# Some Placeholder Title

**A Master Thesis**

written by

**Victor Melbye Staven**
vista17@student.sdu.dk

The code for this project is available at
`https://github.com/vmstavens/in_hand_pose_estimation`

**University of Southern Denmark**
The Technical Faculty

Word Count : 932
September 21, 2022

## Abstract

Some abstract text explaining the goal, methods and conclusion of the project.

# Contents

Acknowledgements

My acknowledgements

## Acronyms

**CP** correspondence problem.
**CV** computer vision.
**DL** deep learning.
**DOF** degrees of freedom.
**PE** pose estimation.
**ROS** Robot Operating System.
**SOTA** state of the art.

## Terms

**computer vision (CV)** is a scientific field which deals with how computers can gain a high-level understanding from digital inputs in the form of images or videos.

**correspondence problem (CP)** the problem within computer vision (CV) to localize the projection of the same object in 3D space in 2 or more images.

**deep learning (DL)** is a technique for data driven pattern recognition.

**docker container** , a docker container is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another[1].

**pose estimation (PE)** is a scientific picipline which deals with how computers can gain a high-level understanding from digital inputs in the form of images or videos.

**Robot Operating System (ROS)** some description.

Chapter 1

# Introduction

The developments in robotics as a field has over the past years provided automation solutions to execute repetitive manual tasks with high efficiency and reliability \cite. One of the most common tasks being pick and place tasks which involves picking un an object from one position and placing is in another. This is can be parted into the following subparts: Object localization, pose estimation, grasping and placing. In the solutions currently present for industrial use computer vision (CV) is used for object localization and pose estimation (PE) due to the low cost of cameras and the fields maturity. However, while these solutions may be sufficient for certain tasks they fundamentally suffer from the weaknesses introduced by vision techniques. These include a great number of outliers caused by occlusions, reflecting, transparent or homogeneous surfaces, and repetitive structures when solving the correspondence problem (CP). These problems as of the writing of this project have jet to be completely solved. Promising results have been found with the rise of deep learning (DL) which in present time has proven its versatility and provides proof of concept solutions for narrow cases in pose estimation of transparent [2] and reflective objects [3]. This is relevant since industrial settings often contain transparent and especially reflecting objects as metallic parts tend to appear frequently and have high reflectances. To solve these problems this project aims to perform in-hand pose estimation through only the use of tactile sensors. Specifically this will be done on a Shadow Dexterous Hand [4] with 20 degrees of freedom (DOF). Using tactile inputs rather than visual, eliminates the weaknesses mentioned above. A schematic showing the hand can be seen in Fig. 1. Using this approach, the overall problem can be partitioned into 3 sub-problems labeled problem 1, 2 and 3. Problem 1 involves modeling the contact between the gripper's fingers and the object, also referred to as tactile perception. Problem 2 is to convert the collected data from problem 1 to meaningful surface data, treat these data as features and use them to estimate pose candidates. Finally problem 3 involves in-hand manipulation, such that further information is gained by probing the object. Here new desired surface points are found through intelligent probing such that strong surface features are found to better identify the object's correct pose.

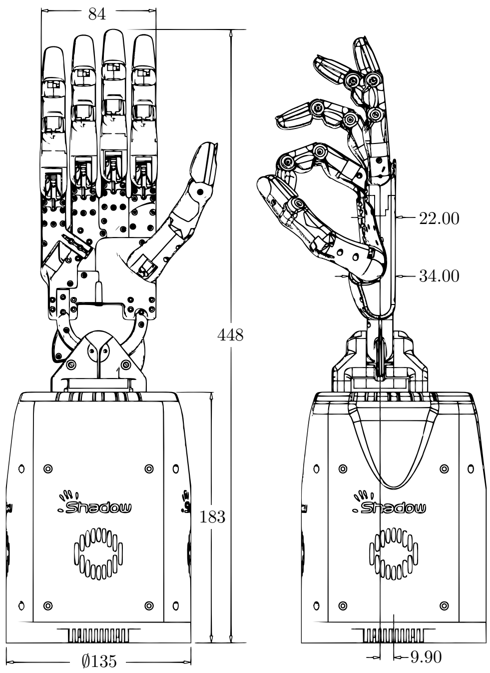


**Fig. 1:** Schematic of Shadow Dexterous Hand from Shadow Robots, based on [5]. The measurements are in mm.

Thus the hypothesis of this projects $H_1$, will be testing if intelligent probing for strong features increases in-hand pose estimation performance, with the null hypothesis $H_0$ being that there is no statistical significant difference in the pose estimation performance of the system if the probing is done randomly or intelligently at a certainty level of 95%. Here pose estimation performance is quantified in terms of mean execution time for estimating the pose with an accuracy greater than 95%.

