Elephace: Elephant Re-identification Using Deep Learning

Deepak Duggirala
Indiana University
deduggi@iu.edu

Gandhali Marunmale
Indiana University
gamarunm@iu.edu

Abstract

Our project's purpose is to create a vision model that can be used to re-identify elephants using images. Researchers can use this model to study elephant movement patterns in zoos and sanctuaries. This non-invasive method of identifying elephants is preferable than catching and attaching devices to them, which necessitates human intervention. Manually recognizing elephants from photos takes time, thus a robust automated model is essential. We created an elephant re-identification model using three deep learning algorithms and ran experiments to determine the best model parameters. Our primary dataset (aka zoo dataset) provided by Prof. Chusyd. We have built a classification model to classify an image into one of the known elephant categories. With ResNet50 as the backbone and our primary dataset, this model's top-1 accuracy was 99.71%. We aimed to recognize elephants with as few images as possible in our next approach, therefore we constructed a few shot learning model. Using just 3 images per class we obtained top-1 accuracy of 81% and top-3 accuracy of 92% on our primary dataset. The performance of few shot on more complex wild elephants dataset was not as good as zoo elephants so we moved to training Siamese network using triple loss (ala FaceNet). With the best hyperparameter choices, we got validation rate of 0.585 at false acceptance rate of 0.01.

1. Introduction

Re-identification of animals is necessary for biodiversity monitoring and ecological research projects. Biologists and anthropologists need animal tracking to monitor their health, behavior, group dynamics and variation of population over time [4]. Biodiversity development and health information is highly valuable for assessing the environmental effect and necessary steps required to preserve the ecosystems.

Animal tracking can be performed by using a device that is attached to the animal. This technique requires human intervention and it is intrusive. It requires capturing animals which can be dangerous and also disruptive for an animal's wild habitat [4]. Another method of tracking animals is by capturing the images and re-identifying animals from those images. The re-identification performed manually can be a time consuming process as multiple and vast numbers of images need to be analyzed. This re-identification requires domain knowledge about the particular species and also its prone to biases related to human judgment [Schneider, Stefan, et al.]. [12]

Automated re-identification of animals provides a better solution to animal tracking without involving disruptive techniques and human errors. There has been ongoing research to extend human face recognition and identification techniques to animal identification using deep learning. There are many challenges while building a state of the art animal re-identification and classification model. These challenges arise due to similar looking animals and very small distinctive features like body size, scars and marks, coloring, etc [Körschens, et al.]. Apart from this, these features could also change over time. For example in the case of elephants, it could lose a tusk and a hole in the ear may become a rip [Körschens, et al.]. Other challenges are variations in image captured such as occlusion, varying viewpoints, different poses and angles of the animal in an image that affect the detection of distinctive features. For example in the case of elephants the feature might not be clearly visible in images because of mud on their bodies or the angle and movement [Körschens, et al.].

In this project, we have build a model for elephant re-identification system using bounding box detection architecture for bounding box predictions followed by multiple deep learning network architectures for elephant re-identification.

2. Background and related work

In the context of animal re-identification, various works have used custom feature engineering and classical computer vision techniques. Recently there has been some work on using the advanced and already mature human face recognition techniques e.g. FaceNet, DeepFace for animal recognition and identification tasks.

Korschens et al [5] used a pre-trained YOLO network

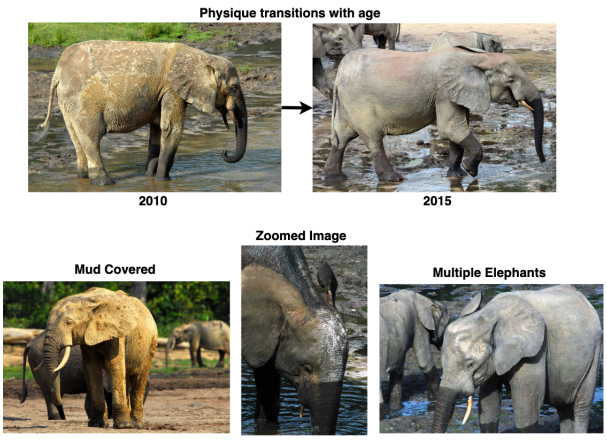

Figure 1. Clockwise: Elephant Physique transitions with age, Elephant covered in mud, zoomed image, multiple animals in an image

