

Smart Sustainability Simulation Game

Case 3: Quality Management- Unit 1
04.06.2024

FIM Research Center for Information Management
Fraunhofer Institute for Applied Information Technology FIT,
Branch Business & Information Systems Engineering

Prof. Dr. Christoph Buck
Prof. Dr. Hans Ulrich Buhl
Prof. Dr. Torsten Eymann
Prof. Dr. Gilbert Fridgen
Prof. Dr. Henner Gimpel
Prof. Dr. Björn Häckel
Prof. Dr. Robert Keller

Prof. Dr. Wolfgang Kratsch
Prof. Dr. Niklas Kühl
Prof. Dr. Anna Maria Oberländer
Prof. Dr. Maximilian Röglinger
Prof. Dr. Jens Strüker
Prof. Dr. Nils Urbach
Prof. Dr. Martin Weibelzahl

www.fim-rc.de/en
www.wirtschaftsinformatik.fraunhofer.de/bise



Augsburg

Bayreuth

Frankfurt

Luxembourg

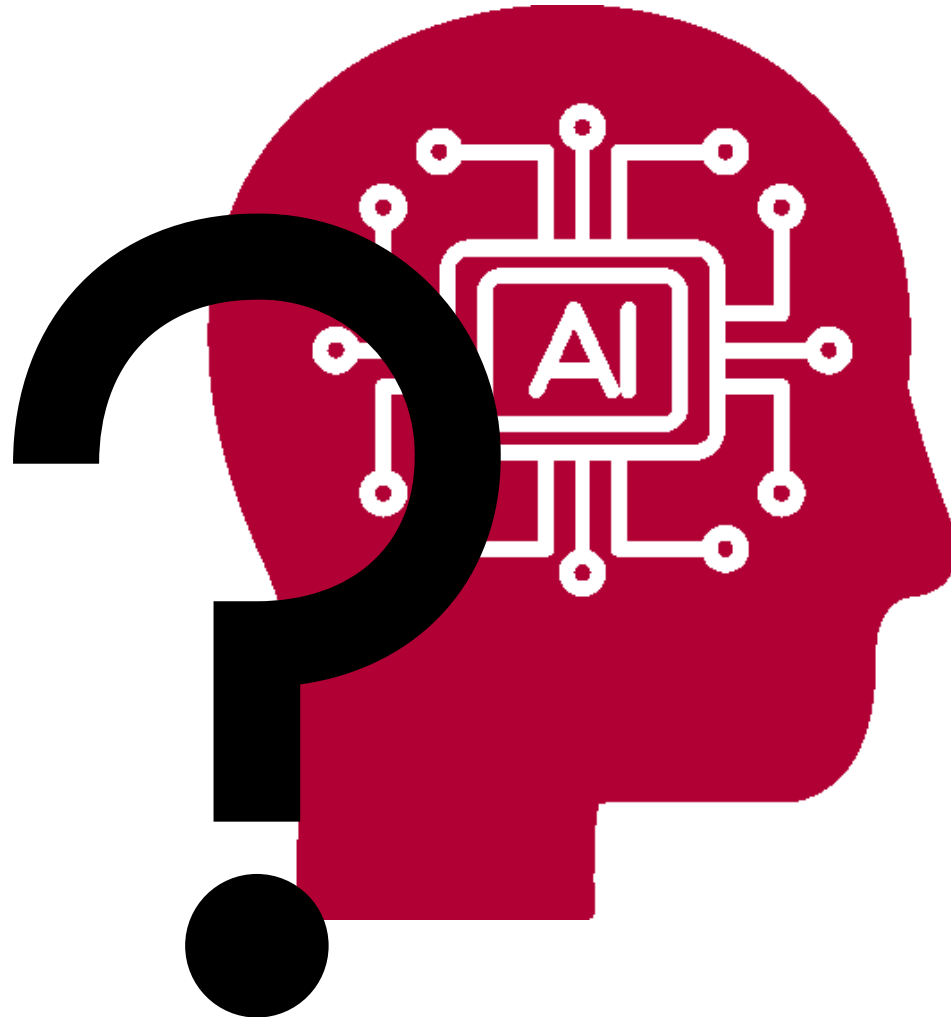
Munich

Stuttgart

01

Organizational information

Time for Feedback



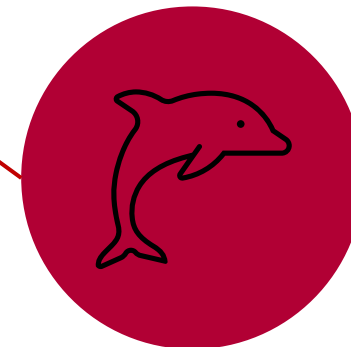
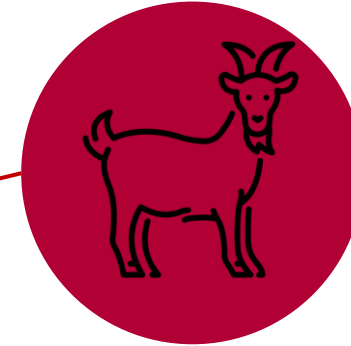
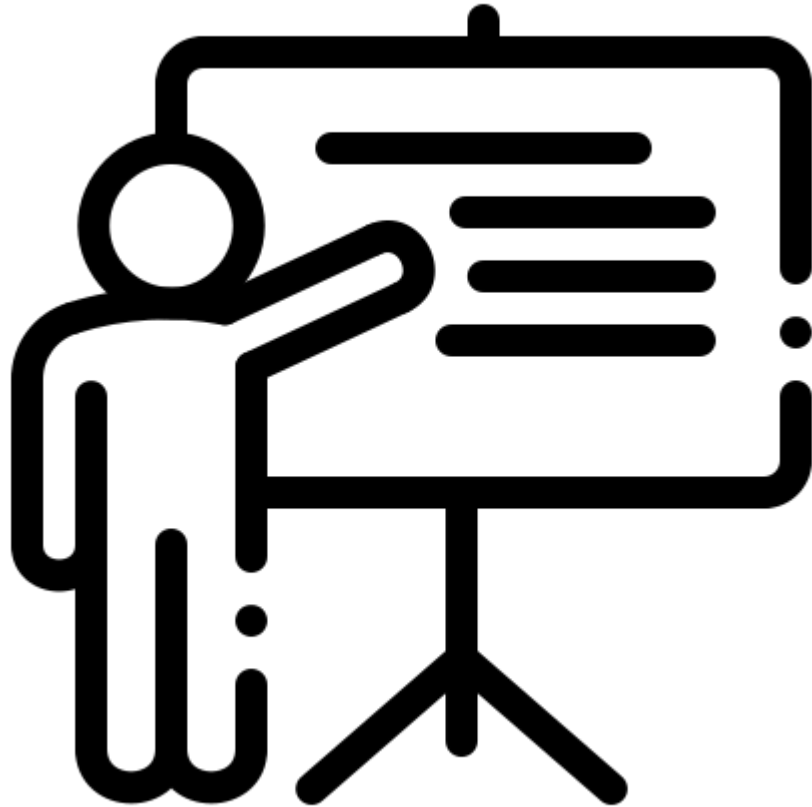
How was the
second case?

