Worcester Polytechnic Institute

Multi-View Patient Perception and Body Anatomy Reconstruction

Interim Project Report

Fenil Desai
Pratik Surendra Kumbhare
Sairam Venkataramani
Benjamin Verdesi

Robot Dynamics - RBE 501

Instructor:
Professor Loris Fichera

Lab Advisors:
Professor Haichong Zhang
Xihan Ma

12/13/2021

Abstract

This project proposes a cost-effective robotic system that can quickly generate a point-cloud model of a patient's surface anatomy that can be used during a robotic ultrasound procedure. The point cloud model allows the robotic ultrasound to maintain optimal orientation and applied force to the patient's skin during the procedure and leads to more consistent imaging. An RGB-D Camera is mounted as the end effector of a Franka Emika Panda robotic arm such that the camera can image the critical regions of the patient's body. A model of this robotic system is constructed in a Gazebo-ROS simulation and Moveit is used to control the robot. Kinematic and dynamic models were derived to allow for manipulation of end effector position, orientation, velocity, and acceleration by controlling joint torques. Quintic polynomial trajectory planning is used to generate a smooth path for the camera mounted on the end-effector. The calculations and simulation for the Inverse-Kinematics and Inverse-Dynamics was done using MATLAB and Robotics Toolbox[2] developed by Peter Corke. We obtained depth data from the Gazebo depth camera which was visualized using Rviz.

Introduction

Ultrasound imaging has been an important diagnostic tool for physicians since the 2000's [3]. This is a non- evasive imaging technique using an ultrasound transducer. Compared to other tissue scanning devices such as CT or MRI, ultrasound imaging is low cost and puts the patient at very little risk [4]. However, there are some downsides to this technique. The quality of the ultrasound image highly depends on applied force and the pose of the transducer, a decision made in real time by the sonographer. This can lead to cases where different sonographers will not arrive at the same diagnoses [4]. As well, some ultrasound imaging procedures demand the sonographer to apply high forces for enduring periods of time and can lea d to strain injuries and musculoskeletal pain [5].

One solution to this is to employ an autonomous or teleoperated ultrasound scanning robot. This idea became more prominent with the arrival of the Covid-19 pandemic when it became difficult to have a sonographer present for direct ultrasound imaging of patients [5]. Ideally, these ultrasound imaging robots can perform scans with consistent and reproducible results while minimizing the physical pain to the patient and sonographer. In order to achieve this, it is essential that the robot end-effector maintains the transducer normal to the patient's skin while applying the correct amount of force. To constrain the transducer's movement in this way, it is helpful to have a 3D surface model of the patient's ext ernal body prior to the ultrasound scan procedure. In this way, a trajectory that traverses the patients surfac e perfectly can be mapped and followed by the robotic ultrasound device.

The goal of this project is to propose a cost-effective robotic system that can quickly preform a preoperative scan of the patient's surface anatomy using an RGB-D camera and construct a point cloud model that can be used to maintain a normal orientation of the transducer during the robotic ultrasound imaging operat ion. To sweep over the patient's body to image both sides and the top of the torso, the RGB-D camera is mounted as the end effector of a Franka Emika Panda robotic arm. The camera obtains point-cloud surface data. This surface model obtained can be utilized in the future to plan the motion of the ultrasound transducer.

The first objective of this project is to set up a simulation environment for the Franka Emika Panda robot with a mounted RGB-D camera in MATLAB and ROS. The second objective is to develop the inverse kinematic model and inverse dynamic model of the robot based on the information provided in the official documentation. The final objective is to develop a trajectory of the camera for the point-cloud acquisition of the targeted areas around the patient's lungs and to achieve surface reconstruction and optimisation from point-cloud data captured by the RGB-D camera.

Materials and Methods

For our simulation we have used the Franka Emika Panda Robot. The specifications of the robots were obtained from the data provided by the official documentation of Franka Emika Panda[5]. All the links and their corresponding location are as seen in Fig.1

The columns of the S matrix represent the Screw axis corresponding to the joints in space frame. M is the home-configuration (transformation between the spaceframe and tool-frame). The S and M matrix are-

$$M = \begin{matrix} 1 & 0 & 0 & 0.088 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0.926 \\ 0 & 0 & 0 & 1 \end{matrix}$$

$$S = \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & -1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & -1 \\ 0 & -0.333 & 0 & 0.659 & 0 & 1.033 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.088 \\ 0 & 0 & 0 & -0.0825 & 0 & 0 & 0 \end{matrix}$$

