Brief Paper

GorillaVision – Open-Set Re-Identification of Wild Gorillas

Lukas Laskowski [1,†] *Rohan Sawahn* [1,†] *Maximilian Schall* [1,*] *Dante Wasmuht* [2] *Magdalena Bermejo* [3,4] *Gerard de Melo* [1]

[1] *Hasso Plattner Institute, University of Potsdam, Prof.-Dr.-Helmert-Str. 2-3, Potsdam, Germany*
[2] *Conservation X Labs, 20007 Washington DC, USA*
[3] *SPAC Scientific Field Station Network, Hasso Plattner Foundation (HPF), 14467 Potsdam, Germany*
[4] *Department of Ecology and Environmental Sciences, University of Barcelona, 08028 Barcelona, Spain*
[†] *Equal contribution*
** E-mail: Maximilian.Schall@hpi.de*

Abstract: This paper presents GorillaVision, an open-set re-identification system for gorillas in the wild. Open-set re-identification is crucial to identify and track individual gorillas the system may not have previously encountered, thereby enhancing our understanding of gorilla behavior and population dynamics in dynamically changing wild environments. The system adopts a two-stage approach, in which gorilla faces are first detected with a YOLOv7 detector and subsequently classified with a custom neural network model. The classification model is based on a pre-trained Vision Transformer, which is fine-tuned with Triplet Loss to compute embeddings of gorilla faces. Such embeddings can be relied upon to obtain a similarity measure between gorilla faces and thus also between individual gorillas. Classification is then performed on these embeddings with a k-nearest neighbors algorithm. We evaluate our method on two heterogeneous datasets and show that our approach yields minor gains over the state-of-the-art YOLO detector in a closed-set scenario. In an open-set scenario, our model can deliver high-quality results with an accuracy of 60 to 90%, depending on the dataset quality and the number of individuals. Our code is accessible on `https://github.com/Lasklu/gorillavision`.

1 Deep Learning in the Wild

In light of the current ecological crisis and the increase in space taken up by human society, many wild animals are in danger of becoming extinct. According to the *Red List of threatened species* issued in 2008 by the International Union for the Conversation of Nature, more than 25% of species risk extinction [12, 37]. Primates are particularly endangered, with about 60% of primate species under the threat of extinction [12]. Eastern and western gorillas are classified as endangered species with a decreasing population size [37]. To adopt appropriate conservation measures, a thorough understanding of the population in terms of abundance, species distribution, and its behavior in the habitat is essential [8]. This, among other things, leads to a need to monitor the population. However, many established methods to achieve this require expert knowledge and effort to manually process collected data, for instance, tagging gorillas or analyzing DNA to re-identify them. Modern re-identification approaches for wildlife use camera traps that capture images or videos of wildlife without being intrusive. Typically, substantial expertise is necessary to discern the identity of individual animals in such footage. Over the past years, computer vision has advanced numerous deep learning approaches to re-identify objects, humans, and animals [8, 9, 24, 25, 28]. These approaches exhibit substantial potential for large-scale automated monitoring of wildlife. However, changes in illumination, image resolution, and environmental interference make re-identification challenging. Moreover, not all individuals of a wild population are typically known when starting the monitoring process (*open-set problem*), as new gorillas may be born or unknown individuals may appear during monitoring. Novel approaches for re-identification do support open-set re-identification, in which individuals not present in the training set can be distinguished during classification. There is prior work on deep learning approaches to re-identifying primates, such as chimpanzees, lemurs, or golden monkeys [9]. However, open-set re-identification for gorillas in the wild has not previously been addressed. In this paper, we present **GorillaVision**, a two-stage system for re-identifying gorillas in the wild that is able to generalize to gorillas not included in the training set. The first stage detects the faces of gorillas based on the *YoloV7* Object Detection System developed by Wang et al. [38]. The second stage uses a Vision Transformer-based architecture along with a *Triplet Loss* to compute high-dimensional embeddings, which serve as input for a k-nearest-neighbor classifier. The latter is used to label the detected gorillas. With the help of an automated re-identification system of gorillas, existing databases of wildlife footage could be analyzed to aid further research in gorilla conservation measures.

Fig. 1: Example of an image captured by a camera trap in the rainforest in the Republic of Congo.

2 Background & Related Work

Object Detection: Detecting relevant objects of interest within an image is one of the most fundamental tasks in computer vision. Region-based deep Convolutional Neural Networks (R-CNNs) are among the most popular methods for object detection, with numerous variants seeking to improve their efficiency, performance, and

robustness against lighting conditions and varying environments. Examples of these include Mask-RCNN, Faster R-CNN, and Granulated R-CNN [3, 14, 16, 26]. A faster approach compared to R-CNN is the YOLO Real-Time Object Detection system, which is based on a single pass of the image through the model. The approach divides the image into a grid and predicts weighted probabilities for bounding boxes over this grid [27, 39]. Other approaches with comparable performance include the Single Shot MultiBox Detector (SSD) and RetinaNet [19, 21].

Human Face Identification: Over the past three decades, human face recognition and identification have been a prominent area of research within computer vision [1]. Modern face re-identification approaches often begin with face recognition. This is then followed by face identification, which usually involves learning meaningful features of the input image with the help of CNNs and then extracting a generalized representation of the faces [24]. Such representations are mapped to a vector space in which images belonging to the same class have a low Euclidean distance and images from different classes have a high distance. This is a form of distance metric learning, in which the goal is to minimize intra-class variance in the learned representations, while maximizing inter-class variance [18, 29]. In order to identify to which individual a face belongs, a nearest neighbor search can be conducted with optimized search algorithms [4]. This approach allows identifying individuals that are not present in the training set. DeepFace is one of the earlier deep learning approaches for face re-identification, which was outperformed by VGGFace and FaceNet [31, 32, 34]. FaceNet incorporates a *Triplet Loss*, which seeks to minimize the distance between samples from the same class and maximize the distance between samples from different classes, thus directly learning a generalizing similarity network [29]. A comparable approach is the *Contrastive Loss* for Bi-Encoder Networks [5, 29], which, however, was found to be outperformed by Triplet Loss for the domain of human face re-identification [17].