Any questions?

02

Case 2: Your results

Case 2: Presentation of results



03

Case 3: Quality Management - Unit 1

Overview of the cases

Case 1: Material procurement

- What materials should I buy and when?
 - Value chain level: Procurement
- Time Series Analysis

Case 2: Predictive Maintenance

- How often and when should I maintain my machine?
 - Value chain level: Operations/production
- Predictive Analytics

Case 4: Recycling

- How much effort do I put into recycling?
 - Value chain level: After-sales-services
- Process Mining

Case 3: Quality Management

- How to ensure good quality?
 - Value chain level: Operations/production
- Computer Vision



Case 3: Quality Control of Edison Cars AG



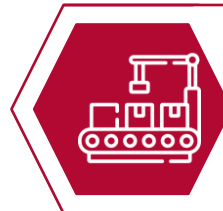
Change in demand

- Increasing end-consumer demand for high quality low-emission vehicles
- Growing competition from China leads to a need for a zero mistakes policy



New product

- Board of Edison Cars AG made the strategic decision: Abandon current combustion-engine-based cars
- New product lines tend to show more errors in production

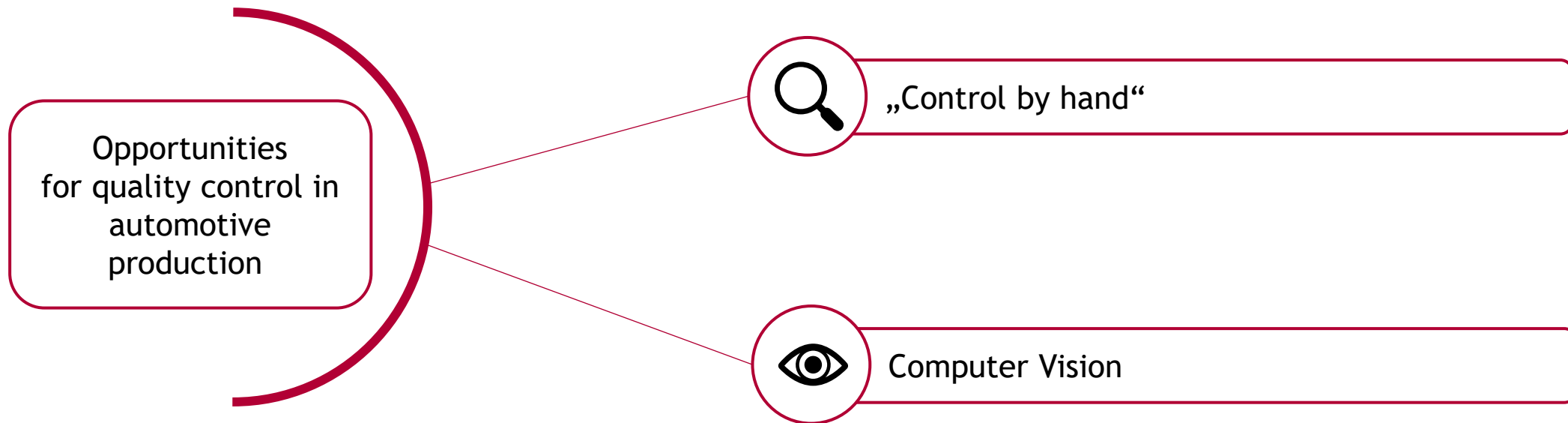


Transformation of production

- Currently, the company is putting a lot of effort in transforming their production to become leaner and more agile
- Development of new production lines with high requirements

Quality Control as a Part of Production

Automotive manufacturers have an inalienable responsibility to deliver high-quality products. General manufacturing Quality Control (QC) is an effort that reviews product quality intending to identify and eliminate defects. Discovering and fixing the defects should typically happen before the products reach the consumer.



Introduction to Computer Vision and Object Detection

What is computer vision?

- Computer vision is a part of AI
- Enables possibility to retrieve information out of digital images, videos and other visual impacts
- Possibility to check thousands of products during a production process
- Mostly implemented through deep learning, machine learning or neural networks

What is object detection?

- Computer vision technique for locating instances of objects in pictures or videos
- Typically implemented through machine or deep learning models

Case 3: Quality Management of Edison Cars AG - Task

Goal of the task of case 3 is to develop an algorithm for the **quality control** of an e-car production. This algorithm should **detect scratches** on the back side of a car and decide whether it is more efficient to **exchange the part** with the scratch or **correct it**.



The company management wants you to implement their quality control system

How to implement a working object detection

1

Select an algorithm and a pre-trained model

Do not start from scratch. We recommend using YOLOv8 since it is commonly used, easy to implement and backed by a vast amount of tutorials online.