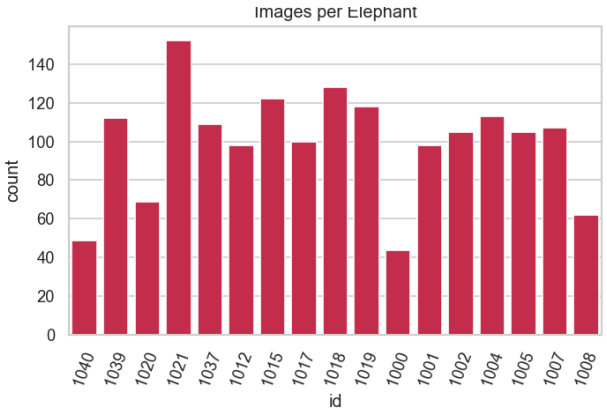

Figure 2. Zoo Dataset: Images per Elephant

for bounding box predictions to automatically locate an elephant's heads in images using pre pre-trained YOLO network.Used a human in the loop approach to select the best bounding box and then the bounding box is cropped and fed into a pre-trained ResNet 50 for feature extraction from the middle layer followed by pooling. PCA is used to reduce the number of features and then SVM is used for classification. To improve the accuracy, results of multiple images of the same elephant are aggregated by averaging the class wise confidence. The Top 1 accuracy for classification is 56% and 74% with two image aggregation. For the same task but using traditional methods [Ardovini et al][Schneider review] - performed re-identification of elephants using multi-curve matching technique and achieved top-1 accuracy for 75%.

For primate re-identification Debayan Deb et al [4] implemented a variation of a sphereface deep recognition to build embeddings of faces which then can be compared using similarity functions. Instead of using an object detection network to identify the faces, they manually annotated landmarks to align the faces. They achieved a closed set accuracy of 75.82% on the chimpanzee dataset.

For identification of giant pandas, [Jin Hou et al] [9] used significant data augmentation and trained a VGGnet for classification tasks. They were able to accurately identify 90% of panda individuals.

For the difficult task of Nyala identification, [Zyl et al] implemented an end to end system consisting of Faster R-CNN model for bounding box detection and then siamese network using ResNet-152 backbone for image embedding. Their top1 accuracy for zebra dataset is 74.1% and top-10 accuracy of 85% for Nyala animal identification.

[Freytag et al] uses matrix logarithm transformation on bilinear pooling to increase the discriminability of features in Alexnet and VGGfaces networks. Using these techniques, they achieved the state of the art for chimpanzee faces classification. For a similar task, Gorilla faces classification [Brust et al] uses YOLO and R-CNN for bounding box detention along with AlexNet for feature extraction upon which they used SVM for class label prediction.

3. Methods

3.1. Dataset

Our primary dataset is provided by Prof Daniella Chusyd, Assistant Professor, School of Public Health. This dataset consists of 17 individual zoo elephants with a total of 1691 images. We refer this dataset as zoo dataset. Graph 2 shows distribution per class of these images. On an average there are 99 images per elephant. The standard deviation is 28 images. Minimum images per class is 44 and maximum is 152. The histogram 3 depicts these variations. After initial analysis of the elephant images, it is found that there is little variation in images of some of the elephants as shown in figure 4.

To enhance the variability in images, we researched and found Elpephants dataset [10]. We acquired this elephant data by requesting Matthias Koerschens. This dataset consists of 274 individual elephants with a total of 2074 images. We refer to this dataset as wild dataset. On an average we have 7 elephants per class. The histogram 5 depicts the distribution of images per class. This dataset has elephant images taken from various angles and backgrounds. Thus they provide variation which will help the network learn nuisance in the dataset. The figure 6 depicts three different views captured for an elephant.

