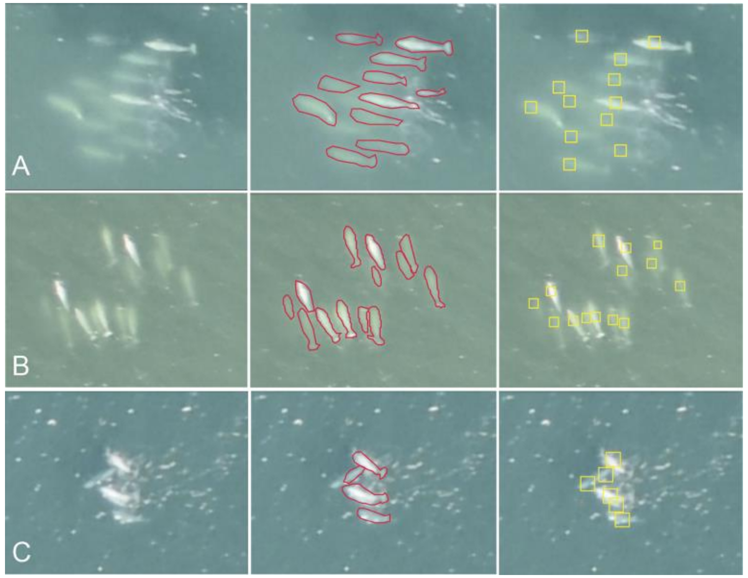
Semester Project Final Presentation
Marine Detect

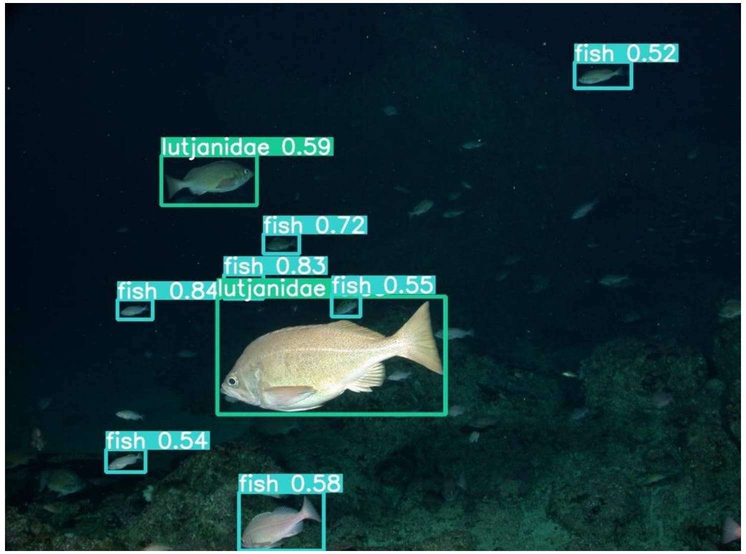
CS 3820-002: Introduction to Artificial Intelligence

Group 5: Kaylie Aguila & Phoenix Coln
Professor: Dr. Armin Moin
TA: Himon Thakur

Project Overview

- Marine Detect: Analyze images and video of underwater marine life to determine the name and species of animals through training and pattern recognition.
- Problem: Analyzing smaller marine life species from underwater, more detailed images, and identifying them with their general specific names.
- Solution: Marine Detect creates a way to study underwater living organisms in marine environments up close and differentiate them from one another, while tracking and annotating multiple subjects in an image or video.
- AI Subdiscipline: Machine Learning/Pattern Recognition





https://www.frontiersin.org/journals/marine-science/articles/10.3389/fmars.2023.1099479/full

Datasets

6 Mussels: https://universe.roboflow.com/beesknight/mollusk-ull9y

Common Dolphin: https://universe.roboflow.com/underwater-fish-f6cri/dophin

Clownfish: https://universe.roboflow.com/itk-amrull/amphiprion_percula

Flounder: https://universe.roboflow.com/belltree86/flounder

Leopard coral grouper: https://universe.roboflow.com/unsoed-d5e9h/plectropomus-leopardus-otdg9

