Final Report

April 18, 2025

Project: Simulating the Kinematics & Dynamics of a 7-DOF Franka Emika Panda Robot to Perform Fetal Ultrasound in Remote Settings

Team #21: Aly Khan Nuruddin, Brandon Tsang, Sumaiya Hossain, Victor Jiang

Instructor: Tim Salcudean

THE UNIVERSITY OF BRITISH COLUMBIA

Word Count: 5310

Table of Contents

Abstract

This project aims to develop a computer vision-guided robotic gripper system using the 7-DOF Franka Emika Panda for remotely-operated fetal ultrasound imaging. The implementation focuses on lower abdomen scanning for gynecologic sonography, with particular emphasis on improving diagnostic accuracy in low-resource settings. The system's design attempts to incorporate safety mechanisms and force sensing capabilities, drawing from established robotic ultrasound protocols. The simulation implemented in MATLAB successfully demonstrated a proof-of-concept for possible integration within clinical environments. Further, possible denoising methods showcased immense potential for obtaining high-quality ultrasound images.

Introduction

Ultrasound imaging is a fundamental tool in medical diagnostics, widely used for its ability to provide real-time, non-invasive insights into internal organs and tissues. It is essential that precise and consistent positioning of the ultrasound probe is achieved to produce accurate and reliable images. Even small variations in probe positioning can drastically impact the clarity and diagnostic quality of images, particularly for tasks that require repetitive scanning or high precision. Ultrasound is predominantly implemented as a high throughput device, with specific applications for detecting cardiac defects and fetal abnormalities. However, rural communities often lack the necessary infrastructure and medical expertise needed to execute these diagnostic tests on a regular basis. Additionally, significant healthcare accessibility challenges exist in such inaccessible, remote locations, with Micks T. et al (2016) estimating that 72% of residents lack access to medical ultrasound. Sadly, these barriers to point-of-care ultrasound use increase the likelihood of delayed or inaccurate diagnoses, thereby significantly impacting patient outcomes.

Methodology

The Franka Emika Panda robot is a 7-degree-of-freedom (DoF) collaborative robot (cobot) widely used for applications requiring precise and safe manipulation, like medical robotics. Motivated by this need, accurate modeling of its kinematics and dynamics is essential for control, simulation, and planning around image-derived cues or predefined anatomical landmarks. Due to the project's complexity, this implementation is primarily focused on simulating the robot's motion in MATLAB and describing the system's conceptual framework.

System Overview

The full system design consists of 6 primary components:

1. **Robotic Manipulator**: The Franka Emika Panda, selected for its compliance, precision, and integration with existing robotics toolchains. In a full implementation, an off-the-shelf (or 3D printed) adapter that can hold a convex or linear array ultrasound probe would be attached to the end effector to facilitate the fetal scanning.
2. **Control Layer:** A MATLAB script that simulates the forward kinematics (FK), inverse kinematics (IK), and trajectory planning necessary to move the robot to a desired pose.
3. **Simulation:** A MATLAB simulation that verifies our work done in the Control Layer with the intention of showcasing the clinical feasibility of such a proof-of-concept.
4. **Computer Vision Module:** A theoretical algorithm that detects visual markers in a scene to determine target positions for probe placement relevant for autonomous applications.
5. **Force Control Module:** A theoretical approach using torque sensors to ensure patient safety by limiting the amount of force the robot can apply on the scanning surface.
6. **Image Processing Module:** A Python notebook that assesses the performance of denoising techniques using synthetic ultrasound images from an open-source repository.

For the purposes of this report, only the robot's kinematic model and motion planning in joint space were developed and simulated in MATLAB. The computer vision and force control components are described theoretically to demonstrate an understanding of how the system could be extended in future work, while still holistically capturing its core functionalities.

II. Control Layer

The process of forward kinematics involves determining the position and orientation of a robot's end-effector based on its known joint angles. It is one of the most fundamental calculations in robotics for tasks such as trajectory planning, control and workspace analysis. We use Denavit-Hartenberg (DH) parameterization to describe the geometry of the Franka Emika Panda robot, which consists of 7 revolute joints, with a flange for end-effector articulation and power.

Moreover, both Cartesian and joint space limits must be considered for the Franka Emika when modeling its motion. While these limits are important to respect so that the robot is not damaged, it is equally relevant to ensure that the device operates within its defined workspace, thus minimizing harm to patient or nearby staff especially in sensitive medical environments.

Once the pose of the robot's end-effector is computed, the Jacobian matrix $J(q)$ is needed to map joint velocities to Cartesian space end-effector velocities. The Jacobian formulation, specific to the Franka Emika robot, can be found using well documented methods and those taught in class.

After applying control inputs of joint position to the robot's control system, the joint velocities are calculated using numerical differentiation. Simultaneously, a physical sensor can be used to measure the current end-effector pose, which includes both its position and orientation in space.

For dynamic modeling, the Newton-Euler algorithm will be applied to compute the joint torques required to achieve a desired motion. The algorithm will calculate the velocities and accelerations of each link recursively from the base to the end-effector, using joint parameters and kinematics. After determining the net forces acting on each link, the transmitted joint force is computed, and the necessary joint torque is derived by projecting these forces onto the joint axes.