3.2. Elephant Detection and Extraction

In order to accurately identify elephant in an image, it is essential to isolate it from surrounding which might contain

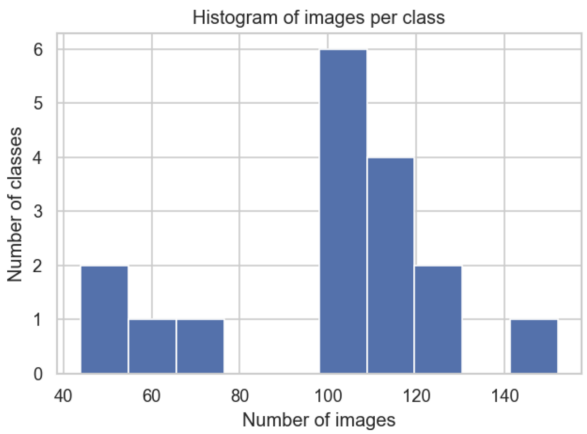

Figure 3. Zoo Dataset: Histogram of images per class

Figure 4. Zoo Dataset: Images of elephant id 1001

other elephants, animals or people. For this reason, a pretrained object detection model is used to obtain bounding boxes for all the elephants in the image. If there are multiple elephants detected, then the elephant with the bounding box of largest area is considered to be the subject and extracted.

Yolov4 [1], Faster R-CNN Inception ResNet V2 1024x1024 [2], and SSD MobileNet v2 320x320 [3] pretrained models were tried to detect and get bounding boxes for the elephants. The Yolov4 pretrained model is obtained from [16] and Faster R-CNN Inception ResNet V2

Figure 5. Wild Dataset: Histogram of images per class

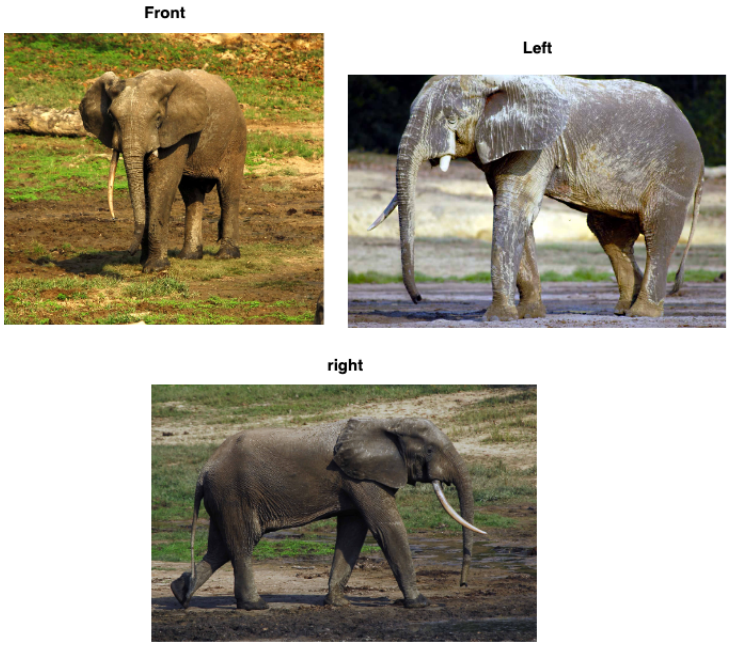

Figure 6. Wild Dataset: Images of elephant id 35

1024x1024 and SSD MobileNet v2 320x320 are obtained from Tensorflow Object detection zoo [15]. It is found that for these datasets, bounding boxes obtained from Yolov4 often times did not include the head of the elephant see fig:7. For this reason, Faster R-CNN and SSD MobileNet are used to extract the elephant for the next phase.

To re-identify an elephant given some of its images we consider three approaches - classification, few-shot recognition, and Siamese network for similarity measurements.

