Hand-Eye Calibration of Depth Cameras based on Planar Surfaces
Andersen, Rasmus Skovgaard; Madsen, Ole; Moeslund, Thomas B.

Publication date:
2014

Document Version
Accepted author manuscript, peer reviewed version

Link to publication from Aalborg University

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

? Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
? You may not further distribute the material or use it for any profit-making activity or commercial gain
? You may freely distribute the URL identifying the publication in the public portal?

Take down policy
If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from vbn.aau.dk on: december 15, 2018
Hand-Eye Calibration of Depth Cameras based on Planar Surfaces

Rasmus S. Andersen	extsuperscript{1}, Ole Madsen	extsuperscript{1}, and Thomas B. Moeshlund	extsuperscript{2}

	extsuperscript{1} Dep. of Mechanical and Manufacturing Systems, Aalborg University, Denmark,  
	extsuperscript{2} Dep. of Architecture, Design and Media Technology, Aalborg University, Denmark  
\{rsa,om\}@m-tech.aau.dk, tbm@create.aau.dk

1 INTRODUCTION

For robots to be able to perform advanced tasks, it is a necessary to use various sensors. This is the case both for industrial tasks (pick-and-place operations, bin-picking), for home service robots (identification of humans, navigation), and for military robots (local/global navigation, obstacle identification). There are three typical ways to mount sensors relative to the robot:

1. Mounted on the robot in a fixed or movable position (pan/tilt).
2. Mounted in the environment in a fixed or movable position.
3. Mounted on the end-effector of the robot arm, that is supposed to interact with the environment.

The calibration between the sensor and the robot is essential for all of these mountings. In this work we focus on calibrating a depth camera to the end-effector; also known as hand-eye calibration. Hand-eye calibration is necessary for all sensors mounted on an end-effector. The most popular sensor type to mount on end-effectors is visible light cameras, and calibration of these have therefore been investigated thoroughly. Depth cameras is another popular choice, which have also been used on robots for several decades. Especially since the launch of Microsoft’s Kinect in 2010, their popularity have increaead [El-laithy et al., 2012]. The depth sensor in the Kinect works by projecting infrared structured light onto the scene. The depth is measured by capturing the known projected pattern, and based on this compute the depth. Other technologies for capturing depth images include Time-of-Flight (ToF) [Fuchs, 2012] and stereo vision.

2 Methods for Hand-Eye Calibration of Depth Cameras

The problem is illustrated in Figure 1. The unknown transformation is the one between tool and camera, while the transformation between the base and tool is assumed to be known.

Several approaches to depth camera hand-eye calibration exist. One popular approach is the TurtleBot calibration algorithm, which is available through ROS. This works specifically for Kinect-like cameras by first pose estimating
a calibration board using the RGB/D sensors, and afterwards localizing it in
the base frame by moving the robot tool to the four corners on the board. The
problems with this approach include that it relies both on the imperfect internal
RGB-D calibration of the Kinect and of the model of the tool.

Use of the tool can in some cases be avoided when using a calibration board
[Tsai and Lenz, 1989, Hvilshøj et al., 2010]. For the Kinect, a transformation be-
tween the depth camera and a visible light camera is known beforehand. Thus,
the visible light camera can be calibrated first and used to indirectly calibrate
the depth camera. This approach of course relies on a transformation, which is
not perfectly known. Another approach that can be used for a Kinect is to di-
rectly calibrate the internal infrared camera using an infrared light source and a
calibration board. A problem with this is that the internal depth computations
of the Kinect are circumvented. Also, it only works for depth cameras that is
based on an infrared camera.

A few methods focus specifically on depth cameras. In [Pomerleau et al., 2011]
the motion of the depth camera is continuously logged based on ICP. The hand-
eye transformation can then be computed by comparing to the movement of the
end-effector. A problem with this approach is the matches do eventually drift,
causing the calibration to be less precise. In [Kahn et al., 2014] it is instead sug-
gested to design a 3D shape to be optimal for 3D pose estimation from a point
cloud, and use this to find the camera pose. The pose of the object relative
to some world frame (such as the robot’s base frame) must however be known
beforehand. This is a severe limitation for general purpose hand-eye calibration.

2.1 Suggested Approach

The approach that we suggest here is to do calibration of depth cameras using
only the point clouds as in [Kahn et al., 2014], but to estimate equations for a
simple planar surface instead of carefully designed 3D shapes. The advantage is
that planes can be found in point cloud very fast and reliably using standard techniques such as RANSAC. Also, a sufficiently planar surface is nearby in most locations. A plane is estimated for the same surface from multiple positions, and using these, the tool-camera transformation can be found by minimizing an overdetermined system of non-linear equations. A total of 10 parameters are estimated; 6 for the tool-camera transformation and 4 for the plane. The idea is illustrated in Figure 1.

3 PRELIMINARY RESULTS AND CONCLUSIONS

An initial implementation has been tested using 12 measurement points. In this it is assumed, that the plane normal is known to be vertical, and thus only 7 degrees of freedom is estimated. For each point, an equation for the plane is determined and logged along with the position of the robot end-effector. The learning process was stopped when \( \Delta \theta \) got below a predefined threshold, which happened after 143 iterations.

Our next step is to develop a full implementation and to evaluate its performance both with regards to precision and speed. We intend to publish the implementation as a publicly available ROS package.

ACKNOWLEDGEMENTS

This research was partially funded by the European Union under the Seventh Framework Programme project 260026 TAPAS.

References


