WILDLIFE MONITORING

Understanding wildlife populations is crucial for conservation efforts.

Can Machine Learning scale monitoring efforts?

PRESENTED BY PATRICK BROWN

OUR OBJECTIVE

Produce an model to detect if
wildlife is present in a given image.

WHY

Cost
Scalability
Interference

METHODS

Deep Learning Modeling

- Classification - Wildlife
- Regression - Bounding Box

RESOURCES

Config/Training

- https://github.com/ultralytics/yolov5

Datasource/Processing

- https://universe.roboflow.com/hcl/all_animals21

TOOLS

NumPy/Pandas

- Data Handling

PyTorch/Keras

- Modeling

HuggingFace

- Hosting

Google Colab

- Compute

Roboflow

- Dataset

THE DATA

Source

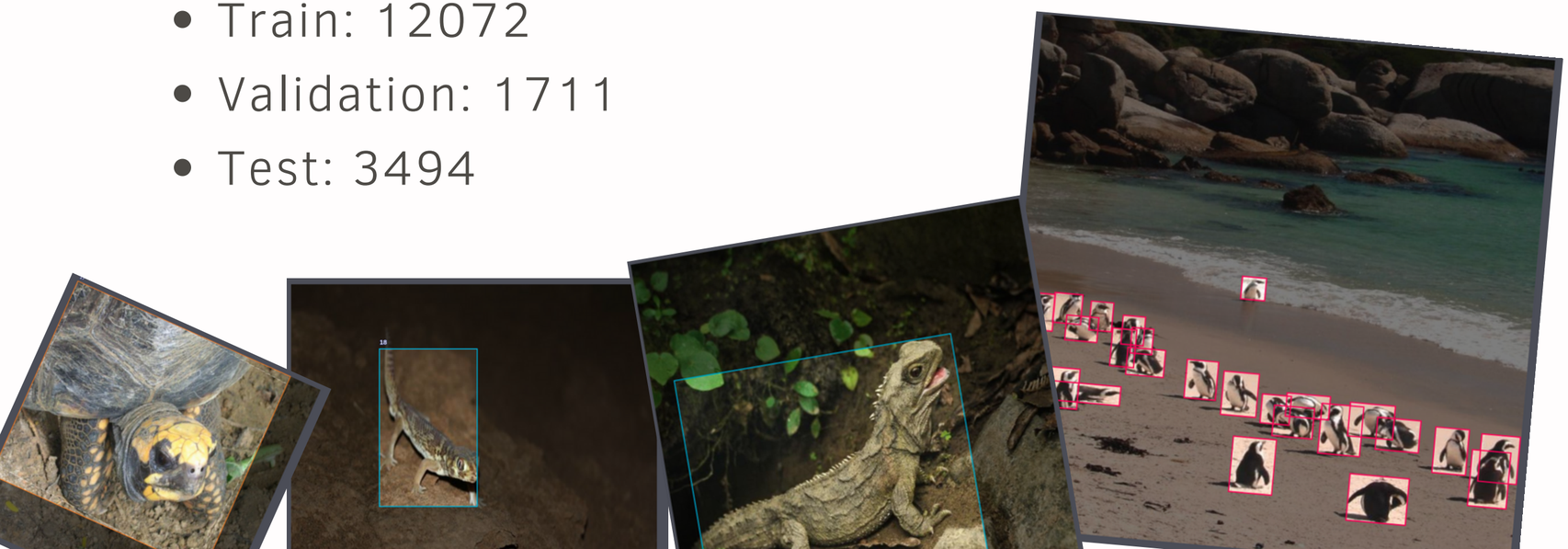
- https://universe.roboflow.com/hcl/all_animals21

Classes

- 20 (Lion, Penguin, Gorilla etc)
- Avg 1090 Images Per Class

N = 17277

- Train: 12072
- Validation: 1711
- Test: 3494



How it looks:

- Label: 7 (Lion)
- X: 341.5
- Y: 321
- Width: 467
- Height: 636

THE MODEL

Yolo V5

- You Only Look Once

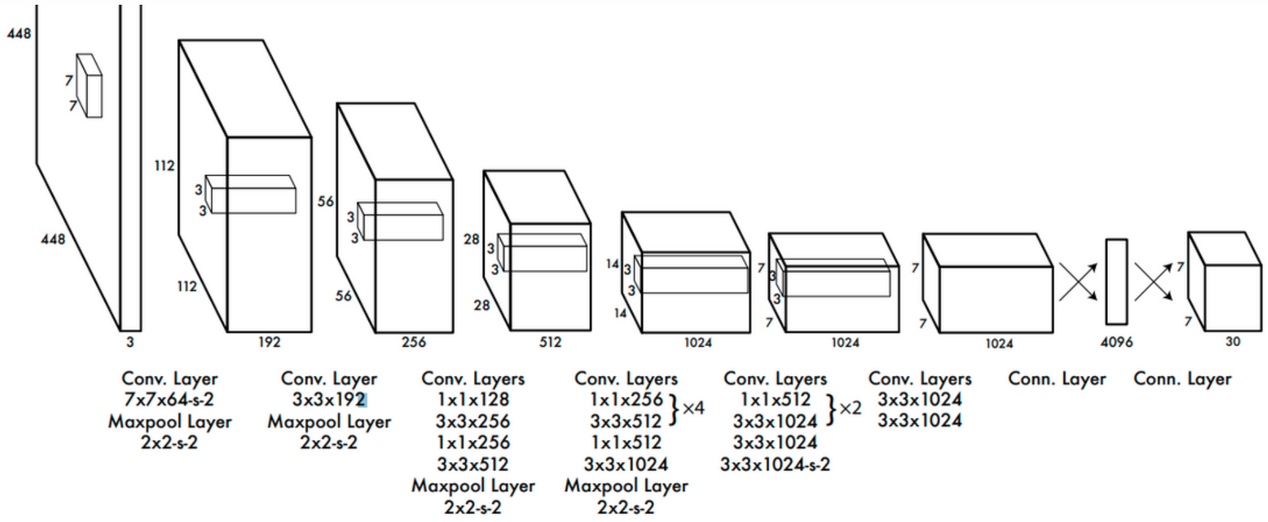
https://arxiv.org/pdf/1506.02640.pdf

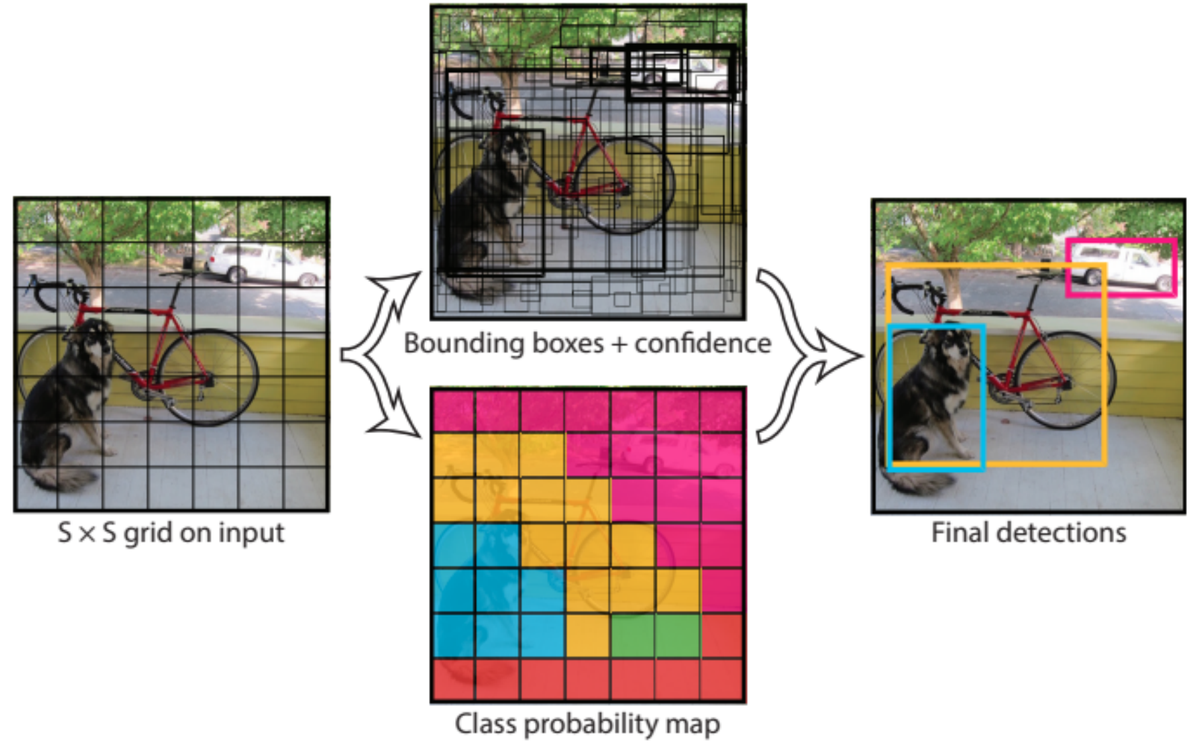
OURS

Layers: 283

Parameters: 7309034

Classes: 20





TRAINING

Run Time

- 3h 47min 58s

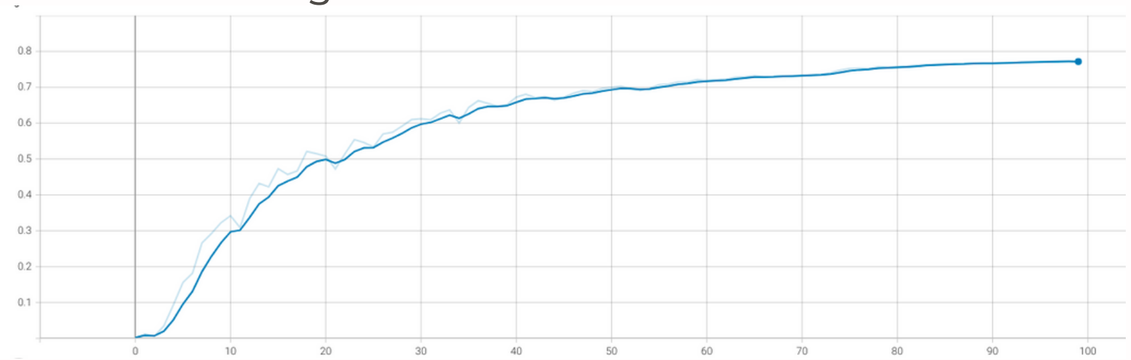
Epoch

- 100

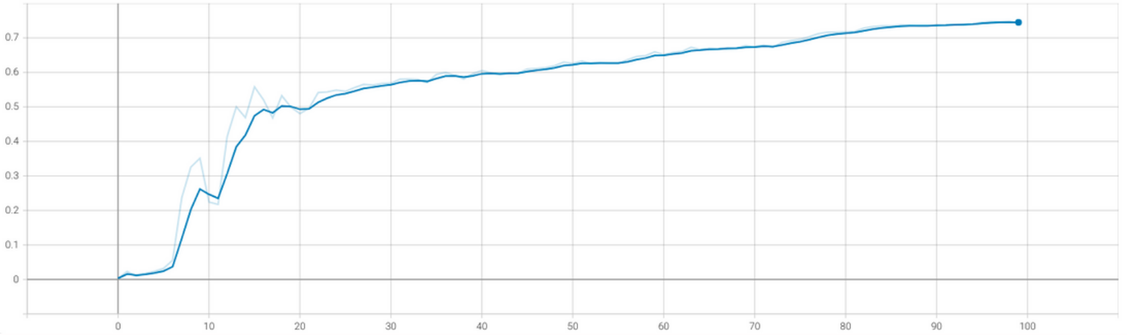
Metrics