Lastly, in torque-controlled robots such as the Franka Emika Panda, the joint torque is essential for ensuring precise and stable motion. This is achieved using the inverse dynamics model, which joint torques using joint position q, joint velocities $\dot{q}$, and joint acceleration $\ddot{q}$.

III. Simulation

The behavior of the automated ultrasound scanner will be emulated over a flat soft-tissue surface using the Franka Emika robot. The goal is to validate inverse kinematics (IK)-based control for surface-level scanning over the abdomen for fetal imaging, ultimately laying the foundation for a future vision-guided control framework with applications in remote medical settings.

The simulation is performed in MATLAB using the `loadrobot` API, which initializes the Franka model with gravity and a defined end-effector, `panda_hand`. The scanning area is discretized into a grid of target positions, with 10 rows by 10 columns, each spaced at 5 cm intervals. These dimensions were designed to capture the average anatomical proportions of the mid-section, simulating a planar surface. To mimic clinical scanning behavior, a zig-zag pattern is used where every alternate row is traversed in reverse column order. This path structure minimizes unnecessarily large re-orientations and improves efficiency of the rasterization. While this trajectory is a fairly crude simplification compared to real-world sonography practices, it is sufficient to capture the baseline ability for the robot to traverse across a 2-dimensional plane.

A key part of the simulation involves defining the probe orientation perpendicular to the surface. This is achieved using an axis-angle rotation that flips the end-effector 180° around the X-axis, ensuring that the probe tip points downward. The robot is required to reach each target pose with this fixed orientation. The gripper center point is offset by 11 cm along the local Z-axis, and represents the cumulative length of the physical probe attached to the end-effector.

An inverse kinematics solver, `inverseKinematics`, is used for each grid point to compute the corresponding joint configuration that enables the robot to reach the desired pose. Moderate importance is given to the orientation in the IK weighting to allow for some flexibility during solution search. The robot begins from its home configuration and updates its internal IK guess sequentially using the solution from the previous pose to ensure smooth and continuous motion.

A 3D environment is constructed using MATLAB's `surf`, `scatter3`, and `plot3` functions to visualize the soft-tissue plane, which is demarcated with a light brown skin-tone color, as well as grid points with red markers, and the probe path in black. The ultrasound transducer is modeled as a cylinder of 1 cm radius and 5 cm height. During the animation, the robot's joints are updated to transition between scan points, with real-time rendering of its dynamically positioned probe.

A GIF animation is generated by capturing each frame during the robot's motion. The simulation loop stores the probe's trajectory and applies transformation matrices to move and rotate the phantom in accordance with the current end-effector frame. To provide a clear sense of progression, the path traced by the probe tip is drawn incrementally over the simulated tissue.

This controlled simulation is intended to explore whether the Franka Emika robot can execute position and orientation-constrained motion across a defined scanning surface. By integrating pose tracking, environment modeling, and inverse kinematics, this framework provides a reliable basis for future development of automated or vision-based ultrasound navigation systems.

IV. Computer Vision (Theoretical)

There are several practical reasons why camera vision control is important in robotic sonography. For example, in situations where direct human contact should be minimized, such as during infectious disease outbreaks like the COVID-19 pandemic, remote or automated scanning provides clear safety benefits for both patients and healthcare providers.

Moreover, computer vision supports more repeatable and automatable sonography by enabling the system to consistently position and orient the probe based on identifiers like anatomical features. This reduces operator variability and improves image quality across scanning sessions.

In a full-scale implementation, the robot would rely on one or more cameras or depth sensors to detect visual cues such as anatomical outlines or reference markers placed on the patient's body. Using computer vision techniques, potentially with the help of libraries like OpenCV, the system can estimate the desired position and orientation of the probe. This pose information is then passed on to an inverse kinematics solver, which moves the robotic arm accordingly.

One major challenge in this process is achieving reliable accuracy across different patients. Human bodies vary in shape, size, and skin tone. Thus, even small errors in visual pose estimation could cause the robot to move in ways that are uncomfortable or unsafe for patients.

One way to reduce this risk is to integrate force sensing into the system. This allows the robot to respond to contact feedback in real time and apply only an appropriate amount of pressure.

V. Force Sensing (Theoretical)

In a real clinical setting, the ability to precisely control and measure forces during robotic-assisted procedures is an obvious requirement. Force control mechanisms will allow the robot to respond appropriately to physical contact and variations in the patient's anatomy, ensuring the robot's end-effector maintains a safe and controlled force profile.

Limiting End-Effector Force Using Torque Sensors

One of the most fundamental aspects of force control is the ability to monitor and limit the forces exerted by the robot's end-effector. This can be achieved using torque sensors embedded in the robot's joints or force-torque sensors at the wrist or end-effector. These sensors measure the forces applied during interaction with the environment, during probing or contact with patients.

By continuously monitoring these forces in real-time, the robot's control system can actively adjust motion commands to prevent excessive force, thus ensuring no harmful contact is made with the patient. For instance, if the end-effector applies too much pressure against the patient's skin, the control system could adjust the robot's trajectory or reduce speed to mitigate injury risk.

Admittance and Impedance Control

Another technique involving impedance control via a stiffness controller can also be considered. Ideally, this will regulate the robot's stiffness and damping properties to control its behavior during interaction with the patient. By adjusting its impedance, the robot can vary its response based on the resistance encountered in its environment. In a similar vein to using torque sensors, when probing soft tissue, the robot can adjust its stiffness to be more compliant, reducing the risk

of discomfort or injury. This can be accomplished on the Franka Emika robot via `SetJointImpedance` and `SetCartesianImpedance` listed under `franka_control`.

