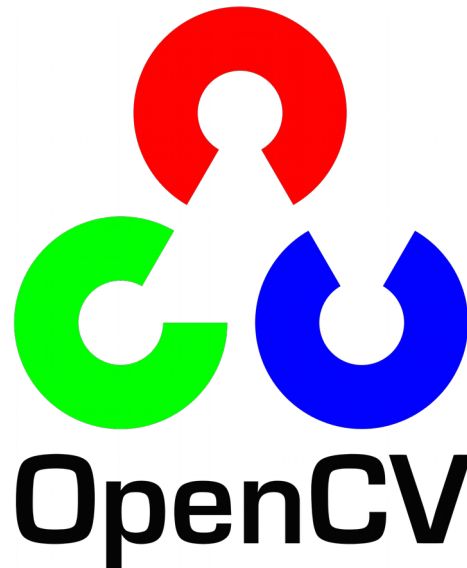# Field Limits Detection

- ROS
- OpenCV
- Nao v5

The Project

In this project, we will attempt to make **Nao** understand where the limits of the field it will be in are, and make it stay behind those limits. We split the work in two nodes. One for image processing and one for moving the Nao.

The field we will work on is at the Intelligent Systems Laboratory of **Technical University of Crete** :
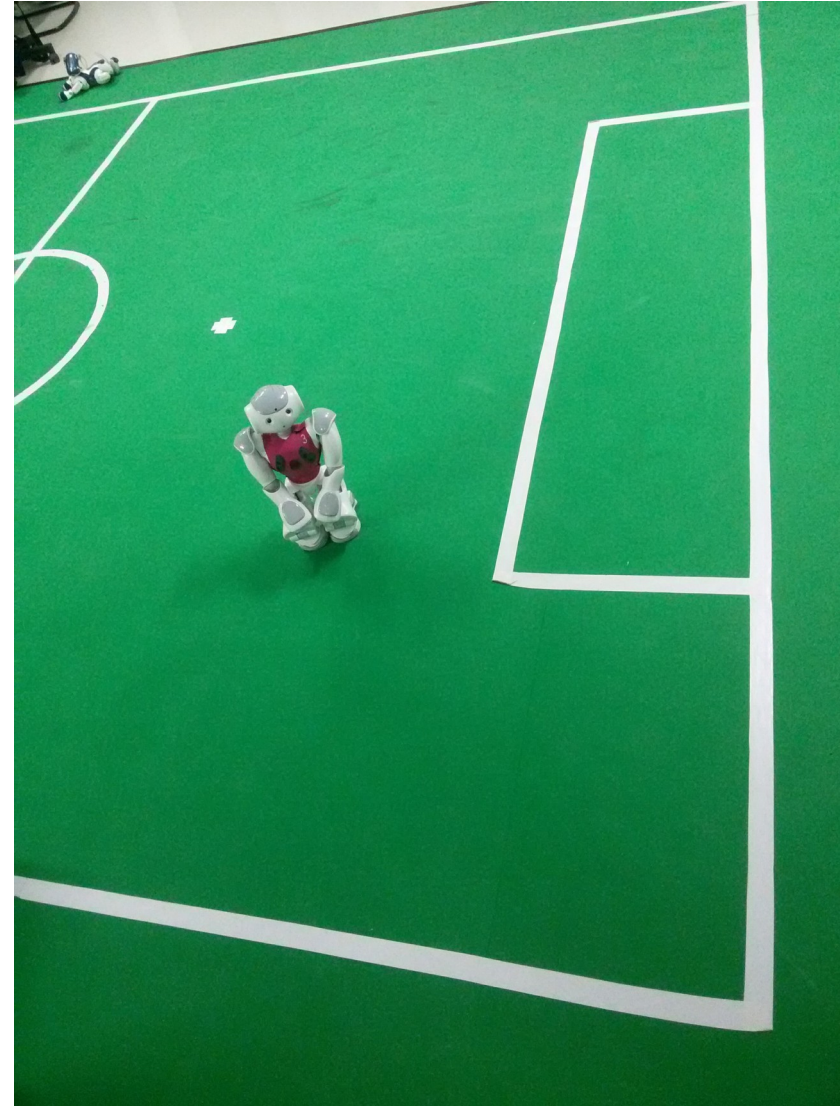
Line finder Node

