

ECE 445

Senior Design Laboratory Design Document

A Comprehensive Approach to Tumor Detection using RGB, NIR, and Immersive 3D Visualization

University of Illinois at Urbana-Champaign

Team Members:

- ❖ Zach Mizrachi (zdm3)
- ❖ TJ Shapiro (tylers5)
- ❖ Yue (Amy) He (yuehe4)

TA: Jason Zhang

February 20, 2024

Contents

1	Introduction.....	3
1.1	Problem.....	3
1.2	Solution.....	3
1.3	Visual Aid.....	5
1.4	High-level requirements.....	6
2	Design.....	7
2.1	Block Diagram.....	7
2.2	Casing Module: Physical Design.....	9
2.3	Imaging Module: Data Collection and Stabilization Subsystem.....	10
2.4	Vision Module: Part 1 - Structure From Motion and 3D Modeling.....	13
2.5	Vision Module: Part 2 - Apple Vision Pro Projection Subsystem.....	15
2.6	Tolerance Analysis.....	17
3	Cost and Schedule.....	18
3.1	Cost Analysis.....	18
3.2	Schedule.....	19
4	Ethics and Safety.....	21
4.1	Potential Safety Concerns.....	21
4.2	Mitigating Concerns.....	22
5	Citations.....	22

1 Introduction

1.1 Problem

The prevailing method for tumor removal, traditional surgery, presents significant challenges due to its reliance on the surgeon's visual and tactile feedback, which are prone to human error. This approach often results in suboptimal outcomes due to limitations in tumor visibility and the potential for unintended damage to surrounding tissues. Additionally, traditional surgery may necessitate excessive tissue manipulation to enhance tumor visibility, thereby increasing patient morbidity. Consequently, there is a critical need for innovative solutions that mitigate these limitations and enhance the precision and efficacy of tumor removal procedures.

1.2 Solution

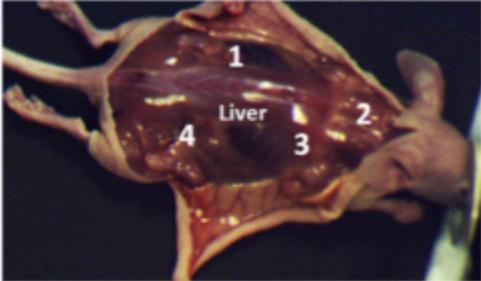
Our proposed solution addresses the limitations of traditional tumor removal procedures by introducing a compact and mobile camera system designed to augment the surgeon's capabilities. The system will focus on two primary tasks: tumor detection through segmentation from surrounding tissue and 3D reconstruction of the identified tumor. To achieve this, we will integrate a tumor-detecting pen system, validated in a pilot study, which utilizes fluorescent drugs to differentiate between tumor and healthy tissue.

In practice, the system will be deployed in conjunction with an Apple Vision Pro to provide real-time visualization of the surgical site. Utilizing augmented reality, the system will highlight the tumor based on near-infrared (NIR) reflection, guiding the surgeon's movements for precise tumor removal. Simultaneously, this augmented reality visualization will facilitate the capture of accurate images for

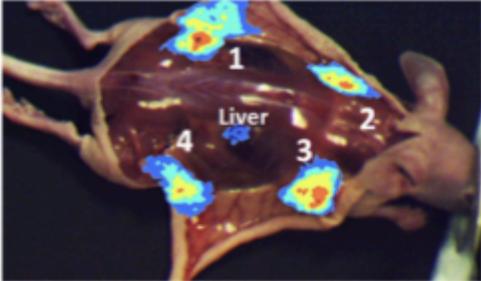
subsequent reconstruction. Post-operatively, advanced image processing techniques will be employed to generate a detailed 3D model of the tumor, providing the surgeon with enhanced visualization and understanding of the surgical area. By improving the surgeon's visual capabilities and facilitating informed decision-making throughout the tumor removal process, our solution aims to enhance surgical outcomes and patient care.

