Fine-tuning based deep convolutional networks for lepidopterous genus recognition

Juan A. Carvajal[1], Dennis G. Romero[1], Angel D. Sappa[1,2]

[1]Escuela Superior Politécnica del Litoral, ESPOL,
Facultad de Ingeniería en Electricidad y Computación,
Campus Gustavo Galindo Km 30.5 Vía Perimetral, P.O. Box 09-01-5863,
Guayaquil, Ecuador
[2]Computer Vision Center, Universitat Autónoma de Barcelona,
08193-Bellaterra, Barcelona, Spain.

Abstract. This paper describes an image classification approach oriented to identify specimens of lepidopterous insects recognized at Ecuadorian ecological reserves. This work seeks to contribute to studies in the area of biology about genus of butterflies and also to facilitate the registration of unrecognized specimens. The proposed approach is based on the fine-tuning of three widely used pre-trained Convolutional Neural Networks (CNNs). This strategy is intended to overcome the reduced number of labeled images. Experimental results with a dataset labeled by expert biologists, is presented—a recognition accuracy above 92% is reached.

1 Introduction

The ever increasing computational capability of mobile devices together with the advances in multimedia technology has opened novel possibilities that goes beyond communication between users. One of this novel possibilities lies on the image acquisition and recognition, from anywhere at any time. In this sense, different tools have been developed to contribute with biodiversity categorization, driven by recognition techniques based on features obtained from images, for a variety of applications: registration of known animal species; identification of unrecognized genus; or academic tools for learning about animal species.

In biology, for example, students learn to identify insect species as part of their formal instruction. For this task, it is necessary to consider tools that provide summarized information in order to be presented to students, which represents a challenge for teachers who must summarize large amounts of information about class, family, genus and details that distinguish between them. A suitable method for automatic recognition of insect species would facilitate the registration of biodiversity, by enabling a wider use of these records and the knowledge we have about the species, also by reinforcing the inclusion of different areas of research such as life sciences, computer science and engineering.

The current work is a continuation of previous studies about automatic identification of lepidopterous insects, which discussed an image feature extraction

methodology based on the butterfly shape. The current approach goes beyond classical pattern recognition strategies by using convolutional neural networks (CNNs). More specifically, a fine-tuning scheme of pre-trained CNNs is considered. The manuscript is organized as follow. Related works are presented in Section 2. Then, the proposed approach, which is based on the usage of different CNN's architectures, is introduced in Section 3. Experimental results are provided in Section 4. Finally, conclusions are given in Section 5.

2 Related work

The problem addressed in this paper is related with biology. This discipline covers a specific approach oriented to the study and description of living beings, either as individual organisms or species. These species have in some cases significant differences between them, facilitating their identification, although this is not always easy when categorized by genus.

The study presented in [1] describes a system for visual identification of plants, by using pictures taken by the user, returning additional images of some specie along with a description thereof, using a mobile device. Some other studies have been carried out identifying automatically patterns through traditional classification algorithms. In a recently published paper [2], three classifiers were selected (k-NN, MLP, SVM) and evaluated considering metrics such as simplicity, popularity and efficiency. Different tests were conducted by varying the training set, considering in all the cases 75% for training and 25% evaluation, from 2 to 8 classes, the validation method used was Leave-One-Out. The best result was obtained with SVM reaching a 75.5% of accuracy. This work was evaluated using the same dataset than the one used in the current work, but just 8 classes where considered, instead of 15 as in the current work.

Over the past few years, deep CNNs have revolutionized large-scale image recognition and classification. Virtually all of today's high achieving algorithms and architectures for image classification and recognition make use of deep CNN architectures in some way [3] [4]. In large part, these advances have been made possible by large public image repositories and the use of high performance GPUs. CNNs are a set of layers in which each layer performs a non-linear transformation function learned from a labeled set of data. The most important type of layers are convolutional and fully connected layers. Convolutional layers can be thought as a bank of filters that are convoluted to produce a feature map as an output. When relating CNNs to multilayer perceptrons (i.e., classical neural networks) these banks of filters can be seen as shared weights for all of the neurons in a layer. Fully connected layers, while also applying convolution, connect every single neuron of a layer, thus creating a large number of weights, and having an n-dimensional vector as an output [5]. Other widely used layers are pooling layers, which produce a sub-sampling of the input, and ReLu layers, which apply the rectifier activation function to add non-linearity to the network. For training purposes, a layer called dropout can be inserted between fully connected layers

to avoid over-fitting [6]. These dropout layers set to zero the output of a neuron randomly, preventing the dependence of that neuron to particular other neurons.

