Visual Re-Identification of Wildlife using MegaDescriptor.

Kunal Maurya 2020215
Mohammad Shariq 2020220
Arun Yadav 2020033

INDRAPRASTHA INSTITUTE *of* INFORMATION TECHNOLOGY DELHI

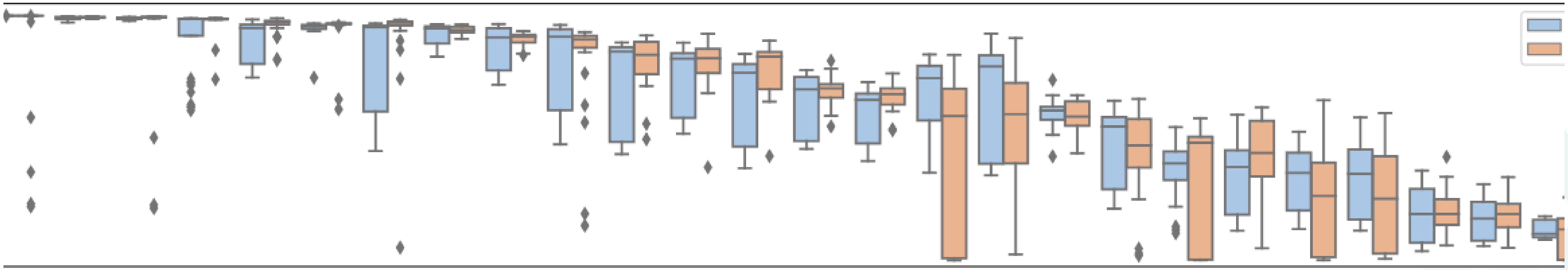
Problem Statement

The absence of standardized algorithmic procedures, evaluation metrics, and dataset utilization in the realm of automated animal re-identification is a significant challenge. Wildlife datasets are scattered across literature, often employing species-specific methods, which impedes comparability, reproducibility, and generalization of results. This lack of standardization and unity in research practices creates obstacles for the development, evaluation, and real-world application of automated animal re-identification methods.

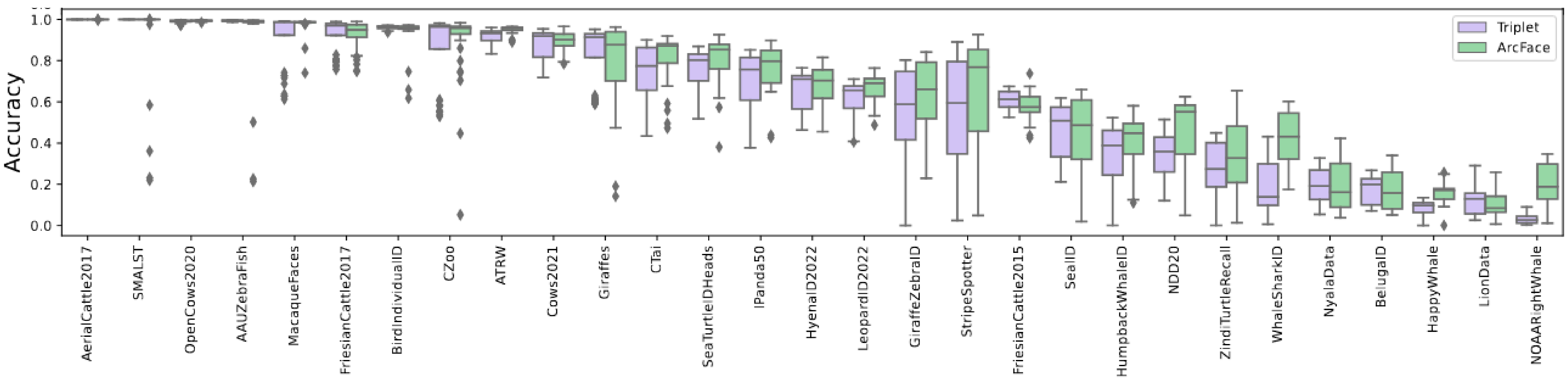
Papers reviewed model aspects and evaluation metrics.

The paper evaluates various aspects of the model, including architecture, loss functions, and evaluation metrics. Here are some key aspects covered in the evaluation:

- Model Architecture: The paper explores different backbone architectures for the re-identification model, specifically comparing a transformer-based backbone (Swin-T) with a CNN-based backbone (EfficientNet-B3).