1.3 Visual Aid

NIR + Visible Light Illuminated Object



Tumors Overlayed in Augmented Reality



Tumor
Detection
Device

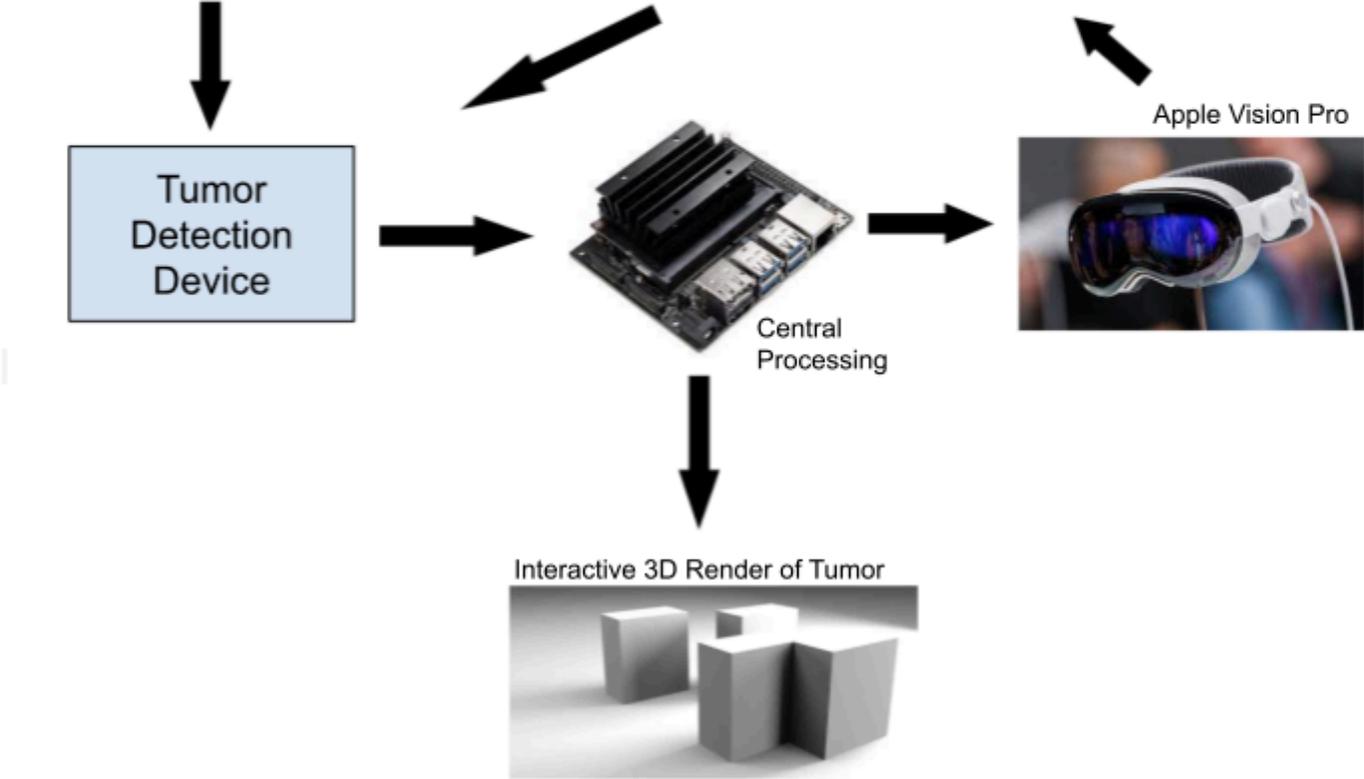
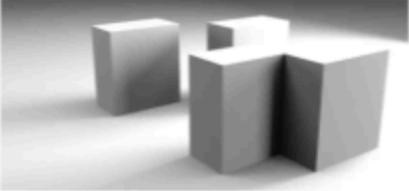