Real-Time Haptic Feedback

Another aspect of force control in clinical robotics is providing real-time haptic feedback to the human operator. In a teleoperation scenario with the Franka, this feedback helps the operator make safer adjustments and avoid actions that could lead to unsafe situations. By providing haptic feedback through a control interface such as a robotic arm or glove, the human operator can adjust their actions more effectively, responding intuitively to changes in force and torque.

VI. Image Processing Module

Synthetic ultrasound image processing was carried out using a dataset derived from CT scans, sourced from the public repository at https://github.com/MXHsj/rus_sim1.0. These images were generated to mimic ultrasound synthesis for scenarios where real-time data acquisition would not be feasible, such as during experimentation with the Franka Emika robot on phantom models.

However, the absence of real ultrasound ground truth images necessitated the use of CT scans as a reference for evaluating post-processing efficacy. Specifically, the simulated ultrasound images were inherently noisy and therefore underwent three denoising techniques: Wavelet filtering using the `BayesShrink` method with soft thresholding and multi-level decomposition, Wiener filtering leveraging local variance estimation and Morphological filtering through median rank operations with a disk-shaped structuring element. All preprocessing steps involved converting grayscale pixel intensities to `float32` values normalized to [0,1] for standardized computation.

The methodology involved pairing each synthetic ultrasound image with its corresponding CT slice. Each denoising technique was applied across all ultrasound images, and the resultant filtered outputs were evaluated against their CT counterparts using three key metrics: Structural Similarity Index Measure (SSIM), Peak Signal-to-Noise Ratio (PSNR), and the standard deviation of the noise difference (Noise-STD) between the filtered ultrasound and the CT scan. SSIM quantified perceptual image similarity, while PSNR offered a logarithmic estimate of signal fidelity, and Noise-STD assessed residual discrepancies. A `DataFrame` was generated to

compile these metrics, enabling statistical comparisons via boxplots. This allowed identification of the most robust denoising method under the given constraints. To complement the numerical evaluation, side-by-side visual comparisons were presented for three slices, clearly displaying each denoised output alongside its original noisy version and the ground truth CT.

The results of this pipeline are significant for guiding optimal post-processing strategy for the simulated data with direct implications for real-time robotic ultrasound acquisition. Once the Franka Emika robot captures ultrasound sequences, these denoising techniques can be readily applied to improve the clinical interpretability of the scans. In the absence of ideal acquisition conditions or high-end ultrasound equipment, post-processing enhancement becomes critical. Using reliable filtering pipelines grounded in objective metrics like SSIM and PSNR can bridge the quality gap, helping align the output of robot-guided imaging with diagnostic standards.

Results

I. Modeling of Kinematics & Dynamics

For the forward kinematics and dynamics of the Franka, we start with the Denavit-Hartenberg parameters, which use four variables per joint to describe the robot geometry along its coordinate frame for each link. The parameters include joint angle θ, link offset d, link length a, and twist angle α. Table #1 lists the D-H parameters for each of the 7 joints of the robot and its flange.

Table #1: D-H parameters of the Franka Emika Cobot.

Joint	θ (rad)	d (m)	a (m)	α (rad)
1	θ_1	0.3330	0	0
2	θ_2	0	0	-π/2
3	θ_3	0.3160	0	π/2
4	θ_4	0	0.0825	π/2
5	θ_5	0.3840	-0.0825	-π/2

6	θ_6	0	0	$\pi/2$
7	θ_7	0	0.0880	$\pi/2$
Flange	0	0.1070	0	0

The motions of the Franka Emika robot are bounded by predefined limits in Cartesian space. These include constraints on the maximum allowable velocity $\dot{p}_{max}$, acceleration $\ddot{p}_{max}$, and jerk $\dddot{p}_{max}$ for three types of motion: linear (translational), angular (rotational), and specific movements about the elbow joint. As mentioned previously, it is important to carefully control these to ensure safe, smooth, and physically feasible motion, particularly in medical and teleoperated environments. The Cartesian limits of the Franka Emika robot are listed in Table #2.

Table #2: Cartesian Space Limits of the Franka Emika Cobot

Name	Translation	Rotation	Elbow
$\dot{p}_{max}$	1.7 m/s	2.5 rad/s	2.1750 rad/s
$\ddot{p}_{max}$	13.0 m/s^2	25.0 rad/s^2	10.0 rad/s^2
$\dddot{p}_{max}$	6500.0 m/s^3	12500.0 rad/s^3	5000.0 rad/s^3

In addition to constraints in Cartesian space, the motion of the Franka Emika is also restricted by a set of joint space limits which are predefined. Unlike Cartesian constraints, which affect movement of the end-effector in 3D space, joint space limits directly govern internal rotation and performance of the robot's 7 joints. Joint space constraints cover various parameters, including joint position (q_{max}, q_{min}), joint velocity ($\dot{q}_{max}$), joint acceleration ($\ddot{q}_{max}$), joint jerk ($\dddot{q}_{max}$), joint torque ($\tau_{j\,max}$) and joint torque rate ($\dot{\tau}_{j\,max}$). Successfully controlling these ensures smooth, controlled motion in simulated and real-time operation. Table #3 lists these Joint Space Limits.