Fistularia Commersonii (Bluespotted cornetfish): https://universe.roboflow.com/msc-pt2/msc-pt-2

Lobster: https://universe.roboflow.com/project-ppzen/cd-fc6du

Octopus: https://universe.roboflow.com/csj/octo-vwken

Banded Eagle Ray: https://universe.roboflow.com/universitas-jenderal-soedirman/itk-5b

Black Diamond & Albino Stingray: https://universe.roboflow.com/arandii/black-diamond-sting-ray-detection

Seahorse: https://universe.roboflow.com/andre-rsy7b/seahorse-fazfj

Bulldog, Mako, Tiger, White Shark: https://universe.roboflow.com/bioshark/white-sharks

Squid: https://universe.roboflow.com/brenda-brmqg/squid-image-dataset

Datasets:

- Includes three directories: test, train, validation
- Train: images used to train the model on specific species
- Test: used to evaluate accuracy on unseen data instances
- Validation: used to evaluate the model's performance

Licenses:

- CC0 1.0 Universal/Public Domain: https://creativecommons.org/publicdomain/zero/1.0/
- CC Attribution 4.0 International: https://creativecommons.org/licenses/by/4.0/deed.en

Customer and End-Users

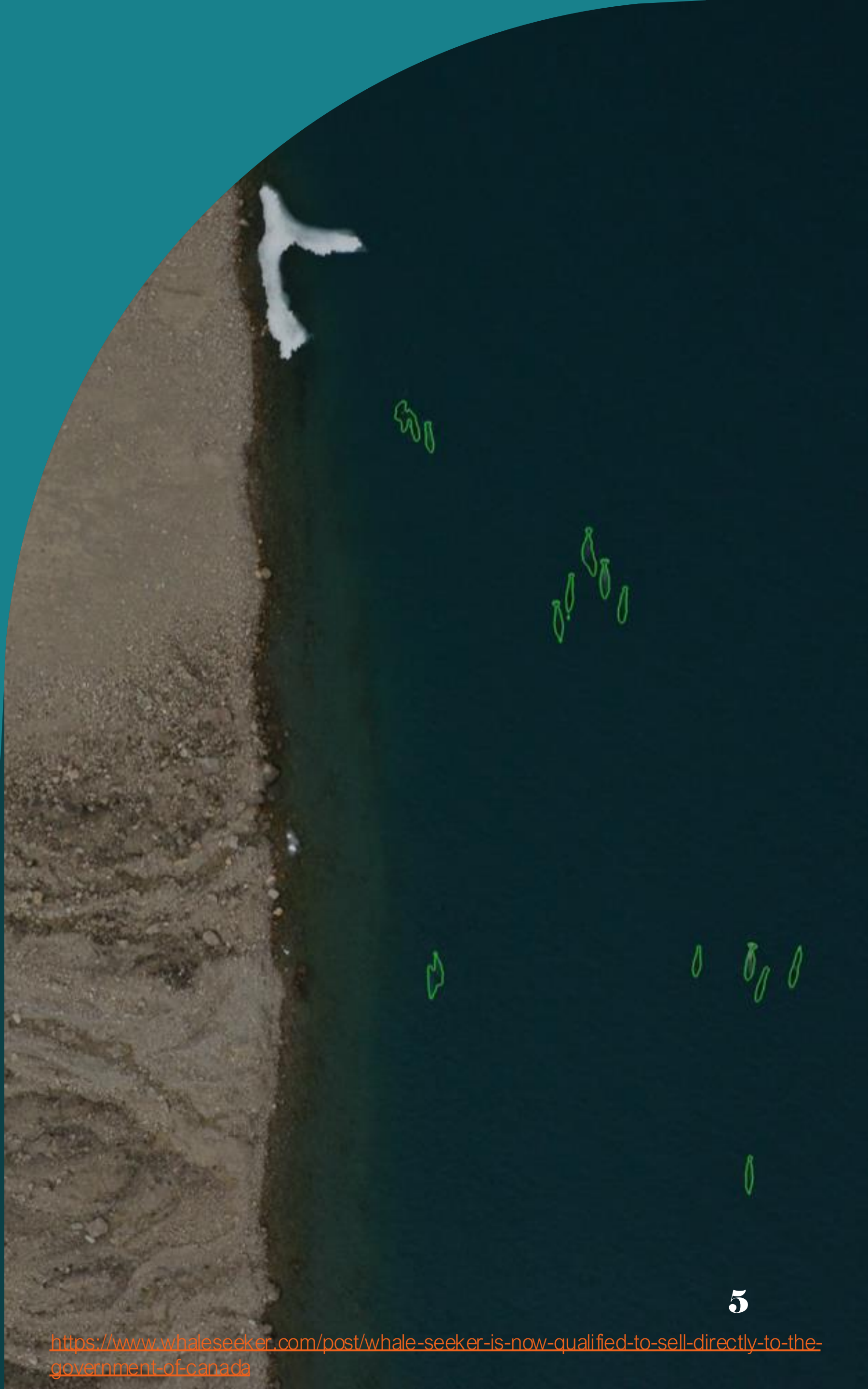
- Customers and End-Users: Researchers, Educators, and Marine life enthusiasts
- Researchers: Allows quicker analyzation of marine life through video or pictures
  - Study marine ecosystems
  - Document marine life in certain areas where images were taken
- Educators: Allows for simple and easy identification of marine life
  - Smart screen at aquariums for identifications
  - Teachers providing easy identifications for students
- Marine life enthusiasts: Easy access to identifications of marine life
  - Fishing
  - Diving