Central
Processing



Apple Vision Pro

Interactive 3D Render of Tumor

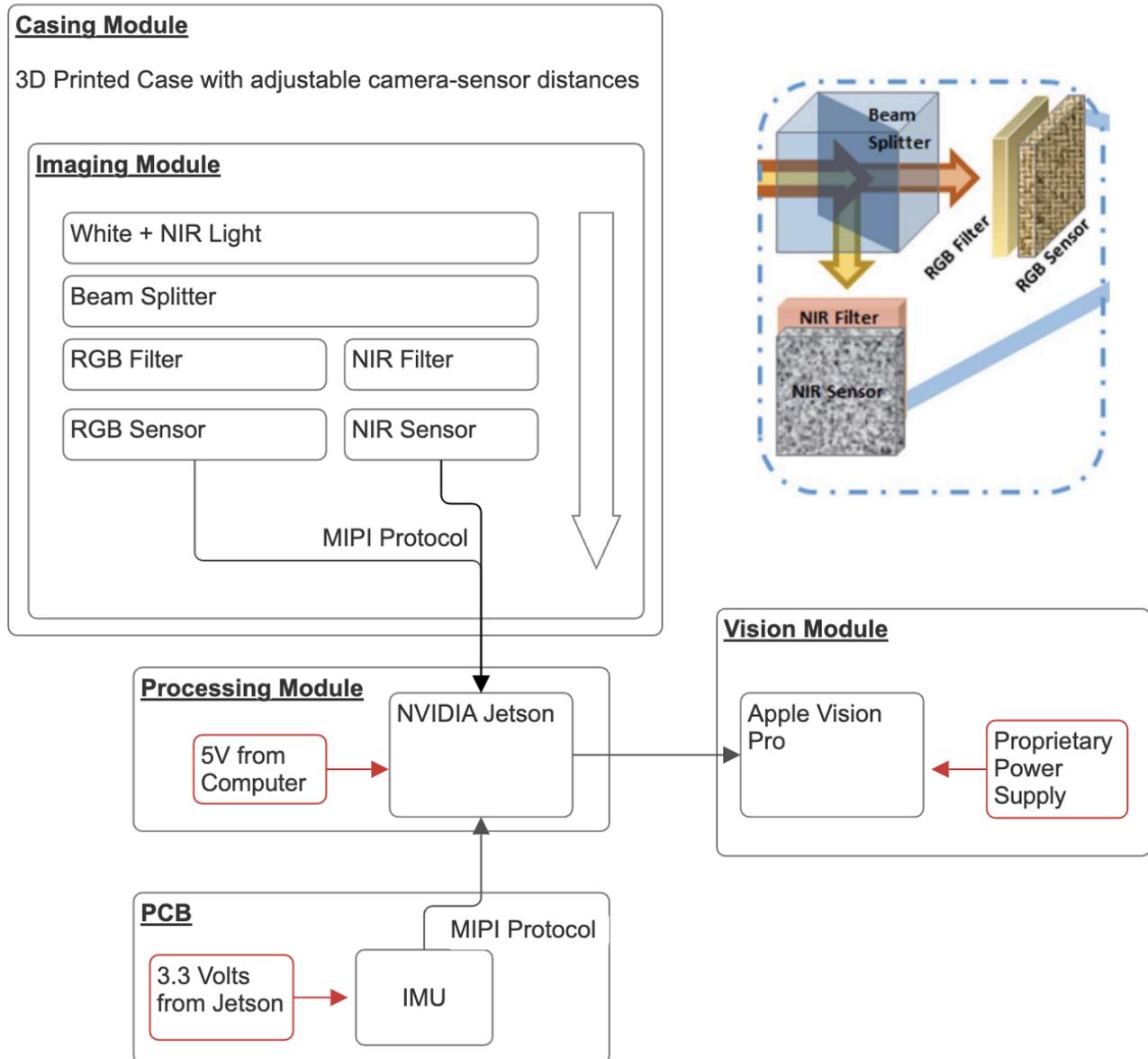


1.4 High-level requirements

The project aims to achieve specific hardware and software requirements for developing a tumor-detecting camera system. Hardware tasks include 3D printing a casing with an adjustable distance from the camera sensors to the beam splitter, assembling electrical components correctly, and integrating an IMU with the PCB. Software objectives involve receiving and validating RGB, NIR, and IMU data on an NVIDIA Jetson, filtering RGB images by the NIR region of interest, and running open Structure-From-Motion software on the Jetson. Additional software goals are to enhance the SFM model using IMU data, perform optimal frame selection with the IMU, and establish communication between the Jetson and Vision Pro for augmented reality tumor highlighting. The system should enable viewing and interaction with the 3D SFM Point Cloud on the Vision Pro, improving surgical visualization and decision-making.

2 Design

2.1 Block Diagram



The casing module will act as the physical container for the sensor suite and the associated electronics.

This casing will allow the surgeon to calibrate the cameras to ensure optimal photos for photogrammetry.