- mAP: 0.771
- Precision: 0.743
- Recall: 0.707

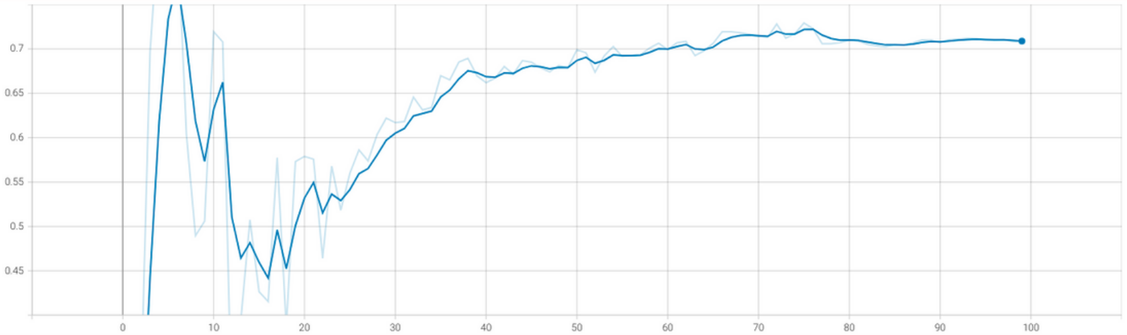
mean Average Precision



Precision



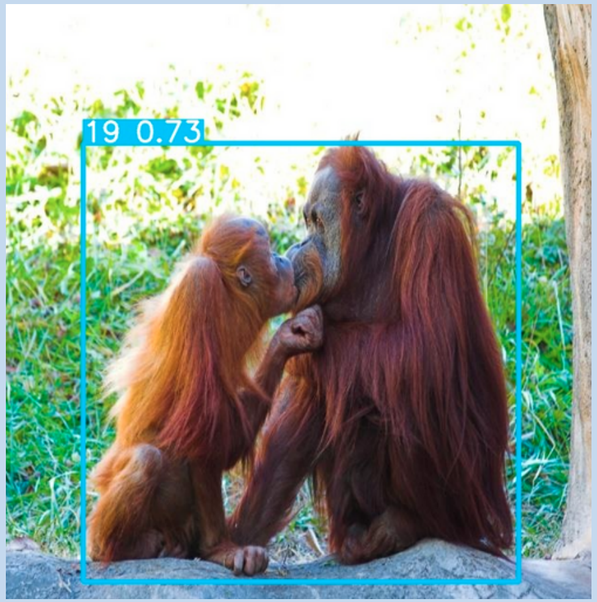
Recall

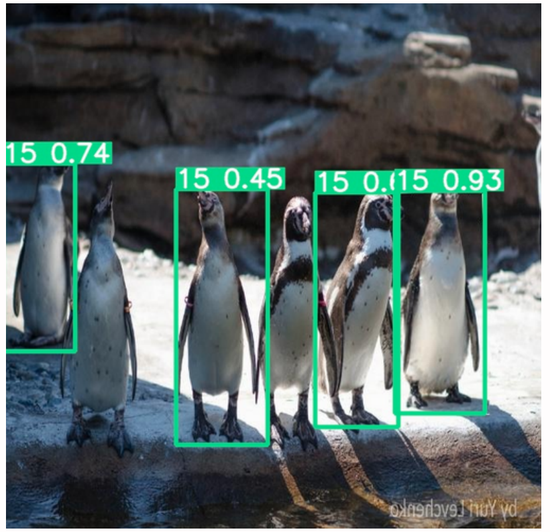


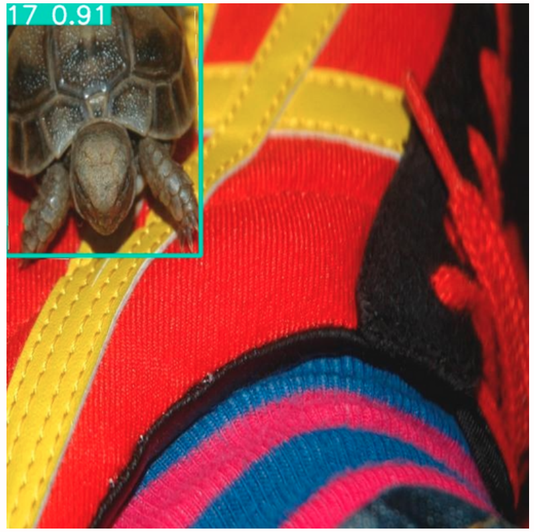
PERFORMANCE

- Overall fairly accurate
- Affected by class imbalance

Class	Images	Labels	P	R	mAP@.5	mAP@.5:.95:
all	1711	1873	0.746	0.707	0.771	0.571
10	1711	477	0.923	0.751	0.911	0.74
