15.01 INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO

SIMPÓSIO DE ENGENHARIA INFORMÁTICA

Semi-automatic Image Tagging for ML Workflows: a GUI for Industrial Applications

Vasco Costa, Lobinho Gomes, Carlos José Campos, Veríssimo Lima, António Sousa e Fernando Carvalho

SIMPÓSIO ENGENHARIA INFORMÁTICA · DEPARTAMENTO DE ENGENHARIA INFORMÁTICA · ISEP · P.PORTO · nei-isep · CELFOCUS · SOFTINGAL WE DO I.T.

Summary

- Introduction
- Tools and Technologies
- Software Development
  - GUI General Architecture
  - Front-end
  - Back-end
- Results
- Conclusions and Future Work

Introduction

Gislótica:

- **It is a company that designs and develops industrial machinery;**
- **Try to integrate machine learning into our equipment;**
- **This project aims to identify potential gaps in the quality control process of cordage manufacturing, such as:**
  - **Excessive twisting or misalignments;**
  - **Frayed or loose fibers;**
  - **Braiding interruptions;**
  - **Inconsistent diameter;**
  - **…because these patterns are not yet available, we use cat faces images as an example for testing the system's functionality.**

Introduction

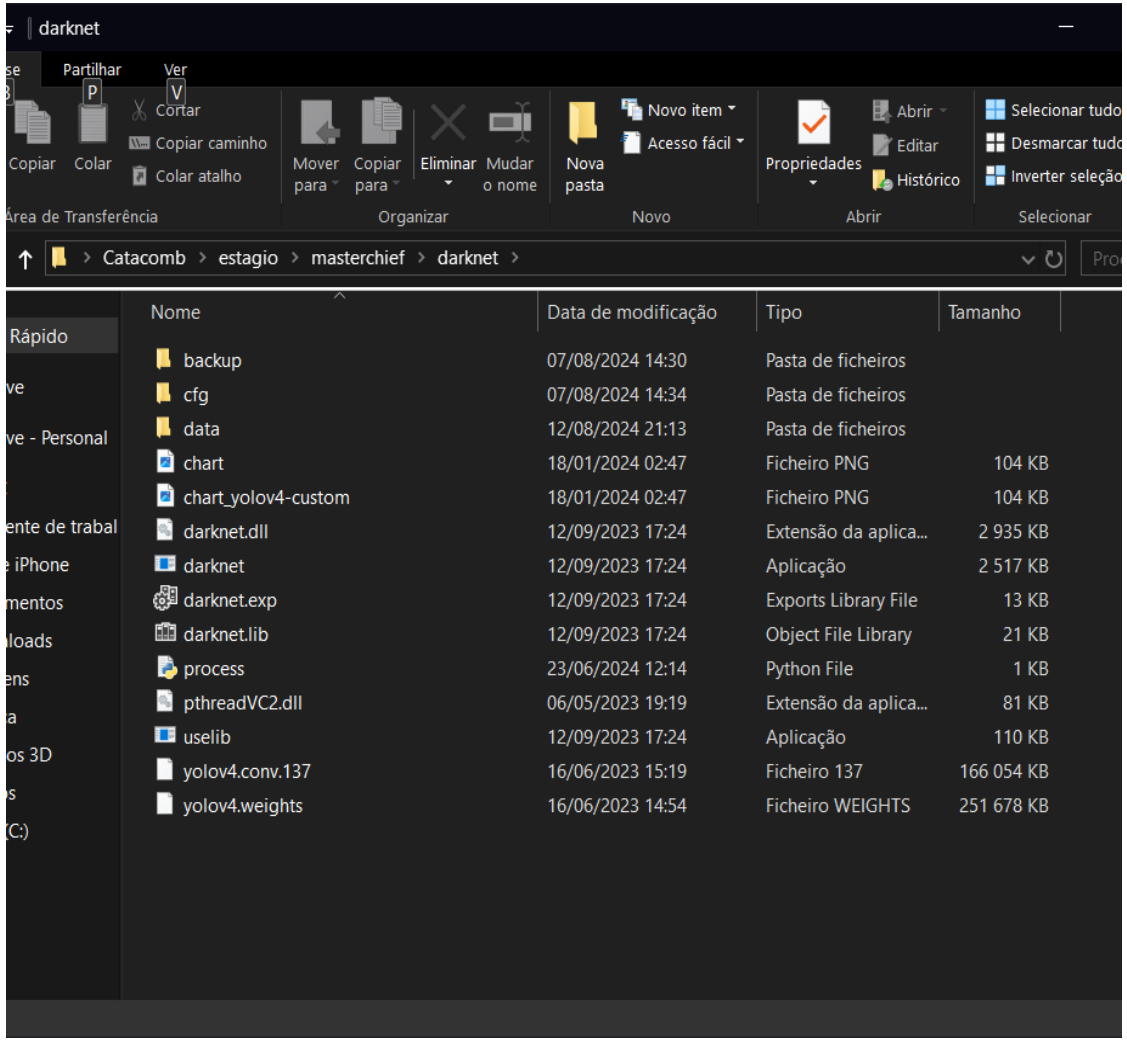
In the context of the LEE - ISEP course, the project aims to:

- **Develop the Masterchief graphical user interface (GUI) in C++ programming language using the QT framework extension and some python scripts;**