Animal Re-Identifcation: Re-identifications methods are not limited to general objects or humans, but have as well been applied to animals in areas such as cattle re-identification [2], pet re-identification [35], and wildlife monitoring [8]. Approaches here mostly focus on using facial features for identification, but are not restricted to these. Moskvyak et al. show the benefits of using pose invariant embeddings from features such as patterns on the body [23]. Guo et al. show the usage of R-CNN for face detection of primates in the wild and identification with tri-attention networks [15]. Triplet Loss has been shown to be a suitable loss function, performing well in learning embeddings that generalize well [24, 28]. Furthermore, the identification of the pose of the individual is able to improve results, by using the estimated pose to normalize the image [20, 40]. Deyban et al. present a new method called *PrimeNet* for re-identification of primates based on faces and show the success for lemurs, golden monkeys, and chimpanzees [9].

Great Ape Recognition and Re-Identification: Gorillas belong to the family of the great apes; however, there is only limited work on gorilla recognition and even less on re-identification, in part due to the lack of a publicly available dataset. More extensive work exists for chimpanzees, with datasets such as the C-Zoo and C-Tai dataset*. Freytag et al. present a system for identifying chimpanzees in the wild using log-Euclidean CNNs to learn facial representations [13]. A further deep learning approach based on Single Shot MultiBox Detectors uses video tracks to exploit available information for facial re-identification of chimpanzees [30]. As mentioned above, PrimeNet has also been shown to successfully achieve a re-identification of chimpanzees [9]. The only work on the topic of gorilla re-identification to the best of our knowledge is the work by Brookes and Burghardt [6] on gorillas from the Bristol Zoo, and the work by Brust et al. [7]. Brookes and Burghardt use a YoloV3 model for face recognition and identification, while Brust et al. use the YOLO model for face detection, extract features with a BVLC AlexNet model, and then use a support vector machine for classification. With this approach, the latter achieved a re-identification accuracy of 62.4%; however, their dataset is not publicly available [8]. Importantly, both approaches lack the ability to identify new gorillas that are not present in the training set (open-set identification), which is what our approach seeks to solve.

*https://github.com/cvjena/chimpanzee_faces

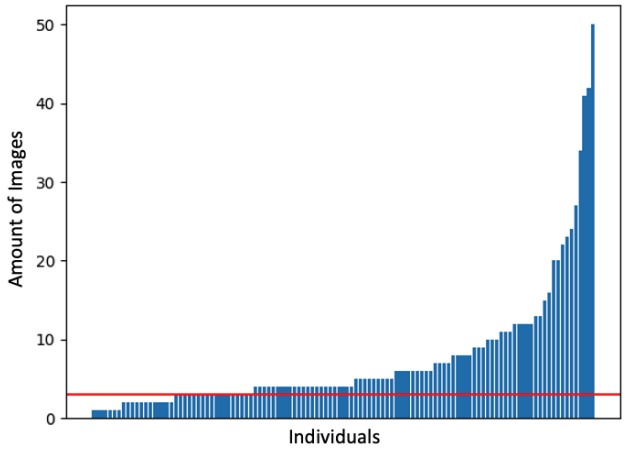

Fig. 2: Amount of images per class in *SPAC-Gorilla* dataset. The red line marks the threshold of three that we require for individuals in our dataset.

3 Datasets

We evaluate our model using data from two sources: *SPAC-Gorilla*, which contains a large number of individuals in the wild with only a few images per individual, and *Bristol**, which is publicly available and features a few zoo-housed individuals, but with a large number of images per individual. Both datasets lack landmark annotations, such as eyes, nose, and mouth, which could be used to align images for improved predictions [9, 23].

Bristol*:* The *Bristol* dataset consists of 5,400 images from seven individuals with an average of around 771 images per individual. The images are frames from video footage captured for over six weeks at the Bristol Zoo and manually annotated by experts from the zoo's primate division [6]. The labels of the dataset represent the bounding boxes around the gorilla faces. However, the gorilla face images have high variance in resolution, with an uneven distribution over the different classes (individuals).

SPAC-Gorilla*:* The *SPAC-Gorilla* dataset contains 823 images from 96 individuals, with an average of eight images per individual. The images in this dataset are manually annotated frames from video footage done by wildlife researchers. The footage was captured at Odzala-Kokoua National Park, Republic of the Congo, over a time span of 6 months with camera trapping devices. The original data exhibits a high variance in the number of images per individual, with some individuals having only one image, while others have 46 images. For simplicity, in the train, validation, and test splits, we prune away individuals with fewer than three images. This also serves to conform with our loss function (see Section 4.2). The data distribution is plotted in Figure 2, with the red horizontal line indicating the threshold of three images per individual.

*https://data.bris.ac.uk/data/dataset/jf0859kboy8k2ufv60dqeb2t8

3.1 Dataset Splits

In the following, we briefly describe the most relevant splits and the hypotheses we attempt to evaluate. All datasets are based on cropped images of gorilla faces and are used for Stage II of our approach, while Stage I merely uses the Bristol Zoo training data.

The datasets are split into three sets for the identification phase: *train*, *database*, and *eval*. The *train* set is further split into training and validation subsets, which are used to train the model to produce embeddings that enable the subsequent analyses. The *database* set consists of samples that are used to train the classification algorithm based on the previously obtained embeddings. The *eval* set is used for final model evaluation and computing quality metrics.

