https://doi.org/10.2352/EI.2023.35.7.IMAGE-280

Challenges and Constraints when Applying Few Shot Learning to a Real-World Scenario: In-the-Wild Camera-Trap Species Classification

Haoyu Chen, Purdue University, West Lafayette, IN USA
Stacy M. Lindshield, Purdue University, West Lafayette, IN USA
Amy R. Reibman, Purdue University, West Lafayette, IN USA

Abstract

Few shot learning (FSL) describes the challenge of learning to classify when there are only a few labeled examples. The goal is to adapt to a new classification task using a minimum amount of new data. However, when applying FSL to real-world problems, there exist a number of constraints and challenges that are not addressed in benchmark datasets.

In this paper, we consider a realistic problem that fits perfectly with the narratives of FSL: to classify animal species that appear in our in-the-wild camera traps located in Senegal, when these species have yet to appear in popular animal datasets. Using the philosophy of FSL, we would train a FSL network to learn to separate animal species, using large public datasets, and then implement the network with our data where there are fewer labeled images. To explore this framework, we construct two separate testing datasets using our data, to reflect 1) challenges due to our unique imagery properties and environments, and 2) assumptions made in benchmark datasets that do not hold in real-world scenarios. We then conduct a comparison between FSL models, which illustrates the drastic difference between testing in benchmark settings and potential implementation on real data.

Introduction

A conventional deep learning classifier requires massive amounts of training data for each of the categories it attempts to classify, which can be difficult to obtain in many real-world problems. Few-shot learning (FSL) is proposed to deal with this challenge. Ideally, a FSL network learns to extract generalized information that separates classes, and is therefore able to adapt to any new task with only a small amount of labeled data [1–3]. Many ideas to achieve this have been proposed, and will be discussed further in Section "Background – Few Shot Learning Methods". Researchers have then constructed several benchmark datasets designed to represent such scenarios, and apply these datasets for comparing results between algorithms. We discuss more about benchmark datasets in Section "Background – Benchmark Datasets and Evaluation Protocols".

In this paper, we present a real-world scenario for few-shot learning — species classification for in-the-wild camera trap data. Our project aims to explore the ecological basis of hunting and meat sharing in female savanna chimpanzees [4, 5]. A key component of the project is species classification of animals detected by numerous camera traps in Senegal. By experimenting with traditional deep learning models, we discovered two major challenges: 1) site-specific species (e.g., green monkey, patas monkey, red-flanked duiker, oribi, giant eland, etc.) are not available with off-the-shelf models such as Species Classification tool[1], and 2) a lack of clean and annotated data to train a large network from scratch. More details about our data are described in Section "Challenges with Real Data".

To deal with the problem of not having an extensive set of labels, we recognize that our case is a good application of few-shot learning. If we apply the logic of few-shot learning to our case, we could utilize public camera trap datasets with annotations for different species. Ideally, a few-shot learning network could learn to extract what separates animal species, and we could apply that knowledge on our camera sites and species, with relatively fewer labels to create.

We constructed two datasets from our camera trap data; the first one follows the traditional few shot evaluation protocol, and the second one incorporates additional challenges we observed with real-world data which were not considered with benchmark datasets. Through testing with these two datasets, we then revealed that a network's performance on benchmark settings may not transfer well into more realistic settings, where some strong assumptions made in benchmark evaluation protocol do not hold.

This work is intended to serve as a starting point of bridging the gap between benchmark evaluations protocol and real-world applications.

Background

Few-Shot Learning Methods

Few-shot learning networks aim to learn to separate, rather than to assign deterministic classifications like traditional one-hot encoding. During testing/implementation, the idea is for user to only label a few images per class (known as the support images), and unknown images will be classified through comparison with the support images.

To formally compare results under the FSL philosophy, researchers constructed benchmark evaluation protocols. A classic few-shot learning evaluation setup can be described as a N-way K-shot classification problem — "N" denotes number of classes, and "K" denotes the number of support images per class. Each of the small sample problems consisting of N classes, K support images (per class), and several query images (assumed to be unknown) is called an "episode".

