Summary: Animal Biometrics and Deep Learning

Fabian Otto*

Technische Universität Darmstadt

ABSTRACT

This summary provides an overview about the current research in Animal Biometrics, especially with regards to Deep Learning. This includes the overall processes, the filtering and preprocessing of images, architectures of networks, post-processing, etc. As the animal individual identification field is not well developed, this also tries to find similar ideas which can be adapted for this problem. This includes species identification as well as general fine-grained object recognition.

1 INTRODUCTION

The main goal in animal ecology is to observe species in their natural habitat. However, observing biodiversity can be expensive, logistically difficult and time-consuming. Many animals are rare, secretive and inhabit remote areas. Animal presence and behavior may vary over broad spatial and temporal scales, and depend on important but infrequently observed events, such as breeding, predation or mortality. Direct observation of these events can be disruptive to wildlife and potentially dangerous to observers. To reduce cost, labor and logistics of observation, ecologists are increasingly turning to greater automation to locate, count and identify organisms in natural environments. Therefore, researchers utilize camera traps, they provide a relatively cheap and easy solution to simultaneously collect large quantities of images in different areas. However, one big problem remains: The evaluation ans information extraction of images. This summary tries to provide an overview of different approaches and techniques, which are used in Computer Vision to solve this problem. Hereby, the main focus is to find existing research, which uses Convolutional Neural Networks (CNN), which were trained to identify individuals within one species and find similar Deep Learning approaches, which can be adapted for this problem.

2 OVERVIEW, PROBLEMS AND CLASSICAL APPROACHES

A good overview about the topic is provided by Weinstein [15]. His survey provides references to several previously conducted research projects involving Computer Vision and Animal Ecology. It also addresses other issues apart from the identification of individuals, such as counting, gender identification, species identification, descriptions of ecological object (e.g. eggs) etc. Further, the survey demonstrates the problems current approaches have to face in order to achieve any form of classification and also presents possible solutions.

As mentioned earlier collecting images is, due to camera traps, efficiently possible. Nonetheless, a significant issue for processing these images is their quality. The quality can be influenced by different illumination, shadows, weather and other image artifacts. A good visualization about common problems can be found from Gómez [6] (see Fig. 1)

In order to achieve better results one suggestion from Weinstein [15] is using image metadata, such as time or location, to assist in image classification.

*e-mail: fabian.otto@stud.tu-darmstadt.de

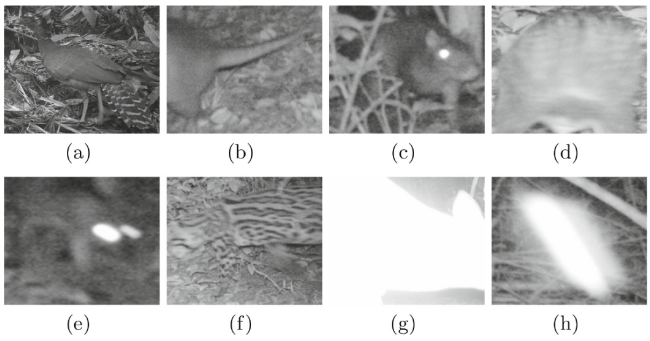

Figure 1: (a) Ideal and scarce case in camera-trap framework. (b) Partial capture of the animal. (c) Background occlusion. (d) Auto-occlusion. (e) Low resolution images. (f) Blurred animal. (g) Overexposed animal (h) Occlusion of the camera lens. [6]

For individuals identification, computer vision algorithms use images of known individuals to match new images based on the similarity of phenotypic patterns. By matching the image features among images, matching algorithms score the likelihood that two images are of the same individual. For animals with unique markings, this can be a low-cost alternative to expensive trapping and tagging programs. Current state-of-the-art systems try to solve this problem with highly adapted solutions, which are most of the time not applicable for other areas/species. One example of such system is Burghardt's [7] approach for identifying Great White Sharks individuals based on their fins. It detects and groups object boundaries at multiple scales into an ultrametric contour map. Afterwards, salient boundary locations are detected and used to partition region boundaries into contour sections, which are classified into fin and background classes based on shape, encoded by normals, and local appearance encoded by opponentSIFT features. Other application areas are quite limited, it might be possible to transfer this system to other fish or similar marine creates, for instance dolphins.