We distinguish two task setups: The *open-set* and *closed-set* scenarios. In the *closed-set* scenario, we sample a certain percentage p of all available images to construct the training set. The remaining $1-p$ percent of the data is used for evaluation. In order to avoid overrepresenting certain classes from our imbalanced dataset, the *evaluation* set is constructed by randomly sampling p percent of the images per individual. In the *open-set* scenario, we select p percent of classes (individuals) for the training. The training set is then formed by taking all available images of each of these individuals. For the remaining $1-p$ percent of individuals, 30% of the images of that individual are used for evaluation, and the remaining 70% constitute the database. Additionally, we always add the training samples to the *database* set for both scenarios. This makes the datasets closer to the real-life scenario, in which as many classes (individuals) as possible are in the database, thus making the classification task more challenging.

3.2 Dataset Subsets

Additionally to *SPAC-Gorilla* and *Bristol* we use several additional subsets for our experiments:

1. *SPAC-Gorilla > 6*: The *SPAC-Gorilla* dataset with all individuals that have more than six images. This dataset is used in combination with *SPAC-Gorilla* to explore how individuals with only a few images (<6) affect the results.
2. *Bristol 10 (20, 50, 100, 200, 400) Images*: These datasets contain a precise predetermined amount of images per individual from the *Bristol* dataset. We aim to use this to investigate to what extent the specific number of images available in the training set influences the prediction results.
3. *Bristol-SPAC-Gorilla*: This dataset contains all individuals from both the *Bristol* dataset and the *SPAC-Gorilla* dataset. However, the *database* and *eval* set contain only images from the *SPAC-Gorilla* dataset. We use this dataset to analyze if the results improve by incorporating more individuals, especially individuals with many more samples, to the training set.

4 GorillaVision

In this section, we provide a comprehensive overview of the architecture of the GorillaVision system, including data preprocessing steps, loss calculation, and details on the specific implementation of face detection and classification.

4.1 Overall Architecture

GorillaVision identifies gorillas based on a two-stage approach, as shown in Figure 3: First, it applies a face detection model on raw images in the wild to detect the presence of gorillas (Detection Phase). For this, we use the real-time object detector YoloV7 [38], which we fine-tune on the 5,400 labeled gorilla faces included in the Bristol Zoo dataset. The detected faces are then cropped, and passed to the second stage, which classifies the detected faces (Identification Phase).

As described in the introduction (see Section 1), there is a requirement that not only known gorillas should be identified but also those that are not yet known to the model. To meet this requirement, the second stage of our system learns embeddings such that images of the same individual are located close to each other, and images from different individuals are located further away. Thus, we build a representation of each individual in a high-dimensional vector space that represents semantic and contextual information, similarly as in the FaceNet architecture [31].

The embedding computation stage consists of several steps: The image of a cropped face of a gorilla serves as input to the model, since experts are also able to re-identify gorillas solely on this information. During training, data augmentation is applied to this input. We use geometric transformations such as rotations and horizontal flips, intensity augmentation such as applying random planckian jitter to change the illumination, and erasing, which covers random parts of the image with black boxes. With this, we aim to account for the low amounts of data per individual and allow our model to generalize better. The pre-processed image is then fed into a Vision Transformer. We use the vanilla Vision Transformer (ViT) proposed by Dosovitskiy et al. [11]. On top of the it, we include a fully-connected linear layer, which calculates the embeddings from the output of the ViT. Finally, we apply a k-nearest-neighbor classifier to the embeddings to determine the label of each gorilla.

4.2 Loss Computation

To train the embedding model, we rely on the *Triplet Loss*, a loss function for learning effective representations of data that is widely used in re-identification. The basic idea is to learn the embeddings such that instances of images of the same gorilla reside closer together in the vector space, while instances of different gorillas remain further apart. This requires a loss function that maximizes the distance between embeddings of images of the same gorilla and minimizes the distance between embeddings of different gorillas. To achieve this, we need three images for the calculation of the loss, which together form a triplet:

- **Anchor Input** A: The reference input, which is compared to two other images
- **Positive Input** P: An image sharing the same label as the anchor input
- **Negative Input** N: An image that has a different label than A (and P)

Let $\mathcal{T}$ be the set of all triplets. Then, for any triplet in $\mathcal{T}$, it should hold that the embedding of the anchor is closer to the embedding of the positive input than it is to the embedding of the negative input by some margin α. This is encouraged by the following constraint:

$$d(f(A), f(P)) + \alpha < d(f(A), f(N)), \forall (A, P, N) \in \mathcal{T}. \quad (1)$$

With this, the triplet loss is calculated as follows:

$$\begin{gathered} \mathcal{L}(A, P, N) = \\ \max(0, d(f(A), f(P)) - d(f(A), f(N)) + \alpha) \end{gathered} \quad (2)$$

Here, the function f computes the embeddings of the respective image, and the function d is a distance measure.

Since iterating over all possible triplets is both computationally infeasible and unnecessary, given that most triplets easily fulfill the constraint, the challenge becomes selecting triplets that result in high-quality embeddings.