Related Works & USP

- FathomNet: drop down menu lookup that uses taxonomic names in image annotations.
  - https://fathomnet.org/fathomnet/#/
  - Source Code: https://github.com/fathomnet
- Whale Seeker - Möbius: uses aerial and satellite image analysis to detect marine mammals and environments
  - https://www.whaleseeker.com/

Unique Selling Point:

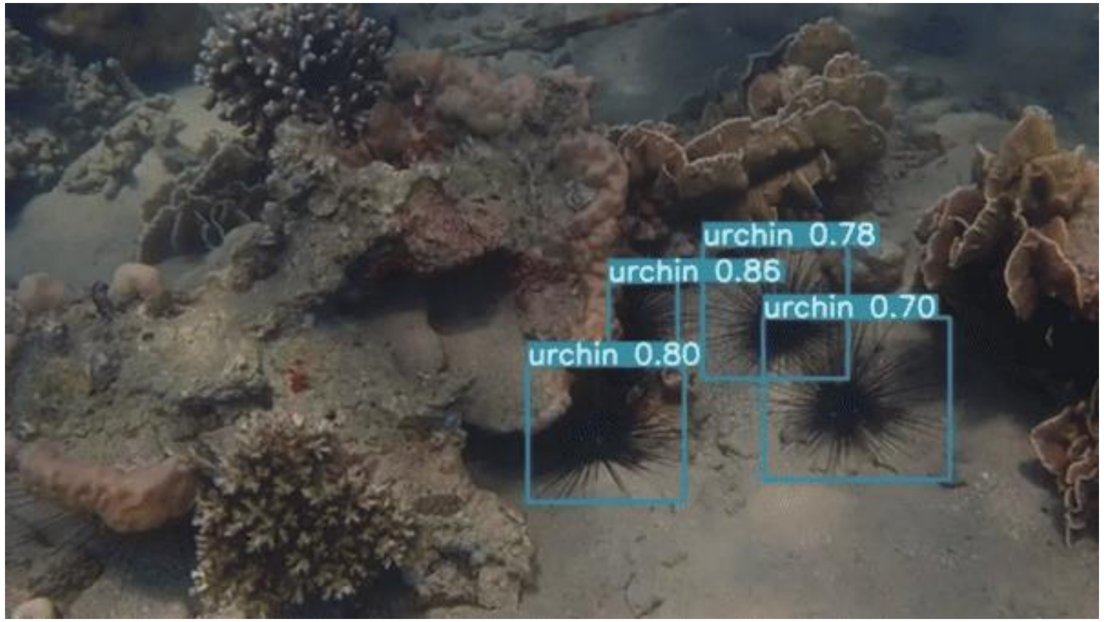
- Simple and easy to read annotations on images
- Analyzing both images and videos
- Broader specification of marine animals
- Open to more audiences than researchers or marine biologists