This node is responsible for finding the lines in front of Nao and after applying a filter to pick only the ones that are unique, it publishes them. Those lines will be used by the Mover Node later. The system implemented is the following :
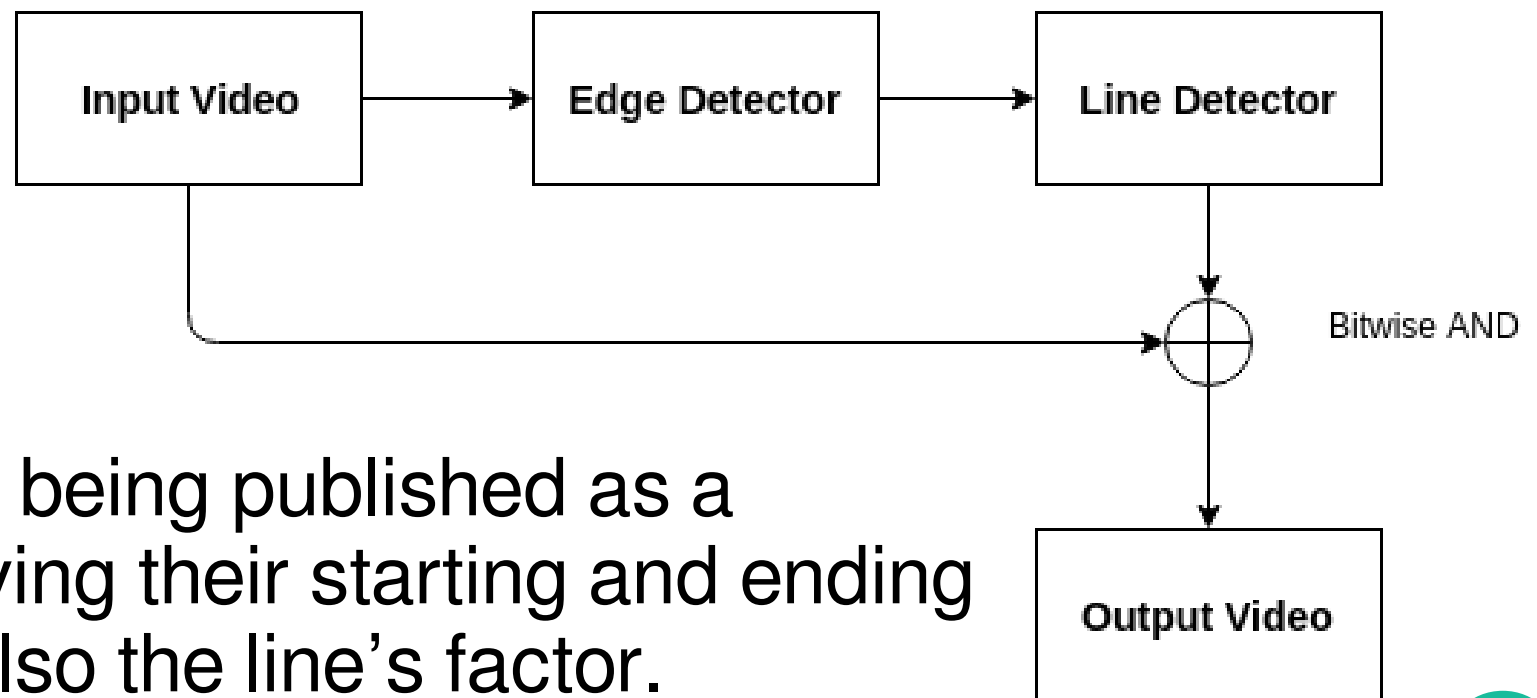
The lines are being published as a message having their starting and ending points, and also the line's factor.