The imaging module is responsible for physically capturing the images, both from visible light and from

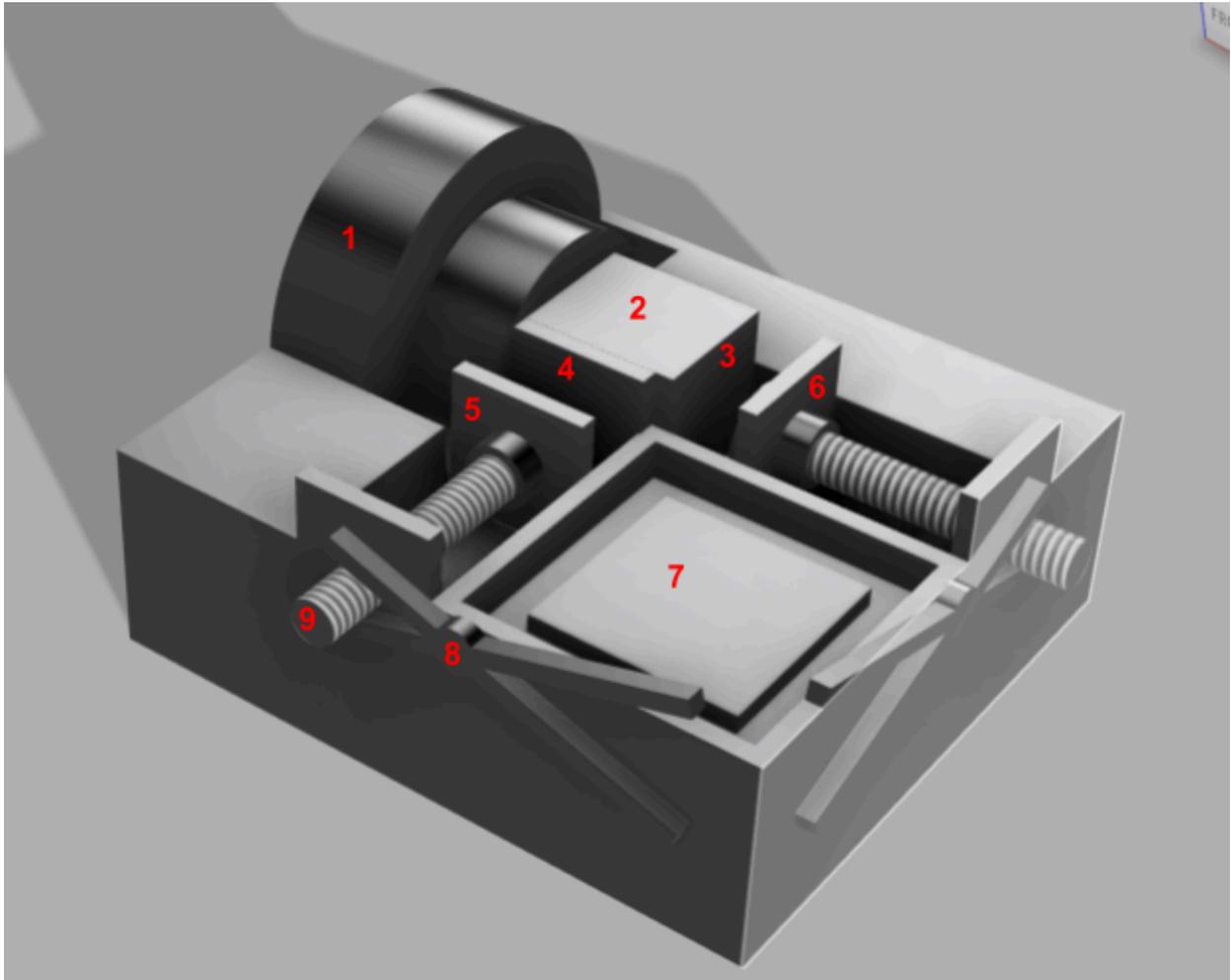
NIR. Further, the PCB module will route the connections from the IMU to an easily accessible output to

be received in the processing module - an NVIDIA Jetson Nano. Lastly, the Vision Module is responsible

for highlighting the tumors in real time via augmented reality, as well as creating a 3D reconstruction of

the tumor via photogrammetry.