For example, in "5-way 5-shot classification", the evaluation program randomly samples 5 classes (5-way), and then 5 sup-

[1] https://github.com/microsoft/SpeciesClassification

port images (5-shot) per class, and (for example) 15 images that are considered unknown. The evaluation program then takes any FSL network, uses it to extract features from these selected images, conducts classification by comparing an unknown image's feature to the support images, and quantifies its performance by top-1 match rate of the 15 "unknown" images. The procedures above constitute one "episode". Typical testing of few-shot learning algorithms is performed with many iterations of episodes, either randomly selected, or pre-determined.

As for training a few-shot learning algorithm, there are two major categories — episodic, or non-episodic. Episodic training partitions the training data into small "episodes" as well, usually having the same N-way K-shot setup as the testing scenario. Within each episode, the network attempts to minimize loss by maximizing the similarity (or minimizing distance) between a query images and its corresponding support images, and sometimes also minimizing the similarity (or maximizing distance) between a query images and support images of other classes. Typical episodic training networks include Matching Net [2] and Proto Net [6].

The episodic training scheme can be also viewed as a kind of meta-learning, where the network learns the task (N-way K-shot classification) through many small sample tasks of the same nature. Hence, meta-learning networks such as [7] and [8] are also used for few-shot learning tasks.

More recently, several non-episodic training algorithms have demonstrated better performance on benchmark datasets. Non-episodic training does not mimic the testing setup (i.e., N-way K-shot episode), but rather takes the entire training set at once, and learns to separate different classes. During testing, they still follow the same episode setup when comparing with other few-shot learning methods. Such networks include [9] and [10].

Benchmark Datasets and Evaluation Protocols

Popular few-shot learning datasets include Omniglot [11], primarily designed for a character recognition task, and Mini-ImageNet [2] — a subset of ImageNet — primarily designed for a general image classification problem. There also exist several alternatives in constructing a few-shot dataset from ImageNet, including tiered-ImageNet [12] and better-tiered-ImageNet [13].

For testing/evaluation, the episode-based method is now the dominant protocol after being proposed in [2] and further elaborated in [14]. As briefly mentioned in the background section, each "N-way K-shot" episode consists of N classes, K support images (per class), and several query images. Besides an equal number of support images per class, we observed that current evaluation protocols sample an equal number of query images per class as well. Note that the assumption of equal number of query images per class are not being explicitly mentioned in papers; instead, we made the observations through multiple well-recognized few-shot learning repositories on GitHub — such as [14][2] and [15][3].

However, the true objective of any machine learning advancement is to help people solve real problems. Hence, benchmark datasets should not be the ultimate goal, but rather a representation of how a system would perform in real world. As researchers advance in developing algorithms that achieve better results, we should also examine the difference between benchmark datasets and potential real-world problems.

Many researchers have already started to explore the gap between benchmark datasets and realistic problems. In [16], the authors argued that despite disjoint classes, training and testing on the same dataset does not necessarily replicate the domain difference likely to be observed in real-world problems. The authors then created a cross-domain testing case by training few-shot learning networks using a benchmark dataset (mini-ImageNet) and testing on a fine-grained classification dataset (Caltech-UCSD Birds 200 [17]), re-partitioned to fit the few-shot learning benchmark style. In [13], the authors criticized current benchmark datasets for lacking realistic meaning between classes in each sub-task (i.e., an episode). They proposed a relevance measure between classes using semantic structures, and use that measure to create a dataset with more relevant class sampling.

In the next section, we will discuss more about limitations of benchmark datasets and challenges we observed with real-world data.

Challenges with Real Data

Real world data is very different from benchmark datasets. Due to factors like camera condition, environment, and target class of interest, real-world data may pose challenges that are not addressed in benchmark datasets.

Our data, specifically, are taken from motion-triggered camera traps around two major sites in Senegal — Assirik and Fongoli. As of now, we have received 6 data shipments, totaling 55,269 videos, 122,228 images from 287 camera locations; the entire data volumn is 2,716 GB (Gigabyte). Details of our data collection can be seen in Table 1; note the term "CT days" (Camera Trap days) is counted as days between when a camera started to record and when it recorded its last video/image. For this paper, we consider 114 camera locations from data shipments 2 and 3, apply an off-the-shelf animal detector[4], and label about 8,000 bounding boxes of detected animals, including some false positives that were mis-detected.