https://www.whaleseeker.com/post/whale-seeker-is-now-qualified-to-sell-directly-to-the-government-of-canada

Works to Use

- Orange: Marine Detect
  - o https://marine.orange.com/en/
  - o Source Code Repository: https://github.com/Orange-OpenSource/marine-detect
  - o GNU Affero General Public License v3.0
  - o Offers image and video algorithm
  - o Python libraries Pillow and CV2 (image reading & processing), Ultralytics (model training)



Tasks

- Task Breakdown
  - Data Ingestion & Preparation
    - Primarily using Roboflow datasets for the different species
  - Model Building
    - Feature Engineering – determine relevant and useful data
  - Model Training
    - Ensemble Learning – separate models for each species
  - Model Evaluation
    - Test the model and evaluate/fine tune model for new instances
  - Performance Monitoring
    - Retraining and adjusting for more optimal solutions
- Repeat pipeline as necessary to optimize outputs
- Keep updating constraints iteratively to optimize inconsistent results

- Kaylie:
  - Data Ingestion, Data segregation
  - Model Building
  - Model Training
  - Retraining
- Phoenix:
  - Data Preparation
  - Model Building, Feature Engineering
  - Model Evaluation
  - Performance analyzing

Timeline

Date	General Task	Task: Kaylie	Task: Phoenix
Oct 20 - 24	Refine Problem Definition as needed alongside Data Ingestion, Preparation, and Segregation	Data Ingestion: Collect public datasets within scope of problem Data Segregation: determine which to use for training and testing	Data Ingestion: Collect public datasets within scope of problem Data Preparation: remove duplicates, check data is not overfit
Oct 30 - Nov 6	Model Building and Feature Engineering	Model Building: implement constraints and rules from features	Feature Selection: determine constraints based on relevance and correlation
Nov 6 - 20	Interleaved Model Training, Evaluation, and Performance Monitoring	Feature Engineering: Iteratively train model and adjust model for inconsistent areas	Feature Importance: evaluate accuracy, make changes for next training and determine weak areas
Nov 20 - 24	Deploy final model and final fixes	Prepare final product: final optimizations on learning	Prepare final product: determine final accuracy and performance
Nov 25 - Dec 1	Fall Break	----	----
Dec 2	Prepare and present final presentation	Split presentation slides	Split presentation slides