The dataset to be used for training and validation is an important element, as in any pattern recognition approach. The current work is specifically oriented to a particular genus of lepidopterous (butterflies) in their last stage of life cycle. The process of collecting samples of these genus takes time and effort before being subjected to studies, in order to determine more relevant characteristics. Experts in this area have generated over time, knowledge bases on families, subfamilies, tribes and genre of butterflies with different relevant information about them. Because of this, biology students now have a lot of information about Lepidopterous, however, this amount of information turns hard to be used as learning resources in a more efficiently and effective way.

There are different datasets to be used in studies of pattern recognition; the most widely used is *ImageNet* [7], which has a considerable quantity of images in different categories; in this dataset there is a category called "Butterfly" that has 2115 images of butterflies classified from four main families: nymphalid, danaid, pierid, lycaenid. Figure 1 shows butterflies obtained from the ImageNet dataset.

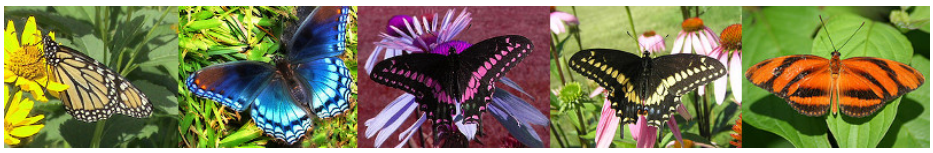

Fig. 1. Images of *butterflies* category taken from *ImageNet* dataset.

Leeds [8] is another dataset; it contains images and textual descriptions for ten categories (species) of butterflies. The image dataset comprises 832 images in total, with the distribution ranging from 55 to 100 images per category. Images were collected from Google Images by querying with the scientific (Latin) name of the species, for example "Danaus plexippus", and manually filtered for those depicting the butterfly of interest (Figure 2).

Fig. 2. Five categories of butterflies from *Leeds* dataset.

One of the richest place in lepidoptera specimens is the *"Sangay National Park"* [1], which has been listed by UNESCO as a World Heritage Site since 1983. Several samples of lepidoptera specimens have been collected by expert biologist in this reserve, some of them not included in *ImageNet* or *Leeds datasets*. The dataset provided by Sangay National Park has 2799 images of butterflies, classified in 32 genus (Table 1). Figure 3 shows an image per category; the inter-class similarity of some of them makes the problem quite challenging.

Table 1. Name and number of images per butterfly's genus in *Sangay National Park dataset* (in bold the 15 selected classes, those that have a larger number of images)

Genus	Images	Genus	Images	Genus	Images
Adelpha	**120**	*Catonephele*	28	*Eunicini but Eunica*	38
Anaeini but Memphis	65	*Cealenorrhini*	27	***Euptychia et al***	**197**
Ancyluris et al	40	*Coloburini*	16	***Eurybie Teratophtalma***	**127**
Anteros	39	*Corades*	58	***Euselasia***	**156**
Anthoptini	68	***Dalla***	**208**	*Euselasiini but Euselasia*	10
Astraptes	**122**	*Dynamini*	21	***Forsterinaria***	**93**
Callicore et al	30	***Emesis***	**86**	*Haeterinae*	120
Calpodini	**89**	*Epargyreus*	31	***Helicopini others***	**83**
Carcharodini	**160**	*Eretris*	37	***Hesperiini***	**236**
Catagrammidi	**80**	***Erynnini***	**111**	*Hesperiini incertae sedis*	33
Eudaminae	**242**	*Eunica*	28		

