the photo database.

1.2 Motivation

Given the importance in identifying reptiles/amphibians in order to study them and preserve their wildlife habitat, we wanted to explore the capacity of a Convolutional Neural Network (CNN) in correctly classifying the images. In addition, we wanted to see if any image processing techniques could enhance the performance of the model.

2 Related Work

2.1 Choosing Image Augmentations for Neural Net Training

There are a number of recent studies in which researchers have experimented with preprocessing images for optimal classification performance that we took inspiration from. One study took place in 2017, where a team of computer scientists trained a variety of classifiers using the well known

MNIST hand written digits dataset, incorporating different image augmentation techniques such as "centering, elastic deformation, rotation and different combinations of them" [6]. Their paper provides compelling evidence that this kind of image augmentation can significantly improve the performance of a CNN. They point to a combination of rotation and elastic deformation as optimal preprocessing steps for MNIST. We are inspired by this study to see what preprocessing steps work best for our reptiles/amphibians data.

Another study from 2017 consisted of a team looking to add some preprocessing steps to their emotion recognizing CNN. The augmentation techniques they explored include resizing, cropping, global contrast normalization and histogram equalization [4]. They found that global contrast normalization had the best accuracy among all the normalization techniques they tried, which inspires us to work with contrast normalization in this project. They also found that they were able to boost the performance of their CNN by cropping their images and adding noise to them, which we kept in mind for this work as well.

2.2 Classifying Animal Species with Neural Nets

When it comes to using CNNs to work with animal classification specifically, we found a few more examples of past work to review. Deep Convolutional Neural Network has previously been proven to be effective in performing image classification. The algorithm with neural networks has been designed to work parallel to the human brain [7]. A previous study has shown success in detecting invasive species using camera-trap images and CNN models [2]. The study achieved a top-1 accuracy of 91.8%, yet also produced the highest missed invasive rates. This study served as a reminder that in order to optimize the performance, the metrics of the model must be tailored specific to our classification goals. Another study achieved 91.30% accuracy in identifying venomous and nonvenomous snakes [5]. Given the success of this binary classification was contained within a single specie, it inspired us to further investigate the possibility of multi-class image classification using CNN models.

3 Methods

3.1 Data

3.1.1 The Vermont Reptile and Amphibian Atlas Samples

The photo samples we received were single page PDFs that usually featured more than one image of the documented animal. In addition, the labels of the animal were given in the form of binomial nomenclature, and were hand-written on the PDF itself. It would require major efforts in Natural Language Processing in order to decipher the image into a workable dataframe. In addition, we were only provided with 200 image samples, which is insufficient for modelling.

3.1.2 Kaggle: Reptile and Amphibian Image Set

Due to the reasons mentioned in the previous section, we decided to forgo using the Atlas image set as our main training data and sought out an additional dataset on Kaggle that was more suitable for our project goal. This dataset consisted of 200-1000 images for a myriad of reptile and amphibian types, which included the five types that we were hoping to use to train our classifier: frogs/toads, salamanders, turtles, snakes and lizards. We did some grouping in order to prepare our data, such as combining frogs and toads, which came as separate classes, into one "Frog/Toad" class, as well as grouping the "Chameleon" and "Iguanas" classes in with the given "Lizard" class (chameleons and iguanas are types of lizards). After this grouping, each of the classes had at least 500 images. In order to ensure class balance for training, we narrowed down the number of images per class in our data frame to exactly 500. The size of individual images were around 300 pixels each.

3.2 Experimenting with Augmentation

As the main goal of this project is to explore the capacity of a Convolutional Neural Network (CNN) to correctly classify images of reptiles and amphibians, we have chosen three main avenues of experimentation. Ultimately, we will experiment with changing the Aperture, Contrast, and

Composition of the training images for our classifier in order to answer the question: How should reptile and amphibian images be preprocessed for classification?

3.2.1 Base Model

We utilized a deep convolutional neural network architecture called Xception for this project. Xception was proposed in a research paper by Francois Chollet in 2017 as an augmentation of the Inception [1]. Inception is a neural network architecture that utilizes multiple branches of convolutional layers with varying kernel sizes to capture features at different scales. Xception reduces the number of parameters and computational complexity of the network, while improving its accuracy and ability to generalize to new data. We chose Xception because its architecture optimizes accuracy in image classification.