3 SPECIES IDENTIFICATION

One of the first approaches for classifying species, which can be seen as similar to identifying individuals, was from Yu [16]. They reached 82% accuracy on their dataset of 7196 images and 18 classes with local and global feature extraction, which were used to train an Support Vector Machine (SVM). However, this solution still requires a lot of manual preprocessing: Removing images without animals, manually cropping all the animals from the images, and selecting only those images that captures the animals' whole body.

The first completely automated approach, which used Deep Learning for species classification was established by Chen [2]. They showed that CNNs are able to outperform the traditional Bag-of-Words technique, if enough data is provided. Before training the network, they used their own automatic segmentation method (Ensemble Video Object Cut) for cropping animals from the images. However, the accuracy, which was achieved on their 20,000 image data set with 20 classes, was only 38%. This can probably be backtracked to their relatively small network architecture, seen in Fig. 2

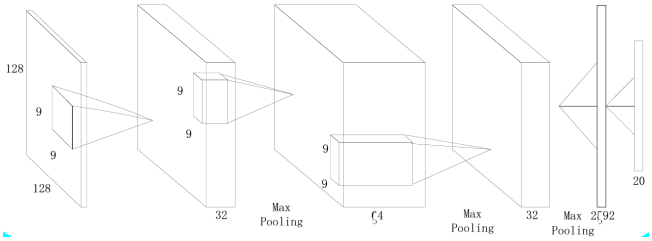

Figure 2: The structure of Chen's CNN used for species recognition. [2]

The work of Gómez [6] is based on the idea of clustering species to smart biologically inspired sets in order to increase the performance even for low quality as well as gray scale images. The data set of 1572 and 2597 images contains animals in the South American jungle, which present a very cluttered back ground, poor illumination conditions, gray scale, and low resolution. In order to increase the trust in the system (necessary due to possible actions e.g. to save species), for each classification a confidence was computed. This confidence had to be above a certain threshold or otherwise the images were given to an expert. The training set in the experiments was augmented using images from the ImageNet dataset. With regards towards their architecture they utilized several state-of-the-art CNN architectures (AlexNet, VGGNet, GoogLenet and ResNet). As preprocessing the images where resized in order to fit the input sizes. As a result, they found deeper architectures work better up to ResNet-101, which was found to be the best option for the given dataset.

A similar approach is from [5], which uses the Snapshot Serengeti dataset. However, they removed 22 classes that have the fewest images. Further, they split up the dataset in order to get unbalanced (rare animals less frequent), balanced (cleaned to get approx. even distribution), conditioned (animals in foreground), and segmented (manually segmented, to simulate perfect segmentation algorithm, which finds (part of) the animal) datasets, which were used during their analysis. The used dataset contains high as well as low quality images such as in Fig. 3.

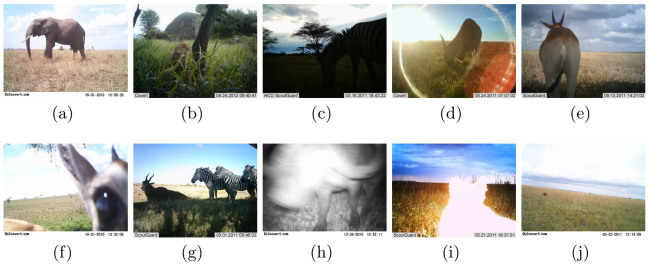

Figure 3: (a) Ideal. (b) Occlusion due to context. (c) Poor illumination. (d) Over-exposed regions. (e) Auto-occlusion. (f) Complex animal poses and unexpected images. (g) Different species in the same image. (h) Blurred. (i) Over-exposed animals. (j) Animals far way from camera. [5]

They also compared several state-of-the-art CNN architectures (AlexNet, VGGNet, GoogLenet and ResNet). The CNNs were all pre-trained with ImageNet and fine-tuned with Stochastic Gradient Descent (SGD) and back-propagation. The Learning rate and step size were reduced to avoid overfitting. They found even the deepest architectures cannot deal with highly unbalanced data, but they still perform better than shallow networks. For balanced datasets the accuracy increases to 70%. the main problems are empty frames or high distance between camera and animal. The conditioned data increased accuracy by 12.9%, the segmented dataset, however, increased the conditioned result only slightly. The suggestion of the paper is to use deeper networks and skip segmentation as well as use sufficiently large datasets.