4.3 Triplet Selection

We categorize triplets into three different types: *Easy triplets* conform to the constraint given in Equation 1. Therefore, they have a loss of zero and do not help in training. Triplets with a negative input closer to the anchor input than the positive input are called *hard triplets*. For these, the condition $d(f(A), f(N)) < d(f(A), f(P))$ holds. Between these lie *semi-hard triplets*, which

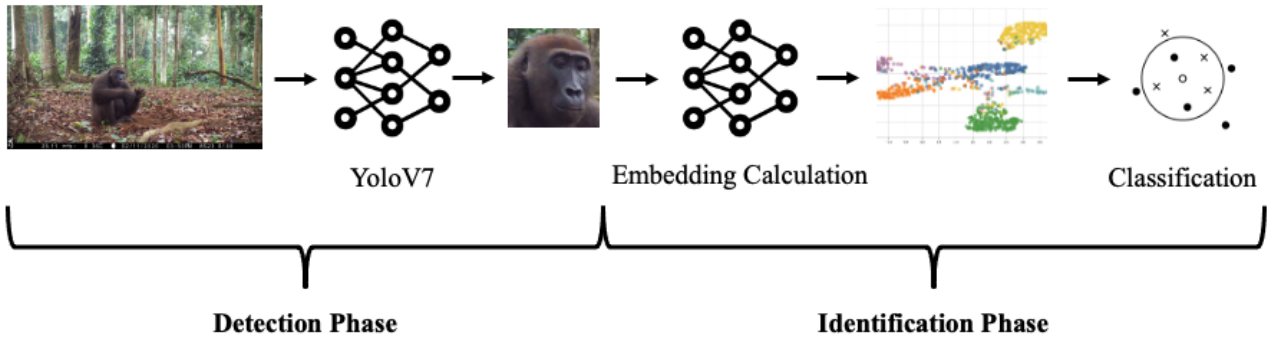

Fig. 3: GorillaVision
The system consists of two stages: The detection phase and the identification phase.

we use for GorillaVision. For such triples, the negative input is farther away than the positive input, but not by the required margin, i.e., $d(f(A), f(P)) < d(f(A), f(N)) < d(f(A), f(P)) + \alpha$.

5 Main Results

5.1 Stage I: Detection

The first stage of the GorillaVision system locates gorilla faces within images. We use YOLOv7, which we fine-tune on *Bristol* using stochastic gradient descent. Our model achieves an F1-Score of 0.97 and mAP@0.5 of 0.991 on the test data and is thus well-suited for our application. The model does not produce any false positives (identification of an object as a gorilla) on the test dataset. Only 1% of the gorilla faces are not correctly recognized as gorillas. Hence, this component yields sufficiently accurate detections for the subsequent identification step in Stage II, which is much more challenging.

5.2 Stage II: Identification

We evaluate our results in two main scenarios: The *closed-set* scenario, in which all possible classes are already present in the training set, and the *open-set* scenario, in which some classes are present only in the test set and not in the training set. We only evaluate the predictions for these new classes in the *open-set* scenario.

We evaluate our GorillaVision model described in Section 4. The implementation is in *PyTorch-Lightning* and is partly based on a *Tensorflow* implementation by Olga Moskyvak*. Additionally, we use a Triplet Loss implementation for PyTorch† with semi-hard sampling that is based on the FaceNet approach [31]. We use the Vision Transformer PyTorch implementation as the backbone of our model, which is pre-trained on ImageNet1K V1. The Vision Transformer we use is the base version with an input size of 32x32 and 86M parameters. We fine-tune the backbone with our datasets by using the Adam optimization algorithm.

5.2.1 Baseline: The only method for gorilla re-identification in the wild is introduced by Brust et al., which yielded an accuracy of 62.4% on an unpublished dataset comprising 2,000 images [8]. In a different context, Brookes et al. pursued the re-identification of gorillas within a controlled zoo setting utilizing the *Bristol* dataset [6] and the YOLOv3 model. Despite our focus on wild gorilla re-identification, we adopted a slightly adapted version of the methodology used by Brookes et al. This decision was motivated by the accessibility of the dataset and the quick model re-implementation.

Our validation confirms that our adapted model performs comparably under the simpler, controlled circumstances of the *Bristol* dataset. Subsequently, we assess the model's efficacy in the more demanding *in-the-wild* context using our *SPAC-Gorilla* dataset. We further improved Brookes et al.'s approach by using the more recent *ultralytics* PyTorch implementation of YOLOv5 for classification‡ instead of the YOLOv3 model. This modification allowed us to replicate the outcomes on the *Bristol* dataset and establish a benchmark for the *SPAC-Gorilla* dataset.

It is important to note that the baseline mentioned above can only be used for closed-set re-identification. Nevertheless, it provides a relevant benchmark for the evaluation of our model.

Table 1 Closed-Set Accuracy.

Dataset	Baseline		**GorillaVision**	
	Top-1	Top-5	Top-1	Top-5
Bristol	0.91	0.90	0.95	0.99
SPAC-Gorilla	0.67	0.79	0.73	0.84
SPAC-Gorilla > 6	0.84	0.94	0.88	0.96
Bristol-SPAC-Gorilla	0.01	0.05	0.49	0.75

5.2.2 Results: All results for our model reported in this section use the Vision Transformer as a backbone without any additional pooling layer or dropout as final layers. Furthermore, a learning rate of 1×10^{-5}, no L2 regularization, a batch size of 128, the same classes (individuals) in training- and validation set, 800 epochs, and an embedding size of 256 are used. The classification is obtained using the k-nearest-neighbours algorithm (w*scikit-learn* implementation) with $k = 5$ using Euclidean distance.

We utilize k-fold cross-validation to compute the reported results for our model. The overall accuracy is calculated as the mean of the accuracy scores obtained across all folds. The cross-validation folds still follow the open-set and closed-set formats specified in Section 3. We use $k = 4$ folds because we have limited data available and want to use as much as possible for training, but at the same time want to ensure that there are enough different individuals in the test set to make the classification more challenging as in the real-life scenario. With this fold size, we use 75% of the available data for training and validation and the remaining 25% for testing for each fold. At the same time, we ensure that in the closed-set approach, every image is in the test set once. In the open-set approach, we ensure that each individual is present in the test set once.