To further elaborate on our challenges with real-world data, we face several major issues. First, most of our data come in as unlabeled videos, without annotations of whether an animal is present, bounding boxes of animals, or what species an animal is. Second, many animal detections yield low-quality images; these causes include distance to camera (too close leading to only partial image of the animal, and too far leading to lower resolution), and occlusion. Third, we observe heavy imbalance between classes; during initial annotating of the 8,000 images, we observed 1,624 baboons (*Papio papio*), but only 247 green monkeys (*Chlorocebus sabaeus*), and 54 patas monkeys (*Erythrocebus patas*). The number of labels for each species is listed in Table 3. In addition, we observe the presence of "distractor" images, including non-animal objects being falsely detected as animals, and other animals occasionally appearing in camera trap videos that are not species-of-interest for this research project.

We divide this section into two major parts: **challenges** that might make classification task harder, and **differences** where some assumptions made in benchmark dataset do not hold.

[2] https://github.com/twitter-research/meta-learning-lstm

[3] https://github.com/RL-VIG/LibFewShot

[4] https://github.com/microsoft/CameraTraps

Shipment	Videos	Images	Size (GB)	Cameras
1	2998	106283	187	48
2	25606	10433	1304	110
3	3489	2392	132	31
4	6894	1410	284	37
5	11523	356	642	39
6	4759	1354	167	22
Total	55269	122228	2716	287

Table 1: Data Shipments We Have Received

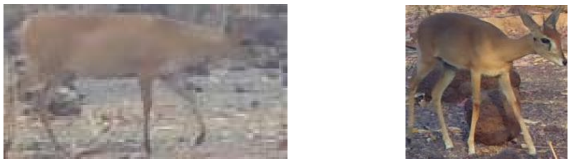
Challenges

The major challenges we face can be categorized into image quality factors, and the overall data context or relationship between images.

We begin with the imagery factors of our data. Unlike many benchmark datasets that obtain generic images from the internet, our data that was captured in the wild suffers much more from poor image quality. Such factors include but are not limited to: **low resolution** due to animals being detected far from the camera; **occlusion** due to plants or other animals; **incomplete animal** due to animals being detected too close to the camera or at the edge of the frame.

Another challenge lies within the general context of the data, namely separation between classes, and variation within the same class. To begin with, we observed many cases of small inter-class (between-class) separation. The cause of such small separation is mostly due to 1) inherent similarity (build, color, etc.) in species or 2) similar background since our data came from stationary camera traps in the same region.

Figure 1 shows an example of such challenges. Despite being from two species, these two images can easily cause confusion even to human viewers, due to their similar build and color. Relatively low quality of the images also harm the separability between species.

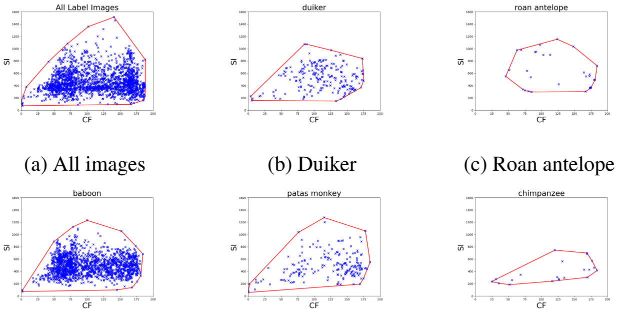


(a) Crowned duiker (*Sylvicapra grimmia coronata*)

(b) Oribi (*Ourebia ourebi*)

Figure 1: Hard-to-distinguish species with similar build and color

On the other hand, intra-class difference can sometimes still be large, meaning that images that belong to the same species can look drastically different, due to the aforementioned imagery factors. To quantify the difference within each class, we apply dataset analysis proposed in [18], and use spatial information (SI) and colorfulness (CF) to measure imagery variation within datasets. In Figure 2, we plot the SI (y-axis) x CF (x-axis) map within all labeled images and within five sample species. The coverage area indicates how diverse the images are, and we can see that the variations within several classes are close to the variation within the entire dataset. This result coincides with what we observed with the actual images, as the same species' image can be very different depending on the environments.