Another approach based in the Snapshot Serengeti dataset can be found from [11, 12]. Their research goal was to identify and count species in the Snapshot Serengeti dataset including all 3.2 million images with 48 species. Additionally, they removed images with multiple labels/multiple animals and trained two CNNs for different tasks:

- animal vs. empty (due to 75% empty frames)
- animal identification (species)

Preprocessing includes rescaling the images and normalizing all three input channels individually (z-transformation). The training is conducted by using SGD with momentum (0.9), weight decay and batch size of 128. The best result was achieved by using VGGNet. Nonetheless, ResNet architectures show similar performance with decreases less than 1% in accuracy. In their results they also provide a comparison between pre-trained CNNs as they were used by [5] and CNNs trained from scratch. The results suggest that the features learned from the ImageNet dataset do not provide an advantage for learning animal classifications. They propose to use pre-trained models on more similar tasks/inputs, so that Deep Learning can also be applied for data sets containing less images than Snapshot Serengeti. Further, their findings show ensembles increase the overall accuracy. One issue with this approach is the low accuracy for rare classes. The paper uses weighted loss, oversampling and emphasis sampling to overcome these problems. However, only some classes show improved results with all three methods.

4 Fine-Grained Object Recognition

All the above approaches try to solve a fine-grained problem with species identification. Even though some species are really similar, this is definitely true for individuals of the same species.

One approach for fine-grained animal identification can be found from Freytag et al. [3]. Their goal is to identify identity, age, age group, and gender of chimpanzee faces in the wild without the necessity of aligned face images. As mentioned in Sect. 2 this is often done by highly specific hand-crafted recognition pipelines, which are prone to noise. In their experiments they utilize the bilinear pooling approach from [9]. Further they use matrix logarithm on top of the CNN bilinear pooling (*LOGM*-transformation). This transformation can be seen as amplifying axes with small variances in data. Intuitively, this is ideal for identification tasks, where small parts of the image are supposed to be discriminative. However, the paper also suggests that for low-quality images bad representations are thereby amplified as well. The overall approach was to extract features, which were found in CNNs, apply bilinear pooling and optionally *LOGM*-transformation and finally pass them to a linear SVM after L2-normalizing it. Training was done by SGD with weight decay (0.0005), momentum (0.9) and learning rate 0.001 to 0.0001. Regarding preprocessing, they found that random crops of 227 px $\times$ 227 px after scaling training images to 256 px $\times$ 256 px performed worse than directly scaling images to 227 px $\times$ 227 px. The experiments support that bilinear pooling and *LOGM*-transformation show worse results on data with more noise. Additionally, they found that pre-trained models on human faces [10] show much worse results than models trained from scratch. Which is similar to [11, 12], where pre-training on ImageNet decreased the performance. Further, feature extraction before the first fully connected layer showed improved results compared to after.

A paper published based on fine-grained recognition is from Branson [1]. It is working with bird species classification on a dataset containing 11,788 images of 200 bird species. The goal was

to utilize pose normalization based on aligning detected keypoints to the corresponding keypoints in a prototype image. Therefore, they used annotated data, which included parts (namely: image, bounding box, body, head) and keypoints in order to learn how the pictures have to be warped (see Fig. 4). Based on that they learn the constellation of parts via the deformable part model (DPM) and normalize the parts accordingly.