3.3. Classification

When the re-identification problem is framed as a classification problem, the model is trained using labelled photos

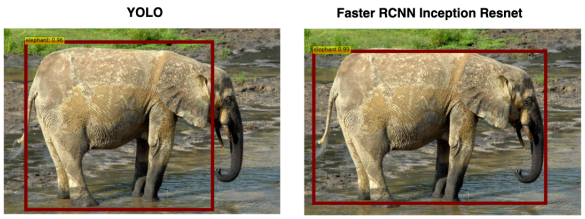

Figure 7. Bounding Box Detected by Yolov4 and Faster RCNN Inception ResNet V2

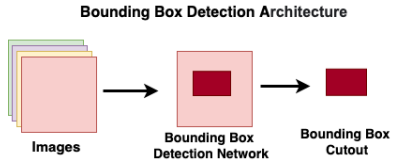

Figure 8. Elephant Bounding Box Detection

of elephants. The train and validation sets will both include disjoint images of the same elephant, and the model must learn discriminative features and use them to predict the labels of the images in the validation set. Our classification model is made up of a pre-trained CNN as a backbone that provides image embeddings and a pair of fully connected layers with softmax that classifies the images into one of the elephants from the train set. The model is trained using cross-entropy loss and L2 regularization for fully connected layers.

3.4. Few Shot Learning

When just a few images are provided for each elephant, the train set used for classification becomes unbalanced, making learning difficult. A few shot learning approach is used to tackle this challenge, in which embeddings from a pre-trained CNN of known images (support set) are compared to the embeddings of unknown images (query set) using a suitable similarity measure [17]. If the support set contains k classes and has n images per class then the problem is called n-shot k-way classification.

The embeddings of all the images in the support set S are obtained and normalized using a pre-trained model f. The mean of all L2 normalized embeddings of a given class will act as its representative. To classify an image from the query set, its embedding is compared to the mean embeddings of all classes, and the class with the closest embedding is selected as the prediction. Dot product or cosine similarity can be used to calculate similarity.

$$w_{S_k} = \sum_{s \in S_k} \frac{\|f(x_s)\|}{|S|}$$

$$z_{S_k}^{(q)} = w_{S_k}^T \|f(x_q)\|$$

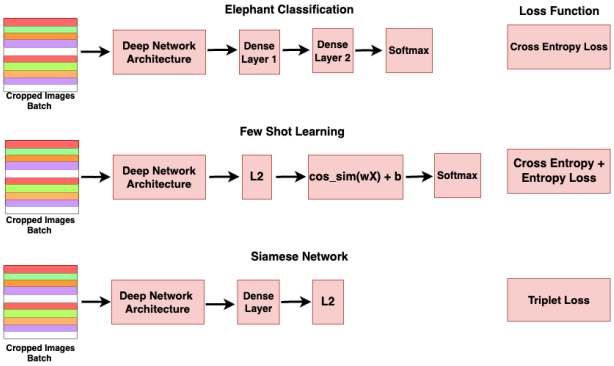

Figure 9. Elephant re-identification architectures

$$\hat{y}^{(q)} = softmax(z^{(q)})$$

w_{S_k} is the mean embeddings of the class k in the support set S. $\hat{y}^{(q)}$ is the prediction for image x_q in the query set Q.

This base model can be improved by learning W as well as a bias vector b from the support set using back propagation.

$$\hat{y}^{(q)} = softmax(\frac{W^T \|f(x_q)\|}{\|W\|} + b_{S_k})$$

Along with cross-entropy loss, using an entropy regularization will improve the performance [20], [18], [19] because it will force the model to be more sure about the predictions it is making.

$$\mathcal{L} = \sum_j \sum_i (y_i^{(j)} log(\hat{y}_i^{(j)}) + \hat{y}_i^{(j)} log(\hat{y}_i^{(j)}))$$

where $y_i^{(j)}$ and $\hat{y}_i^{(j)}$ is the i^{th} component of the true label and predicted labels of j^{th} example respectively.