Closed-Set Results: The results for the closed-set scenario are reported in Table 1. It is important to note that the top-5 accuracy scores on the Bristol dataset are not particularly informative, as there are only 7 classes in the database in total.

As we can observe, our model achieves a higher accuracy than the baseline. When comparing our results for *SPAC-Gorilla* and *SPAC-Gorilla > 6*, we can see that if we only consider individuals with

*https://github.com/olgamoskvyak/reid-manta
†https://github.com/alfonmedela/triplet-loss-pytorch/
‡https://github.com/ultralytics/yolov5

more than 6 images, the results improve significantly, since the more images are available for each individual in our dataset, the better the clusters that can be computed for exactly this individual. The results on the Bristol dataset supports this observation, as it has many images available and the accuracy of the results is very high. Hence, if we have many images of good quality available for training, our model can predict the identity of a gorilla on new images fairly reliably.

The baseline by Brust et al. is found to deliver particularly subpar results on the *Bristol-SPAC-Gorilla* dataset. We assume that this is because the gorillas in the Bristol dataset have more than 5,000 images in total (distributed over seven gorillas), while the gorillas from the *SPAC-Gorilla* dataset only have about 500 images in the training set (distributed over 96 gorillas). Due to YOLO's sampling, they only get little attention during training, leading to poor results.

In Figure 4, we visualize the computed embeddings with Principal Component Analysis (PCA) for the Bristol Zoo dataset. We can see that most embeddings are well-separated and individual clusters are easy to distinguish, with some misclassified samples in each cluster.

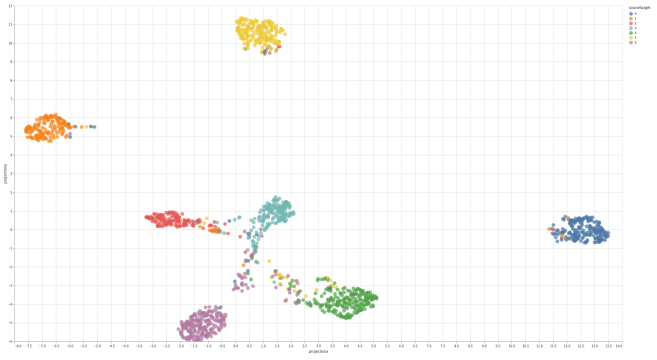

Fig. 4: Closed-set embeddings for the Bristol dataset
Dark blue = 0, Orange = 1, Red = 2, Light Blue = 3, Green = 4, Yellow = 5, Violet = 6.

A manual analysis of misclassified samples shows that those samples that lie in the wrong cluster (close to the center of a wrong cluster and far away from their actual cluster) are primarily images that are very hard to identify, such as the ones shown in Figure 5. Future research could build a system to identify such images and adopt special strategies to reduce their impact.

For the images that lie between clusters 2, 3, 4, 5, we observe a large number of misclassifications because many images of different individuals lie close to each other in this space. From a manual analysis of these images, we could not identify a specific reason why they are not correctly classified within the cluster. Our hypothesis is that most of the affected images can still be deemed part of the correct cluster, but that the inter-cluster distances between clusters 2, 3, 4, 5 are insufficiently large. Hence, fuzzy clusters exist, which can lead to misclassifications.

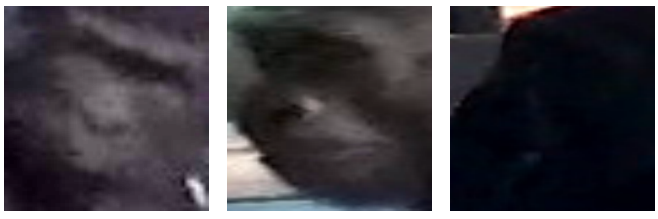

Fig. 5: Samples that are clearly located in a wrong cluster.

Open-Set Results: In the open-set scenario, the challenge shifts from correctly classifying images of known classes to generalizing the classification task to unknown classes, i.e., gorilla individuals. The corresponding results are given in Table 2 and are contrasted with the closed-set results. Due to fluctuations in the obtained result accuracy of $\pm 10\%$, the reported results for the Bristol and *SPAC-Gorilla* dataset are the mean of seven different train-database-eval splits.

Table 2 Closed-Set vs. Open-Set Accuracy. Values in parentheses are the top-5 accuracy.

Dataset	Closed-set		**Open-set**	
	Top-1	Top-5	Top-1	Top-5
Bristol	0.95	0.99	0.80	0.98
SPAC-Gorilla	0.73	0.84	0.63	0.81
SPAC-Gorilla > 6	0.88	0.96	0.80	0.89
Bristol-SPAC-Gorilla	0.49	0.75	0.41	0.68

When comparing our open-set results to the closed-set results, our model is also able to generalize well to new classes. The comparison of *SPAC-Gorilla* vs. *SPAC-Gorilla > 6* and *SPAC-Gorilla* to *Bristol* shows that it is essential to include as many images as possible per individual. However, by comparing the validation loss curves from *SPAC-Gorilla* and *SPAC-Gorilla > 6* and inspecting the embeddings, we can conclude that the significant improvement between these two datasets is also caused by the fact that the k-nearest neighbor classification works better when having more images in the database for each individual. This is the case for *SPAC-Gorilla > 6* with a minimum of six images in the database and two for testing, whereas the *SPAC-Gorilla* dataset sometimes only has two images in the database. Additionally, fewer individuals are in the database, making the classification problem less challenging (37 vs. 115 individuals). Considering these important factors for the comparison of the *SPAC-Gorilla* and *SPAC-Gorilla > 6* results, we conclude that including as many individuals as possible is equally important as having as many images as possible per individual.