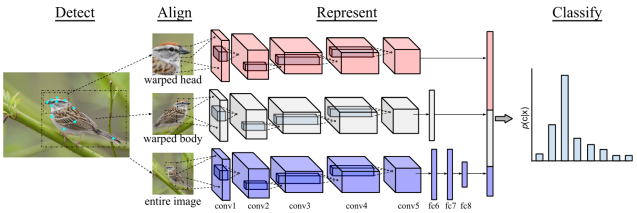

Figure 4: Branson's pipeline overview [1]

They also evaluated a pre-trained model on ImageNet and a fine-tuned version of the same. Additionally, they used a third approach, which fine-tuned only the fully connected layer and fixed the other weights. The newly learned weights were then used for initialization during the training/fine-tuning of the whole network. In general they saw significant improvements due to the use of CNN features compared to handcrafted models. However, their model drops in performance if no ground truth parts are given and detected parts are used. This means, manual annotation of parts is required, which is expensive and often not feasible. Further, they found pre-training with ImageNet to be essential in order to achieve good results, which is in contradiction to [3,11,12] Comparing both fine-tuning methods, the second approach worked more reliable in multiple trails.

The above mentioned bilinear pooling approach from Lin [9] focuses more generally on fine-grained recognition. Among others the same bird data set as in [1] was used. They describe an approach, which has two feature extractors based on CNNs whose outputs are multiplied using the outer product at each location of the image and are pooled across locations to obtain an image descriptor (see Fig. 5]). Their experiments are also different from [1] as they do not assume annotated parts or bounding boxes.

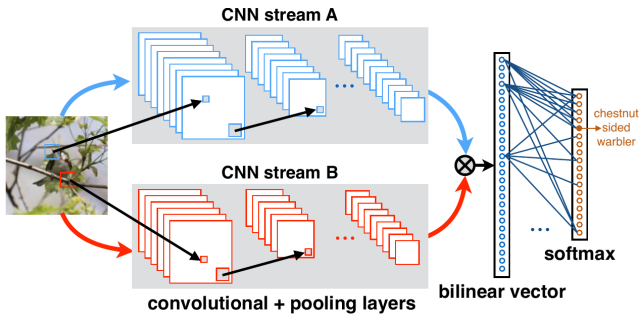

Figure 5: A bilinear CNN model for image classification. At test time an image is passed through two CNNs, A and B, and their outputs are multiplied using outer product at each location of the image and pooled to obtain the bilinear vector. This is passed through a classification layer to obtain predictions. [9]

For their three experiments they used two VGG-M-Nets, two VGG-D-Nets and a combination with one of each. The CNNs are all pre-trained on ImageNet and use only the convolutional layers, which were then combined with bilinear pooling and afterwards l2-normalized. For classification they used logistic regression or linear SVM, which can be replaced by a multi-layer neural network for non-linear outputs. Further, they compare the results for input data with annotated bounding boxes and without bounding boxes. Both results where equally good with 84.1% accuracy and 85.1% respectively for the combination of VGG-M-Net and VGG-D-Net. In addition, the two VGG-D-Nets performed equally well. Problems, which could be found, were the bilinear CNN models that are symmetrically initialized will remain symmetric after fine-tuning. Therefore, they suggest to include Dropout or dimensionality reduction with Principal Component Analysis (PCA) at the end of one CNN. With these changes the performance did not increase.

This bilinear pooling model was refined in a later paper from Lin [8]. As found in the above approach, normalization is key to reliable training and prediction. Therefore, they compare different methods of normalization after the bilinear pooling. They propose matrix logarithm and power normalization as possible alternatives (see Fig. 6.

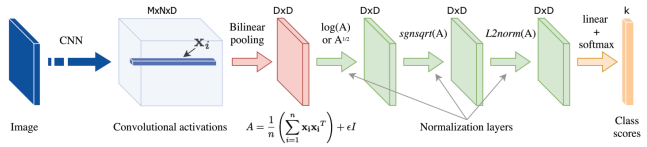

Figure 6: Improved B-CNN architecture with a log(A) or $A^{\frac{1}{2}}$, signed square-root, and l2 normalization layers added after the bilinear pooling of CNN activations. [8]

A big drawback they discovered is based on the computation of these normalizations, they normally require Singular Value Decomposition (SVD). SVD can be computed during the forward and backward pass, but is highly inefficient (approx. the same time as remaining network). They found approximations, based on Denman-Beavers iterations to solve a Lyapunov equation, are a more efficient and suffice for the classification scenario.