(a) All images

(b) Duiker

(c) Roan antelope

(d) Baboon

(e) Patas monkey

(f) Chimpanzee

Figure 2: SI x CF plots of various classes in our data

Differences

In this section, we describe differences between assumptions made in the benchmark evaluation protocol and in real-world data, which adds difficulties to the problem.

As stated in the background section, to compare results for few-shot learning networks, most benchmark datasets simplify the problem. For each episode of the "N-way K-shot classification" task, all the queries, or unknown images, are selected from the N classes — which means these algorithms do not have to consider that an image might not belong to any of the classes. However, real data does not fulfill this assumption; our data contains many images not belonging to any of the species we are interested in. Most of these images are caused by false detections of background objects that are not animals; in addition, a few of these images are other animals not fitting in our research scope. Another key issue to note is that these "distractor" images do not actually belong to one same class, as they do not necessarily share common features. Hence, the problem is more "identify image not belonging to any class" instead of "identify images belong to the 'distractor' class."

In addition, the sampling of "unknown images/queries" in benchmark dataset is also evenly distributed, which is another assumption not applicable to real data. During initial screening and labeling of our data, we randomly selected 8,000 images to be labeled, and the result can be seen in Table 3. As we can see here, we have heavily unbalanced classes — commonly found species like baboons have 1,624 images, while rarer animals like Patas monkey only have 54 images.

Experiments

To address the aforementioned challenges as well as to compare results with different setups, we created two **testing** datasets from the camera trap video we have collected. Notice that as the first step of making this a viable few-shot learning problem, we are not training on our data. Instead, we train using either 1) benchmark few-shot learning datasets, or 2) the publicly available camera trap dataset Snapshot Serengeti [19], which is from a different geographic region and has different animal species. We mostly focus on using Snapshot Serengeti dataset as the training set, as it fit the few-shot learning narrative best — using available data with similar context for training, while only labeling a few images on the target data.

Data Setup

To formally evaluate a network's performance on our data, we constructed two datasets from our images. Senegal-benchmark (**Senegal-B**) mimics a benchmark evaluation protocol, and enables us to focus on easy and straightforward comparison between networks. And Senegal-implementation (**Senegal-I**) is for in-depth exploration of difference between benchmark datasets and potential real-world implementation.

Dataset 1: Benchmark Style

To begin, we selected a small amount images per species and constructed a small dataset, **Senegal-B**. The number of images per species can be found in Table 2. The purpose of this dataset is to address the first part of the challenges with real data (i.e., "Challenges"), and compare FSL networks' performance under our environment.

Species	Images	Species	Images
Baboon	108	Hartebeest	25
Buffalo	74	Oribi	22
Bushbuck	126	Patas monkey	30
Duiker	53	Roan antelope	97
Green monkey	99	Warthog	83
Guineafowl	86		

Table 2: Images of each species for our benchmark-style dataset (Senegal-B)

We consider this dataset to be simple and straightforward since the evaluation protocol is the same as other FSL benchmark evaluation protocols. The main purpose of this dataset is to compare a network's cross-domain performance in our environment, as we attempt to leverage publicly available data to achieve classification on our data.

To select images, we take an extra step to eliminate repetitive images to make the variation within each class larger; e.g., if an animal stays in the same spot and is detected multiple times, only one image will be preserved. We then employ the benchmark style evaluation, using 5-way, 5-shot settings with randomly sampled classes, support images, and query images.

In summary, Senegal-B mimics the typical benchmark style for straightforward comparison, and uses our images to reflect the aforementioned imagery/environment challenges.

Dataset 2: Implementation Style

For **Senegal-I**, we mimic a real implementation process of few-shot learning. Instead of carefully selecting images with reasonable quality and avoiding repetitive images, we directly sampled and labeled 8,000 detected animals from shipment 2 and 3 of our data, while excluding those already selected for **Senegal-B**. Therefore, the two datasets have disjoint image sets.

As mentioned in the second part of the challenges with real data (i.e., "Differences"), several assumptions made in benchmark datasets do not hold anymore with non-curated data. To begin with, unknown images (queries) do not follow a nicely balanced distribution between classes. As can be seen from Table 3, some of the species are more frequently detected than others. In addition, a testing episode in the benchmark datasets' evaluation does not consider images not belonging to any of the classes. However, in our case, we observe about 30% distractor images that do not belong to any of the classes; these images are often background objects, such as branches or rocks that have been mis-detected as an animal. As these images are from multiple different object types, they do not share inherently similar features to be regarded as one single classes.