Table #3: Joint Space Limits of the Franka Emika Cobot

Name	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6	Joint 7
q_{max} (rad)	2.8973	1.7628	2.8973	-0.0698	2.8973	3.7525	2.8973
q_{min}(rad)	-2.8973	-1.7628	-2.8973	-3.0718	-2.8973	-0.0175	-2.8973
$\dot{q}_{max}$(rad/s)	2.1750	2.1750	2.1750	2.1750	2.6100	2.6100	2.6100
$\ddot{q}_{max}$(rad/s^2)	15	7.5	10	12.5	15	20	20
$\dddot{q}_{max}$(rad/s^3)	7500	3750	5000	6250	7500	10000	10000
$\tau_{j\,max}$(Nm)	87	87	87	87	12	12	12
$\dot{\tau}_{j\,max}$(rad/s)	1000	1000	1000	1000	1000	1000	1000

Once the current pose of the end-effector is known, the Jacobian matrix $J(q) \in \Re^{6x7}$ is computed. The Jacobian provides a linear mapping between joint space velocities $\dot{q}_i \in \Re^7$ and the Cartesian space velocity $v \in \Re^6$ of the end effector. This relationship is expressed as:

$$\boldsymbol{v} = \begin{bmatrix} \dot{\boldsymbol{p}} \\ \boldsymbol{w} \end{bmatrix} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}_i$$

where the Cartesian space velocity v depends on the end-effector linear velocity $\dot{p} \in \Re^3$, and angular velocity $\omega \in \Re^3$.The Jacobian matrix itself can be defined using two sub-matrices: $J_{v,}$ which maps joint velocities to linear velocities and $J_{w,}$ which maps joint velocities to angular velocities. This expression is called the geometric Jacobian and is expressed as follows:

$$\mathbf{J} = \begin{bmatrix} \boldsymbol{J}_v \\ \boldsymbol{J}_w \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_{v1} & \cdots & \boldsymbol{J}_{v7} \\ \boldsymbol{J}_{w1} & \cdots & \boldsymbol{J}_{w7} \end{bmatrix}$$

In the case of the Franka Emika robot, all joints are revolute which means each column of the Jacobian matrix is computed based on the geometry of the robot using the following formulation:

$$\left[J_{vi}, J_{wi}\right]^T = \left[{}^0z_{i-1} * ({}^0p_7 - {}^0p_{i-1}),\ {}^0z_{i-1} \right]^T$$

where $({}^{0}p_{7})$ is the position of the end-effector with respect to the base frame, $({}^{0}p_{i-1})$ is the position of each revolute joint with respect to the base frame, and $({}^{0}z_{i-1})$ is the joint axis vector.

This formulation accounts for the contribution of each joint to the total velocity of the end-effector. It allows the robot to execute precision motion trajectories, respond to control commands in real-time and ensure compliance with dynamic constraints.

To model the dynamics of the robot, the basic relationship relating the vector of joint torques τ to the joint position q, joint velocities $\dot{q}$, and joint acceleration $\ddot{q}$ is defined as follows:

$$\tau = F^{-1}(q, \dot{q}, \ddot{q})$$

The Newton-Euler algorithm is commonly used to compute required joint torques, allowing for efficient computation for a given set of joint accelerations. This can be achieved in four steps.

The velocity $\mathbf{V_i}$ and acceleration $\mathbf{a_i}$ of each link i is calculated recursively starting from the fixed base. The velocity of a link depends on its parent link and motion of the joint connecting them:

$$\boldsymbol{V_i} = \boldsymbol{V_{i-1}} + \boldsymbol{S_i}\dot{\boldsymbol{q}}_i$$

The corresponding acceleration is obtained by differentiating the above expression:

$$\boldsymbol{a_i} = \boldsymbol{a_{i-1}} + \boldsymbol{S_i}\ddot{\boldsymbol{q}}_i + \dot{\boldsymbol{S}}_i\dot{\boldsymbol{q}}_i$$

The net force $\boldsymbol{f_i^a}$ acting on the body i is calculated using the following equation of motion:

$$\boldsymbol{f_i^a} = \boldsymbol{I a_i} + \boldsymbol{V_i} * \boldsymbol{I_i V_i}$$

where $\boldsymbol{I}$ is the inertia matrix of link i. Once the forces acting on each link are known, the algorithm proceeds with the backward recursion, starting from the end-effector and moving forward to the base. Thus, the total force transmitted through joint i is calculated as:

$$f_i = f_i^a - f_i^e + f_{i+1}$$

where f_i^e is the external force acting on the body, and f_{i+1} is the force transmitted from the next link in the chain. This step accumulates force from all links and external effects.

Finally, the torque τ_i required at joint i to produce the desired motion is calculated by projecting the resulting force onto the joint axis as shown. This step completes the dynamics computation and yields the joint torque vector, which can be directly used for actuation or control:

$$\tau_i = S_i^T f_i$$

To apply the Newton-Euler method in practice, a set of inertial parameters should characterize each robot link. These include: mass of the link, coordinates of its center of mass, and 6 elements of its symmetrical inertia matrix. However, these parameters are often estimated based on CAD models or approximations. Inaccurate estimations can lead to non-physical results, such as instability in the control loop. Including joint friction models and torque rate limits may further enhance simulation accuracy and controller robustness for practical implementation.