A similar, but different direction as [1] is proposed by Simon [13]. Their goal is to learn model parts completely unsupervised. In order to get part proposal they use pre-trained CNNs with ImageNet. Also other data sets are possible as the CNN can be pre-trained on a weakly related object dataset. The later layers in that CNN are sensitive to increasingly abstract patterns in the image. These patterns can correspond to whole objects or parts of objects, which are used as part proposals. Nonetheless, one problem is the output resolution at these layers. To solve this problem deep neural activation maps are computed, which are used consecutively in order find the part constellations. Therefore, different views of objects, i.e. a selection of part proposals are used. Shift vectors denote the offset to the common root location of an object. This is then formulated as optimization problem of selected view, the common root, the parts proposals associated with a view and if a part is visible or not (see Fig. 7).

As result they mainly found improvement, compared to a plain VGGNet without applying part constellations, when classifying the bird datasets. On other datasets the performance was similar.

In Gebru's [4] paper a domain adaption approach is proposed. It focuses on learning fine-grained recognition with no or partly labeled data. They utilize easily acquirable data, such as data from e-commerce websites, to train the network together with the original data set. The goal is to enable fine-grained recognition even for domains where labeling is not economic. Further, they try to provide an approach, which avoids training/fine-tuning for every specific topic (e.g. dogs, cats, bunnies, etc.) and instead adapt more general models to real world data. For example, images and annotations from a field guide can be used to train a model recognizing various bird species in the wild.

As shown in Fig. 8, the idea is based on two CNNs with shared weights. One is trained on the easily acquirable source data and the other on the rare target data. They also utilize several indepen-

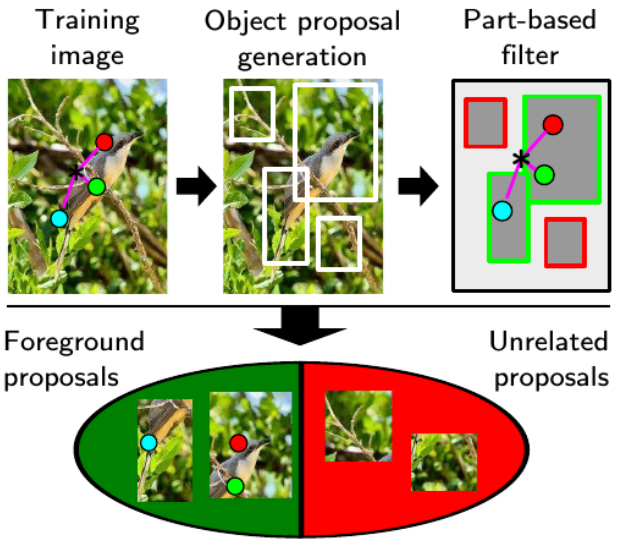

Figure 7: Overview of Simon's approach to filter object proposals for fine-tuning of CNNs [13]

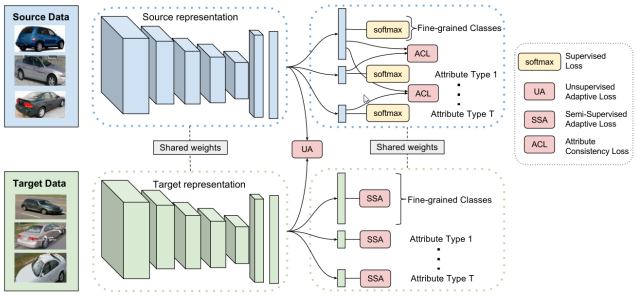

Figure 8: Architecture overview of Gebru [4]

dent softmax classification layers in order to predict the fine-grained classes for source and target. As mentioned, obtaining labels for every single class is infeasible, however, classes often share attributes. For instance, a Beagle and a Jack Russell terrier are both small dogs while a Bearded Collie and Afghan Hound are both shaggy dogs. These attributes are used to in order to identify a target's class more easily. During the learning they also included KL Divergence in order to ensure attributes and class predictions are consistent. In their pre-tests, they found that models trained on the source are infeasible for the target data. However, a joint training increased the accuracy. In general, the added attributes helped to increase performance further, as long as they are visually informative. For instance, WordNet attributes for an office related data set did not yield in the same results.