2

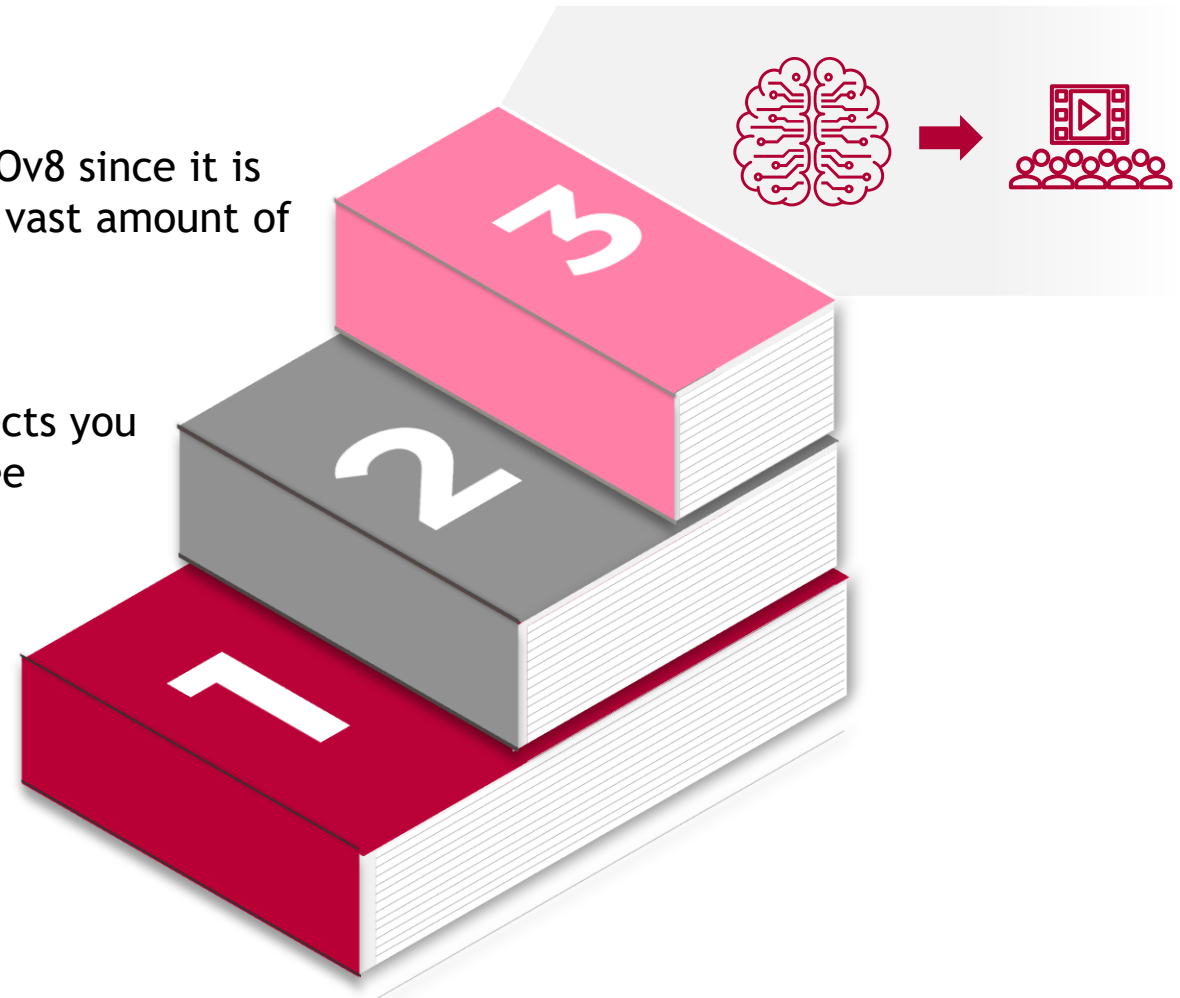
Select and label your data

Continue with labeling your input data with the objects you would like to detect and export as needed. Using free software like Roboflow is helpful for labelling.

3

Train your model and run the inference

Use your labeled data to train your custom model. After that, you might evaluate the performance of your model, improve your labelling, and continue training.



YOLOv8 as a starting point for your object detection



YOLOv8 is a computer vision model that enables you to classify, detect, segment, and track objects. Further, you can use YOLOv8 for human pose estimation, e.g., in sports.