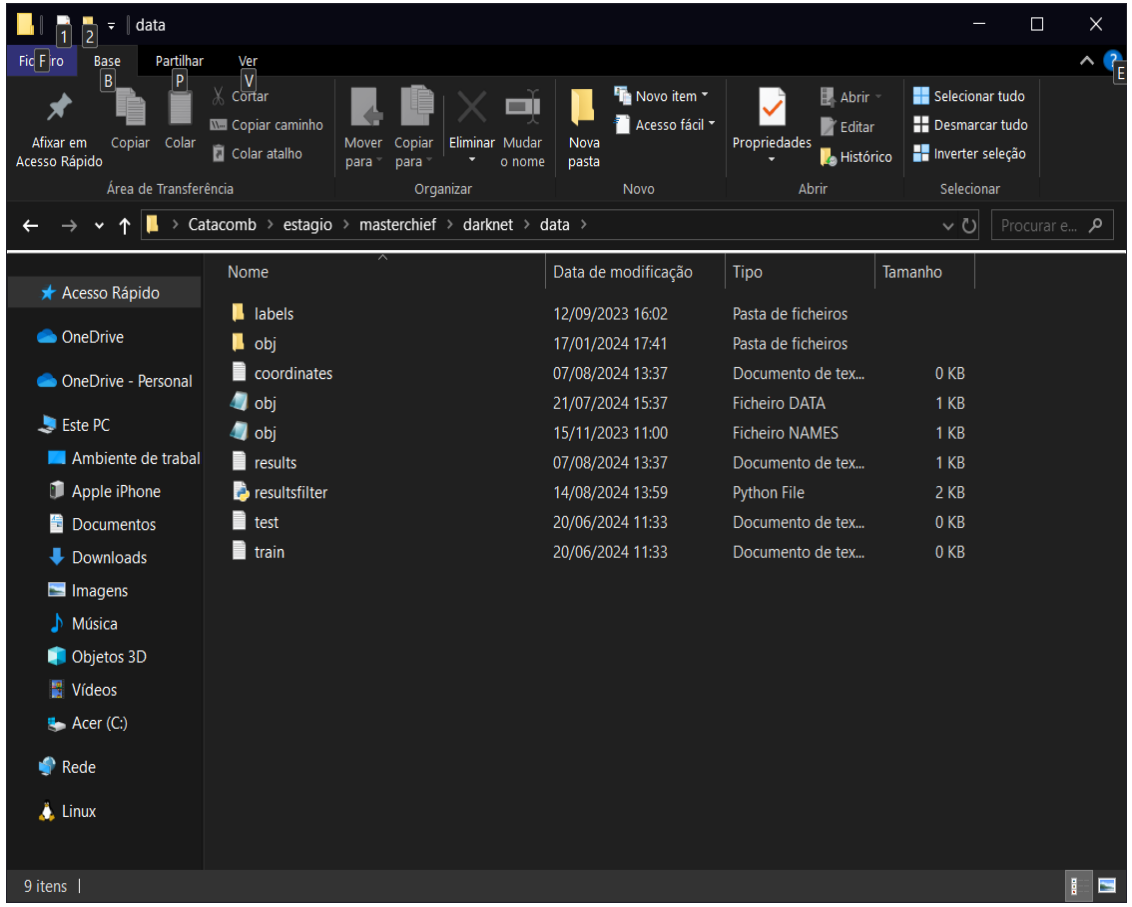
- **The GUI allows:**
  - **Manually tagging a set of patterns (sub-images) by the users;**
  - **Automated use of the Darknet framework for training the ML model based on deep learning, using the yolov4 structure;**
  - **Testing a set of unseen images during the validation phase of training.**

Tools and Technologies

Darknet framework structure, based on https://github.com/AlexeyAB/darknet



Example: **data** folder content



Tools and Technologies

yolov4.weights and yolov4.conv.137:

The yolov4.weights file contains the global combination of weights resulting from the yolov4.conv.137 (the training of first 137 layers of the YOLOv4 neural network, using general features such as edges, textures, shapes, and basic patterns, all of which are essential for object detection tasks).

For training:

./darknet detector train obj/obj.data cfg/yolov4-obj.cfg yolov4.conv.137

For testing:

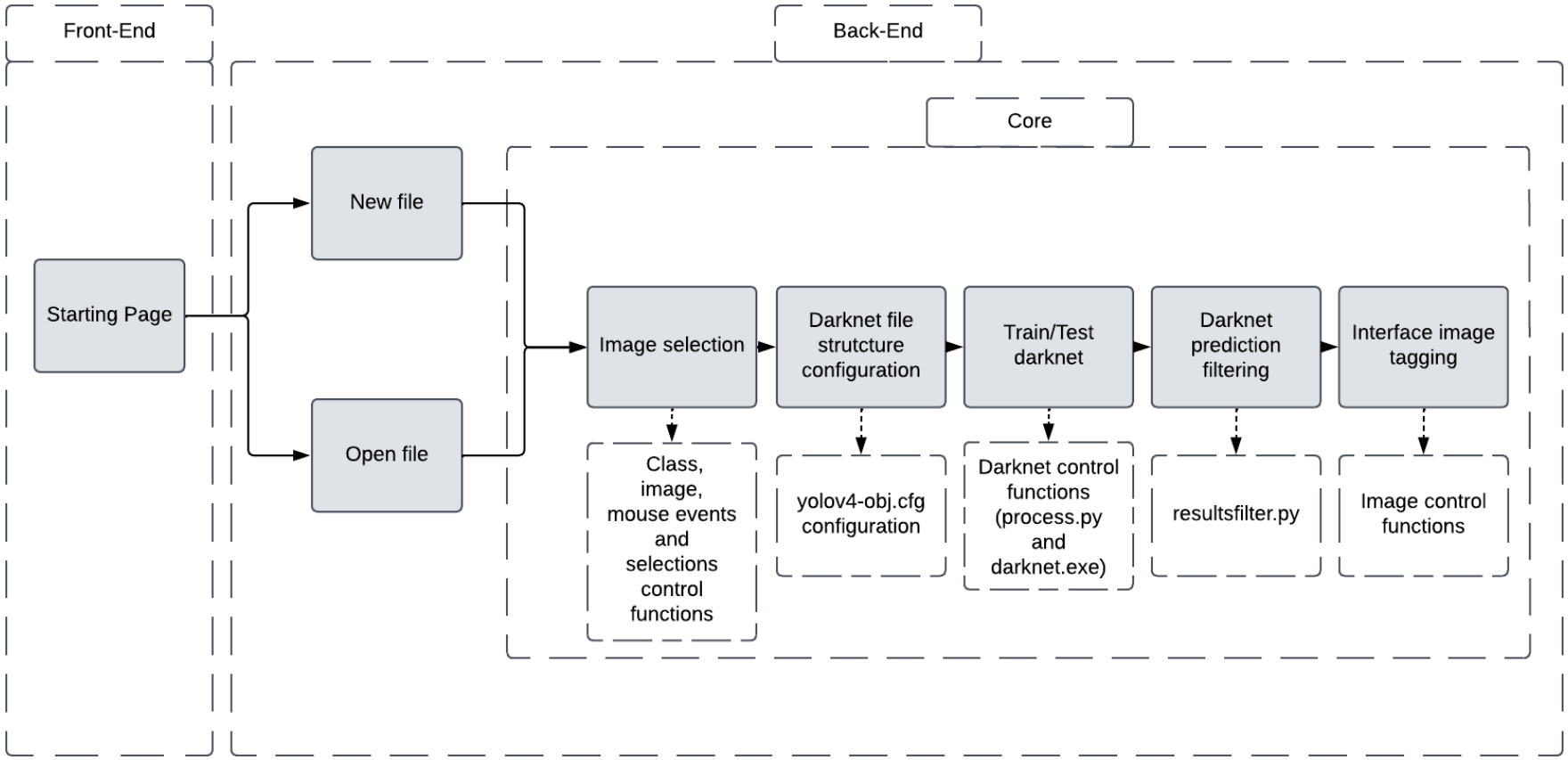
./darknet detector test obj/obj.data cfg/yolov4-obj.cfg backup/yolov4.weights data/test_image.jpg

if the yolov4.conv.137 file is absent from the training command, the yolov4.weigts is calculated from scratch. **In the results, we present the two approaches (Experiment 1 and Experiment 2, respectively).**

Image Test and Training Phases

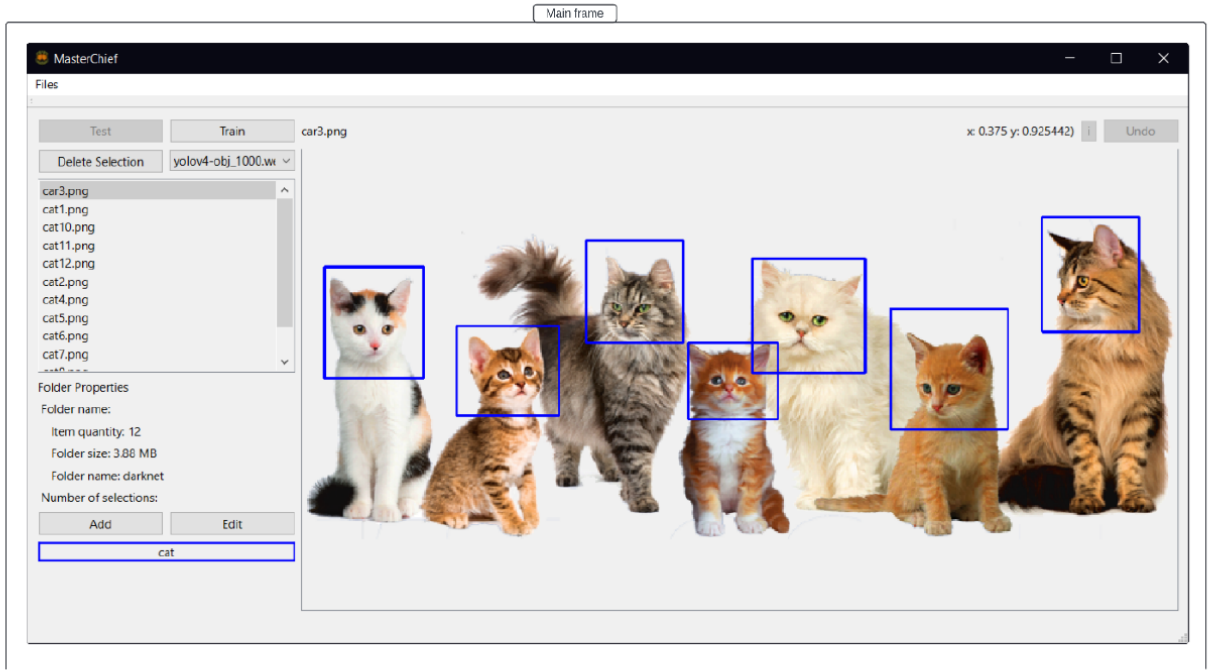
The test and training could be divided in **4 phases**:

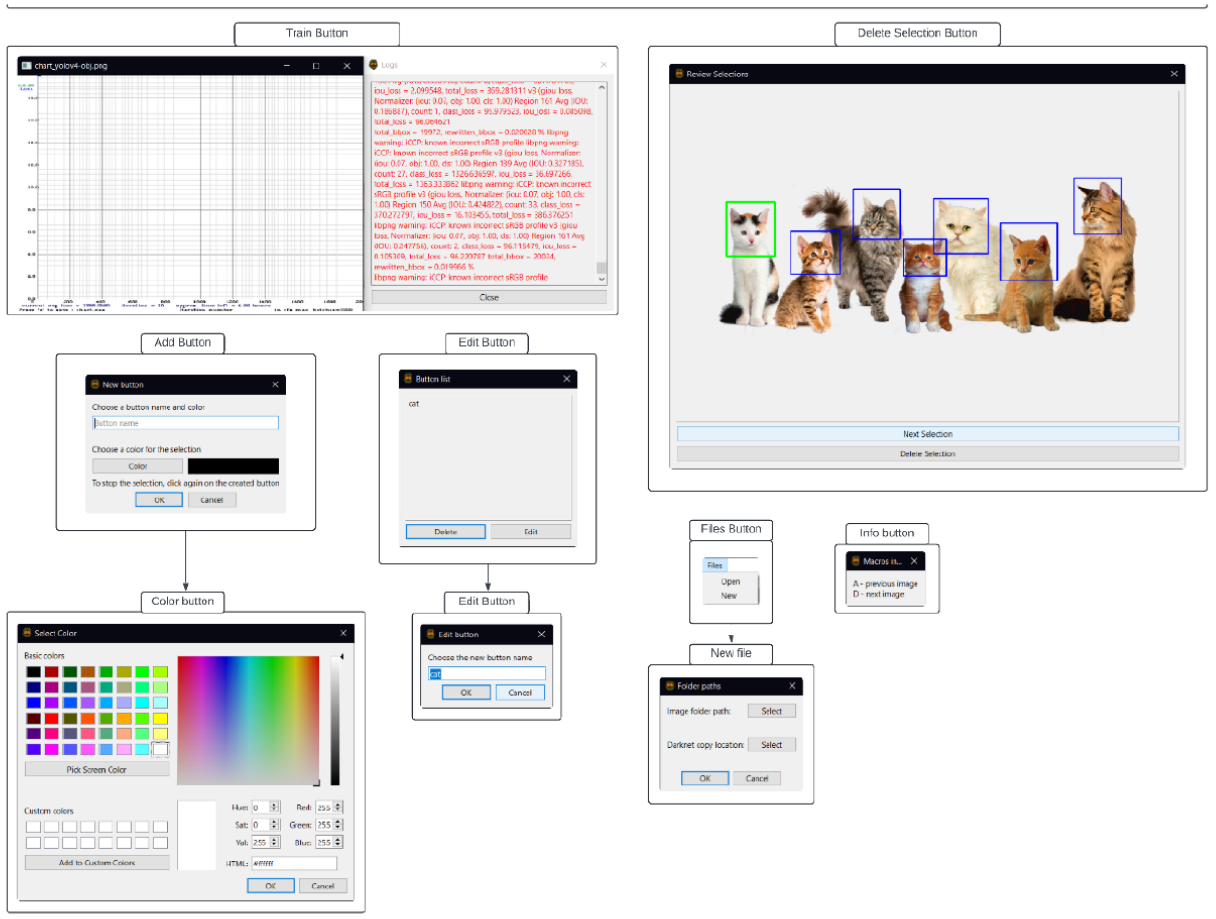
- Upload of the image dataset
- Image tagging
- Weight Training
- ML image prediction