Evaluation Metrics

```
Results for Clam Model:
 Average Precision: 0.59
 Average Recall: 0.66
 Average F1 Score: 0.61
 Average Accuracy: 0.66

Results for Dolphin Model:
 Average Precision: 0.64
 Average Recall: 0.82
 Average F1 Score: 0.69
 Average Accuracy: 0.82

Results for Bluespotted Cornetfish Model:
 Average Precision: 0.67
 Average Recall: 0.83
 Average F1 Score: 0.72
 Average Accuracy: 0.83

Results for Clownfish Model:
 Average Precision: 0.93
 Average Recall: 1.00
 Average F1 Score: 0.95
 Average Accuracy: 1.00
```

```
Results for Flounder Model:
 Average Precision: 0.80
 Average Recall: 0.88
 Average F1 Score: 0.81
 Average Accuracy: 0.88

Results for Leopard Coral Grouper Model:
 Average Precision: 0.93
 Average Recall: 0.96
 Average F1 Score: 0.94
 Average Accuracy: 0.96

Results for Lobster Model:
 Average Precision: 0.56
 Average Recall: 0.54
 Average F1 Score: 0.49
 Average Accuracy: 0.54

Results for Octopus Model:
 Average Precision: 0.54
 Average Recall: 0.60
 Average F1 Score: 0.55
 Average Accuracy: 0.60
```

```
Results for Eagle Ray Model:
 Average Precision: 0.96
 Average Recall: 1.00
 Average F1 Score: 0.97
 Average Accuracy: 1.00

Results for Black Diamond/Albino Stingray Model:
 Average Precision: 0.67
 Average Recall: 0.66
 Average F1 Score: 0.62
 Average Accuracy: 0.66

Results for Seahorse Model:
 Average Precision: 0.44
 Average Recall: 0.59
 Average F1 Score: 0.48
 Average Accuracy: 0.59

Results for Shark Model:
 Average Precision: 0.57
 Average Recall: 0.62
 Average F1 Score: 0.59
 Average Accuracy: 0.62

Results for Squid Model:
 Average Precision: 0.77
 Average Recall: 0.89
 Average F1 Score: 0.80
 Average Accuracy: 0.89
```

Evaluation Metrics – Comparison

Given models: from Orange's Marine Detect

Fish & Invertebrates Model:

- Trained & Validated on ~12,500 images
- Tested on ~500 images

```
Average Precision: 0.17
Average Recall:    0.18
Average F1 Score:  0.17
Average Accuracy:  0.18
```

MegaFauna & Rare Species Model:

- Trained & Validated on ~8,000 images
- Tested on ~300 images

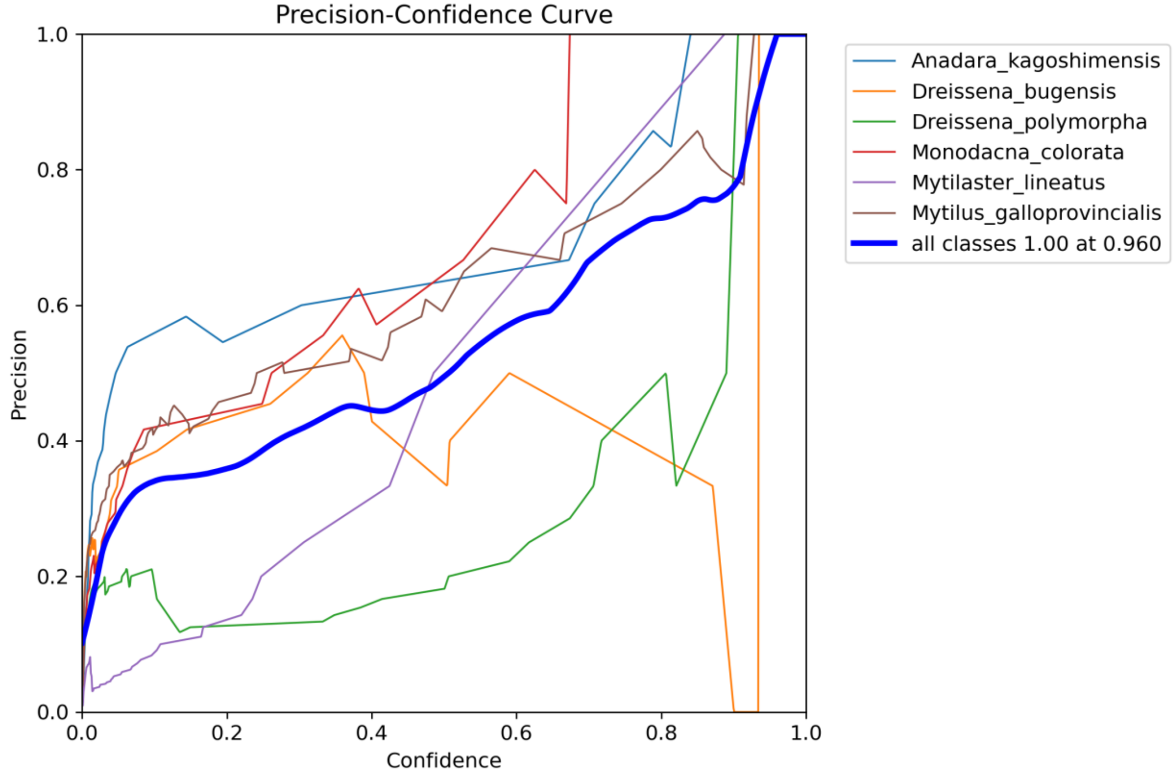
```
Average Precision: 0.67
Average Recall:    0.67
Average F1 Score:  0.66
Average Accuracy:  0.67
```