When comparing the *SPAC-Gorilla* dataset results to the *Bristol-SPAC-Gorilla* dataset results, we can see that using only the *SPAC-Gorilla* data for training performs better. This is because the data is distributed unevenly over the individuals: Since we have more than 5,500 images of individuals from the *Bristol* dataset and only 540 images from the *SPAC-Gorilla* dataset in our training dataset, the batch sampling mainly picks images from the *Bristol* dataset. Hence, the majority of learning is performed on the same individuals rather than on learning features from a more diverse set of individuals, which would allow the model to generalize well.

A manual analysis of misclassified images shows that 11% are of poor quality, and it is nearly impossible to identify anything on them. In 43%, the eyes of the gorilla and the surrounding area are not visible. This leads us to conjecture that the eyes are a critical feature for re-identification. Furthermore, 26% are classified as the wrong individual but are classified as an individual within the correct gorilla group. For tasks such as social network analysis, the model could hence be used to predict to which group a gorilla belongs with higher accuracy.

6 In-Depth Analysis of GorillaVision

In the following, we analyze and compare the properties of the systems in further detail based on additional experiments.

6.1 Backbone Model

As depicted in Figure 3, GorillaVision's embedding learning relies on a large pre-trained backbone. The choice of backbone can significantly impact system performance, so we evaluate the two following models:

InceptionV3: The InceptionV3 model developed by Szegedy et al. [33] is a deep convolutional neural network designed to be particularly efficient while maintaining a high quality. We use a version of InceptionV3 pre-trained on ImageNet [10]. Additionally, we add

Table 3 Comparison of Backbone Models. All results are obtained without optimizations such as data augmentation.

Model	InceptionV3		**ViT**	
Scenario	Closed	Open	Closed	Open
Bristol	0.80	0.39	0.92	0.78
SPAC-Gorilla	0.35	0.34	0.63	0.58
SPAC-Gorilla > 6	0.50	0.16	0.80	0.44

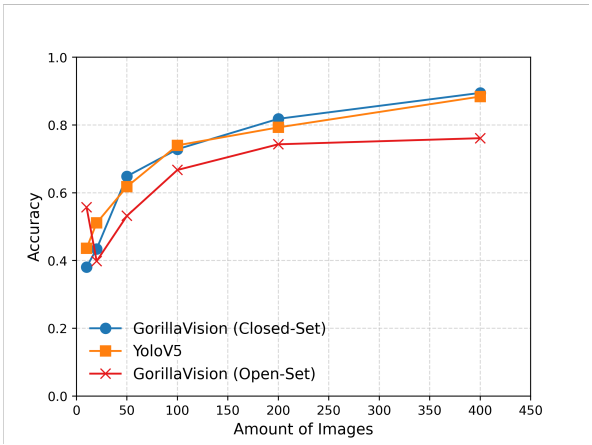

Fig. 6: Effectiveness of GorillaVision on different amounts of training images in the closed- and open-set scenarios.

a global pooling layer after the backbone and before the fully connected layer for embedding computation. This pooling layer takes the averages over each feature map, reducing the dimensionality. This mechanism serves to mitigate overfitting [24].

Vision Transformer (ViT): The Vision Transformer was originally presented by Dosovitskiy et al. [11]. In contrast to InceptionV3 and other traditional vision models, it is not based on convolutional layers for feature extraction but uses a self-attention mechanism introduced by Vaswani et al. [36]. Its core idea is splitting the image into a two-dimensional array and then considering the associations between these arrays using attention.

For both approaches, we truncate the top layer of the backbones to obtain features instead of label probabilities for different classes. As we can see in Table 3, the Vision Transformer outperforms the InceptionV3 backbone model. The ViT increases the accuracy on average in the closed-set scenario by 42.5%. This improvement is even more significant in the open-set scenario, as the ViT nearly doubled the accuracy compared to InceptionV3. This improvement in accuracy comes with a higher computational cost. While the InceptionV3 model has 24 million parameters, the ViT has 86 million.

6.2 Number of Samples Experiments

Labeling images is a very time-consuming task and, in certain cases, can only be performed by experts. For example, in the case of gorillas, when there are many individuals (96 individuals in the case of the *SPAC-Gorilla* dataset), there are likely to be individuals that resemble each other and are easily confused by non-experts. For these reasons, the available training data is limited, and an automated identification solution should also achieve high-quality results using limited amounts of data.

To investigate this further, we used the *Bristol* dataset to analyze how well our method (open- and closed set) performs compared to the baseline with varying amounts of images per individual in the training data. We used $k \in \{10, 20, 50, 100, 200, 400\}$ images for training. As a baseline, we used the YOLOv5 model for classification that is similar to the approach of Brooks et al., as described in 5.2.1. Figure 6 plots the results of this experiment. It can be seen that YoloV5 performs slightly better than GorillaVision on small amounts of data, up to 50 frames per individual. YoloV5 achieves an accuracy of 44% on 10 images per individual and an accuracy of 51% on 20 images per individual. GorillaVision only achieves accuracies of 38% and 43%, respectively. However, from 50 images and upwards, GorillaVision achieves a comparable, if not better, performance to the baseline.

The open set scenario behaves differently from the closed set scenario: If the accuracy is still 53% at ten images per individual, it drops sharply to 40% at 20 images. This might be due to variations in classification difficulty for this small amount of data. It then increases up to 200 images per individual, to approximately 74%. This shows that the GorillaVision model in the open-set scenario is successfully able to learn generalized embeddings of gorilla faces with limited amounts of data. Nonetheless, the amount of training data clearly has an impact on the quality of the model.

6.3 Discussion

Given the results, we see several avenues to further improve the results of GorillaVision.