5 Conclusions and Ideas

Based on the experiments describe previously, a baseline for evaluation needs to be established. A good start for this could be to use a pre-trained VGGNet or ResNet (maybe even use Alex and GooLeNet) in order to predict individuals. In the next steps fine-tuning these models should be tried to validate, which architectures can actually help to solve this problem. If the quality seems to be too low, approaches like Deep Image Prior [14] could be worth a look. Otherwise different crops and flips of the images could improve training size and generalization. In the next stages validating the bilinear pooling model on the dataset is an option. Hereby, alternatives of the original implementation could be used, i.e. using different CNN architectures and combining prediction and features extraction in a single network instead of using SVMs afterwards. Part detection, part constellation and adaption approaches currently seem quiet complicated to implement using Keras and TF. Further, especially the later was published recently and does not provide high accuracy scores. First, the above ideas should be evaluated.

References

[1] S. Branson, G. V. Horn, S. J. Belongie, and P. Perona. Bird species categorization using pose normalized deep convolutional nets. *CoRR*, abs/1406.2952, 2014.

[2] G. Chen, T. X. Han, Z. He, R. Kays, and T. Forrester. Deep convolutional neural network based species recognition for wild animal monitoring. In *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 858–862, Oct 2014. doi: 10.1109/ICIP.2014.7025172

[3] A. Freytag, E. Rodner, M. Simon, A. Loos, H. S. Kuhl, and J. Denzler1. Chimpanzee faces in the wild: Log-euclidean cnns for predicting identities and attributes of primates. *GCPR: Pattern Recognition*, 9796, 2016.

[4] T. Gebru, J. Hoffman, and L. Fei-Fei. Fine-grained recognition in the wild: A multi-task domain adaptation approach. *CoRR*, abs/1709.02476, 2017.

[5] A. Gómez, A. Salazar, and F. V. Bonilla. Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks. *CoRR*, abs/1603.06169, 2016.

[6] A. Gomez Villa, G. Diez, A. Salazar, and A. Diaz. Animal identification in low quality camera-trap images using very deep convolutional neural networks and confidence thresholds. vol. 10072, pp. 747–756, 12 2016.

[7] B. Hughes and T. Burghardt. Automated visual fin identification of individual great white sharks. *CoRR*, abs/1609.06323, 2016.

[8] T.-Y. Lin and S. Maji. Improved Bilinear Pooling with CNNs. In *British Machine Vision Conference (BMVC)*, 2017.

[9] T.-Y. Lin, A. RoyChowdhury, and S. Maji. Bilinear cnn models for fine-grained visual recognition. In *International Conference on Computer Vision (ICCV)*, 2015.

[10] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. 1:41.1–41.12, 01 2015.

[11] M. S. Norouzzadeh, A. Nguyen, M. Kosmala, A. Swanson, C. Packer, and J. Clune. Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. *CoRR*, abs/1703.05830, 2017.

[12] M. S. Norouzzadeh, A. Nguyen, M. Kosmala, A. Swanson, C. Packer, and J. Clune. Automatically identifying wild animals in camera trap images with deep learning. *CoRR*, abs/1703.05830, 2017.

[13] M. Simon and E. Rodner. Neural activation constellations: Unsupervised part model discovery with convolutional networks. *CoRR*, abs/1504.08289, 2015.

[14] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky. Deep image prior. *CoRR*, abs/1711.10925, 2018.

[15] B. G. Weinstein. A computer vision for animal ecology. *Journal of Animal Ecology*, 87(3):533–545. doi: 10.1111/1365-2656.12780

[16] X. Yu, J. Wang, R. Kays, P. A. Jansen, T. Wang, and T. Huang. Automated identification of animal species in camera trap images. *EURASIP Journal on Image and Video Processing*, 2013(1):52, Sep 2013. doi: 10.1186/1687-5281-2013-52