3.5. Siamese Network

Facenet paper
triplet loss

$$\mathcal{L} = \sum_i^N \left[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right]$$

batchall
batch hard
hard positive: $x_i^P = \text{argmax}_{x_i^p} \|f(x_i^a) - f(x_i^p)\|_2^2$
hard negative: $x_i^N = \text{argmin}_{x_i^n} \|f(x_i^a) - f(x_i^n)\|_2^2$
batch partial hard

	Zoo Elephants			Wild Elephants		
	top-1	top-3	top-5	top-1	top-3	top-5
ResNet50	0.997	0.997	0.997	0.210	0.325	0.383
InceptionV3	0.979	0.991	0.994	0.159	0.243	0.272

Table 1. Accuracy results on Zoo and Wild elephants datasets using Classification for re-identification

4. Experiments and Results

4.1. Elephant Classification

For elephant classification model, we used two backbone models of ResNet50 and InceptionV3. We found that both the model performed well on zoo elephants dataset with top-1 accuracies of 0.9971 and 0.979 respectively. This model did not perform well on wild dataset with top-1 accuracies of 0.21 and 0.16 respectively. The top-1, 3, and 5 accuracies for both zoo and wild dataset are shown in Table 1

4.2. Few Shot Learning

In Few shot learning, we have used two backbone architectures of ResNet50 and InceptionV3.Both these model performed well on zoo elephants and didn't perform well on wild elephants. Table 2 and 3 shows the accuracy results for zoo and wild elephant dataset respectively. Next, we trained the models with and without image augmentation.Image augmentations consisted of left-right flip, small random changes to saturation and hue as shown in figure 10. We observed that performing image augmentation improved the accuracy by 17%. We experimented with preserving the aspect ratio of images by padding them as shown in figure 11 but it caused degradation of performance. We fine-tuned the model with entropy regularization and observed that the accuracy improved by 35%. Graph 12 shows the performance of model for various support set with and without fine-tuning implementation. For few shot learning, we have used support set size of 1, 2, 3, 5, 7, and 9. Graph 13 shows the Top-1 accuracy for model for various support sizes. It was observed that the support size of just 3 performed well with top-1 accuracy of about 81% on zoo dataset. Furthermore, the top-1 accuracy of 99% was achieved with the support set of 7.

4.3. Siamese Network

In Siamese Network, we have experimented with two backbone architectures of ResNet50 and InceptionV3 along with image augmentation. While training siamese network, we experimented with various triplet loss margins of 0.2, 0.5, 0.75 and selected margin of 0.5. We used triplet loss strategies of batch triplet loss and hard triplet loss. We observed the model performance using two optimizer: Adam and SGD and chose Adam optimizer. The model's performance was then tested with different embedding sizes: 64,

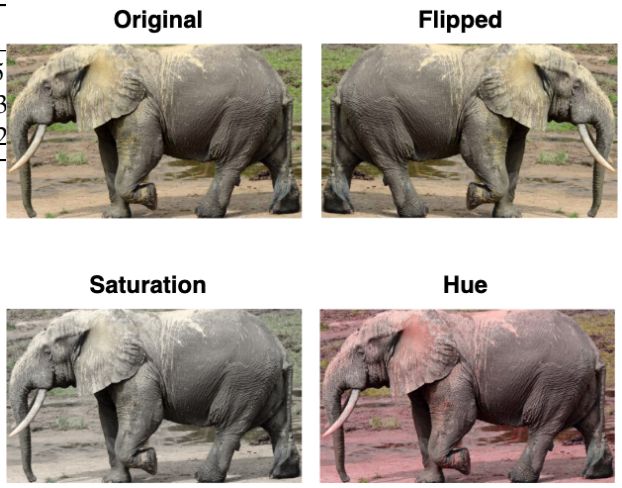

Figure 10. Image augmentations performed are right-left flip, saturation and hue changes