Detecting New Individuals: Our approach lays the foundation for new individuals to be detected without retraining the model. Its potential extends beyond gorilla re-identification; the techniques applied here can be applied to other wildlife species as well. However, the final verdict of whether to classify an individual as unknown still needs to follow, e.g., by learning a threshold for when the distance between a prediction and all its neighbors in the database is too large. New individuals could then be added to the database with a new label.

Video Data: Since footage of wildlife is often captured as video data, a pipeline to process videos would improve the results further. If an individual can be identified and tracked over multiple frames, the results could then be aggregated to determine the most likely identity of the gorilla in a more robust manner. This approach has also been applied in multiple related scenarios and has led to significant improvements in results [6, 30].

Body Data: We could incorporate the body of gorillas for improved identification. Gorillas exhibit substantial variation in their bodily appearance, especially in datasets that include a diverse set of gorillas in different age groups. We have already worked on detecting the body of gorillas, but this has not led to satisfying results yet due to bad ground truth data. More work could be done to improve the results here. Furthermore, a similar approach proposed by Makowski et al. [22] could be applied. For biometric identification, they considered different ways of including body information and were able to improve their results.

7 Conclusion

This paper presents **GorillaVision** – an open-set re-identification system for gorillas in the wild. It employs a two-stage approach by detecting a gorilla face and then classifying the cropped image. GorillaVision is the first gorilla-specific model that identifies known and unknown gorillas. The system can flexibly use different kind of backbone models. We show that a Vision Transformer backbone model outperforms CNN-based approaches and is suitable for learning high-quality embeddings that aid in distinguishing different individuals.

We show on two different datasets that the system attains comparable results to the state-of-the-art in a closed-set scenario. With sufficient training data, in terms of images per individual and the total number of individuals, our model can produce high-quality results in an open-set scenario, which is a key desidaratum for re-identification. Furthermore, an in-depth analysis investigated several abbreviations of the model and how they affect the results.

We aim to further facilitate research and usage in the field by releasing our source code.

8 References

1 Insaf Adjabi, Abdeldjalil Ouahabi, Amir Benzaoui, and Abdelmalik Taleb-Ahmed. Past, present, and future of face recognition: A review. *Electronics*, 9(8), 2020. ISSN 2079-9292. doi: 10.3390/electronics9081188. URL https://www.mdpi.com/2079-9292/9/8/1188.

2 Luca Bergamini, Angelo Porrello, Andrea Capobianco Dondona, Ercole Del Negro, Mauro Mattioli, Nicola D'alterio, and Simone Calderara. Multi-views embedding for cattle re-identification. In *2018 14th international conference on signal-image technology & internet-based systems (SITIS)*, pages 184–191. IEEE, 2018.

3 Puja Bharati and Ankita Pramanik. Deep learning techniques—r-cnn to mask r-cnn: a survey. *Computational Intelligence in Pattern Recognition: Proceedings of CIPR 2019*, pages 657–668, 2020.

4 Nitin Bhatia and Vandana. Survey of nearest neighbor techniques. 2010. doi: 10.48550/ARXIV.1007.0085. URL https://arxiv.org/abs/1007.0085.

5 Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. *Advances in neural information processing systems*, 6, 1993.

6 Otto Brookes and Tilo Burghardt. A dataset and application for facial recognition of individual gorillas in zoo environments, 2020. URL https://arxiv.org/abs/2012.04689.

7 Otto Brookes, Stuart Gray, Peter Bennett, Katy V. Burgess, Fay E. Clark, Elisabeth Roberts, and Tilo Burghardt. Evaluating cognitive enrichment for zoo-housed gorillas using facial recognition. *Frontiers in Veterinary Science*, 9, 2022. ISSN 2297-1769. doi: 10.3389/fvets.2022.886720. URL https://www.frontiersin.org/articles/10.3389/fvets.2022.886720.

8 Clemens-Alexander Brust, Tilo Burghardt, Milou Groenenberg, Christoph Kading, Hjalmar S. Kühl, Marie L. Manguette, and Joachim Denzler. Towards automated visual monitoring of individual gorillas in the wild. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 2820–2830, 2017. doi: 10.1109/ICCVW.2017.333.

9 Debayan Deb, Susan Wiper, Alexandra Russo, Sixue Gong, Yichun Shi, Cori Tymoszek, and Anil K. Jain. Face recognition: Primates in the wild. *CoRR*, abs/1804.08790, 2018. URL http://arxiv.org/abs/1804.08790.

10 Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

11 Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. URL https://arxiv.org/abs/2010.11929.

12 Alejandro Estrada, Paul A. Garber, Anthony B. Rylands, Christian Roos, Eduardo Fernandez-Duque, Anthony Di Fiore, K. Anne-Isola Nekaris, Vincent Nijman, Eckhard W. Heymann, Joanna E. Lambert, Francesco Rovero, Claudia Barelli, Joanna M. Setchell, Thomas R. Gillespie, Russell A. Mittermeier, Luis Verde Arregoitia, Miguel de Guinea, Sidney Gouveia, Ricardo Dobrovolski, Sam Shanee, Noga Shanee, Sarah A. Boyle, Agustin Fuentes, Katherine C. MacKinnon, Katherine R. Amato, Andreas L. S. Meyer, Serge Wich, Robert W. Sussman, Ruliang Pan, Inza Kone, and Baoguo Li. Impending extinction crisis of the world's primates: Why primates matter. *Science Advances*, 3(1):e1600946, 2017. doi: 10.1126/sciadv.1600946. URL https://www.science.org/doi/abs/10.1126/sciadv.1600946.

13 Alexander Freytag, Erik Rodner, Marcel Simon, Alexander Loos, Hjalmar Kühl, and Joachim Denzler. Chimpanzee faces in the wild: Log-euclidean cnns for predicting identities and attributes of primates. volume 9796, pages 51–63, 09 2016. ISBN 978-3-319-45885-4. doi: 10.1007/978-3-319-45886-1_5.