2.2 Casing Module: Physical Design



1 - Lens 2 - Beam Splitter 3 - Visible Light Filter 4 - NIR Filter 5 - NIR Sensor

6 - CMOS Sensor 7- PCB + IMU 8 - Clamp 9 - Adjustable Screw

Top lid for enclosing the electronics not shown for viewing

2.3 Imaging Module: Data Collection and Stabilization Subsystem

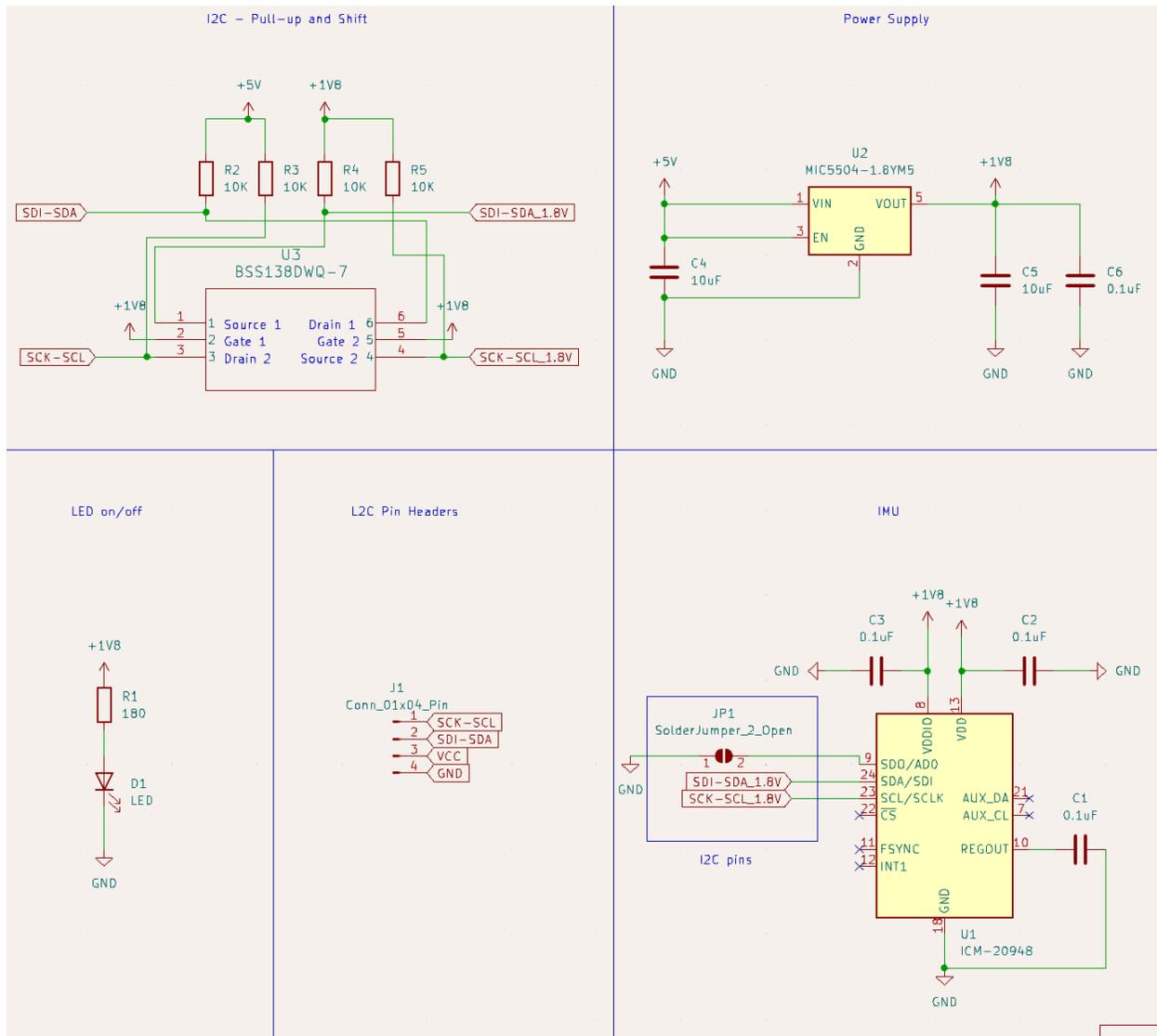
This subsystem is responsible for collecting all necessary data and ensuring the accuracy and validity of this data. Specifically, it consists of two camera sensors, their respective filters, a beam splitter, a lens, and a PCB board with IMU sensor. The beam splitter splits the incoming light into two beams, and the filters select for NIR and visible light, respectively. The output of the system are the NIR and Visible Light Images which are sent to the Jetson Nano over MIPI protocol. The IMU collects the 9-degree-of-freedom (9-DOF) data of the pen, including the orientation estimation (heading, pitch, and roll) in addition to linear acceleration and angular velocity measurements. The IMU output is sent to the Jetson Nano over I2C protocol.