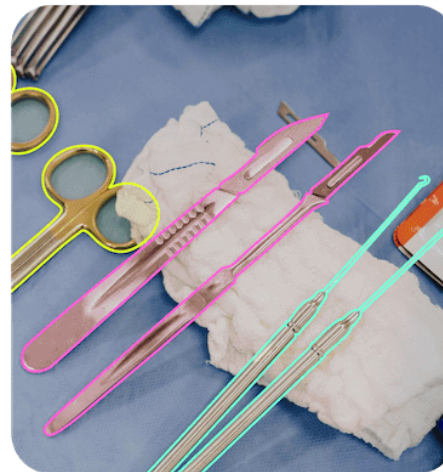
Classify



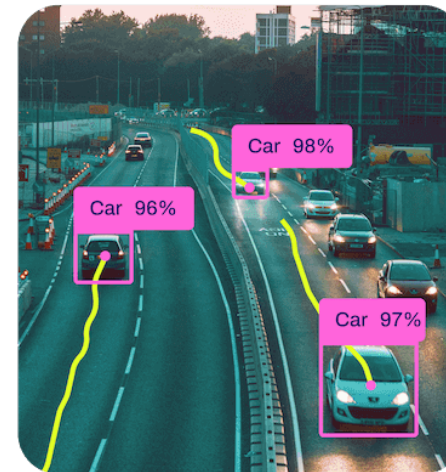
Detect



Segment



Track



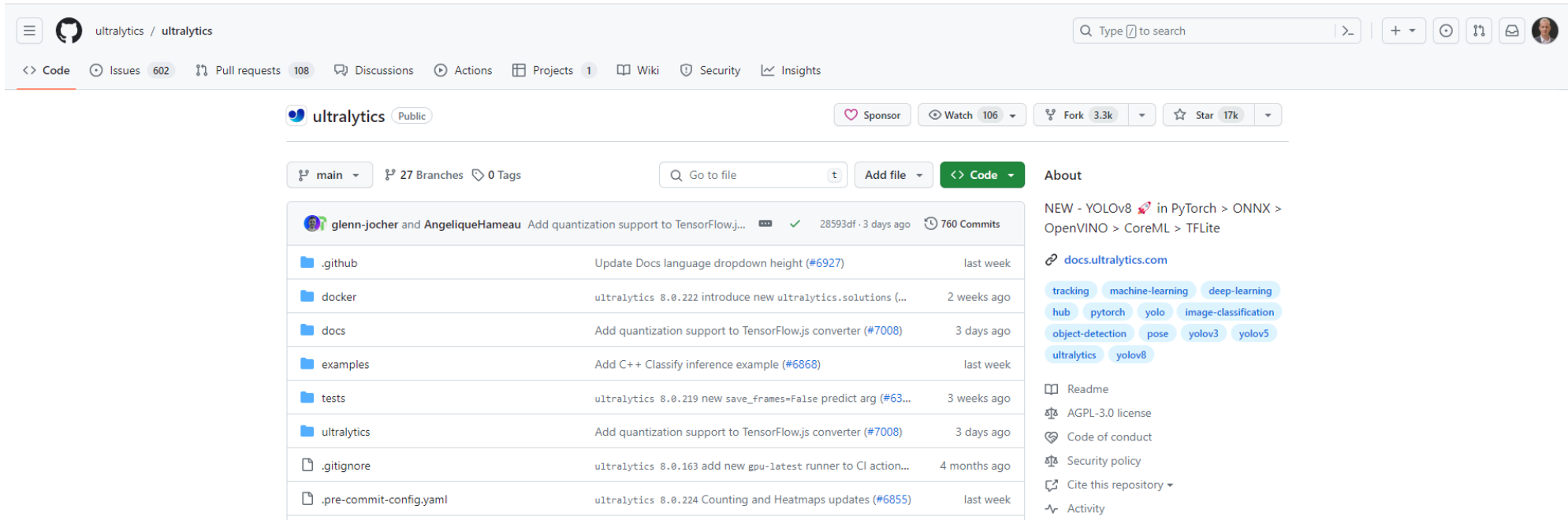
Pose



YOLOv8 as a starting point for your object detection



You can find helpful documentation on the installation and use here:
<https://github.com/ultralytics/ultralytics>



The screenshot shows the GitHub repository page for `ultralytics / ultralytics`. The repository is public and has 106 watchers, 3.3k forks, and 17k stars. The main branch is selected, showing 27 branches and 0 tags. The repository is managed by `glenn-jocher` and `AngeliqueHameau`, with 760 commits. The file list includes:

File	Description	Time
<code>.github</code>	Update Docs language dropdown height (#6927)	last week
<code>docker</code>	ultralytics 8.0.222 introduce new <code>ultralytics.solutions</code> (...)	2 weeks ago
<code>docs</code>	Add quantization support to TensorFlow.js converter (#7008)	3 days ago
<code>examples</code>	Add C++ Classify inference example (#6868)	last week
<code>tests</code>	ultralytics 8.0.219 new <code>save_frames=False</code> predict arg (#63...	3 weeks ago
<code>ultralytics</code>	Add quantization support to TensorFlow.js converter (#7008)	3 days ago
<code>.gitignore</code>	ultralytics 8.0.163 add new <code>gpu-latest</code> runner to CI action...	4 months ago
<code>.pre-commit-config.yaml</code>	ultralytics 8.0.224 Counting and Heatmaps updates (#6855)	last week

The right sidebar shows the `About` section with a new feature announcement: "NEW - YOLOv8 in PyTorch > ONNX > OpenVINO > CoreML > TFLite". It also includes links to `docs.ultralytics.com`, a list of tags (tracking, machine-learning, deep-learning, hub, pytorch, yolo, image-classification, object-detection, pose, yolov3, yolov5, ultralytics, yolov8), and a list of repository files (Readme, AGPL-3.0 license, Code of conduct, Security policy, Cite this repository, Activity).

Jupyter Notebook



Use our Jupyter Notebook as a starting point for this week's task

1. Installation und Dependencies

```
In [ ]: !pip install torch torchvision torchaudio
```

```
In [ ]: !pip install ultralytics
```

```
In [1]: import torch
from matplotlib import pyplot as plt
import numpy as np
import cv2
from ultralytics import YOLO
```

2. Loading the model

```
In [2]: model = YOLO("yolov8n.pt")
```

3. Make detections

```
In [3]: img = "https://media.npr.org/assets/img/2010/08/23/trafficjam-695199b627097a111557672a2520e2b222f48ffa-s1100-c50.jpg"
```

```
In [4]: results = model.predict(img)
```

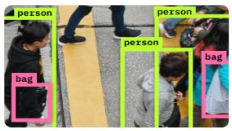

Image annotation using Roboflow Annotate

Create a project for object detection

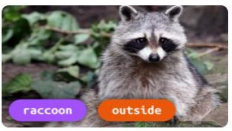
Create New Project

S3GCase3 / New Public Project


Project Type



Object Detection
Find multiple things and their specific location.



Classification
Assign labels to the entire image.



Instance Segmentation
Detect multiple objects and their actual shape.