For our baseline model (before introducing changes in aperture, contrast and composition), we trained a classifier with a basic set of prepossessing steps. These steps include using OpenCV's cvtColor() function to convert the image to the standard, more easily processed, RGB format, as well as resizing. These steps brought the shape of each sample in our data to (128,128,3). For the following augmentations, we kept these basic steps as well and just added upon them.

3.2.2 Aperture

We anticipated that by applying an physical layer of an aperture lens to the images could enhance features of the images and optimize classification. A aperture lens with 50 pixels radius is applied to the input in the Fourier domain. The results are then batch normalized and the processed images are trained and tested on the base model.

Figure 1: Example of original image

Figure 2: Example of image after applying 50px aperture lens

3.2.3 Contrast

We were inspired to try brightening the contrast of our training images as this has the potential improve a classifier by enhancing image features, reducing noise, and normalizing the brightness level across the data. To brighten the images, we used the OpenCV function, cv2.convertScaleAbs(). This function adjusts image brightness and contrast using a linear transformation and outputs an 8-bit unsigned integer format image. The parameters we passed to this function include the original image, as well as "alpha" and "beta". "Alpha" is a scaling factor that is applied to the pixel values of the original image, which we set to 1.5. "Beta" is an optional value that gets added to the result of alpha's scaling operation. We set beta to 50.

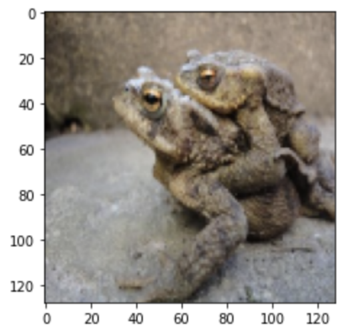

Figure 3: Example of original image

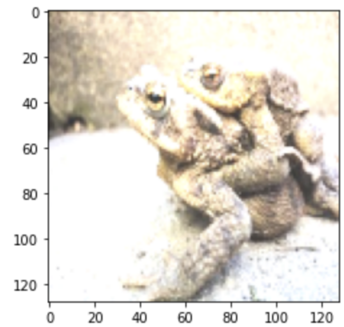

Figure 4: Example of image after applying cv2.convertScaleAbs() with alpha=1.5 and beta=50

3.2.4 Composition

Finally, we wanted to experiment with composition with our training images as sometimes this can improve a classifier by reducing computational complexity, removing irrelevant information, or making the model more generalizable. To change the composition of our images, we implemented a simple center crop method that removed 10 pixels from all four sides of each sample.

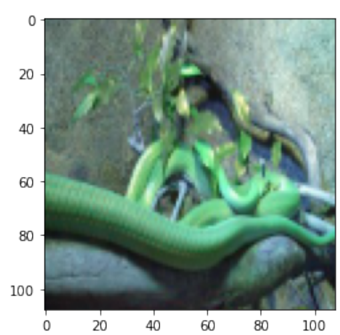

Figure 5: Example of original image

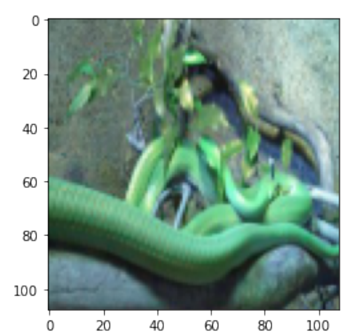

Figure 6: Example of image after center crop

3.3 Experimenting with "Paired Images"

Once we had trained classifiers for each of the previously described augmentation variations using a single image as the unit of observation, we decided some additional work would be helpful in order to determine how well our models would actually apply to the FEMC data. Namely, as the FEMC samples often include more than one image in one sample, we wanted to try to train classifiers using more similarly structured data. However, because we still felt the FEMC data was insufficient for model training (as previously explained) we opted to generate our own data to mimic the FEMC's structure, what we came to refer to as our "Paired Images". Our generator works by beginning with a plain white image of size (128,128,3). Then one of our five reptile/amphibian classes is chosen, and from that class, two images are randomly chosen to get pasted on top of the plain white image. We used this generator to get a new dataset of 500 samples for 5 classes (same classes as before) to use to train the same kinds of classifiers as before (base, aperture, contrast and composition.