Front-end

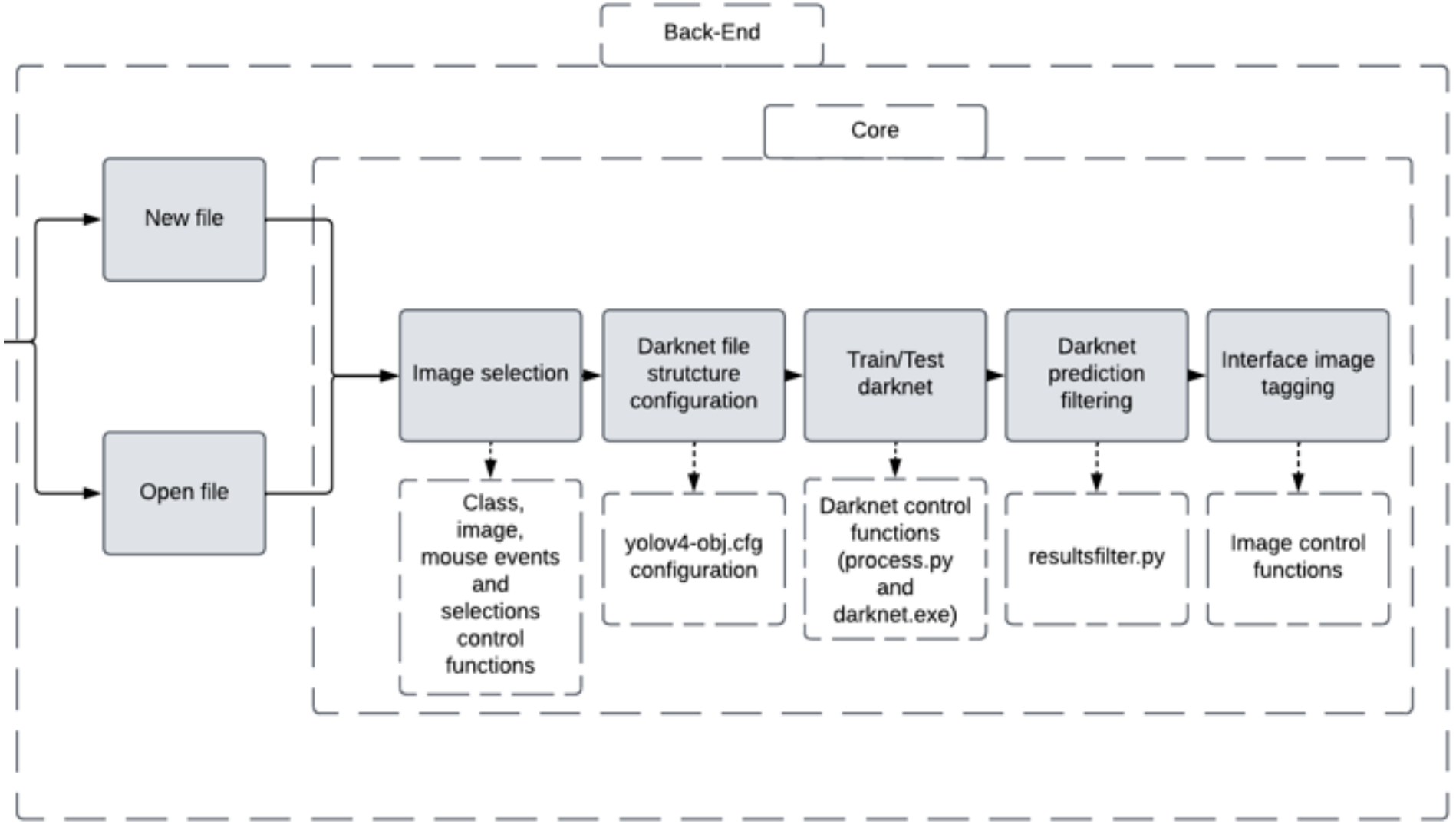
All the project windows and interactable elements.



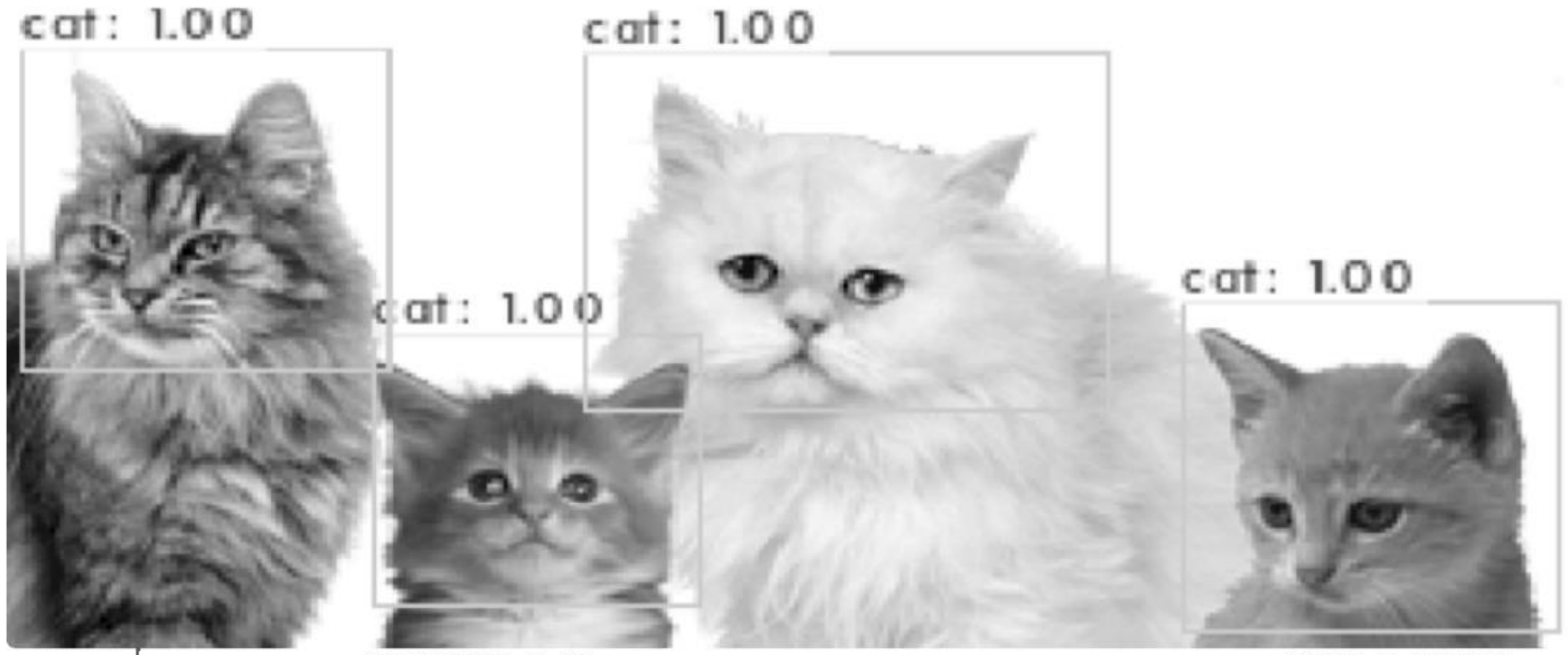


Software Development

Back-end



yolov4 predictions



Primitive data

darknet.exe predictions

```
cat: 100%  (left_x: 8    top_y:  100   width:  74  height:48)
cat: 100%  (left_x: 107  top_y:  132   width:  76  height:51)
cat: 100%  (left_x: 158  top_y:  81    width:  79  height:43)
cat: 100%  (left_x: 217  top_y:  126   width:  58  height:42)
```