Species	Images	Species	Images
Baboon	1,624	Hartebeest	21
Buffalo	566	Oribi	10
Bushbuck	46	Patas monkey	54
Duiker	212	Roan antelope	167
Green monkey	247	Warthog	245
Guineafowl	1,831	Distractors	2,586

Table 3: Images of each species for our implementation-style dataset (Senegal-I)

As we have mentioned earlier, traditional few-shot learning evaluation protocol does not consider distractor images (images not belonging to any class) and always assigns a class to an unknown image. Therefore, for this implementation-oriented dataset, we need an extra step in our evaluation protocol, to decide whether an images should be classified into one of the classes or not.

In this evaluation protocol, we do not randomly sample 5 or 10 classes; instead, we mimic an implementation scenario where all classes are considered at the same time. We select 10 images per class from the relatively-higher-quality **Senegal-B** dataset, and treat them as the support images for each class; the 8,000 images from this dataset (**Senegal-I**) are then treated as queries. Since we have a large amount of distractor images, an additional decision rule must be applied. To start with a straightforward method, we choose K-nearest-neighbor as our decision rule, and require that at least 3 of top 5 matches agree on the same class before an unknown image can be classified.

For comparing results, we will look at three metric:

- **Accuracy** on those that passes the decision rule.
- **False Positive Rate (FPR)** as number of distractor images that pass the decision rule divided by the total number of distractor images. FPR measures a network's effectiveness at eliminating distractor images.
- **True Positive Rate (TPR)** as number of animal images that passes the decision rule **AND** are correctly classified, divided by total number of animal images. TPR measures a network's effectiveness at admitting correct animal images. Due to the very low quality on some of the animal images, we do expect some animal images to not get a classification result.

Evaluation Setup

For evaluation, we choose five networks that we regard as representative.

- ProtoNet [6] is one of the most iconic and well-recognized few-shot learning structures;
- Baseline and Baseline++ are proposed in [16] and are reported to be effective at cross-domain few-shot learning cases;
- RFS [9] discarded the commonly used episodic training scheme in few-shot learning and achieved good results;
- R2-D2 [8] is the representative of meta-learning, where few-shot learning is only one of many potential applications of the network.

To minimize variations between training settings, we used ResNet-12 [20] as the backbone and trained for 1,000 epochs. For networks using episodic training, we choose 5-way 5-shot in training. For all of them, we apply 5-way 5-shot setup during testing. Under Few-Shot Learning's philosophy, we train the network using an off-the-shelf large dataset, with the goal that it learns to extract relevant information for separating image classes. Specifically, we choose two publicly available datasets for training:

1) **Mini-ImageNet** [2] is the standard benchmark dataset representation; previously, the cross-domain experiment in [16] has demonstrated the effectiveness of a potential cross-domain FSL scenario by training on mini-ImageNet, and testing on CUB-Birds-200 dataset.

2) **Snapshot Serengeti** [19] is a large camera trap dataset taken in Tanzania, with 49 labeled species. We used the metadata available to extract animal bounding boxes and labels, and constructed a training set similar in few-shot learning style.

We will then test on the two testing datasets that we created using our data collected from Senegal.

Results and Discussions

Senegal-B

The networks' performance when trained with mini-ImageNet (left column) and when trained with Snapshot Serengeti (right column) are shown in Table 4.

Network	Mini-ImagNet	Snapshot Serengeti
ProtoNet [6]	62.161	60.972
RFS [9]	62.961	71.557
Baseline [16]	67.544	71.981
Baseline++ [16]	54.386	56.494
R2-D2 [8]	63.931	67.631

Table 4: Top-1 accuracy (%) when trained on mini-ImageNet / Snapshot Serengeti, and tested on Senegal-B

As can be seen, four out of five networks (RFS, Baseline, Baseline++, R2-D2) performed better when trained with Snapshot Serengeti than when trained with mini-ImageNet. This shows the importance of creating a smaller domain gap when considering cross-domain problems.