Line finder Node

The stages of the image processing are being presented :
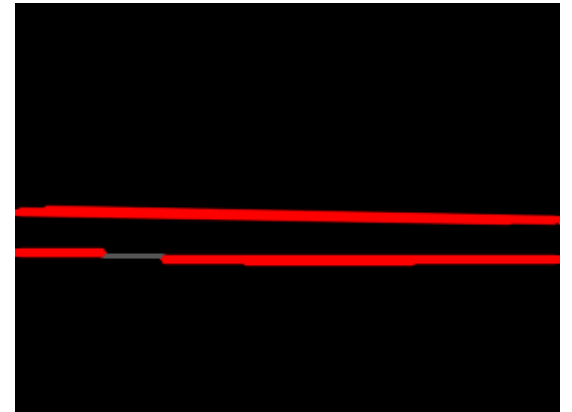
Applying the Edge Detector

Applying the kernel 3x3 Blurring

Applying the Probabilistic Hough Transform with the extra filter for line clarification

In every camera frame, the extra filter we added cleans about the 30% of the lines the probabilistic Hough transform produces.
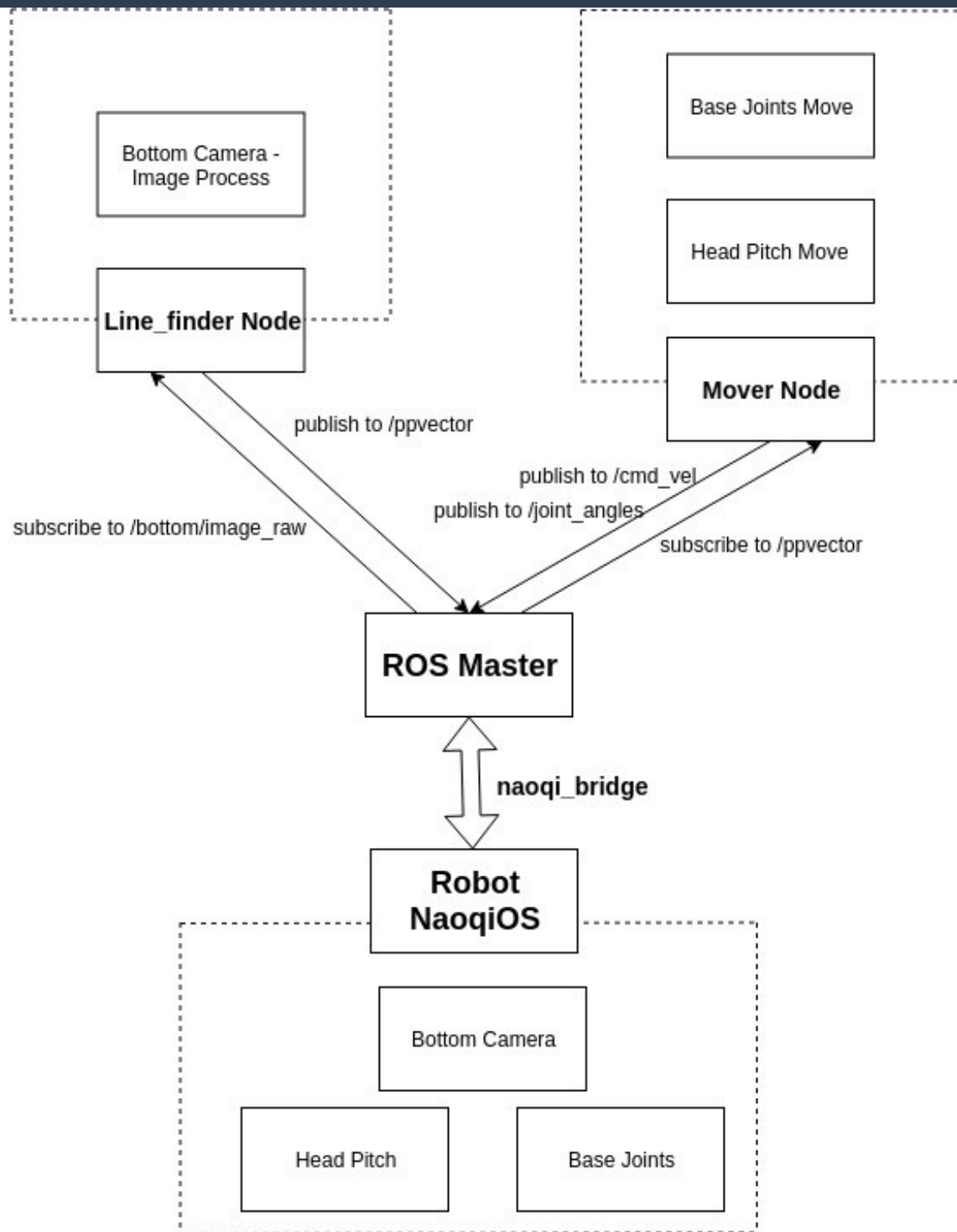
Mover Node

In this node we first subscribe to the topic created by the Line_finder Node. Each line we collect is being checked to see it's position in relation to Nao. If the line has a negative factor and is in front of Nao, then the Nao turns <u>right handedly</u> to avoid going out of limits and then continues moving forward. Accordingly, if the factor of the line is positive.