Our individual species models:

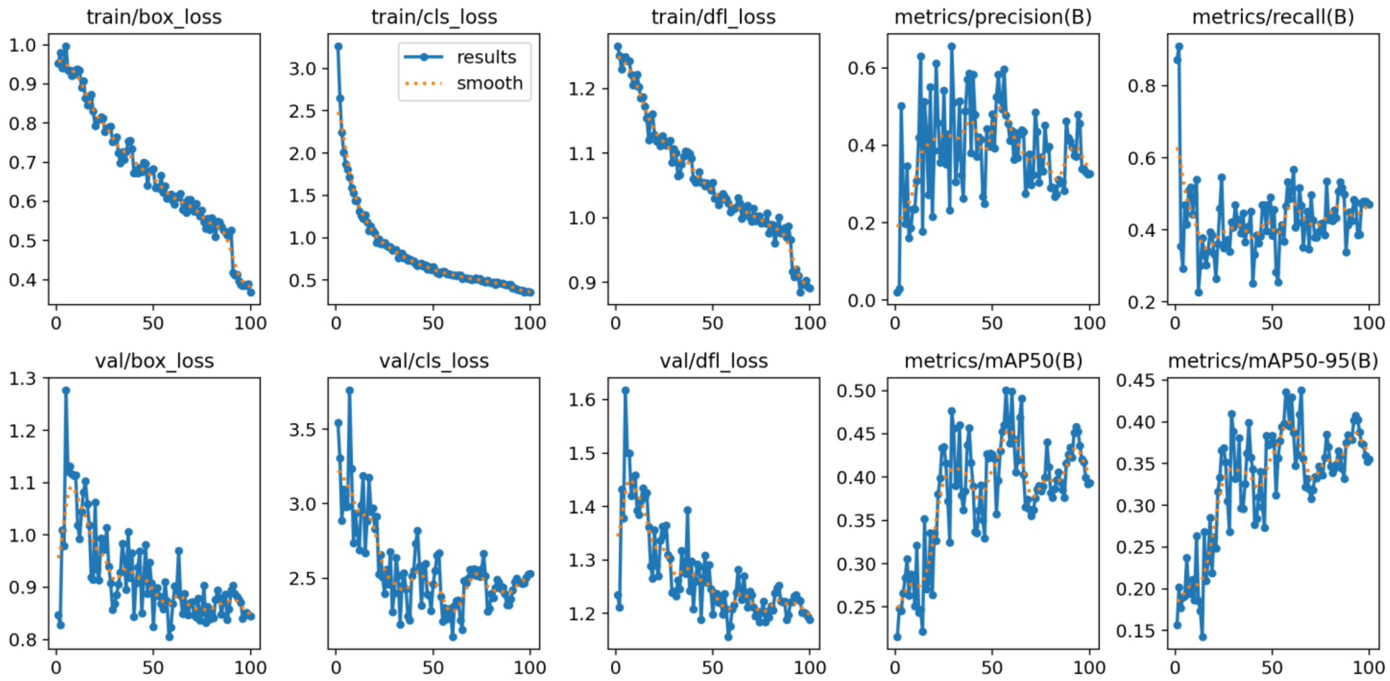
- Trained & Validated on ~5,800 images
- Tested on ~300 images

```
Average Precision: 0.70
Average Recall:    0.77
Average F1 Score:  0.71
Average Accuracy:  0.77
```