The Snapshot Serengeti training set we constructed (49 classes, 8,558 images) is smaller than mini-ImageNet's training set (64 classes, 38,400 images); in addition, Snapshot Serengeti has less variation within class when compared to mini-ImageNet, because its images are taken from stationary cameras rather than internet images from various sources. However, Snapshot Serengeti has similar camera settings (stationary camera traps) as well as a similar classification problem (animal species, despite different species) to our data, and this is why we believe networks trained with Snapshot Serengeti outperform their counterpart when trained with mini-ImageNet.

Interestingly, of all networks, only ProtoNet's performance degrades when trained with Snapshot Serengeti, where the domain gap is smaller. In the original paper [16] where Baseline and Baseline++ are proposed, it is noted that Baseline performed better in cross-domain scenarios; the same is observed here. Baseline performs best with both small domain gap (Snapshot Serengeti to Senegal-B) and large domain gap (mini-ImageNet to Senegal-B). Baseline++, however, performs quite poorly for both domain gap scenarios. On the other hand, RFS performs almost as well as the best (Baseline) for the small domain gap scenario, despite average performance with large domain gap.

Senegal-I

The networks' performance when trained with mini-ImageNet and tested with Senegal-I, under implementation scenarios, is shown in Table 5. Similarly, networks' performance when trained with Snapshot Serengeti and tested with Senegal-I is shown in Table 6. We use the same K-nearest-neighbor decision rule for all five networks.

With the implementation settings, the effect of domain gap is even more significant. All networks show much better true positive rate (TPR) when trained with Snapshot Serengeti, which means they are all better at correctly identifying animal images and correctly classifying them. As expected, with higher TPR, the accuracy among images that pass the decision rule is also higher when trained with Snapshot Serengeti. However, at the same time, all networks show higher false positive rate (FPR), which means they admit more of the distractor images that do not belong to any class. Notably, Baseline shows an much higher FPR, which means that it is admitting most of the distractor images.

When trained with mini-ImageNet, R2-D2 obtained best overall performance, and RFS is a close second. Baseline, despite its best performance on Senegal-B, shows very poor performance with implementation settings; specifically, it obtains extremely high FPR and low accuracy among admitted images.

When trained with Snapshot Serengeti, RFS obtained the best overall performance, showing lowest FPR, highest TPR, and best accuracy among images that pass the decision rule.

The difference between a network's performance on benchmark style evaluation (Senegal-B) and implementation style evaluation (Senegal-I) clearly show that benchmark datasets are inadequate for reflecting a network's true usefulness in real applications. When we remove some of the strong assumptions in the benchmark dataset, a network's behavior may change drastically; hence, designing a network purely to fulfill benchmark settings may not transfer well to the real world.

Conclusion and Future Work

In this paper, we present a real-world problem — animal species classification in in-the-wild camera trap videos — that fits the Few Shot Learning (FSL) narrative, but also poses new challenges. We then constructed two datasets using our data. The first dataset follows benchmark datasets' settings; the second dataset mimics real implementation settings, where we need to consider additional system components like potential false positive from

Network	Accuracy	FPR	TPR
ProtoNet [6]	0.286	0.14	0.079
RFS [9]	0.417	0.049	0.065
Baseline [16]	0.253	0.24	0.087
Baseline++ [16]	0.245	0.071	0.038
R2-D2 [8]	0.441	0.089	0.069

Table 5: Network performances when trained on mini-ImageNet and tested on Senegal-I

Network	Accuracy	FPR	TPR
ProtoNet [6]	0.369	0.26	0.16
RFS [9]	0.562	0.14	0.25
Baseline [16]	0.417	0.32	0.24
Baseline++ [16]	0.212	0.44	0.12
R2-D2 [8]	0.516	0.19	0.22

Table 6: Network performances when trained on Snapshot Serengeti and tested on Senegal-I

system pipeline, and where some strong assumptions set in benchmark datasets do not hold. After conducting experiments on our datasets using popular FSL networks, we found that performance under benchmark settings may not reflect a network's usefulness under more realistic scenarios. Our goal is to use these results to inspire people to explore more about how an algorithm would behave outside benchmark datasets, and design algorithms to accommodate real-world challenges.