While the forward kinematics and inverse kinematics were successfully incorporated, similar success was not achieved in the dynamics and control areas. The MATLAB code was supposed to simulate the framework, but ended up not outputting expected results. The `inverseDynamics` function that was used to calculate the joint torques was able to run until the end without any major errors. However, the values of the torques obtained were very small and practically zero. The expected output was, at the very least, supposed to occupy 4 decimal places for a fully defined space instead of the observed 14 decimal places. This signifies incorrect assumptions for dynamic parameters or incorrect setup of the parameters used elsewhere in the code.

Additionally, the system did not perform as expected when trying to implement dynamics and PD control using the Newton-Euler method. This was primarily due to incorrect assumptions about the inertial parameters of the robot link: mass, inertia and center of mass. Moreover, the potential interactions between joints were not fully captured. Even after tuning the control gains,

the system did not converge. Further refinement of both the model parameters and the control structure is required to capture realistic dynamic behavior for clinical imp

II. Simulation

The following animation captures a simplified, zig-zag motion of the Franka Emika Panda robot across the grid-like planar phantom, as implemented in MATLAB (Figure #1). The continuous motion of the robot is captured within this GIF file: ultrasound-probe-scan.gif.

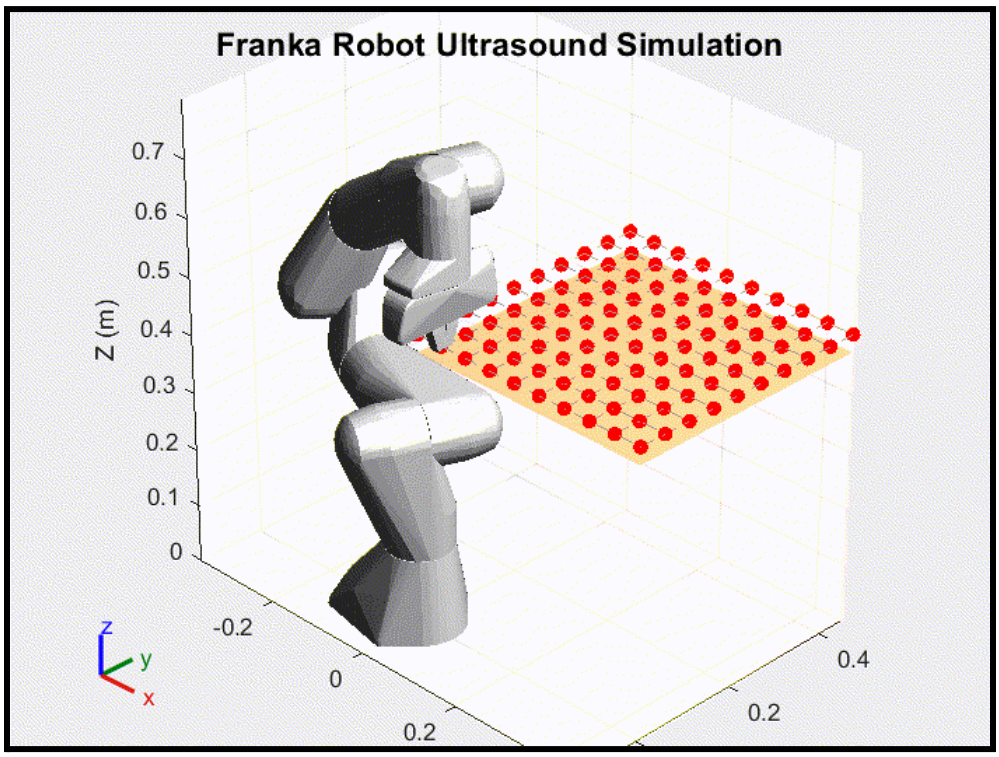

Figure #1: Franka Emika Panda Simulation for Ultrasound Over a Planar Tissue Surface.

Hence, the ultrasound simulation for the Franka Emika Panda robot successfully demonstrates the feasibility of using automated systems for perioperative scanning in clinical settings. The setup features a 7 DOF robot equipped to hold and manipulate an ultrasound probe across a planar surface, which represents a patient's abdomen. The red markers, spaced at 1 cm intervals on a 10x10 grid, visually represent the scan area and ensure consistent probe positioning during the raster scan. The black line illustrates the robot's programmed path, tracing a systematic raster pattern across the surface to mimic a comprehensive ultrasound examination. The consistency

and precision of the robot's movements along this path suggest that the robot can perform repeatable ultrasound scans, which is crucial for accurate diagnostics and monitoring.

The results of this simulation support the potential of robotic assistance in ultrasound procedures. The controlled raster scan pattern ensures uniform coverage of the simulated abdomen, reducing the likelihood of human error and improving consistency of image acquisition. The 1 cm spacing between scan points provides sufficient resolution for detailed imaging, while the robotic arm's precise movements minimize variations in probe pressure and angle, factors that can affect image quality. Additionally, the Franka robot's ability to execute pre-programmed scan paths demonstrates its adaptability for various clinical applications, such as routine abdominal scans, targeted lesion monitoring, and guided interventions, which are crucial for fetal monitoring.