- Loss Functions: Two main loss functions are evaluated - ArcFace and Triplet loss. The performance of these loss functions is compared across different datasets to determine their effectiveness in the re-identification task.



- Evaluation Metrics: The evaluation of the model's performance is based on metrics such as accuracy, precision, recall, and F1 score. These metrics are used to assess the effectiveness of the model in identifying and re-identifying animals across multiple datasets.

DataSet	**Accuracy**
MacaqueFaces	99.04
LionData	25.16
NyalaData	36.45
StripeSpotter	98.17
IPanda50	86.91
CZoo	99.05

Analysis Experiment

In the analysis experiment, three datasets are required to evaluate the model's performance using evaluation metrics and loss:

- Same Dataset : This includes datasets used for fine-tuning the model, such as MacaqueFaces, LionData, NyalaData, StripeSpotter, IPanda50, and CZoo.
- Seen Dataset: This comprises datasets not used for fine-tuning, such as OpenCow2020.
- Unseen Dataset: This consists of datasets not used for training or fine-tuning, including polarbearvidID, MPDD, DogfaceNet, and Cowdataset.

Result

Same Dataset

We achieved similar accuracy for MacaqueFaces, StripeSpotter, IPanda50, and CZoo as mentioned in the paper, except for LionData and NyalaData, where there was a small difference.

DataSet	**Accuracy**	**Precision**	**Recall**	F1 Score
MacaqueFaces	100.00	1.0	1.0	1.0
LionData	8.00	0.519	0.08	0.071
NyalaData	11.00	0.545	0.11	0.118
StripeSpotter	98.00	0.983	0.98	0.975
IPanda50	85.00	0.916	0.85	0.84
CZoo	100.00	1.0	1.0	1.0

Seen DataSet

We obtained good accuracy for the seen data with the OpenCows dataset because the model was trained on a similar kind of dataset (FriesianCattle2015 and FriesianCattle2015).

DataSet	**Accuracy**	**Precision**	**Recall**	F1 Score
OpenCows2020	100.00	1.0	1.0	1.0

Unseen DataSet

In the unseen dataset, we achieved good accuracy for CowDataset and polarBearVidID because the model was trained on a similar kind of dataset (Cows2021, OpenCows2020, FriesianCattle2015, and FriesianCattle2015). This is in contrast to DogfaceNet and MPDD, which are datasets for dogs, for which the model was not trained or fine-tuned.

DataSet	**Accuracy**	**Precision**	**Recall**	F1 Score
CowDataset	99.00	0.99	0.99	0.98
DogFaceNet	56.00	0.98	0.56	0.57
MPDD	3.00	0.99	0.03	0.0257
PolarBearVid ID	100.00	1.0	1.0	1.0

Hyperparameter Tuning

- Extensive grid searches were conducted to identify optimal hyperparameters for minimizing the performance variability inherent to metric learning metrics

- For the Swin-B backbone using ArcFace, the optimal settings were identified (learning rate of 0.001, margin (m) of 0.5, and scale (s) of 64), achieving a median performance of greater than 80 percent.

- Notably, some configurations underperformed dramatically, likely due to issues in training convergence.

- In comparison, settings using the Triplet loss demonstrated higher variability in performance, suggesting that while Triplet loss can achieve competitive results, its performance is more sensitive to hyperparameter settings.

DataSet	Th	SIFT	ArcFace Loss	TripletLoss
MacaqueFaces	0.8	66	100	98.98
IPanda50	0.8	25	82	85
LionData	0.6	21	10	6
NayalData	0.7	5	19	21
StripeSpotter	0.7	95	55	79

New Dataset Hyperparameter Tuning Results

In the unseen dataset, **DogFaceNet** from wildlife datasets.

DataSet	Th	SIFT	ArcFaceLoss	TripletLoss
DogFaceNet	0.7	11	57	67

Individual Work

- Arun: Report writing, Training ArcfaceLoss and TripletLoss, and finding accuracy for LionData and MacaqueFaces Datasets.

- Kunal: Finding datasets and Inference, F1 score, precision,recall, and accuracy using DeepFeatures, report writing

- Shariq: All evaluation metrics, losses, and hyperparameter Tuning, Ablation studies, and report writing

THANK YOU