Show More

Project Name

S3G Case 3

What are you detecting?

scratches

License

CC BY 4.0

Cancel

Create Public Project

Label the scratches: Work as precise as possible to achieve best results

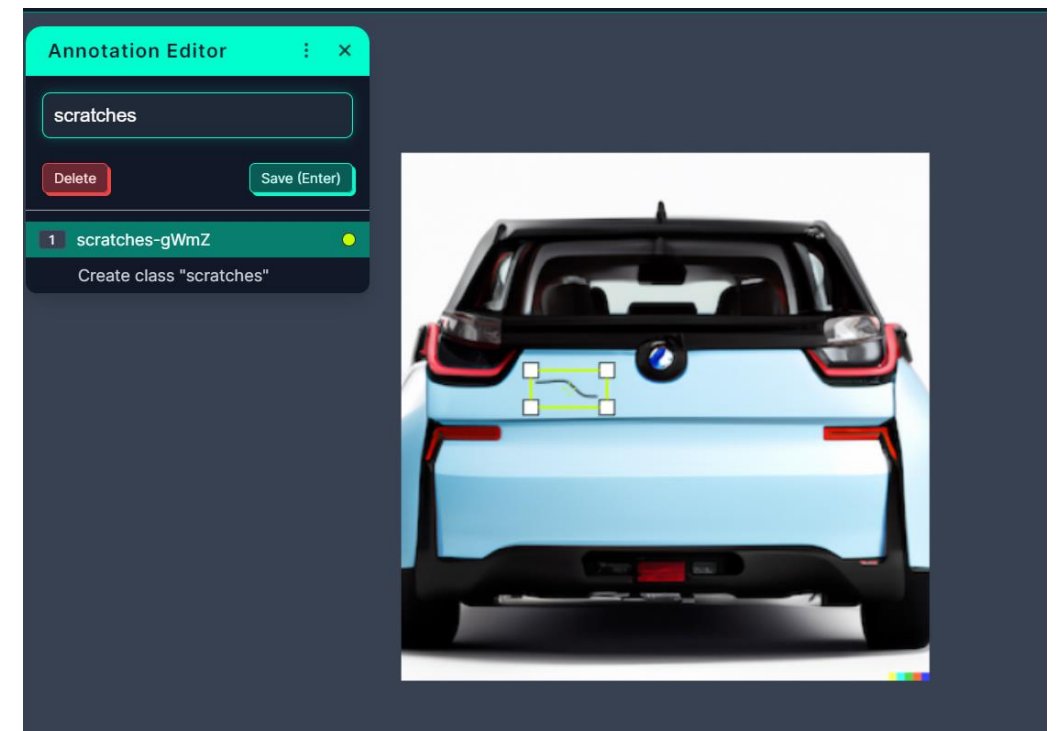
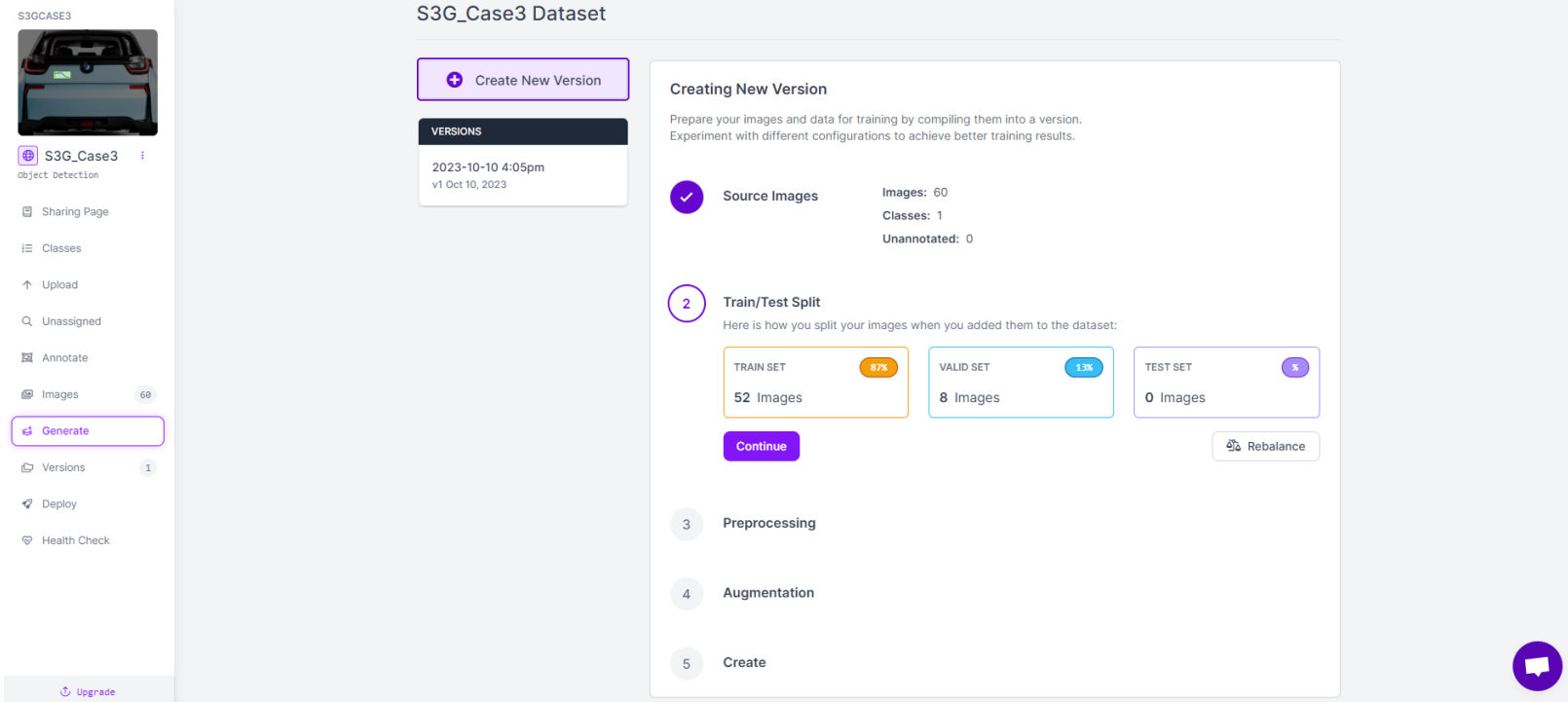