17	1711	617	0.974	0.779	0.945	0.891
18	1711	164	0.612	0.738	0.724	0.378
19	1711	119	0.491	0.765	0.664	0.443
2	1711	425	0.967	0.616	0.908	0.732
20	1711	71	0.512	0.592	0.476	0.24







CONCLUSIONS

Trainability

<4hr

Size

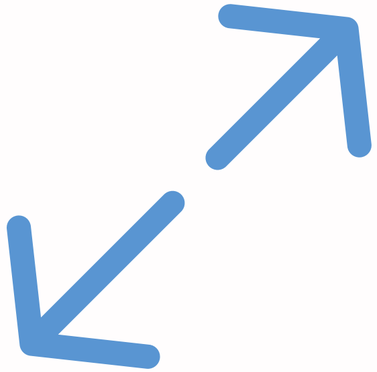
Parameters
7309034

Accuracy

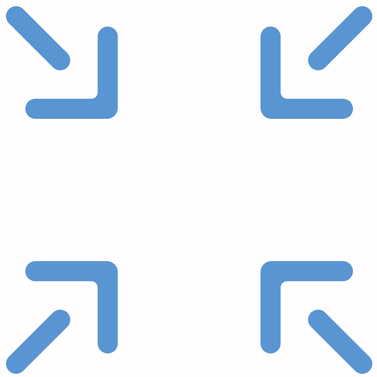
Precision
0.743

NEXT STEPS

- Expand Dataset



- Shrink Model



- Deploy in Hardware