In conclusion, the simulation depicts that automated systems such as the Franka Emika Panada robot have immense potential to enhance the efficiency, precision, and repeatability of ultrasound procedures, paving the way for improved diagnostic accuracy and patient outcomes. This proof-of-concept supports further research and development into robotic ultrasound solutions, potentially leading to widespread adoption of these technologies in healthcare environments.

III. Image Processing

Quantitative evaluation of the denoising methods using SSIM, PSNR, and Noise_STD metrics highlighted distinct differences in their performance, as observed in Figure #2.

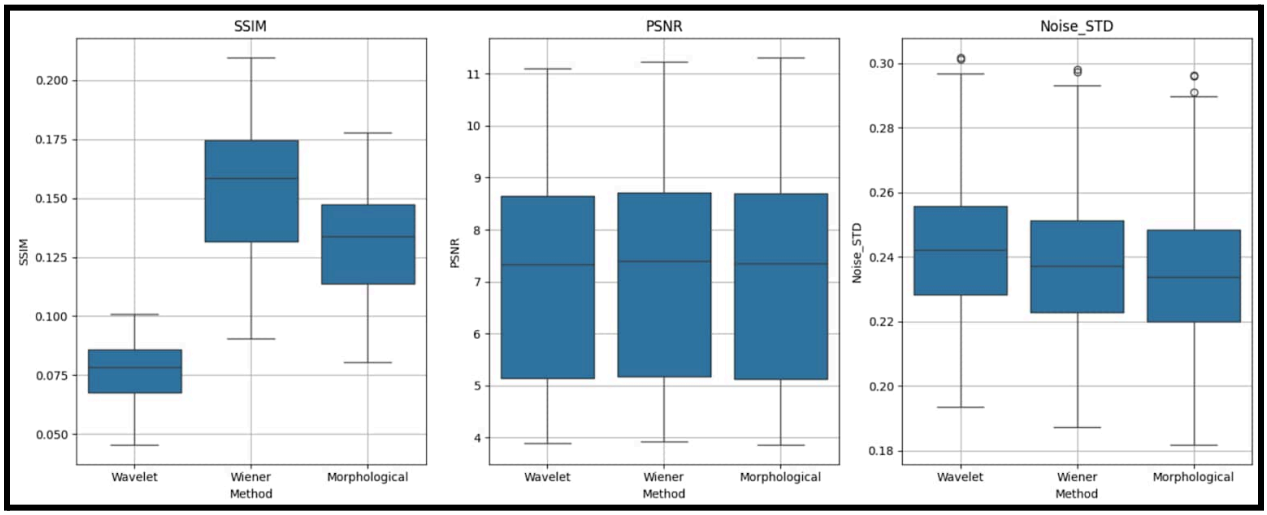

Figure #2: Boxplots for SSIM, PSNR and Noise_STD with Regards to the Denoising Methods.

The boxplots show that the Wiener filter achieved the highest median SSIM, indicating that it best preserved structural similarity compared to the ground truth CT image. Meanwhile, the Morphological method demonstrated intermediate SSIM values, while the Wavelet approach consistently yielded the lowest SSIM, suggesting less effective structural preservation. Despite these differences, all three methods exhibited similar PSNR distributions, with medians around 7 to 8 dB across a broad range, indicating that their overall signal-to-noise enhancement is comparable but varies across the dataset. In terms of noise suppression, the Morphological method achieved the lowest median Noise_STD, closely followed by the Wiener filter, while the Wavelet approach resulted in slightly more residual noise.

Visual comparisons further support these quantitative findings as indicated in Figure #3.