Filtered data

resultsfilter.py output

```
0 8 100 74 48
0 107 132 76 51
0 158 81 79 43
0 217 126 58 42
```

Dataset

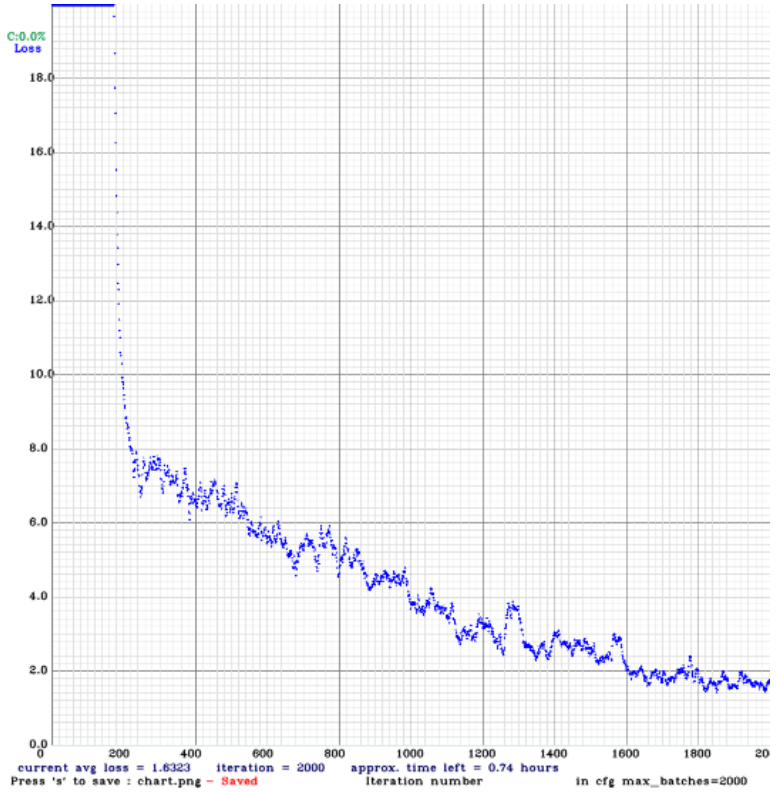
A dataset of 68 manually tagged sub-images was used for training, extrated from 19 images

Images used for testing

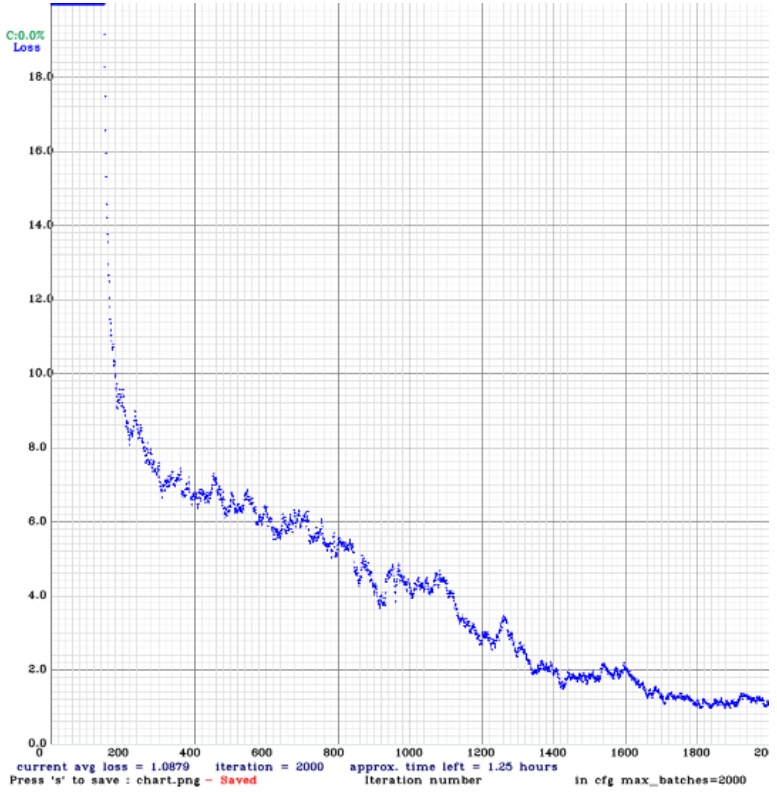
Training Loss Chart

10% of the dataset is used for validation

yolov4 trained using pre-trained weights (Experiment 1)



yolov4 trained from scratch (Experiment 2)



Yolov4-obj.cfg:

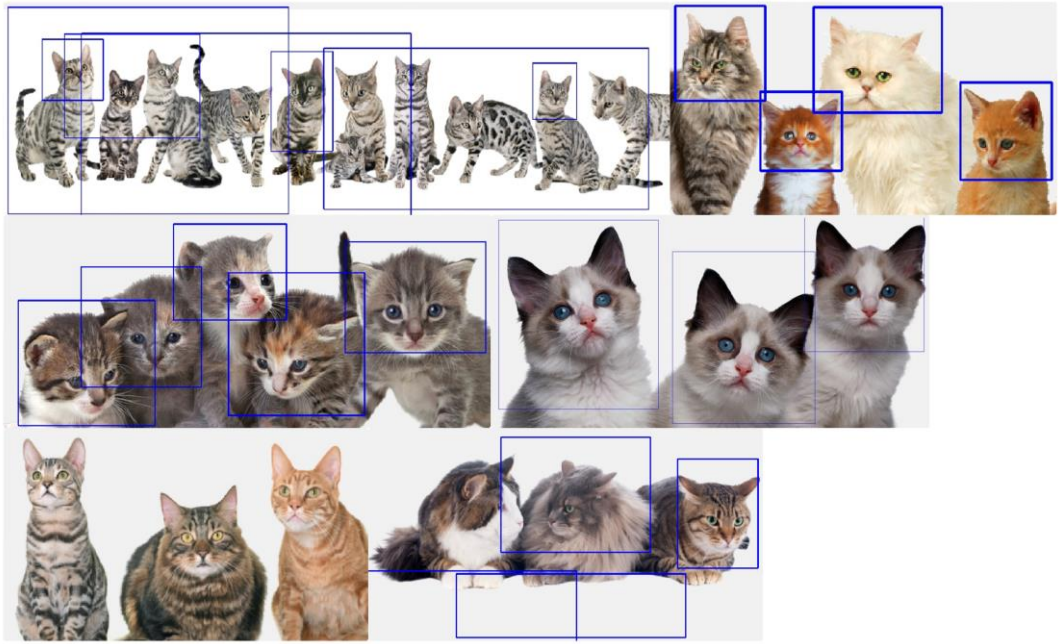
- num_classes=1
- Filters = (num_classes+5)*3
- max_batches=2000*num_classes
- subdivisions=64

Observations:

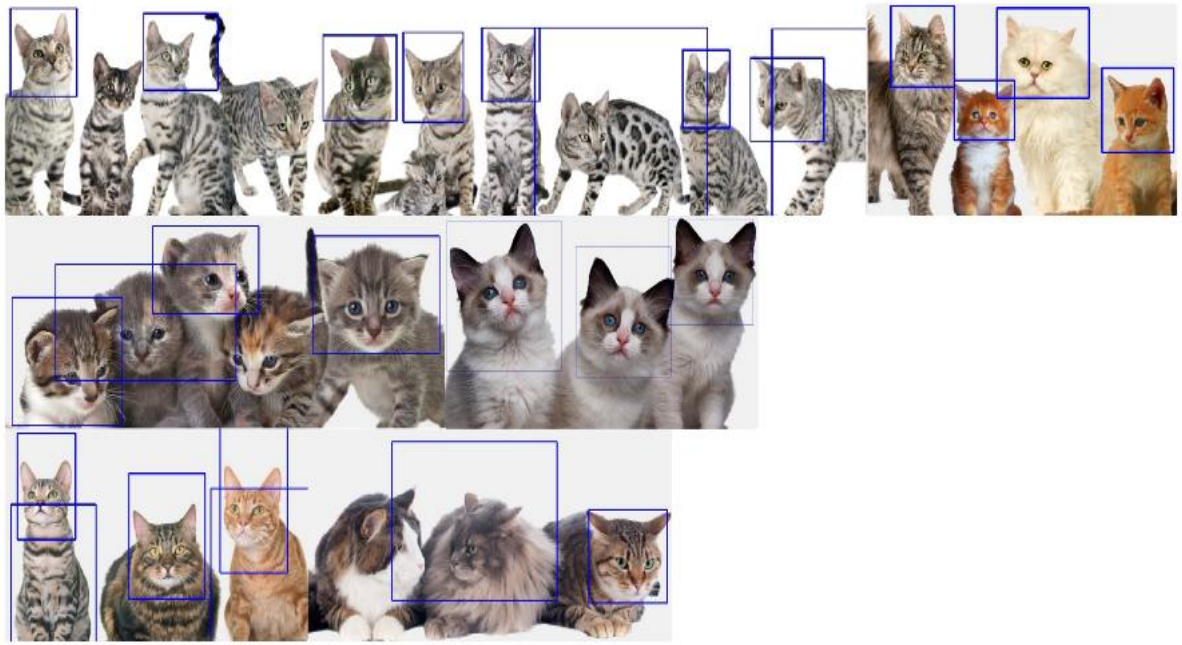
- both experiments show that the model learns well with decreasing loss;
- Oscillations suggest some generalization difficulties, more so in Experiment 1, which stabilize after 2000 iterations and decrease towards the end in Experiment 2.
- The lower average loss in Experiment 2 suggests more accurate predictions than in Experiment 1.