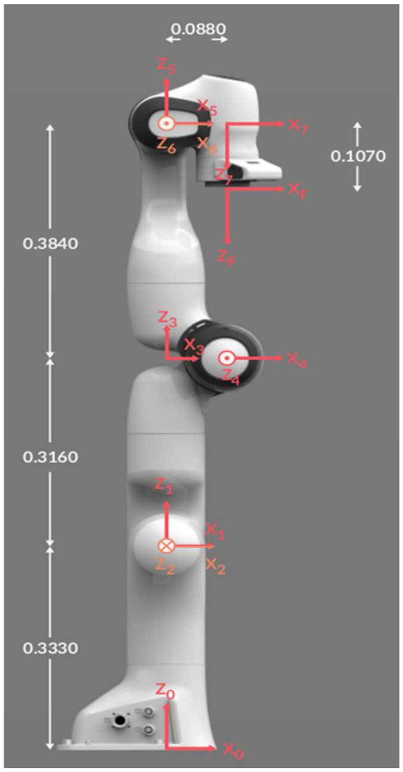

Figure 1: Panda in Home Configuration

For implementing forward kinematics, we have used Product of Exponentials method (PoE). Which can be mathematically represented as –

$$T = e^{[S1]\theta 1} e^{[S2]\theta 2} e^{[S3]\theta 3} e^{[S4]\theta 4} e^{[S5]\theta 5} e^{[S6]\theta 6} e^{[S7]\theta 7} M$$

Where, T is the transformation Matrix, Si is the screw axis corresponding to the ith joint and θi is the ith joint variable.

Inverse Kinematics –

To implement inverse-kinematics we have used NEWTON-RAPHSON method. It is an iterative method to find a solution for the joint values of the robot for the desired position of the end-effector. The Newton-Raphson method is mathematically formulated as –

$$\Delta Q = pinv(J) * (Target_{Twist} - Current_{Twist})$$

Where, ΔQ is the vector containing the change in joint variables, $pinv(J)$ is the Moore-Penrose pseudo-inverse of the Jacobian matrix.

To further improve the solutions obtained for the inverse kinematics of this robot, we implemented the Levenberg-Marquardt method (Damped least square method.) The damped least square methods is given as:

$$J_{star} = (J') * pinv(J * (J') + (\lambda * \lambda) * I)$$

$$\Delta Q = pinv(J_{star}) * (Target_{Twist} - Current_{Twist})$$

Where, λ is the damping factor.

Inverse Dynamics –

To find out the inverse dynamics solution for the we have implemented the recursive NEWTON-EULER formulation of dynamics.

- Forward Iteration:

Goal is to find the ν_i and $\nu_i dot$

Where, ν_i is the body twist of the ith link and $\nu_i dot$ is the time-derivative of the body twist.

$$\nu_i = A_i * (\theta d_i) + [Ad_{T_{i,i-1}}]\nu_{i-1}$$

$$\nu_i dot = A_i * (\theta_i dd) + [ad_{\nu_i}]A_i * (\theta_i d) + \left[Ad_{T_{i,i-1}}\right] * (\nu_{i-1}\mathrm{d})$$

Where, A_i is the screw-axis of the ith joint expressed in the local frame {i},

(θd_i) is the ith joint velocity,

$[Ad_{T_{i,i-1}}]$ is the Adjoint Transform,

$\nu_{i-1}$is the twist of the previous joint,

$(\theta_i dd)$ is the joint acceleration of the ith joint,

$[ad_{\nu_i}]$ is the lie bracket of twist ν_i,

$(\nu_{i-1}\mathrm{d})$ is the time-derivative of the body twist of the previous (i-1th) joint.

- Backward Iteration:

Input - ν_i and $\nu_i dot$

Output – F_i (wrench of the ith joint)

$$F_i = G_i \nu_i dot - [ad_{\nu_i}]' G_i \nu_i + [Ad_{T_{i,i-1}}]' * F_{i+1}$$

Where, $$G_i = \begin{matrix} I_b & O \\ O & m * I \end{matrix}$$

G_i is the Space Inertia Matrix, and $I_b$is the rotational inertia matrix and O is a 3*3 matrix with all the elements zero and m is the mass of the ith link, and I is a 3*3 identity matrix.

To find the joint torques, we will do the following step –

$$\tau_i = F_i' * A_i$$

Where, τ_i is the required torque for the ith joint.

Experiment

The motion plan for the depth camera was decided to be a semi-circular path based on the constraints related to the robot position and patient position. The robot will first traverse the semi-circular path above the chest area and then translate to the lower chest area and traverse a semi-circular path again. This path was used to perform inverse kinematics and find out the joint positions for every point in the path. The joint position matrix was then used to generate a quintic polynomial trajectory and the velocity and acceleration profile was obtained. The inverse dynamics was implemented using these velocity and acceleration profile to compute the torque matrix. This trajectory is planned in such a way that the end-effector is always normal to the path.