3 Proposed approach

This section presents the different approaches proposed to perform the classification of 15 lepidoptera species. All these approaches are based on the usage of deep convolutional neural networks (CNNs). Even though a deep CNN can achieve very high accuracy, one of its drawbacks is the need of having a relatively large dataset to train a CNN from scratch. Solutions to this problem have been proposed, such as using the output of a layer (before the last one) from a pre-trained CNN network as a feature extractor. These features can be then used as inputs to train a classical classifier, for instance a SVM [9]. Another way to tackle the limitation related with the size of the dataset consists of taking a network already trained and *adapt* it for the current work. In the adaptation process, which is referred to in the literature as *fine tuning*, the output layer of the pre-trained CNN network is randomly initialized; then the network is trained again but now with the dataset of the new classification problem [10]. Intuitively, the fine tuning is taking advantage of the filters originally learned by the network, instead of starting from random initialization as would be the case when training from scratch. Moreover, the filter learned at the early layers are usually very generic (edge detectors, blob detectors) than can generalize to a variety of problems.

[1] Ecological reserve in Ecuador: http://www.sangay.eu/index.php?lang=en

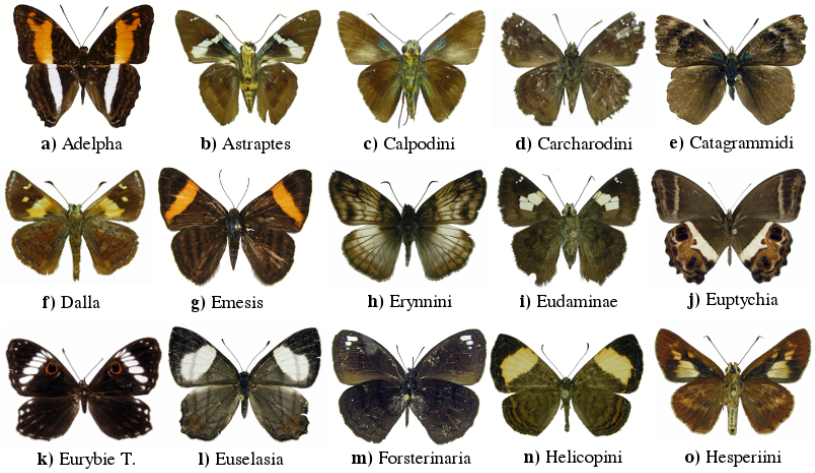

Fig. 3. Illustration of butterflies categorized by genus, for space limitation just one image per category is depicted (note the similarity between classes: (c), (d) and (e)).

Having in mind the two possibilities mentioned above, and being aware of the limitation of our dataset (we are working with real dataset taken by expert biologists that cannot be that easily extended), we have decided to use pre-trained networks. The selected networks have been trained on the *Imagenet* dataset for the Imagenet Large Scale Visual Recognition Challenges. Imagenet classes include various types of animals, insects and more importantly different species of Lepidopterous, assuring that some filters have learned to identify different species of butterfly. Filters along the network that have learned this, generalize well to our classification problem. In conclusion, the fine tuning of the filters, specially the ones in the last fully connected layer, which were previously randomly initialized, takes advantage of whatever filters the pre-trained network had already learned. When training it with the new dataset, it learns to differentiate the different species of Lepidopterous in the new dataset.

In this section we present the three pre-trained networks selected to be fine tuned with our dataset. These networks have been trained to perform classification on Imagenet ILSVRC. Originally the networks had an output of 1000 classes, which correspond to the Imagenet classes. They were modified to output the 15 classes from our dataset. A brief description of them is given below.

AlexNet. This architecture has been introduced in [3] and inspires much of the work that is being done today on deep CNNs. It has eight layers with learned parameters, five convolutional and three fully connected. In its original implementation it was trained on two GPUs. It was the winning model of the ILSVRC-2012 challenge. An illustration of this CNN is depicted in Fig. 4.