Correlation Maps



Training Results



epoch,	train/box_loss,	train/cls_loss,	train/dfl_loss,	metrics/precision(B),	metrics/recall(B),	metrics/mAP50(B),	metrics/mAP50-95(B),
1,	0.95305,	3.2648,	1.266,	0.01998,	0.87179,	0.21572,	0.15683,
2,	0.9798,	2.6502,	1.2513,	0.02865,	0.91026,	0.24804,	0.20166,
3,	0.94066,	2.247,	1.2297,	0.50221,	0.35418,	0.24571,	0.1772,
4,	0.95443,	2.0092,	1.2461,	0.28773,	0.29297,	0.26568,	0.19016,
5,	0.99699,	1.8715,	1.2492,	0.19712,	0.46874,	0.28332,	0.19147,
6,	0.93513,	1.8056,	1.2426,	0.34727,	0.41471,	0.30535,	0.23723,
7,	0.93429,	1.715,	1.2424,	0.16016,	0.50923,	0.26344,	0.19785,
8,	0.92058,	1.5892,	1.2214,	0.18641,	0.51872,	0.28797,	0.19405,
9,	0.92364,	1.5291,	1.2054,	0.23556,	0.44349,	0.28734,	0.21091,
10,	0.93628,	1.4341,	1.2049,	0.23549,	0.4929,	0.25105,	0.18643,
11,	0.93806,	1.4463,	1.2218,	0.30915,	0.54028,	0.32124,	0.26305,
12,	0.93446,	1.3113,	1.2024,	0.42015,	0.22652,	0.24348,	0.18583,
13,	0.89016,	1.2641,	1.1848,	0.63092,	0.3005,	0.24658,	0.17364,
14,	0.90734,	1.2308,	1.1871,	0.17784,	0.377,	0.22184,	0.14272,
15,	0.86399,	1.2726,	1.172,	0.51302,	0.32938,	0.3515,	0.26854,
16,	0.84571,	1.1847,	1.1542,	0.43867,	0.30282,	0.27091,	0.20961,
17,	0.86001,	1.0834,	1.1202,	0.27202,	0.3417,	0.27982,	0.2331,
18,	0.87284,	1.1375,	1.1578,	0.55125,	0.39315,	0.33552,	0.28549,
19,	0.82984,	1.0803,	1.1609,	0.21552,	0.38641,	0.26369,	0.2189,
20	0.79361	1.0556	1.1284	0.38586	0.33368	0.33474	0.26211

Training Results

- box_loss: accuracy for finding the center of an object
- cls_loss: how accurate an object is classified
- dfl_loss: how accurate an object is detected
- mAP50: Mean Average Precision at 0.5 IoU threshold