Image annotation using Roboflow Annotate

Generate your training data, including preprocessing and augmentation



The screenshot displays the Roboflow Annotate interface for the 'S3G_Case3 Dataset'. On the left, a sidebar menu includes options like 'Sharing Page', 'Classes', 'Upload', 'Unassigned', 'Annotate', 'Images' (60), 'Generate' (highlighted with a red arrow), 'Versions' (1), 'Deploy', and 'Health Check'. The main area shows the 'S3G_Case3 Dataset' with a 'Create New Version' button and a 'VERSIONS' table listing a version from 2023-10-10. The 'Creating New Version' workflow is shown with steps: 1. Source Images (60 images, 1 class, 0 unannotated), 2. Train/Test Split (Train Set: 52 images, 97%; Valid Set: 8 images, 13%; Test Set: 0 images, 0%), 3. Preprocessing, 4. Augmentation, and 5. Create. A 'Continue' button is visible under the Train/Test Split step, and a 'Rebalance' button is at the bottom right of the workflow area.

Image annotation using Roboflow Annotate

Especially with small datasets, preprocessing and augmentation might lead to better results

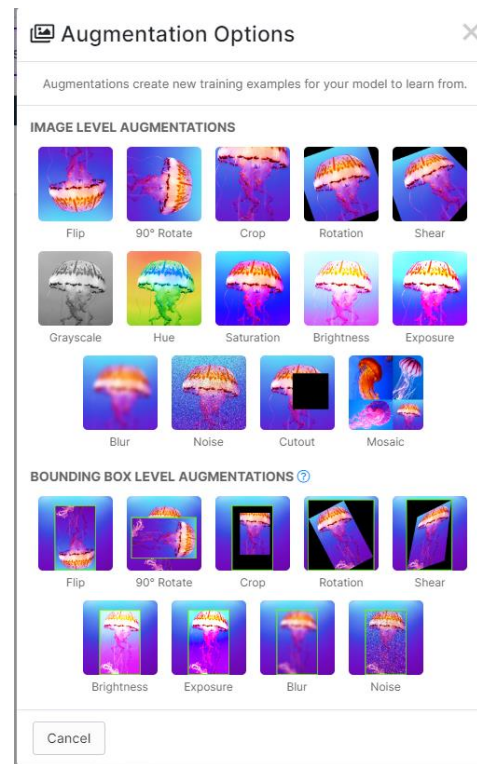
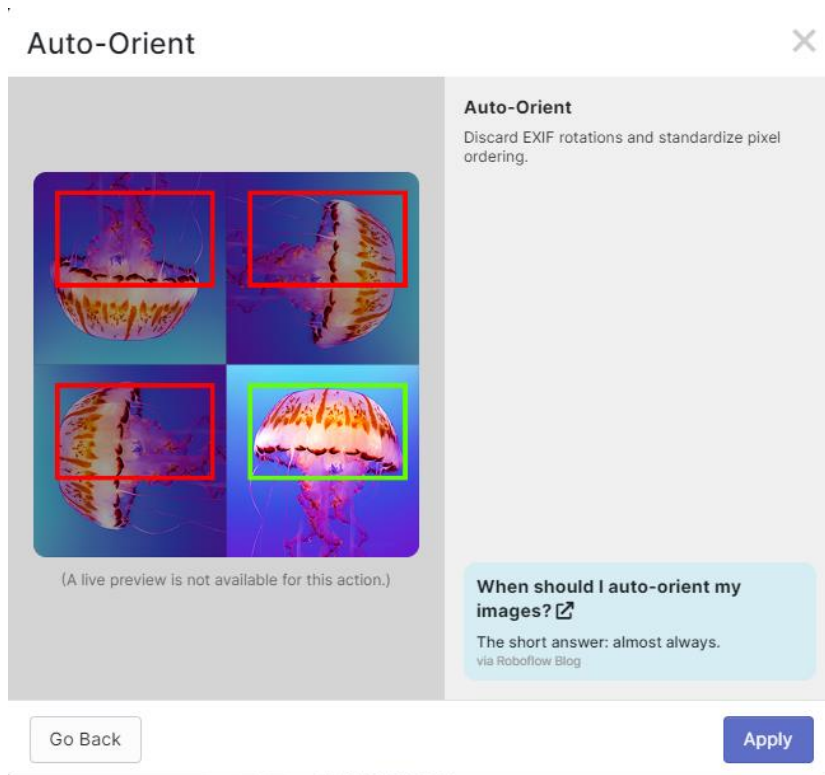
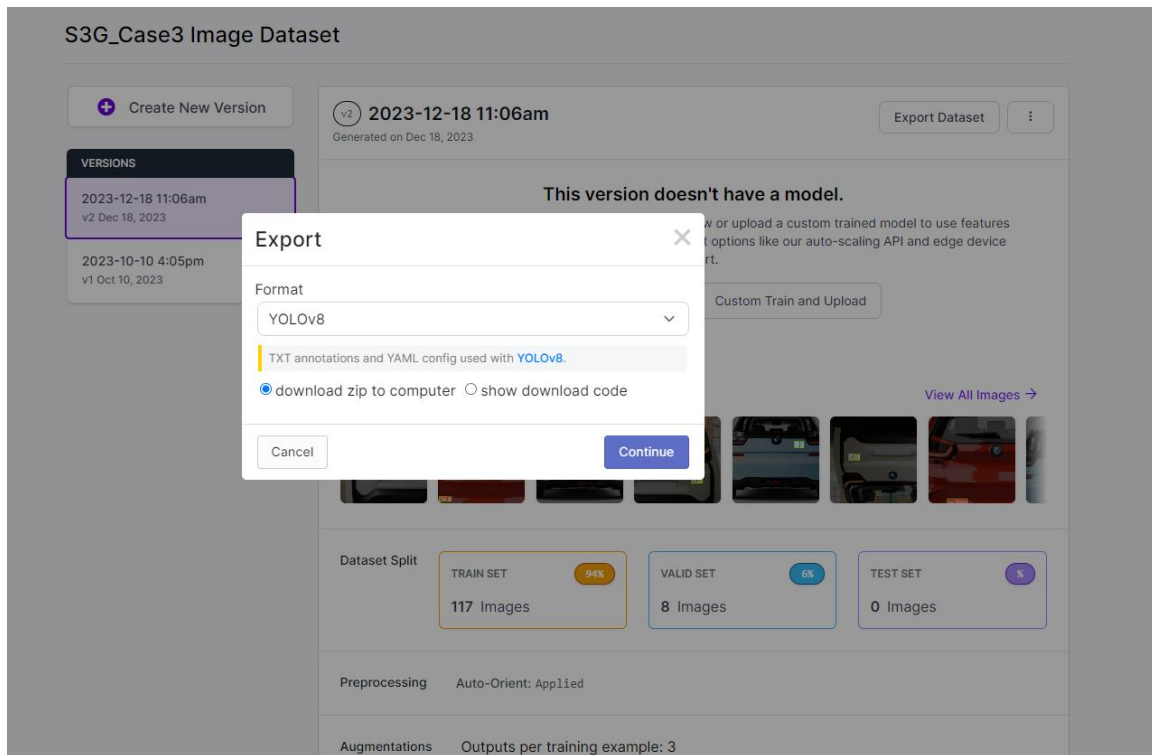