VGG-F.This network has been introduced in [11] and is very similar in architecture to Alexnet [3], but it is considerably faster when training and evaluating. Because Alexnet was trained on two GPUs, it used sparse connections between

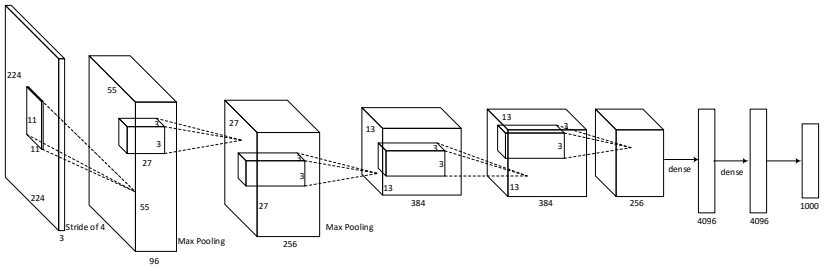

Fig. 4. Illustration of the architecture of the AlexNet CNN (see [3] for details about the configuration of each layer).

convolutional layers. VGG-F, on the other hand, has dense connectivity between these layers [11]. Originally trained on ILSVRC-2012 challenge dataset.

VGG-Very Deep Convolutional Networks. This network has been presented in [12]. This work proposes two deep networks with 16 and 19 layers with learned parameters respectively, with the last three being fully connected. Besides the increased depth of the networks, they also introduce very small convolution filters. Initially submitted to ImageNet ILSVRC Challenge 2014.

4 Experimental Results

This section presents results obtained with the fine tuning of the three pre-trained network presented above; the lepidoptera dataset introduced in Section 2 has been considered. Our training and evaluation of the networks was done with the MatConvNet toolbox for Matlab. This toolbox is efficient and friendly for computer visions research with CNNs [13]. The usage of GPUs has also become widespread when training CNNs, allowing for faster computation and deeper networks. In our case, when training networks as deep as a VGG-Very Deep Convolutional Network the usage of GPU is completely necessary. While this is not mandatory for the other two pre-trained networks, it certainly speeds up the computation. We used the NVIDIA GeForce GTX950 GPU.

Regarding the data, we apply data augmentation so that the network sees more training and test images. Specifically, we mirror a random number of the images presented to the network on each epoch. When mirroring there is no change done to the species of the lepidoptera. Even though it is not as good as having more independent samples, data augmentation improves performance [11]. 25 percent of the images in each category were used for testing, while the other 75 percent were used for training purposes. All images were normalized and re-sized when given as input to each network. All networks originally accept color images of size 224x224, but in the particularly case of the AlexNet network, because an Matconvnet implementation of the network that was imported from Caffe was used, the color images had to be re-sized to 227x227.

Figure 5 depicts the percentage of error versus the number of epochs for the four networks fine-tuned with the 15 classes presented in Table 1. In the case of the less deep networks it can be apreciated that less epochs are needed for the training and test error to converge compared to the VGG-Very Deep Networks. Furthermore, in the deeper networks, it can be seen that there is more over fitting and the validation error oscillates more. We can attribute this behavior to the much larger number of parameters which are learned from the training set in the deeper networks. While this contributes to a very small training error, it makes it harder and longer for the test error to converge. Quantitative results are presented in Table 2, note that in all the cases the recognition ratio is higher than 92%, which is more than acceptable and considerably better that previous recently presented works [2].