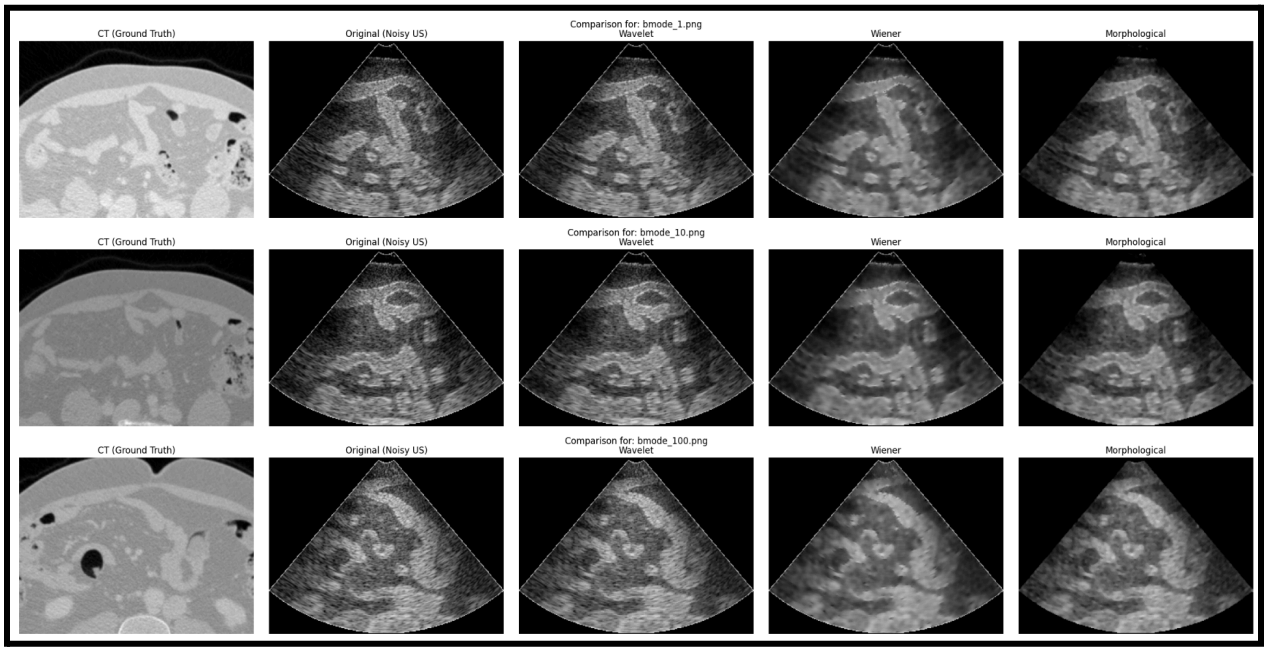

Figure #3: Side-by-Side Images for CT, Original & Denoised Ultrasound.

The original noisy ultrasound image displayed prominent speckle noise that obscured anatomical details. After denoising, the Wavelet method retained more image texture but left considerable noise, which is consistent with its lower SSIM and higher Noise_STD. However, the Wiener filter provided a favorable balance, effectively reducing noise while maintaining essential

anatomical structures, resulting in images that visually resemble the CT ground truth more closely. Lastly, the Morphological method produced the smoothest images with the most effective noise removal, but this came at the cost of some blurring and loss of fine detail. Across multiple representative cases, these trends are consistent, with the Wiener filter offering the best compromise between noise suppression and structure preservation. The Morphological method excels in noise reduction but may sacrifice anatomical clarity, while the Wavelet method, although preserving some texture, is less effective at both noise removal and structure retention.

These results collectively highlight the strengths and limitations of each approach for ultrasound image denoising, emphasizing the importance of balancing noise reduction with the preservation of diagnostically relevant structures. Additionally, it is important to consider that the outcomes for these image processing methods could have been significantly different for inter-modality comparison with non-noisy ultrasound images instead of the provided cross-modality CT scans.

Discussion and Conclusions

This project aimed to develop a computer vision-guided robotic gripper system utilizing the Franka Emika Panda for remote fetal ultrasound imaging, with a focus on improving diagnostic accuracy in low-resource settings. While the theoretical controller design and system modeling were enough to create a simulation, many of the ideas that were pertinent to the dynamics and control elements of the robotic operation were ultimately not very successful.

In general, the approach we have undertaken uses well-established techniques in the field of robotics and have been successfully applied to a variety of computational platforms. Specifically, the theoretical framework for forward kinematics, inverse kinematics, and dynamic modeling used in this project is primarily based on the work of Trabelsi et al. (2021). The research group was able to apply this analytical approach to their project and ultimately were able to publish their simulator for use as a validation tool with medical applications at the Prime Institute. Given their success, it seemed reasonable to conclude that our approach, which followed the same conceptual underpinnings, should have led to a similarly successful outcome.

The primary setback in this project was the inability to resolve the dynamics of the Franka robot. While we were able to execute the simulation of the Franka using the forward and inverse kinematics, the lack of a functioning control system prevented our project from being fully operational. Despite this, the simulation provides a solid basis that captures the proof-of-concept for future iterations. Additionally, the image processing methods highlighted practical denoising strategies to improve the quality of diagnostic information for assessment by sonographers.

Another important takeaway was the critical need for early validation and testing as we go, instead of designing and implementing the modules in one sitting. In hindsight, validating the kinematic and dynamic models earlier in the process could have allowed us to identify the roadblocks present in simulating Frank Emika's control systems much earlier. This would have given us sufficient time to focus our efforts on resolving the issue, potentially enabling us to simulate more complex movements to better demonstrate the practical viability of our work.

Future Work

Expanding the scope of the project to include force control and computer vision capabilities is the natural next step. This would likely require moving beyond computational simulation and developing the robot in a physical environment for iterative testing and controller refinement.

In terms of established literature, many studies have been conducted in the past both conceptualizing and validating robot-assisted sonography. For example, Najafi and Sepehri (2011) designed a robotic wrist for remote ultrasound imaging that requires only 4-DOF. They manufactured the robot to be singularity-free within its workspace, with all joints fully kinematically decoupled. This design choice was supported by Taylor and Stoinovici (2003), who suggest that for medical applications, robots with decoupled joints and a remote center of motion (RCM) are safer for interactions with patients. By decoupling the kinematic chains, each motion of the ultrasound probe is controlled by only one independent joint. This is an interesting avenue to explore, as while higher DOF robots like the Franka Emika offer greater flexibility, it can introduce complexities that might compromise safety within medical settings.

Robot admittance control would likely be the best approach to dynamically influence motion. Admittance control uses a predefined relationship between force and position to regulate the

robot-environment interaction. For example, Carriere et al. (2019) applied admittance control in co-operated robotic sonographies. Similarly, Dimeas and Aspragathos (2016) focused on the stability of admittance control by detecting unstable behaviors and altering gains in real-time, a method also explored by Landi et al. (2017) to stabilize interaction between robots and humans.