Test Results

Yolov4 trained using pre-trained weights (Experiment 1)

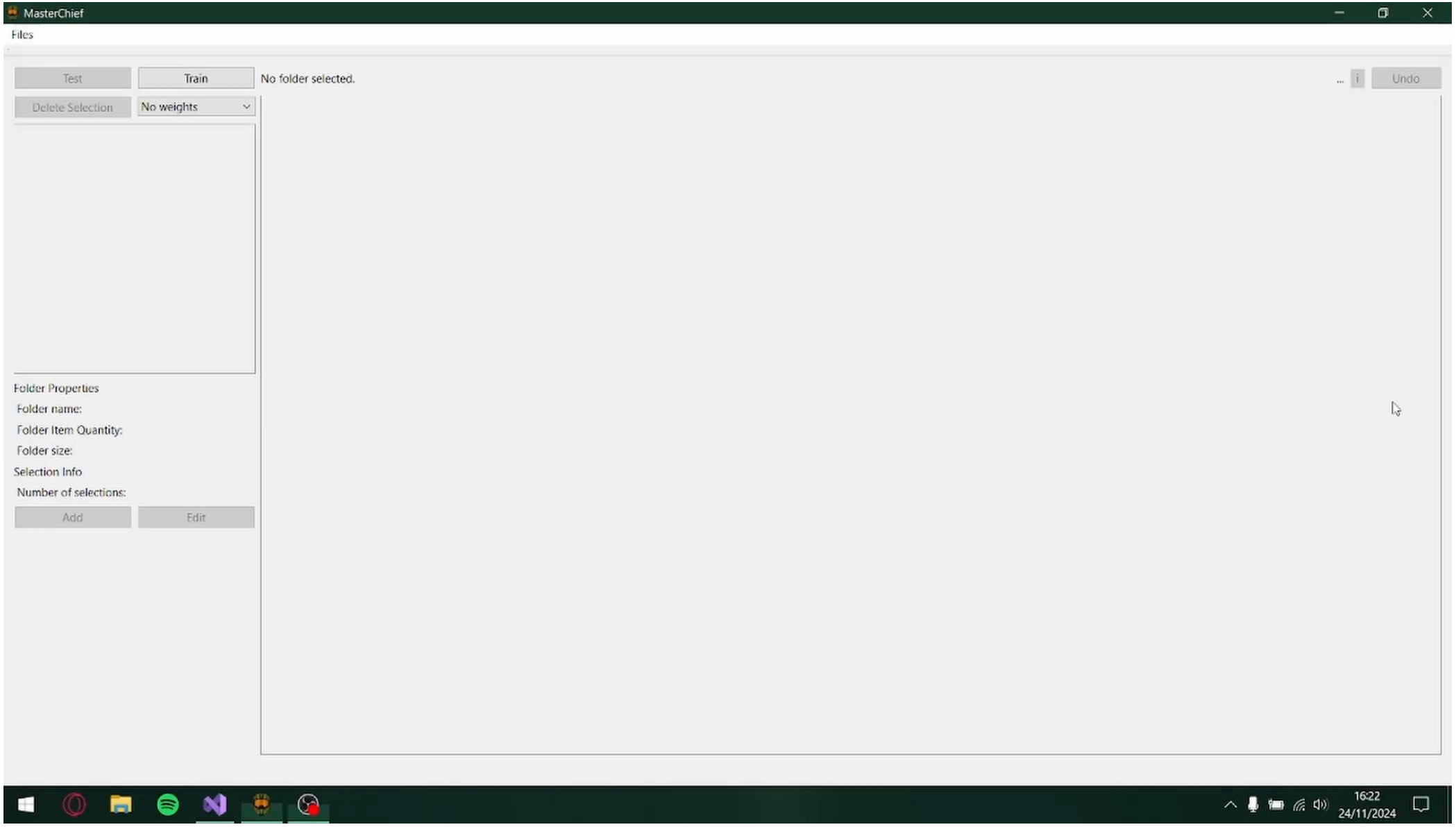


Yolov4 trained from scratch (Experiment 2)



Performing a qualitative analysis on a small test set images, reveals that the identification of cat faces in Experiment 2 is a bit more effective than Experiment 1.

Results


MasterChief
Files
Test
Train
No folder selected.
Undo
Delete Selection
No weights
Folder Properties
Folder name:
Folder Item Quantity:
Folder size:
Selection Info
Number of selections:
Add
Edit
16:22
24/11/2024

Conclusions and Future Work Perspectives

Conclusions:

- The project successfully achieved its goal of developing a robust and scalable system for ML-based object detection;
- By integrating an intuitive GUI with a powerful back-end framework, the system streamlines dataset annotation, neural network training, and continuous model improvement;
- It proved effective in optimizing object detection workflows, with seamless integration between the front-end and back-end, ensuring annotated data is efficiently incorporated into the training pipeline;
- Performance benchmarks confirmed the system's ability to handle large datasets while maintaining responsiveness during both inference and training;
- Two experiments were conducted, one using the pre-trained YOLOv4 model and the other with the same model trained from scratch, referred to as Experiment 1 and Experiment 2, respectively. While the pre-trained model reduces a typical desired training time, the initial experimental results suggest that training from scratch improves the model's ability to detect target patterns a bit more effectively.

Future Work Perspectives:

- Future work will focus on experimenting with the pattern detection system in the cordage industry and potentially optimizing the system's performance by integrating existing advanced YOLO models.
- It would then be useful to use the validation loss during the training phase to detect overfitting, select an appropriate learning rate, and apply an early stopping criterion whenever the validation loss stabilizes, even if the training loss continues to decrease.

SIMPÓSIO DE ENGENHARIA INFORMÁTICA

Thank you for you time!

SIMPÓSIO ENGENHARIA INFORMÁTICA · DEPARTAMENTO DE ENGENHARIA INFORMÁTICA · ISEP · P.PORTO · nei-isep · CELFOCUS · SOFTINSAL WE DO I.T.