The simulation setup is performed in Gazebo by creating a Gazebo world where the patient is lying on a bed and the robot is placed on a table. The motion planning is performed using the MoveIt motion planning framework. The packages used for implementing the motion plan were provided by Franka Emika's[…] official website and Erdal Pekel's blog[…]. Erdal has used joint position controller which takes in joint positions as input and provides joint torque as output. Using MoveIt's provided functions, a function is written in python to make the robot go to a particular set of joint angles. The PID control is done through ROS_control and a smooth motion of the robot is achieved.

The joint position matrix which was calculated from inverse kinematics is then provided to this set_joint_angles() function as argument to implement the intended motion plan.The depth camera is added to the panda hand and provides the depth information through the Rviz interface. The depth information is published on the panda_camera/depth/image_raw topic.

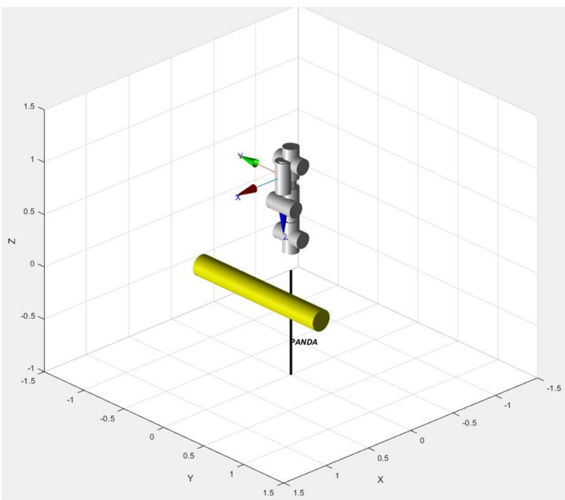

Figure 2: Panda and patient simulated in MATLAB

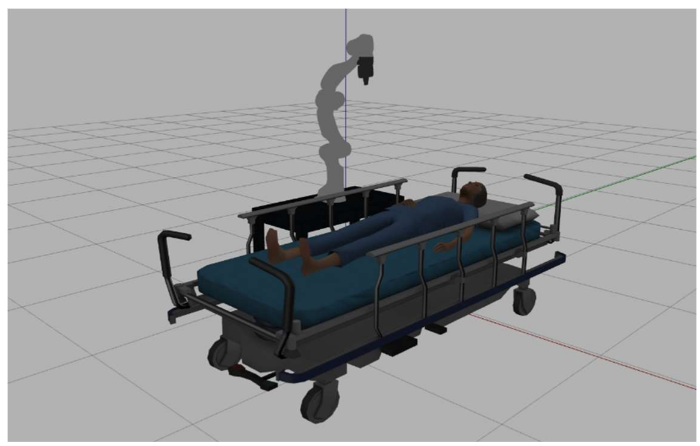

Figure 3: Panda and patient simulated in Gazebo

Results

From the dynamical simulation of the panda robot, we plotted the resulting joint torque, velocity and position profiles. These plots are shown below in figures 4-6.