Additionally, future experimentation is required for simulating the kinematic-based behavior of the Franka Emika robot using specialized software such as ROS and Gazebo. Moreover, grid optimization can be implemented for the image processing module to identify the best set of parameters for maximization of denoising and improvement of scan quality. Further comparisons with ground truth ultrasound images could also yield more clinically relevant results.

In conclusion, while the project faced challenges in setting up the dynamics and control system for the Franka Emika robot, the lessons learned and progress achieved has laid a solid foundation for future development. By revisiting the control approach, we can work to complete these sections and fully validate the concept of using basic kinematic modeling to guide an ultrasound probe. Once this is achieved, it would pave the way for further advancements, eventually leading to full-scale use in remote clinical settings, with applications in fetal imaging amongst others.

Statement of Contribution

Aly Khan conducted the image post-processing and performed the Franka simulation in the **Results** section using MATLAB/Python code, which can be found in **Appendix #3, 4**. He also wrote the **Introduction** section, contributing to **Methodology**, and **Discussion** as well.

Victor researched similar projects/studies and contributed to the design of the Franka simulation. He also wrote the **Abstract**, **Methodology**, and **Discussion and Conclusions** sections in its entirety as well as minorly contributed to the **Results** section.

Sumaiya and Brandon developed the top level approach for the forward kinematics, inverse kinematics, dynamics, etc. found in the **Results** section. They also wrote the MATLAB code which can be found in **Appendix #1, 2**.

Kindly note that the group transitioned from 5 to 4 members due to one of the students dropping the course midway through the project, which provided a minor setback to our workflow.

References

Akbari, M., Carriere, J., Meyer, T., Sloboda, R., Husain, S., Usmani, N., & Tavakoli, M. (2021). Robotic Ultrasound Scanning With Real-Time Image-Based Force Adjustment: Quick Response for Enabling Physical Distancing During the COVID-19 Pandemic. Frontiers in Robotics and AI, 8. https://doi.org/10.3389/frobt.2021.645424

Carriere, J., Fong, J., Meyer, T., Sloboda, R. S., Husain, S., Usmani, N., & Tavakoli, M. (2019). An Admittance-Controlled Robotic Assistant for Semi-Autonomous Breast Ultrasound Scanning. *2019 International Symposium on Medical Robotics (ISMR)*, pp. 1-7. https://doi.org/10.1109/ismr.2019.8710206

Dimeas, F., & Aspragathos, N. (2016). Online Stability in Human-Robot Cooperation with Admittance Control. *IEEE Transactions on Haptics*, *9*(2), 267–278. https://doi.org/10.1109/TOH.2016.2518670

franka_ros. — Franka Control Interface (FCI) documentation. (n.d.). Frankaemika.github.io. https://frankaemika.github.io/docs/franka_ros.html

Jiang, Z., Salcudean, S. E., & Nassir Navab. (2023). Robotic ultrasound imaging: State-of-the-art and future perspectives. *Medical Image Analysis*, *89*, 102878–102878. https://doi.org/10.1016/j.media.2023.102878

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

Landi, C. T., Ferraguti, F., Sabattini, L., Secchi, C., & Fantuzzi, C. (2017). Admittance control parameter adaptation for physical human-robot interaction. *2017 IEEE International Conference on Robotics and Automation (ICRA)*. https://doi.org/10.1109/icra.2017.7989338

Micks, T., Sue, K., & Rogers, P. (2016). Barriers to point-of-care ultrasound use in rural emergency departments. *CJEM, 18*(6), 475–479. https://doi.org/10.1017/cem.2016.337

Najafi, F., & Sepehri, N. (2011). A robotic wrist for remote ultrasound imaging. *Mechanism and Machine Theory, 46*(8), 1153–1170. https://doi.org/10.1016/j.mechmachtheory.2011.03.002

Robot and interface specifications — Franka Control Interface (FCI) documentation. (n.d.). Frankaemika.github.io. https://frankaemika.github.io/docs/control_parameters.html

Taylor, R. H., & Stoianovici, D. (2003). Medical robotics in computer-integrated surgery. *IEEE Transactions on Robotics and Automation*, *19*(5), 765–781. https://doi.org/10.1109/tra.2003.817058

Trabelsi, A., Sandoval, J., Ghiss, M., & Laribi, M. A. (2021, May 1). *Development of a Franka Emika Cobot Simulator Platform (CSP) Dedicated to Medical Applications*. ResearchGate. https://doi.org/10.1007/978-3-030-75259-0_11

Vander Ende, J. E. C., Labossiere, R. A., & Lawson, J. (2023). Utilisation and barriers of PoCUS in a rural emergency department - A quality improvement project. *Canadian journal of rural medicine : the official journal of the Society of Rural Physicians of Canada = Journal canadien de la medecine rurale : le journal officiel de la Société de médecine rurale du Canada, 28*(4), 170–178. https://doi.org/10.4103/cjrm.cjrm_90_22

Appendix

Appendix #1: Direct Dynamics Parameterization

Please find the following link to access the MATLAB code: directdynamics_parameterization.m

Appendix #2: Direct Dynamics Control

Please find the following link to access the MATLAB code: directdynamics_control.m

Appendix #3: Franka Robot Ultrasound Simulation

Please find the following link to access the MATLAB code: franka_simulation.m

Appendix #4: Image Denoising

Please find the following link to access the Jupyter notebook: image_denoising.ipynb