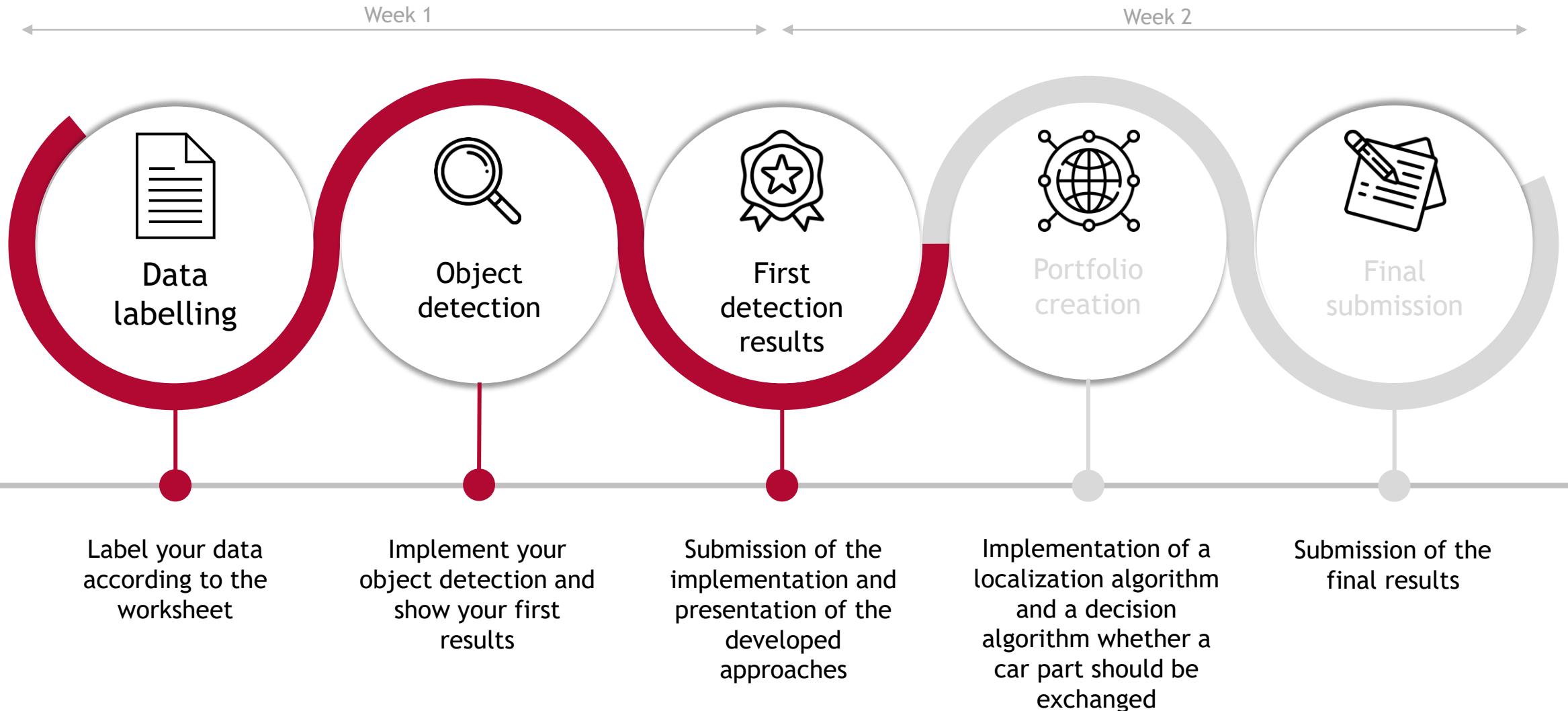
Image annotation using Roboflow Annotate

Export your dataset to the desired format



Name	Status
train	✓
valid	✓
data.yaml	✓

Case 3: Time schedule



Case 3: Submission

The following documents must be emailed to **s3g@fim-rc.de** as one zip folder by **10:00 AM** on **10.06.2024**:

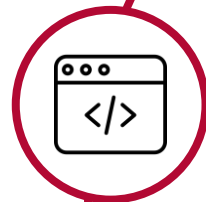
Model

Exported as described in the worksheet



Code

for the object detection incl.
description of additional
packages for the setup as well
as version information of used
packages and libraries