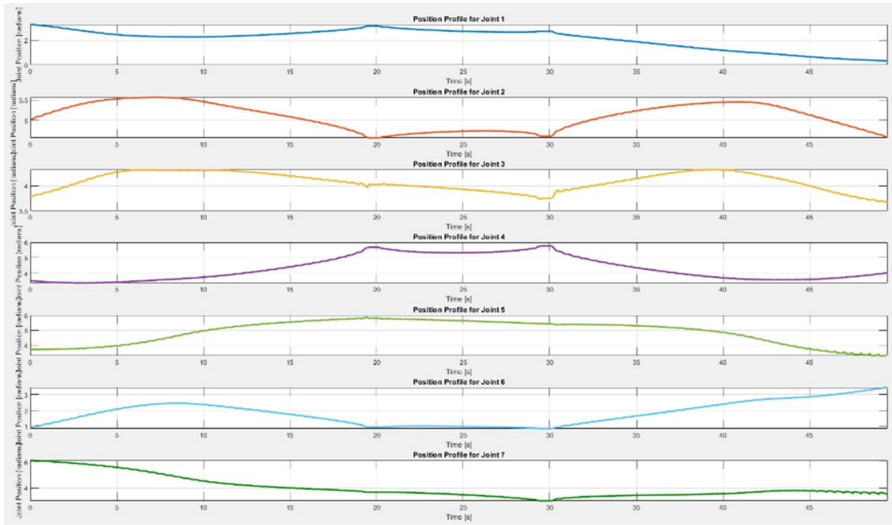

Figure 4: Joint angles over time from forward dynamic simulation in radians

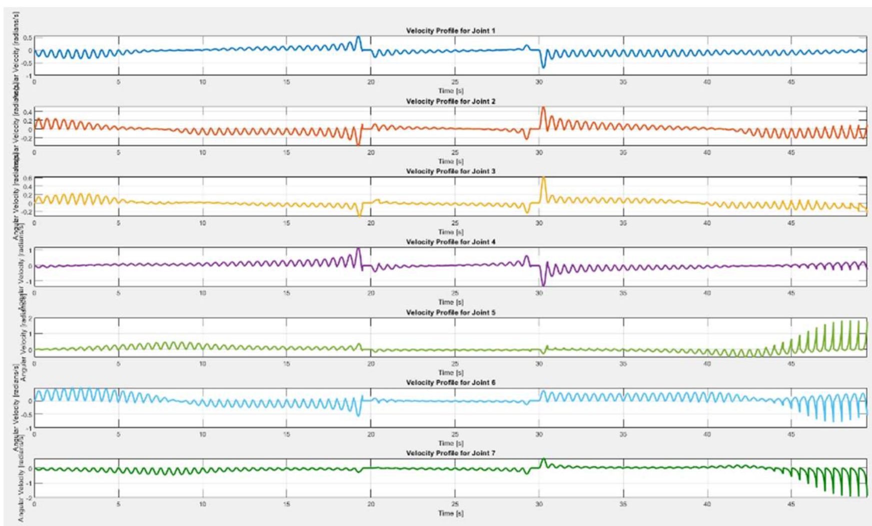

Figure 5: Joint angular velocities over time from forward dynamic simulation in radians/second

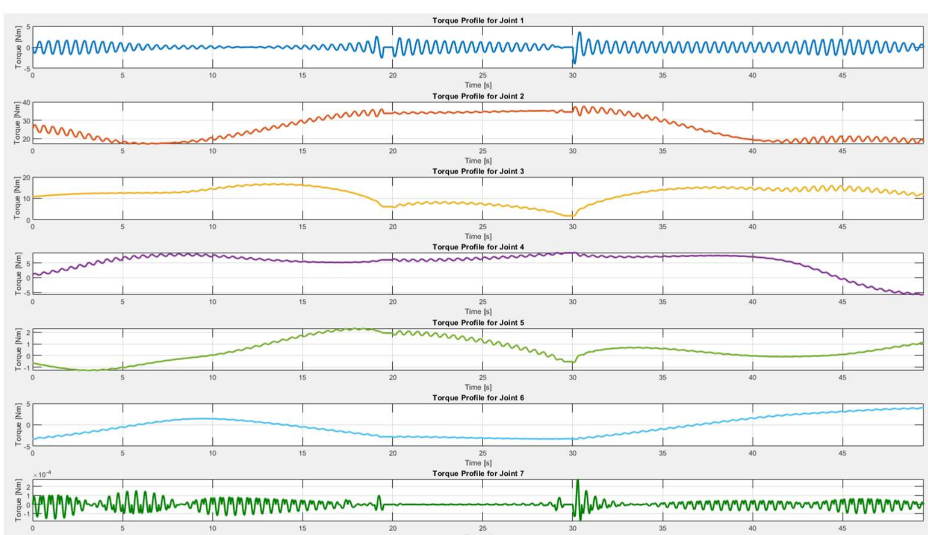

Figure 6: Joint torques over time from inverse dynamic simulation in Newton.meters

Discussion

This work relies on the quality of the motion plan and the quality of depth cloud data acquired by the depth camera. The accuracy of the motion plan depends on the calculations of screw axes and home configuration. Thus, if the home configuration is changed then all the resulting calculations have to be altered to help achieve the intended motion plan.

One of the limitations of this work is limited field of view of the depth camera which means that the robot need to be very close to the patient to get the depth data. This work also considers just one kind of intended motion plan which makes the approach limited. Different sort of motion plans may help in achieving better depth information data to help produce a better point cloud.

Conclusion

This project concluded that the 7-DoF Franka Emika Panda robot could feasibly be implemented to obtain point cloud data of a patient's surface anatomy. The Panda robot was able to span the desired path while pointing its end effector toward the patient's body. The results have shown that using quintic polynomial trajectory planning, a smooth torque trajectory can be implemented to traverse the patient's body. The torques required were shown to be within typical bounds for the joint motors. The Panda robot would be a good choice to implement for the multi-view patient perception and body anatomy reconstruction and would greatly increase the accuracy of robotic ultrasound procedures.