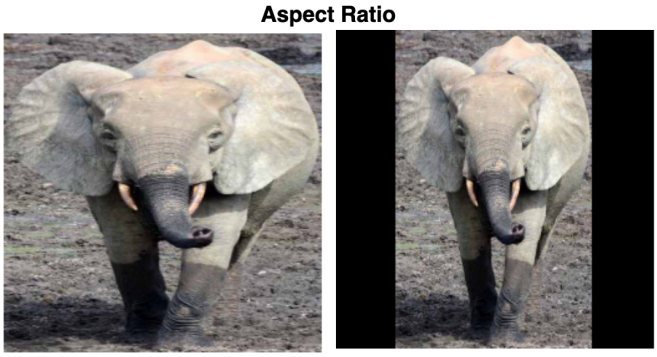

Figure 11. Image Padding: To preserve Aspect Ratio

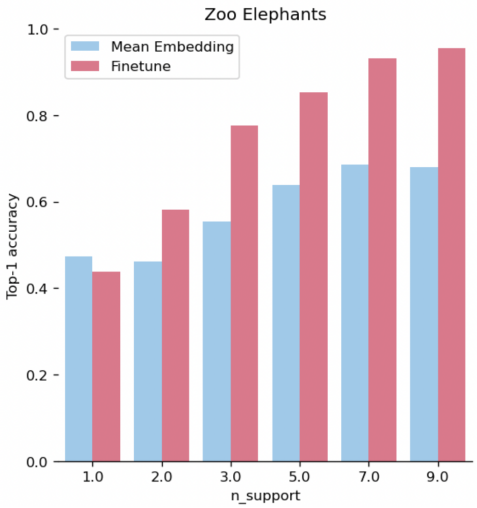

Figure 12. Fine-tuning with entropy regularization

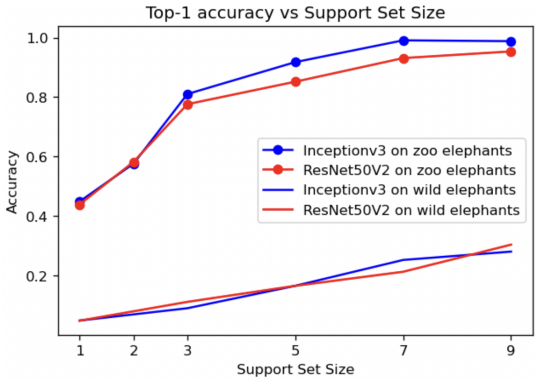

Figure 13. Graph of Top-1 Accuracy with support set size

	ResNet50			InceptionV3		
n	top-1	top-3	top-5	top-1	top-3	top-5
1	0.439	0.572	0.703	0.449	0.585	0.659
3	0.776	0.9	0.943	0.810	0.923	0.959
5	0.852	0.928	0.955	0.918	0.966	0.976
7	0.931	0.978	0.989	0.991	0.995	0.995
9	0.953	0.985	0.992	0.998	0.998	0.993

Table 2. Accuracy results on Zoo elephants datasets using Few Shot for re-identification

	ResNet50			InceptionV3		
n	top-1	top-3	top-5	top-1	top-3	top-5
1	0.047	0.086	0.106	0.048	0.088	0.112
3	0.111	0.179	0.223	0.089	0.170	0.220
5	0.165	0.264	0.323	0.165	0.255	0.321
7	0.212	0.342	0.413	0.252	0.408	0.478
9	0.303	0.446	0.525	0.280	0.460	0.510

Table 3. Accuracy results on Wild elephants datasets using Few Shot for re-identification

Embedding Size	VAL
64	0.441
128	0.513
256	0.585

Table 4. Effect of embedding size on Validation rate (VAL) at a fixed False Acceptance Rate (FAR) of 0.01

128 and 256, with the 256 embedding size providing the best results. Table **??** shows the VAL rate for embedding sizes of 64, 128 and 256. With the hyperparameter choices of embedding size: 256, margin: 0.5, Batch size: 128, Optimizer: Adam, we got validation rate of 0.585 at false acceptance rate of 0.01.

