

# Practical course robotics project proposal

Marc Tuscher, Ralf Gulde

May 13, 2019

## 1 Introduction

Recent advancements in robotics and automated systems have led to the expansion of autonomous capabilities and more intelligent machines being utilised in ever more varied applications [LS05, JYL<sup>+</sup>14]. While manipulating objects is relatively easy for humans, reliable grasping arbitrary objects remains an open challenge for general-purpose robots. Resolving it would advance the application of robotics to industrial use cases, such as part assembly, binning, and sorting. The capability of adapting to changing environments is a necessary skill for task generalised robots [KKGB12, KP10]. A grasp describes how a robotic end-effector can be arranged to securely grab an object and successfully lift. Grasp planning relates to the path planning process that is required to securely grasp the object and maintain the closed gripper contacts to hold and lift the object from its resting surface [BK00]. Initially, deriving the grasp-points of a perceived object is an essential aspects. Consequently, accurate and diverse detection of robotic grasp candidates for target objects should lead to a better grasp path planning and improve the overall performance of grasp-based manipulation tasks.

In this work we propose a deep learning strategy to detect grasp points from sensor data and plan collision free robot paths to manipulate arbitrary objects. In the past decade, deep learning has achieved major success on detection, classification, and regression tasks [BRF13, KSH12]. Its key strength is the ability to leverage large quantities of labelled and unlabelled data to learn powerful representations without hand-engineering the feature space. To conclude this work we experimentally validate our approach using a humanoid robotic system and a structured light RGB-D seonsor.

This project proposal is organized as follows: In section 2 the goal of the work is described. Further, in section 3 a formulation and the proposed mehtods are described. Section 4 thematizes the milestones of implementing the methods in a robotic demonstrator system to evaluate the proposed approach.

## 2 Goal

Robot grasping can be subdivided into two related problems: perception and planning. A typical example of this approach is the ROS grasp pipeline [CJCH12]. Herein, a CAD model is mapped to a point cloud of the object. Optimal grasp poses are then derived from the CAD model. This works well in ideal scenarios, however, in most realistic applications the scenario is far from ideal. Our main goal is to set up a framework which extracts robust grasp poses from RGBD-Images and grasps an object lying on a table.

## 3 Problems and methods

This section gives an overview over the general problems of the objective and the methods we will use to overcome these difficulties.

### 3.1 Localization of objects

needs to be infered if more than one object is in the scene. We will use a SSD for that.

	<b>Marc</b>	<b>Ralf</b>
16.05.19	Basic setup, process RGBD images from Baxter/PR2 with GQCNN	Ralfas asdasdasda sdas dasd asdasd as dasd
23.05.19	Grasp object with default grasping policy	
06.06.19	Tune control, recalculate online if object moves	
13.06.19	Experiment with different grasping policies	

### 3.2 Localization of robust grasp points

Sampling grasp points from RGBD-Images is a major problem and has been studied by a wide variety of approaches.

### 3.3 Grasp planning

Reason about how to move the manipulator into the desired grasp pose. The planning component highly relies on the RAI framework for robot manipulation.

- Problems:
  - Localization of objects.
  - Localization of robust grasp points.
- Methods:
  - Robot control using *rai* framework
  - Sampling of grasp candidates using a grasping policy (e.g. *CrossEntropyRobustGrasping-Policy*<sup>1</sup>)
  - evaluation of grasp candidates with pretrained *GQCNN*<sup>2</sup>
  - localization of objects using bounding boxes (if necessary)

## 4 Milestones

TODO

### Requirements

- Baxter
- PR2
- Optional: machine with GPU for further training. We can also bring our own desktop pc.
- Details about camera intrinsics
- RAI framework running on machines with nvidia graphics card (but this problem might be related to the newest nvidia-driver.)

<sup>1</sup><https://github.com/BerkeleyAutomation/gqcnn/blob/a0930e9d2fef3c930c41dd91cde902d261348fbe/gqcnn/grasping/policy/policy.py#L627>

<sup>2</sup><https://github.com/BerkeleyAutomation/gqcnn>

## References

- [BK00] A. Bicchi and V. Kumar. Robotic grasping and contact: a review. 1:348–353 vol.1, April 2000.
- [BRF13] Liefeng Bo, Xiaofeng Ren, and Dieter Fox. Unsupervised feature learning for rgb-d based object recognition. pages 387–402, 2013.
- [CJCH12] Sachin Chitta, E. Gil Jones, Matei Ciocarlie, and Kaijen Hsiao. Perception, planning, and execution for mobile manipulation in unstructured environments. *IEEE Robotics and Automation Magazine, Special Issue on Mobile Manipulation*, 19(2):58–71, 2012.
- [JYL<sup>+</sup>14] Zhangfeng Ju, Chenguang Yang, Zhijun Li, Long Cheng, and Hongbin Ma. Teleoperation of humanoid baxter robot using haptic feedback. pages 1–6, 2014.
- [KKGB12] George Konidaris, Scott Kuindersma, Roderic Grupen, and Andrew Barto. Robot learning from demonstration by constructing skill trees. *The International Journal of Robotics Research*, 31(3):360–375, 2012.
- [KP10] J. Kober and J. Peters. Imitation and reinforcement learning. *IEEE Robotics Automation Magazine*, 17(2):55–62, June 2010.
- [KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [LS05] M. Lopes and J. Santos-Victor. Visual learning by imitation with motor representations. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 35(3):438–449, June 2005.