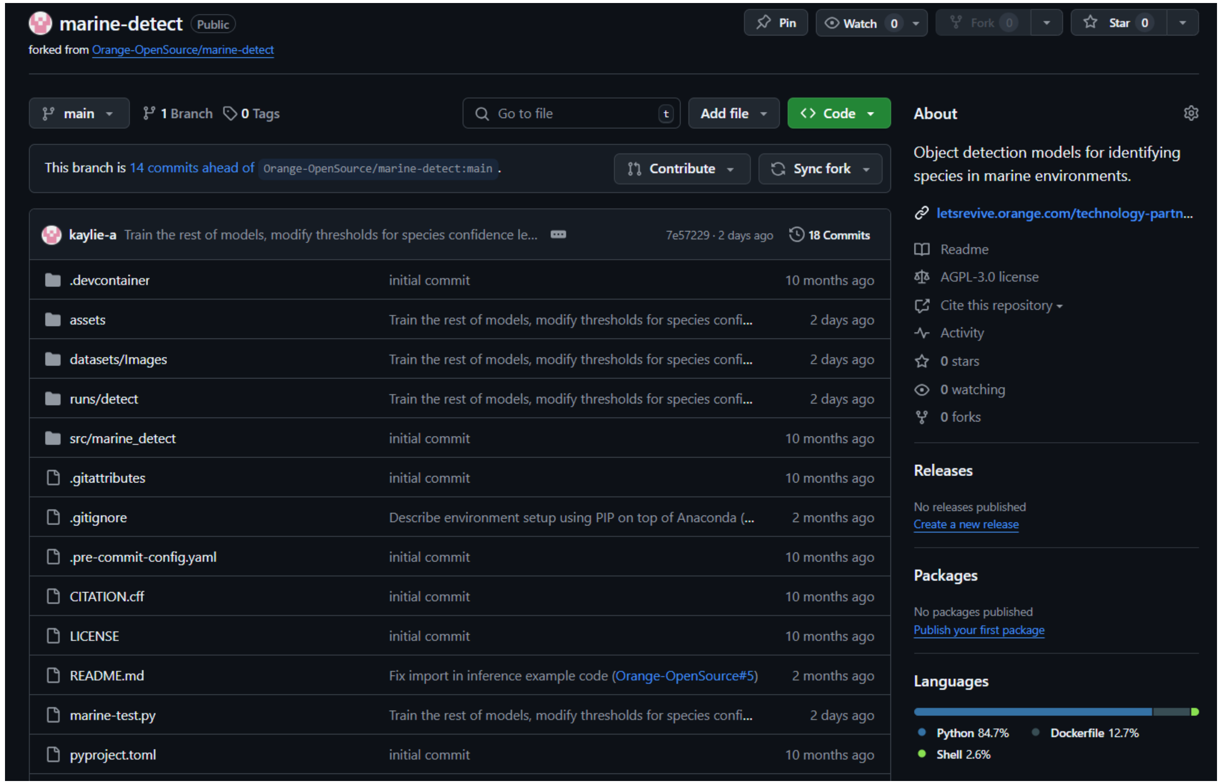
Start of training: (epoch 1)

- box_loss: 0.95305
- cls_loss: 3.2648
- dfl_loss: 1.266
- precision: 0.01998
- recall: 0.87179
- mAP50: 0.15683
- mAP50-95: 0.15683

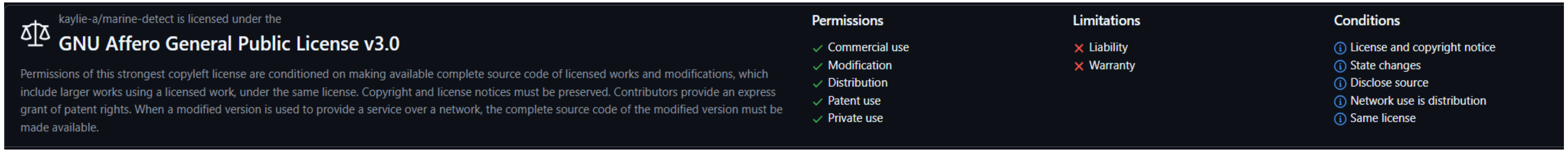
End of training: (epoch 100)

- box_loss: 0.36764
- cls_loss: 0.35566
- dfl_loss: 0.8909
- precision: 0.32715
- recall: 0.47081
- mAP50: 0.39354
- mAP50-95: 0.35545

GitHub Repository



- https://github.com/kaylie-a/marine-detect
- Repository forked from https://github.com/Orange-OpenSource/marine-detect
- Using the same license: GNU Affero General Public License v3.0



Overlay with Course Content

- Using Machine Learning/Pattern Recognition Subdiscipline to recognize certain marine life animals (L03)
- Machine Learning Pipeline based tasks that determine model development and training (L09)
- Feature Selection, Importance, and Engineering – reduce redundancy and use relevant and useful data, and provide accurate and precise measureables (L10)
- Evaluation/Performance Metrics – helped determine which species models were weaker and needed more training (sklearn) (L11)
- Ensemble Learning – built multiple models for better accuracy and predictions on individual species types (L12)

Questions?