Visual servoing system integration

2018-03-07

Maarten De Munck, maarten@kapernikov.com

7/03/2018

Table of Contents

1 Context

At Kapernikov, we help our customers to get value out of their data. For our asset intensive clients, we collect, translate, structure, verify and correct data on their assets to enable effective and efficient asset management. For our industrial customers, we use sensor measurements (including images and video) to maximize the yield and minimize the downtime of their production lines.

2 Description

For one of our clients, we developed a visual servoing system for their mobile handling robots. This system recognises a finished product and calculates its 3D position based on images from a camera mounted on the mobile robot. The configuration and performance evaluation of the current prototype is done using text files. To implement the system on multiple robots and integrate it in a production environment, Kapernikov would like to provide a web interface for configuration and monitoring, using an embedded web server on the mobile robot, a cloud based solution or a combination of both.

3 Proposal

During this internship, you will develop a system for configuration, short time (for troubleshooting) and long term (collecting performance statistics) monitoring of the ROS[1]-based visual servoing system.

The current prototype is configured using text (YAML) files. This is the most common approach to initialise the ROS parameter server, although the ROS API provides the functionality required to implement alternative configuration approaches.

Debugging and performance monitoring of the visual servoing system is done by analysing the files written by our custom logger. This includes text (CSV) files containing the ROS messages with sensor measurements and image analysis results, images (JPG) to record the environment and (optionally) video (AVI) files with the complete image stream on the local filesystem. The mobile robot can use a Wi-Fi connection to import and export data, although some areas in the plant are not covered.

Our client explicitly wishes to limit the access to the visual servoing system's configuration and software. Full access (including SSH) is allowed for the engineering department, but the maintenance department should only have access to a limited set of configuration parameters, debugging information and statistics.

You will develop a system to configure the visual servoing system and visualise its state and performance, either using an embedded web server on the mobile robot for the configuration and a reporting extension on the existing plant-wide logging infrastructure or using a (new) cloud-based solution.

At this moment, we don't propose a specific technology or software stack (although, in general, we prefer open protocols and open source software), but we will help you find and choose the most appropriate ones. Of course, you can also count on us to write the low-level interfaces to pass the configuration from an application on the robot to the ROS system and to give you the results you need in a format you can handle.

Although we'll be there for you when you need any help, we think you need at least some experience developing modern web based applications to tackle this challenge.

4 Practical

We propose a six week internship on a full time basis. The location is, as we agree, in Leuven, Brussels or Aalst. Of course, Kapernikov will cover all costs incurred during the project and provide a fair wage in accordance to your (relevant) experience.

[1] Robot Operating System, http://www.ros.org/