For future work, we believe there are additional factors to be explored for implementing FSL into real-world problems. We will continue investigating potential factors that may affect implementation of FSL-style classification, including training data selection, support image selection, and decision rules for eliminating distractor images.

Acknowledgments

This work is funded by NSF grant number 2022314, under the HUNTRESS (HUnting, Nutrition, Tool-use, Reproductive Ecology, and meat Sharing in Savanna chimpanzees) project.

We thank the Direction des Parcs Nationaux, Direction des Forêts et Chasses, Recherche Chimpanzé Assirik, Fongoli Savanna Chimpanzee Project, Papa Ibnou Ndiaye, and Jill Pruetz.

References

[1] Li Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 594–611, 2006.

[2] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al., "Matching networks for one shot learning," *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[3] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Computing Surveys (CSUR)*, vol. 53, no. 3, pp. 1–34, 2020.

[4] Stacy Lindshield, Stephanie Bogart, Mallé Gueye, Papa Ndiaye, and Jill Pruetz, "Informing protection efforts for critically endangered chimpanzees (Pan troglodytes verus) and sympatric mammals amidst rapid growth of extractive industries in Senegal," *Folia Primatologica*, vol. 90, pp. 124–136, 03 2019.

[5] Jill Pruetz, P. Bertolani, Kelly Boyer Ontl, Stacy Lindshield, Mack Shelley, and Erin Wessling, "New evidence on the tool-assisted hunting exhibited by chimpanzees (Pan troglodytes verus) in a savannah habitat at Fongoli, Sénégal," *Royal Society Open Science*, vol. 2, 04 2015.

[6] Jake Snell, Kevin Swersky, and Richard Zemel, "Prototypical networks for few-shot learning," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[7] Chelsea Finn, Pieter Abbeel, and Sergey Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," *International Conference on Machine Learning*, pp. 1126–1135, 2017.

[8] Luca Bertinetto, João F. Henriques, Philip H. S. Torr, and Andrea Vedaldi, "Meta-learning with differentiable closed-form solvers," *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, 2019.

[9] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola, "Rethinking few-shot image classification: a good embedding is all you need?," *European Conference on Computer Vision*, pp. 266–282, 2020.

[10] Steinar Laenen and Luca Bertinetto, "On episodes, prototypical networks, and few-shot learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 24581–24592, 2021.

[11] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum, "Human-level concept learning through probabilistic program induction," *Science*, vol. 350, no. 6266, pp. 1332–1338, 2015.

[12] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele, "Meta-transfer learning for few-shot learning," *CVPR*, pp. 403–412, 2019.

[13] Etienne Bennequin, Myriam Tami, Antoine Toubhans, and Céline Hudelot, "Few-shot image classification benchmarks are too far from reality: Build back better with semantic task sampling," *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, LA, USA, June 19-20, 2022*, pp. 4766–4775, 2022.

[14] Sachin Ravi and Hugo Larochelle, "Optimization as a model for few-shot learning," *International Conference on Learning Representations (ICLR)*, 2017.

[15] Wenbin Li, Ziyi Wang, Xuesong Yang, Chuanqi Dong, Pinzhuo Tian, Tiexin Qin, Huo Jing, Yinghuan Shi, Lei Wang, Yang Gao, and Jiebo Luo, "Libfewshot: A comprehensive library for few-shot learning," *arXiv preprint arXiv:2109.04898*, 2022.

[16] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Wang, and Jia-Bin Huang, "A closer look at few-shot classification," *International Conference on Learning Representations (ICLR)*, 2019.

[17] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "Caltech-UCSD Birds 200," Tech. Rep. CNS-TR-2011-001, California Institute of Technology, 2011.

[18] Stefan Winkler, "Analysis of public image and video databases for quality assessment," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 6, pp. 616–625, 2012.

[19] Alexandra Swanson, Margaret Kosmala, Chris Lintott, Robert Simpson, Arfon Smith, and Craig Packer, "Snapshot Serengeti, high-frequency annotated camera trap images of 40 mammalian species in an African savanna," *Scientific data*, vol. 2, no. 1, pp. 1–14, 2015.

[20] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.