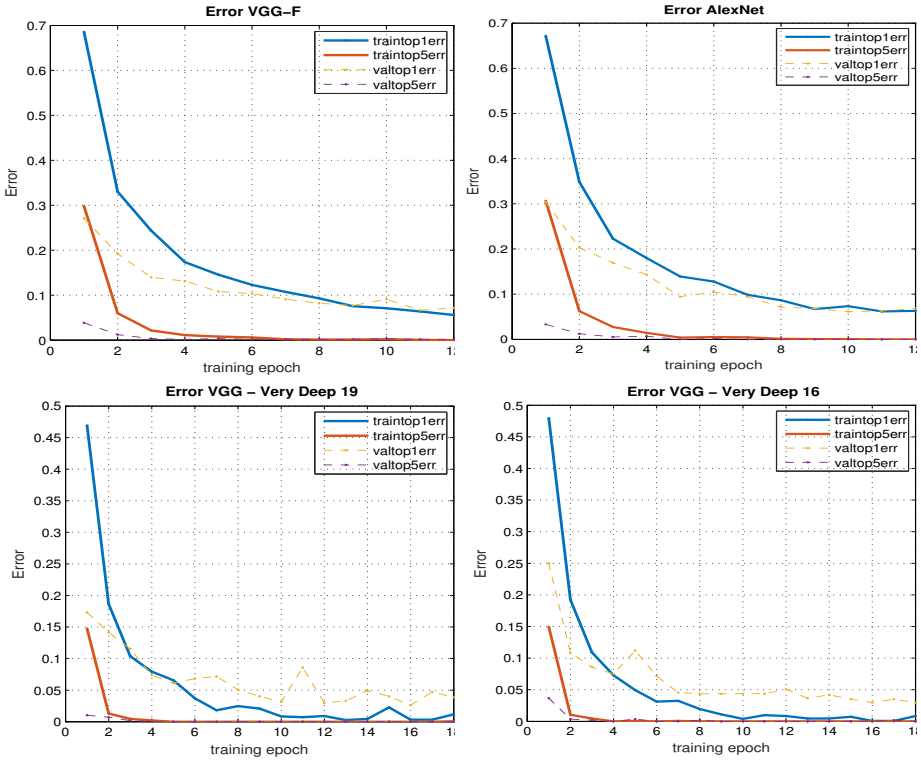

Fig. 5. Percentage of error vs number of epochs for each of the fine tuned networks (top1 error percentage for the train and test sets are shown in full blue and dotted yellow, respectively).

Table 2. Results obtained with 15 classes from Sangay National Park dataset. Error values are percentages relative to the total number of images in the dataset.

Pre-trained CNN	Training Error	Validation Error
AlexNet	6.30 %	6.81 %
VGG-F	5.59 %	7.17 %
VGG - Very Deep 16	0.8 %	2.97 %
VGG - Very Deep 19	1.17 %	3.84 %

5 Conclusions

This paper tackles the challenging problem of lepidopterous insects recognition by fine-tuning three pre-trained CNNs. This strategy is proposed to overcome the lack of large data set to be used during the training stage. Fortunately, the fine-tuned networks have been trained with ImageNet dataset, which contains a category called "Butterfly" with more than 2000 images. Although the lepidoptera specimens considered in the current work are not included in ImageNet, we assume the pre-trained networks already have some capabilities to discriminate them. Experimental results shows that, even thought the size of training set is quite reduced, an acceptable performance is reached (note that in all the cases more than 92 % of recognition ratio has been reached). Future work will be focused on evaluating different data augmentation strategies to enlarge the training set.

References

1. Belhumeur, P.N., Chen, D., Feiner, S., Jacobs, D.W., Kress, W.J., Ling, H., Lopez, I., Ramamoorthi, R., Sheorey, S., White, S., et al.: Searching the world's herbaria: A system for visual identification of plant species. In: Computer Vision–ECCV 2008. Springer (2008) 116–129
2. No author given: Our previous paper. No journal given **nn** (nn)
3. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. (2012) 1097–1105
4. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Computer vision–ECCV 2014. Springer (2014) 818–833
5. Reyes, A.K., Caicedo, J.C., Camargo, J.E.: Fine-tuning deep convolutional networks for plant recognition. In: Working notes of CLEF 2015 conference. (2015)
6. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.R.: Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580 (2012)
7. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE (2009) 248–255
8. Wang, J., Markert, K., Everingham, M.: Learning models for object recognition from natural language descriptions. In: Proceedings of the British Machine Vision Conference. (2009)

9. Razavian, A., Azizpour, H., Sullivan, J., Carlsson, S.: Cnn features off-the-shelf: an astounding baseline for recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. (2014) 806–813
10. Bengio, Y.: Deep learning of representations for unsupervised and transfer learning. Unsupervised and Transfer Learning Challenges in Machine Learning **7** (2012) 19
11. Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A.: Return of the devil in the details: Delving deep into convolutional nets. In: British Machine Vision Conference. (2014)
12. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
13. Vedaldi, A., Lenc, K.: Matconvnet: Convolutional neural networks for matlab. In: Proceedings of the 23rd Annual ACM Conference on Multimedia Conference, ACM (2015) 689–692