References

[1] Tirth Ginoya, Yaser Maddahi, Kourosh Zareinia, "A Historical Review of Medical Robotic Platforms", Journal of Robotics, vol. 2021, Article ID 6640031, 13 pages, 2021. https://doi.org/10.1155/2021/6640031

[2] Center for Devices and Radiological Health. (n.d.). *Ultrasound Imaging*: FDA. U.S. Food and Drug Administration. Retrieved October 29, 2021, from https://www.fda.gov/radiation-emitting-products/medical-imaging/ultrasound-imaging

[3] Nightingale, R., Bell, D. *Ultrasound transducer*. Reference article, Radiopaedia.org. (accessed on 28 Oct 2021) https://doi.org/10.53347/rID-54038

[4] Fang, T. Y., Zhang, H. K., Finocchi, R., Taylor, R. H., & Boctor, E. M. (2017). Force-assisted ultrasound imaging system through dual force sensing and admittance robot control. International journal of computer assisted radiology and surgery, 12(6), 983–991. https://doi.org/10.1007/s11548-017-1566-9

[5] Hanning, J., Silsby, M., Almy, Z., & McCann, E. (2021). *A Robotic Platform for At-Home Ultrasound Diagnostic Imaging.* : Worcester Polytechnic Institute.

[6] Q. Huang, J. Lan and X. Li, "Robotic Arm Based Automatic Ultrasound Scanning for Three-Dimensional Imaging," in *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 1173-1182, Feb. 2019, https://doi.org/10.1109/TII.2018.2871864

[7] Kaminski, J. T., Rafatzand, K., & Zhang, H. K. (2020). Feasibility of Robot-Assisted Ultrasound Imaging with Force Feedback for Assessment of Thyroid Diseases. Proceedings of SPIE--the International Society for Optical Engineering, 11315, 113151D. https://doi.org/10.1117/12.2551118

[8] Gao, S., Ma, Z., Tsumura, R., Kaminski, J., Fichera, L., & Zhang, H. K. (2021). Augmented immersive telemedicine through camera view manipulation controlled by head motions. *Proceedings of SPIE, the International Society for Optical Engineering*, 1159815.

[9] R. Tsumura *et al*., "Tele-Operative Low-Cost Robotic Lung Ultrasound Scanning Platform for Triage of COVID-19 Patients," in *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4664-4671, July *2021, https://doi.org/10.1109/LRA.2021.3068702.*

[10] Chang, Shu-Wei & Chiang, Yi-Te & Chang, Fan-Ren. (2010). SLERP-based optimal TRIAD algorithm. Proceedings of the SICE Annual Conference.

[11] *Zhang L, Xia H, Qiao Y. Texture Synthesis Repair of RealSense D435i Depth Images with Object-Oriented RGB Image Segmentation.* Sensors. 2020; 20(23):6725. https://doi.org/10.3390/s20236725

[12] P. Li, R. Wang, Y. Wang and W. Tao, "Evaluation of the ICP Algorithm in 3D Point Cloud Registration," in IEEE Access, vol. 8, pp. 68030-68048, 2020, https://doi.org/10.1109/ACCESS.2020.2986470.

[13] *Author: Erdal pekel*. Erdals blog. (n.d.). Retrieved October 29, 2021, from https://erdalpekel.de/?author=1.

[14] *Installation on linux*¶. Installation on Linux - Franka Control Interface (FCI) documentation. (n.d.). Retrieved October 29, 2021, from https://frankaemika.github.io/docs/installation_linux.html.

[15] *Moveit tutorials*¶. MoveIt Tutorials - moveit_tutorials Noetic documentation. (n.d.). Retrieved October 29, 2021, from https://ros-planning.github.io/moveit_tutorials/.

[16] Erdalpekel. (n.d.). *Erdalpekel/panda_simulation: This repository contains a sample ROS package that launches the Franka Emika Panda Robot in the gazebo simulation environment.* GitHub. Retrieved October 29, 2021, from https://github.com/erdalpekel/panda_simulation.

[17] I.R.S. (2021, July 1). *GitHub - IntelRealSense/realsense-ros: Intel(R) RealSense(TM) ROS Wrapper for D400 series, SR300Camera and T265 Tracking Module. GitHub.* https://github.com/IntelRealSense/realsense-ros

Authorship

ROS Simulation: Fenil Desai and Ben Verdesi

Kinematics and Dynamics Modeling: Sairam Venkataramani and Pratik Surendra Kumbhare