The development of this project is done in the docker container provided by Shadow Robotics for simulation, control and development of the hand [6]. Here a hardware-simulation agnostic Robot Operating System (ROS) [7] control [8] interface is found, which contains fundamental tools to interact with the robot hand. The dynamic simulation environment Gazebo [9] is likewise packaged as part of the docker container and is thus the one used for this project. To solve the problems presented, the ROS packages in Table 1 will be applied, where `ros_utils` and `in_hand_pose_estimation` will be developed during this project.

| Package | Description |
|---|---|
| `in_hand_pose_estimation` meta | **Project package of the in-hand pose estimation system** |
| `in_hand_pose_estimation` | Integration of the full in-hand pose estimation pipeline |
| `sr_tactile_image` pkg | Extraction of tactile perception |
| `sr_pose_estiamtion` pkg | Estimate the pose of object based on tactile perception |
| `sr_hand_manipulation` pkg | Manipulate object in hand to probe for strong features |
| `sr_common` meta | **Shadow package for commonly used tools** |
| `sr_common` | Implements commonly used tools such as messages |
| `sr_robot_msgs` pkg | Messages used to communicate with the robot hand |
| ... | |
| `sr_core` meta | **Shadow package for core tools** |
| `sr_core` | Implements core features of the hand such as hardware interfacing |
| `sr_hand` pkg | Contains the hand commander for controlling the robot hand |
| ... | |
| `ros_utils` pkg | **Utilities for interfacing ROS/Gazebo/MoveIt/Eigen etc** |

**Table 1:** Software packages used in the in hand pose estimation system.

To present this work, the state of the art (SOTA) solutions to each of the three problems described above will be presented in Chapter 2, where the best fitting methods for this use case will be chosen. In Chapter 3 to Chapter 5 these solutions will be presented, analyzed, and their performance discussed and concluded upon. In Chapter 6 the system integration will be presented and the total performance of the system will be concluded. Finally in Chapter 7 and Chapter 8 the results and methods will be discussed with potential improvement for future iterations and the project til be concluded.

Chapter 2

# State of the Art

## 2.1 Problem 1 - Tactile Perception

What is tactile perception? Why is it relevant?
How is a tactile sensor constructed [10] what different types exist and which one is present in the model provided.
*"Representations of tactile data are commonly either inspired by machine vision feature descriptors"*
often used in computer vision context, where each tactile image
Addressing the problem

## 2.2 Problem 2 - Pose Estimation

## 2.3 Problem 3 - In-Hand Manipulation

Chapter 3

# Tactile Perception

## 3.1 Introduction

Here we write the introduction for problem 1.

## 3.2 Related Work

Here we cite the related work by `\cite{source-label}` like this [10]

Chapter 4

# Pose Estimation

## 4.1 Introduction

Here we write the introduction for problem 2.

## 4.2 Related Work

Here we cite the related work by `\cite{source-label}` like this [10]

Chapter 5

# In-Hand Manipulation

## 5.1 Introduction

Here we write the introduction for problem 3.

## 5.2 Related Work

Here we cite the related work by `\cite{source-label}` like this [10]

Chapter 6

# System Integration

## 6.1 Introduction

Here we write the introduction for the system integration.

Chapter 7

# Discussion

Chapter 8

# Conclusion

# Bibliography


[1] docker. *What is a container*. URL: https://www.docker.com/resources/what-container/.

[2] Xu, Chi et al. "6DoF Pose Estimation of Transparent Object from a Single RGB-D Image". In: *Sensors* 20.23 (2020). ISSN: 1424-8220. DOI: 10.3390/s20236790. URL: https://www.mdpi.com/1424-8220/20/23/6790.

[3] He, Zaixing et al. "6D Pose Estimation of Objects: Recent Technologies and Challenges". In: *Applied Sciences* 11.1 (2021). ISSN: 2076-3417. DOI: 10.3390/app11010228. URL: https://www.mdpi.com/2076-3417/11/1/228.

[4] Straszheim, Troy et al. *Shadow Robot*. URL: https://www.shadowrobot.com/dexterous-hand-series/.

[5] Components, ROS. *The Shadow Dexterous Hand*. URL: https://www.roscomponents.com/en/robotic-hands/117-shadow-dexterous-robotic-hand.html.

[6] Robot, Shadow. *The Shadow Robot Company*. URL: https://github.com/shadow-robot.

[7] Quigley, Morgan et al. "ROS: an open-source Robot Operating System". In: *ICRA Workshop on Open Source Software* 3 (Jan. 2009).

[8] Chitta, Sachin et al. "ros_control: A generic and simple control framework for ROS". In: *The Journal of Open Source Software* (2017). DOI: 10.21105/joss.00456. URL: http://www.theoj.org/joss-papers/joss.00456/10.21105.joss.00456.pdf.

[9] Koenig, N. and Howard, A. "Design and use paradigms for Gazebo, an open-source multi-robot simulator". In: 3 (2004), 2149–2154 vol.3. DOI: 10.1109/IROS.2004.1389727.

[10] Chi, Cheng et al. "Recent Progress in Technologies for Tactile Sensors". In: *Sensors* 18.4 (2018). ISSN: 1424-8220. DOI: 10.3390/s18040948. URL: https://www.mdpi.com/1424-8220/18/4/948.

Appendix A

# Appendix A Title