

NaO Goalkeeper

Robocup PG by Patrick, Esra, Theo

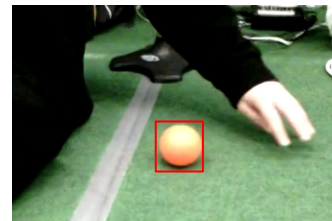
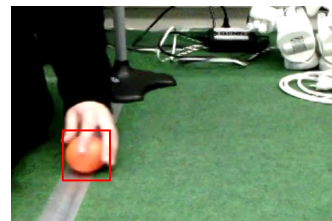
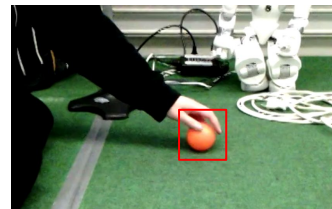
Goalkeeper Challenges

- Fast Object Detection
 - The robot has to detect the red ball in real time
- Trajectory Prediction
 - Predict where the ball is going
- Communication between NAO and CNN
 - Nao is not able to perform the calculations for fast object detection



Fast Object Detection

- Tracking & Prediction in real time requires fast object detect
- Use a small version of a YOLO Detector
- This allows us to make predictions in 2ms
- But: Pretrained YOLO does not work for the red ball



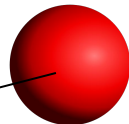
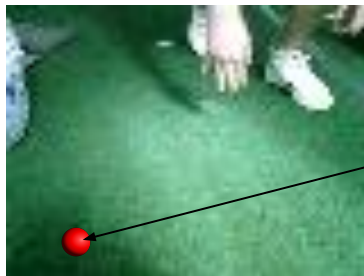
Redball Dataset

- Around 1000 images recorded in the lab
- Annotate 700 real images with ball with our small annotation tool
- Use the remaining 300 images to create synthetic images
- 4000 annotated images in total

Real Image

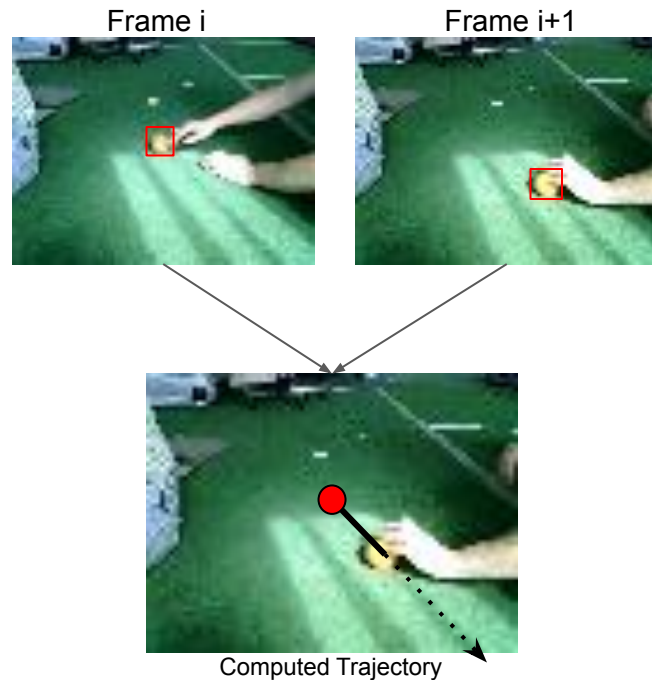


Synthetic Image



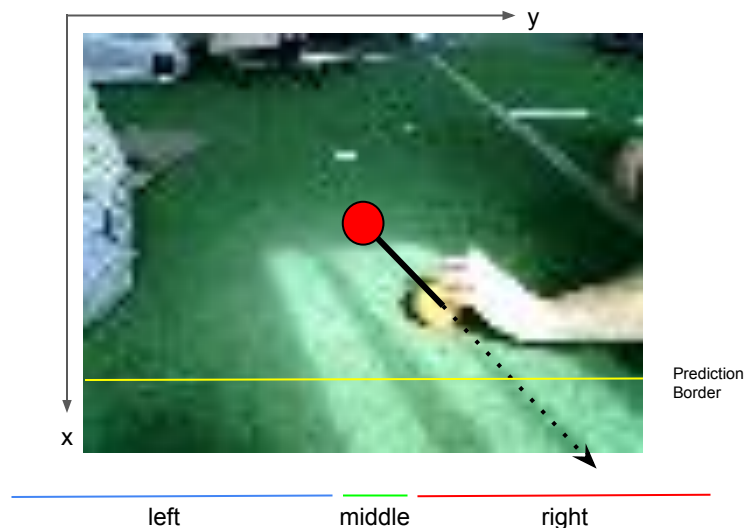
Trajectory Prediction

- Assumption: Once the ball is shot it does not change direction
- Compute ball displacement from consecutive detections
- Use the resulting vector to predict where the ball will leave the image
 - Scale the displacement vector until the x coordinate reaches the image height
 - y coordinate of the scaled vector yields the pixel where the ball leaves the image



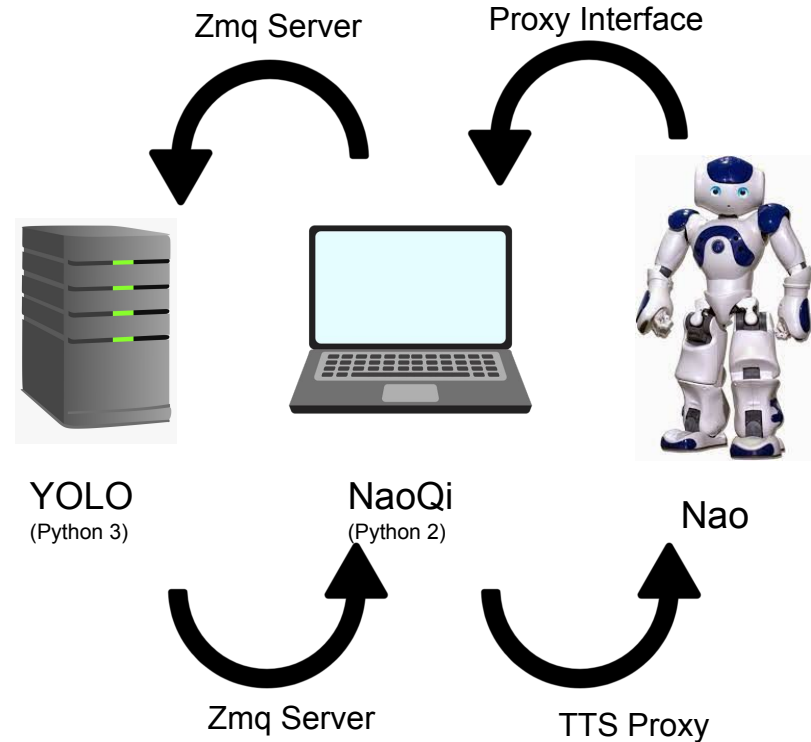
Trajectory Prediction

- Discard movements in the wrong direction
 - Fast balls have larger displacements
 - If the length of the vector is larger than a set threshold or the ball crosses the prediction border
- Output where NAO needs to defend the goal



Communication between NAO and CNN

- Naoqi needs Python2 and YOLO runs with Python3
- Create YOLO Server that receives images and outputs the object center



Implementation