This way it will stay in the field's limits and keep moving freely.

How ROS and NaoqiOS allow communication :

Bottom Camera - Image Process

Line_finder Node

Base Joints Move

Head Pitch Move

Mover Node

publish to /ppvector

subscribe to /bottom/image_raw

publish to /cmd_vel

publish to /joint_angles

subscribe to /ppvector

ROS Master

naoqi_bridge

Robot NaoqiOS

Bottom Camera

Head Pitch

Base Joints

Before and after roslaunch

After every change we make to our project we make sure to build it in catkin workspace. **CmakeLists.txt** file is for package, nodes, message and service declaration while **package.xml** is for package dependencies.
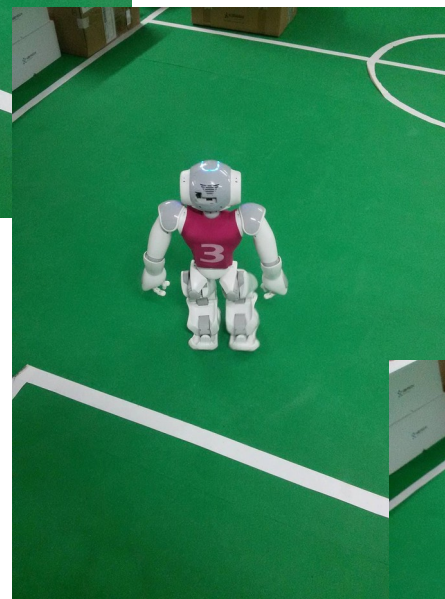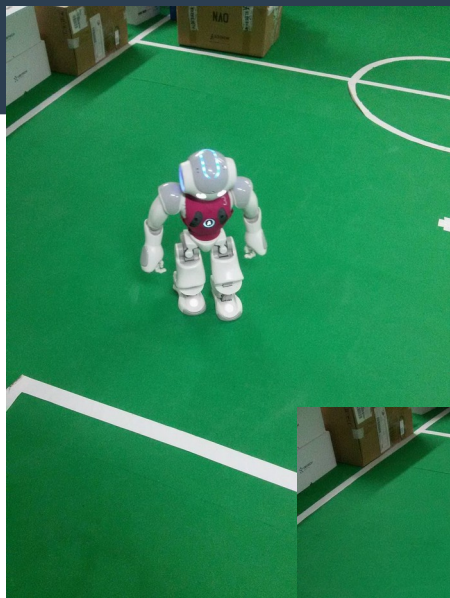
HeadPitch

29.5°   -38.5°
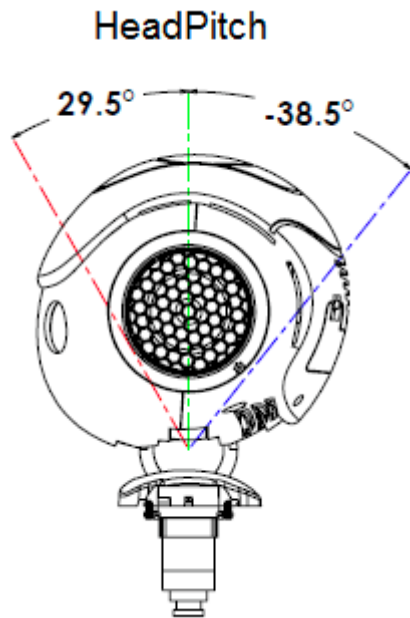
Before we do the roslaunch of our project we make sure Nao will look down by publishing a message to the **HeadPitch**. That message will move its head around 20 degrees front.

Results & Demonstration

1.

2.

3.

4.

5.

6.

Field Limits Detection NAO - ROS

Thank you for your time!

Here is a recent photo of my cat!

Nektarios Sfyris – Autonomous Agents