

## Group Project: Image Guided Robotic Needle Placement

Medical needles are widely used for minimally invasive procedures ranging from biopsies to drug insertion. Compared to manual needle placement, robots offer the potential to make the procedures more precise and safer. Using 3D imaging techniques and trajectory planning the needle can be driven to the desired target while keeping a safe distance to bones and other delicate structures.

In this project, you will develop an image guided robotic needle placement system using the Robot Operating System (ROS) framework. You will use a depth camera (Kinect Azure by Microsoft) mounted on a robot arm (Panda by Franka Emika) to record 3D images of a chest phantom. You will need to find the transformation between the robot's endeffector and the camera with an eye-in-hand calibration. Using this transformation you will be able to stitch the individual images to a combined scan while the robot drives the camera around the phantom. You will then register this scan to a high resolution model, obtained from computer tomography. Within the high resolution model a target for the needle will be given. You will need to perform trajectory planning to find a collision free and kinematically feasible path to the target. Lastly, you will exchange the camera for a needle (mock-up) and let the robot perform the insertion.

**Project Groups:** The project work will be conducted in groups of eight students. We will form the groups based on the information provided in the registration form. To be assigned to a project group you need to submit the registration form either via e-mail to stefan.gerlach@tuhh.de by the 11th of April.

**Lab Access:** You will start developing your solution with the help of a simulator. Towards the end of the semester each group will be assigned one weekly time slot to work in the robot lab. The lab will be available to you during your lab times as well as by appointment. In your first lab session you will receive safety instructions for working in our lab. Therefore, all group members must be present and on time for their time slot.

**Requirements:** The following requirements are **mandatory** and need to be fulfilled in order to successfully complete this project:

- Attend to the first lecture on **1st of April**. If you cannot make it, write an email to stefan.gerlach@tuhh.de beforehand.
- Submit the registration form for the project by the **6th of April**.
- Submit a project plan including a sketch of the necessary interfaces between the components of your ROS application, tasks assigned to each group member and a weekly time plan by the **25th of April**.
- Submit **weekly reports** via e-mail to stefan.gerlach@tuhh.de starting from the week of 2nd of May. These reports should include the contribution of each team member, the progress achieved in each part of the project as well as any problems you may have encountered. Team members not mentioned in more than three of the weekly reports will be considered inactive and excluded from the project.
- Present your needle insertion system between the **5th and 8th of July** during your last lab session.

- Describe your results and choices in a scientific paper consisting of approximately 5 pages due by the **15th of August**. State and explain every assumption made. Provide sources for all formulas you use.

**Subgoals:** In order to develop the overall system on time, we strongly recommend you to meet the deadlines for the following subgoals:

- Working forward kinematics of the Panda: **2nd of April**
- Working inverse kinematics on the Panda: **6th of June**
- Working camera and hand-eye calibration: **13th of June**
- Model recording and registration: **13th of June**
- Planning of feasible needle paths: **20th of June**

**Consultation Hours:** You may consult the teaching assistants during lab times or by appointment.

**Recommendations:** Remember to test scripts concerning the robot in the simulation environment available via Stud.IP before running them in the lab.

Consider the following work packages as suggestions on how to split the work within your group. You can always find alternative solutions as long as you keep the above requirements. All solutions should be implemented as ROS nodes, so that they can be easily integrated in the final application.

#### **Work package 1:** Robot kinematics

In the final application you will need to move the robot

- for recording of poses for the hand-eye calibration
- for scanning the chest phantom from different sides
- for the needle insertion

The robot works in the joint space, however, you will need to move in Cartesian space at least for the linear motion during needle insertion. Therefore, it is important to convert the joint positions to a robot pose in Cartesian space and vice versa. Solve and implement the solutions to the Direct and Inverse Kinematics. For testing and verification of your implementation use the simulator. You can get the current joint positions from the `/joint_states` topic. You can get the current cartesian position of the end effector using the transforms published by the `robot_state_publisher` on the topic `/tf`. From the console, you can look it up with `roslaunch tf_echo panda_link0 panda_link8`. Also read up on ROS `tf`.

#### **Work package 2:** Trajectory planning

In order to move the robot to a desired target you need to send the robot a continuous trajectory of joint positions. When the robot should perform a movement in cartesian space, you need to call the inverse kinematics to get the associated joint positions. You should plan the full trajectory first, before starting to send the robot commands. You can find an example on how to send continuous joint commands to the robot in the file `~/catkin_ws/src/franka_ros/franka_example_controllers/src/test_move.cpp`. For the actual needle placement the trajectories of the robot and the needle need to keep a safe distance to bones. You can test the trajectory planning in the simulator.

**Work package 3:** Camera calibration

The Kinect Azure has an RGB and a depth camera. You need to calibrate the Kinect so that it captures the true geometries of the world. This involves the intrinsic calibrations of both the RGB and the depth camera, as well as finding the transformation between the RGB and the depth camera frame, called extrinsic calibration. We will provide you a set of images capturing a checkerboard calibration object, from which you can calculate the calibration. We will also provide a reference calibration for comparison.

**Work package 4:** Eye-in-hand calibration

In order to stitch the pointclouds of the chest phantom, you need to know the transformation between the mounted camera ("eye") and the robot's end effector ("hand"). For this purpose you will need to move the robot to different poses and estimate the pose of the checkerboard in the images. Then, you will need to implement an algorithm to solve the hand-eye calibration with the set of robot and checkerboard poses as an input. We will provide you a set of robot poses, corresponding images of the camera capturing a checkerboard object, and the true eye-in-hand transformation as a reference and to test your implementation without needing the real robot. For the final application, you will need to implement the sampling of robot poses and the actual robot movement. However, you can test this in the simulator.

**Task 5:** Model recording and registration

In order to obtain a 3D Scan of the chest phantom you will need to drive the camera around the phantom. Based on the known robot poses and the transformation from end effector to camera you can then stitch the individual pointclouds to one combined scan. Afterwards, you can register it to a high resolution model of the phantom which you will obtain from us in the form of a CAD file. For the start of the development we will provide you a set of 3D pointclouds and the robot poses at which these pointclouds were captured, so you will be able to stitch the pointclouds. Subsequently, you will be able to perform the registration to the CAD model. Lastly, you will need to select the target for the needle.

**Bonus Points:** You may receive up to 10 bonus points that will be added to the points you obtain in the written examination for

- active participation during lab sessions,
- regularity of weekly reports,
- quality of your final presentation and the subsequent discussion,
- quality of submitted software (reasonable structure, sufficient commenting, performance during final presentation),
- plausibility of chosen algorithms and quality of their implementation/incorporation into your work,
- your ability to critically evaluate results,
- creativity and implementation of additional features,
- content and completeness of the final report including list detailing contributions of each author,
- and quality of writing (structure, appropriate use of figures and tables, no grammar or spelling errors).