14 Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

15 Songtao Guo, Pengfei Xu, Qiguang Miao, Guofan Shao, Colin A. Chapman, Xiaojiang Chen, Gang He, Dingyi Fang, He Zhang, Yewen Sun, Zhihui Shi, and Baoguo Li. Automatic identification of individual primates with deep learning techniques. *iScience*, 23(8):101412, 2020. ISSN 2589-0042. doi: https://doi.org/10.1016/j.isci.2020.101412. URL https://www.sciencedirect.com/science/article/pii/S2589004220306027.

16 Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

17 Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *CoRR*, abs/1703.07737, 2017. URL http://arxiv.org/abs/1703.07737.

18 Mahmut KAYA and Hasan Şakir BİLGE. Deep metric learning: A survey. *Symmetry*, 11(9), 2019. ISSN 2073-8994. doi: 10.3390/sym11091066. URL https://www.mdpi.com/2073-8994/11/9/1066.

19 Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection, 2017. URL https://arxiv.org/abs/1708.02002.

20 Ning Liu, Qijun Zhao, Nan Zhang, Xinhua Cheng, and Jianing Zhu. Pose-guided complementary features learning for amur tiger re-identification. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 286–293, 2019. doi: 10.1109/ICCVW.2019.00038.

21 Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: Single shot MultiBox detector. In *Computer Vision – ECCV 2016*, pages 21–37. Springer International Publishing, 2016. doi: 10.1007/978-3-319-46448-0_2. URL https://doi.org/10.1007%2F978-3-319-46448-0_2.

22 Silvia Makowski, Lena A. Jäger, Paul Prasse, and Tobias Scheffer. Biometric identification and presentation-attack detection using micro- and macro-movements of the eyes. In *2020 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–10, 2020. doi: 10.1109/IJCB48548.2020.9304900.

23 Olga Moskvyak, Frederic Maire, Feras Dayoub, and Mahsa Baktashmotlagh. Learning landmark guided embeddings for animal re-identification. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, March 2020.

24 Olga Moskvyak, Frederic Maire, Feras Dayoub, Asia O. Armstrong, and Mahsa Baktashmotlagh. Robust re-identification of manta rays from natural markings by learning pose invariant embeddings. In *2021 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8, 2021. doi: 10.1109/DICTA52665.2021.9647359.

25 Hung Nguyen, Sarah J. Maclagan, Tu Dinh Nguyen, Thin Nguyen, Paul Flemons, Kylie Andrews, Euan G. Ritchie, and Dinh Phung. Animal recognition and identification with deep convolutional neural networks for automated wildlife monitoring. In *2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 40–49, 2017. doi: 10.1109/DSAA.2017.31.

26 Anima Pramanik, Sankar K Pal, Jhareswar Maiti, and Pabitra Mitra. Granulated rcnn and multi-class deep sort for multi-object detection and tracking. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 6(1):171–181, 2021.

27 Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement, 2018. URL https://arxiv.org/abs/1804.02767.

28 Stefan Schneider, Graham W. Taylor, Stefan Linquist, and Stefan C. Kremer. Similarity learning networks for animal individual re-identification – beyond the capabilities of a human observer, 2019. URL https://arxiv.org/abs/1902.09324.

29 Stefan Schneider, Graham W. Taylor, Stefan Linquist, and Stefan C. Kremer. Similarity learning networks for animal individual re-identification – beyond the capabilities of a human observer, 2019. URL https://arxiv.org/abs/1902.09324.

30 Daniel Schofield, Arsha Nagrani, Andrew Zisserman, Misato Hayashi, Tetsuro Matsuzawa, Dora Biro, and Susana Carvalho. Chimpanzee face recognition from videos in the wild using deep learning. *Science Advances*, 5(9):eaaw0736, 2019. doi: 10.1126/sciadv.aaw0736. URL https://www.science.org/doi/abs/10.1126/sciadv.aaw0736.

31 Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

32 Sefik Ilkin Serengil and Alper Ozpinar. Lightface: A hybrid deep face recognition framework. In *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pages 1–5, 2020. doi: 10.1109/ASYU50717.2020.9259802.

33 Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *CoRR*, abs/1512.00567, 2015. URL http://arxiv.org/abs/1512.00567.

34 Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708, 2014. doi: 10.1109/CVPR.2014.220.

35 Alžběta Turečková, Tomas Holik, and Zuzana Oplatkova. Dog face detection using yolo network. *MENDEL*, 26:17–22, 12 2020. doi: 10.13164/mendel.2020.2.017.

36 Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017. URL https://arxiv.org/abs/1706.03762.

37 Jean-Christophe Vié, Craig Hilton-Taylor, and Simon N Stuart. *Wildlife in a changing world: an analysis of the 2008 IUCN Red List of threatened species*. IUCN, 2009.

38 Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, 2022. URL https://arxiv.org/abs/2207.02696.

39 Wang Yang and Zheng Jiachun. Real-time face detection based on yolo. In *2018 1st IEEE International Conference on Knowledge Innovation and Invention (ICKII)*, pages 221–224, 2018. doi: 10.1109/ICKII.2018.8569109.

40 Yuan Yao, Abhiraj Mohan, Eliza Bliss-Moreau, Kristine Coleman, Sienna M. Freeman, Christopher J. Machado, Jessica Raper, Jan Zimmermann, Benjamin Y. Hayden, and Hyun Soo Park. Openmonkeychallenge: Dataset and benchmark challenges for pose tracking of non-human primates. *bioRxiv*, 2021. doi: 10.1101/2021.09.08.459549. URL https://www.biorxiv.org/content/early/2021/09/10/2021.09.08.459549.