5. Discussion

For elephant detection architectures, we used Yolov4 [1], Faster R-CNN Inception ResNet V2 1024x1024 [2], and SSD MobileNet v2 320x320 detection networks and observed that Faster R-CNN and SSD generated precise bounding boxes compared to YOLOV4.

Features obtained from deep learning architectures pretrained on ImageNet are good differentiators of elephants from other classes. We observed that these features served as a good prior for distinguishing elephant categories.

We observed that image augmentation consisting of left-right flip, small random changes to saturation and hue improved the performance of models. For few shot models, image augmentation helped to improve accuracy by 17%

We showed that few shot learning model uses fewer images per class to train and performs better than classification model using same deep learning architecture as backbone.

In few shot learning, we observed that using as few images as three per elephant class generated good results with top-1 accuracy of 81% and top-3 accuracy of 92%. With use of 7 images per elephant class, we achieved top-1 accuracy of about 99%

We observed that using entropy regularization to fine-tune few shot learning increased accuracy by 35%.

We observed that siamese model overfits the data. The training of siamese network will require larger dataset to gain higher validation rate.

References

[1] Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. "Yolov4: Optimal speed and accuracy of object detection." arXiv preprint arXiv:2004.10934 (2020). 3, 6

[2] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems 28 (2015). 3, 6

[3] Liu, Wei, et al. "Ssd: Single shot multibox detector." European conference on computer vision. Springer, Cham, 2016. 3

[4] Deb, Debayan, et al. "Face recognition: Primates in the wild." 2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS). IEEE, 2018. 1, 2

[5] Korschens, Matthias, Bjorn Barz, and Joachim Denzler. "Towards automatic identification of elephants in the wild." arXiv preprint arXiv:1812.04418 (2018). 2

[6] Freytag, Alexander, et al. "Chimpanzee faces in the wild: Log-euclidean CNNs for predicting identities

and attributes of primates." German Conference on Pattern Recognition. Springer, Cham, 2016.

[7] Brust, Clemens-Alexander, et al. "Towards automated visual monitoring of individual gorillas in the wild." Proceedings of the IEEE International Conference on Computer Vision Workshops. 2017.

[8] Van Zyl, Terence L., Matthew Woolway, and B. Engelbrecht. "Unique animal identification using deep transfer learning for data fusion in siamese networks." 2020 IEEE 23rd International Conference on Information Fusion (FUSION). IEEE, 2020.

[9] Hou, Jin, et al. "Identification of animal individuals using deep learning: A case study of giant panda." Biological Conservation 242 (2020): 108414. 2

[10] Korschens, Matthias, and Joachim Denzler. "Elpephants: A fine-grained dataset for elephant re-identification." Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. 2019. 2

[11] Schneider, Stefan, Graham W. Taylor, and Stefan Kremer. "Deep learning object detection methods for ecological camera trap data." 2018 15th Conference on computer and robot vision (CRV). IEEE, 2018.

[12] Schneider, Stefan, et al. "Past, present and future approaches using computer vision for animal re-identification from camera trap data." Methods in Ecology and Evolution 10.4 (2019): 461-470. 1

[13] Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[14] Koch, Gregory, Richard Zemel, and Ruslan Salakhutdinov. "Siamese neural networks for one-shot image recognition." ICML deep learning workshop. Vol. 2. 2015.

[15] Tensorflow Object Detection Zoo: https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md

[16] https://github.com/sicara/tf2-yolov4 3

[17] Chen, Liu, Kira, Wang, & Huang. A Closer Look at Few-shot Classification. In ICLR, 2019. 3

[18] Dhillon, Chaudhari, Ravichandran, & Soatto. A baseline for few-shot image classification. In ICLR, 2020. 4

[19] Chen, Wang, Liu, Xu, & Darrell. A New Meta-Baseline for Few-Shot Learning. arXiv, 2020. 4

[20] Few-Shot Learning Lectures, Shusen Wang: https://www.youtube.com/watch?v=U6uFOIURcD0 4

4