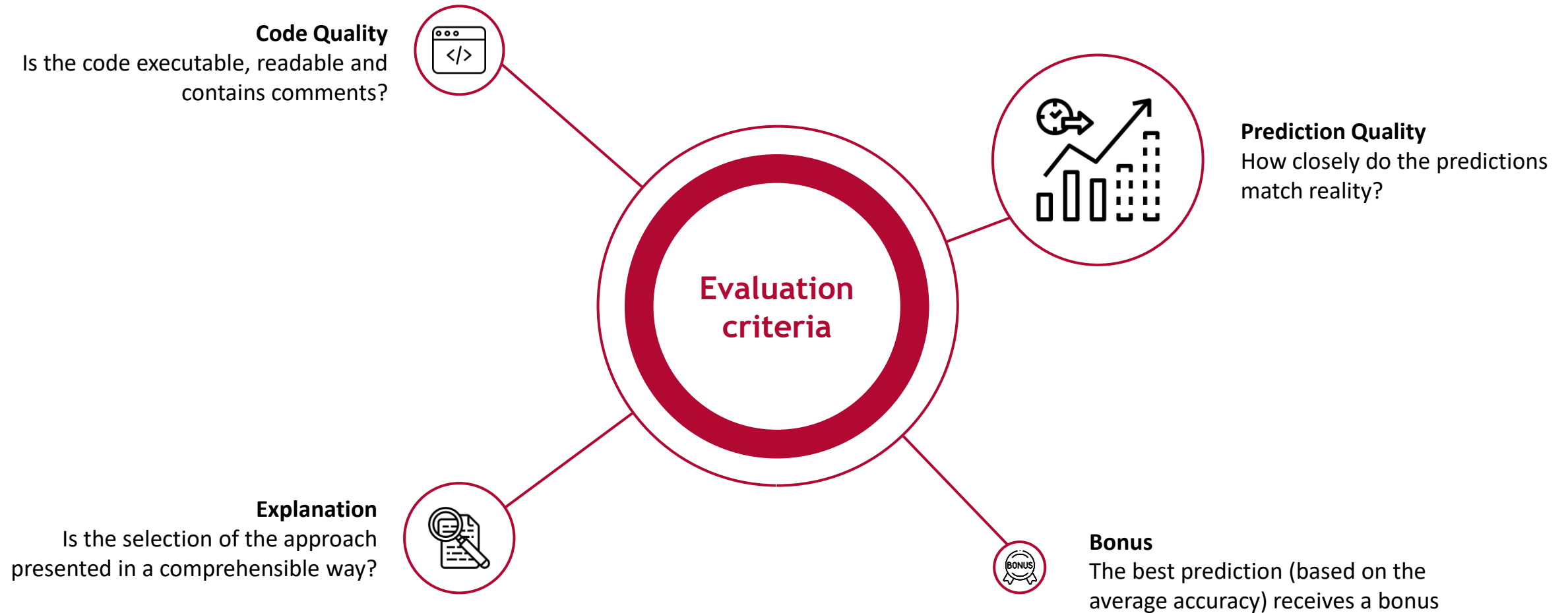
Submission

Presentation

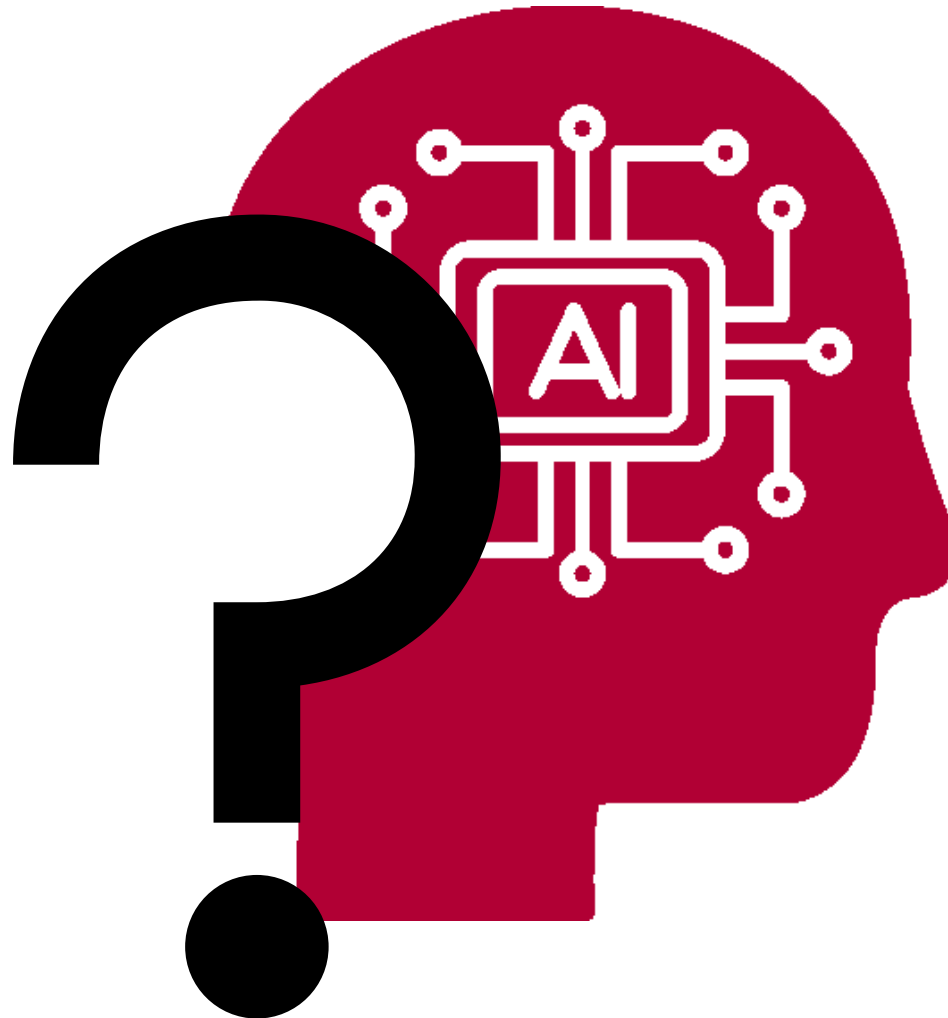
Explaining your approach to
the object detection



Case 3: Evaluation criteria



Case 3: Any Questions?



Any Questions?