Figure 7: Example of original FEMC image

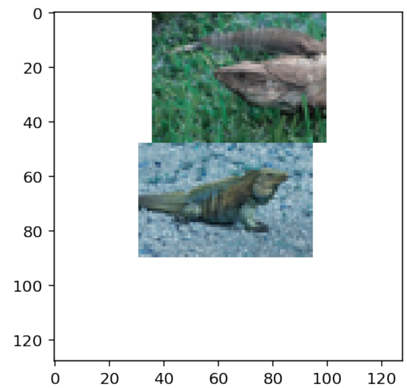

Figure 8: Example of generated pairs image

4 Results

The table below shows the results for the single image classifier using an XCeption model with different image preprocessing techniques. Altering the image contrast and image composition did slightly improve the testing accuracy from the base model, while applying an aperture lens worsened model performance. The area under the curve (AUC) were all either 0.94 or 0.96, indicating excellent performance in terms of avoiding false positive classifications.

Table 1: Results for single image classifier

Preprocessing Technique	Training Acc.	Testing Acc.	AUC
Base Model (N/A)	85.26%	75.18%	0.96
Aperture	84.55%	70.17%	0.94
Contrast	82.82%	75.89%	0.95
Composition	86.34%	76.13%	0.94

Similar to above, the table below shows the results for the image classifier trained and tested on paired images. Changing image contrast only marginally improved performance compared to the base model, while changing aperture and image composition significantly decreased classification accuracy. The AUCs were all above 0.9, suggesting great model performance.

Table 2: Results for paired image classifier

Preprocessing Technique	Training Acc.	Testing Acc.	AUC
Base Model (N/A)	74.74%	72.90%	0.97
Aperture	72.90%	65.13%	0.95
Contrast	73.79%	67.86%	0.96
Composition	65.49%	65.97%	0.92

5 Discussion

Across both single and paired image classifiers, the attempt at improving model performance using image preprocessing techniques were generally unsuccessful. For single images, performing a center crop improved the classification accuracy marginally. However, given the fluctuation in testing accuracies with different testing attempts, whether one method is definitively better than the other is still inconclusive. When comparing single and paired image classifiers, the ROC curve and AUC improved with paired images. With testing accuracies, single image classifiers outperformed the model using paired images.

6 Future Work

Although the variety of classifiers we trained all performed reasonably well, the answer to our original question of which preprocessing techniques should be used for reptile/amphibian image classification remains mostly unanswered. Future work that can be done for this problem includes further experimentation with some additional preprocessing methods. There are a few methods in particular that have been used in some of the related projects we researched for this project that we didn't get to explore, such as image rotation, elastic deformation, and combining methods together.

Aside from nailing down the best preprocessing methods to use with the images, we also think that there is more that can be done in order to improve classification accuracy with the FEMC files. We found that generally our classifiers had higher validation accuracy when single images were used for training (as opposed to our paired image generator solution), so it would be might worth taking the extra steps to translate those files into that format. This could be done by first fitting an image segmantation model to the original files that can be used to locate single images within an FEMC Atlas entry. These predicted single images could then be used for training instead of the entire noisy FEMC Atlas file.

References

[1] Chollet, François. "Xception: Deep Learning with Depthwise Separable Convolutions." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251-1258.

[2] Larson, Julia, "Assessing Convolutional Neural Network Animal Classification Models for Practical Applications in Wildlife Conservation" (2021). Master's Theses. 5184.

[3] Y. LeCun, Y. Bengio and G. Hinton, "Deep Learning," Nature **521**, 436–444 (2015).

[4] Pitaloka, Diah Anggraeni, et al. "Enhancing CNN with Preprocessing Stage in Automatic Emotion Recognition." Procedia Computer Science, vol. 116, 2017, pp. 523–529.

[5] Progga, Nagifa Ilma, et al. "A CNN Based Model for Venomous and Non-Venomous Snake Classification." Applied Intelligence and Informatics, 2021, pp. 216–231.

[6] Tabik, Siham, et al. "A Snapshot of Image Pre-Processing for Convolutional Neural Networks: Case Study of Mnist." International Journal of Computational Intelligence Systems, vol. 10, no. 1, 2017, p. 555.

[7] Chauhan, R., Ghanshala, K.K., Joshi, R.: Convolutional neural network (CNN) for image detection and recognition. In: 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), pp. 278–282. IEEE (2018)