Popular sience article
Overhead Localization of Mobile Robots

Mikael Borg
Department of Automatic Control, LTH, Lund University

March 7, 2013

Abstract

The objective of this thesis has been to develop a system that detects a robot and tracks its position with the help of several cameras. A marker is mounted on the robot which enables the camera to detect the robot. When the marker has been detected the tracking of the robot becomes possible.

1 Introduction

Computer Vision is an area that constantly gains more ground on the market and its variety of applications is immense. With the development of Computer Vision the last two decades a new and cheap sensor has been introduced, namely the ccd-camera. Tailor-made software can be made to fit many different working environments, basically the speed of the camera as a sensor lies within the algorithm.

2 Design and Result

Computer Vision is a wide area and there ar numerous algorithms for detections. My goal was to detect a pre known marker and track it, that can be placed on robot equipments. The detection was done with the help of a computer vision library called ARToolkit [2] (*where AR stands for Augmented Reality*). ARToolkit converts the frames into a binary image which is based on a threshold value. The image is then searched for square regions and finds all the squares in the binary images. For each square it finds ARToolkit matches it with a pre attained pattern template, if the square matches the pattern template; A known pattern has been found. My

Figure 1: Tracking marker pattern

goal was to have multiple cameras detecting the same marker and one of them serve as a reference camera, meaning that i would translate the markers coordinates into the reference cameras coordinate system if it went out of sight and one of the other cameras detected the marker. For the calculus to work properly all the three cameras had to be calibrated in order to get the *intrinsic parameters* and the *extrinsic parameters*. Which includes parameters such as *focal length*, *principal point*, *skew*, *pixel error* and an Essential matrix

$$R = \begin{bmatrix} R_{3\times 3} & T_{3\times 1} \\ 0 & 1 \end{bmatrix} \tag{1}$$

where $R_{3\times 3}$, is the rotation matrix that gives the rotation in x, y, z and $T_{3\times 1}$ is the translation vector. A server then process the data that is sent from each camera where the calculus take place. There will always be an area of view for

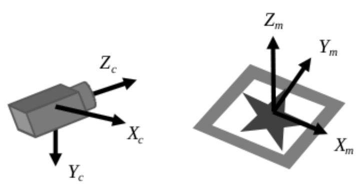

Figure 2: Camera and marker orientation

each camera pair with respect to reference camera that will coincide. This for the calibration between each camera to be possible. By multiplying the rotation matrix and the translation vector from the translating camera to the reference camera one can track over a large distance, given that the marker is detected in one of the cameras. The functionality is demonstrated in a setup shown in Figure 4 [1]. By integrating a NXT brick:developed by Lego group [3], tracking of the robot could be achieved. The communication to the robot is done by bluetooth where my application sends commands to the NXT. A

Figure 3: (*left*) NXT brick CPU [3], (*right*) Omni wheels mounted on the NXT [4]

tracking marker is then placed onto the NXT brick and reference markers are placed on the floor one for each camera. The NXT then moves towards each reference marker for each camera.

Figure 4: Setup for the tracking

References

[1] Mikael Borg , *Overhead localization of mobile robots (2013), ISRN LUTFD2/TFRT--5914--SE*

[2] *ARToolkit, http://www.hitl.washington.edu/ARToolkit/ (2010)*

[3] *Lego Group, http://mindstorms.lego.com/en-us/Default.aspx (2012)*

[4] *Magnus Johansson Bjärenstam, Michael Larsson,* Development of balancing robot with omni wheels (2012), ISRN LUTFD2/TFRT--5897--SE