Sensor and Part List

*For all parts marked N/A, we are receiving them from Professor Gruev and do not have specifics yet

Part	Part # and Link	Function	Reason for Purchase
CMOS Sensor	LI-IMX219-MIPI-FF-NANO-H90	Record visible light images	Cheap, communicates over MIPI which is convenient
NIR Sensor	LI-OV5640-MIPI-AF-NIR	Record NIR images	Cheap, MIPI protocol
Light Filter	N/A	Exclude visible light going into NIR sensor	N/A
NIR Filter	N/A	Exclude NIR light going into visible light sensor	N/A
Beam Splitter	N/A	Split incoming beam into two equal beams	N/A
Lens	N/A	Focus incoming light	N/A

		into the beam splitter	
IMU	ICM-20948	Accelerometer, Gyroscope, Magnetometer, 9 Axis Sensor I2C, SPI Output	Can ship immediately Cheap Substitute of MPU9250 (obsolete)



[Schematic zip file](#)

Requirements	Verification
<p><i>CMOS and NIR Sensors:</i></p> <ul style="list-style-type: none"> - Low latency -> we want less than 0.25 seconds of latency from cameras to the Jetson. With processing and then tumor augmentation, we want to create a system that is as close to real time as possible. - Send photos to the Jetson at a minimum rate of 20 fps. 	<ul style="list-style-type: none"> - Both the latency and the framerate can be measured and verified via software
<p><i>I2C interface:</i></p> <ul style="list-style-type: none"> - The internal registers and memory of the IMU (ICM-20948) can be accessed at 400 kHz 	<ul style="list-style-type: none"> ● Set the IMU as a slave device that communicates to the master system processor (Jetson Nano) ● The Jetson Nano (master) should send the slave address of the IMU (ICM-20948), which is b110100X (7 bits). The LSB is Pin 9, AD0, which can be connected to GND since we're only using one device. ● Start of I2C: HIGH-to-LOW transition of the SDA line (Pin 24) while the SCL line (Pin 23) is HIGH. ● Stop of I2C: LOW-to-HIGH transition of the SDA line (Pin 24) while the SCL line (Pin 23) is HIGH.
<p><i>PCB:</i></p> <ul style="list-style-type: none"> - The SDA and SCL lines typically need pull-up resistors to VDD. - The voltage regulator should decrease the input voltage from 5V to 1.8V for the IMU chip to operate. - The logic voltages of SCL and SDA should be 1.8V because we are using a BSS138DWQ-7 which consists of a pair of MOSFETs that allow pull-up and shift. 	<ul style="list-style-type: none"> ● Use a MOSFET (BSS138DWQ) to implement pull-up resistors as indicated in the schematic. ● Use a voltage regulator as indicated in the schematic to turn 5V to 1.8V.

2.4 Vision Module: Part 1 - Structure From Motion and 3D Modeling

In this subsystem, we apply software techniques to take our 2D images as described in subsystem 2.3 and fuse them to create dense 3D models of the desired images. We plan on utilizing open source photogrammetry software such as [OpenSFM](#) to complete this task.

OpenSFM is an extremely popular structure-from-motion library that is well-documented and supports both large reconstruction projects such as drone based environment modeling as well as dense small-scale modeling as in this study. It will be necessary to edit opensfm.config with our camera parameters as well as algorithmic configurations for small-scale detections.

We also plan on utilizing the IMU to improve the accuracy of the standard OpenSFM reconstruction in two main ways. First, we plan on using the IMU accelerometer values to filter images that are sent into the SFM modeling. This way, any images that come in “blurry” can be filtered out, and high resolution images will be primarily used. Second, by providing roll, pitch, and yaw angles from the IMU directly, we hope to increase the accuracy of the generated model. Within OpenSFM, this can be accomplished as below:

```
org = pose.get_origin()      # save where the camera actually is
pose.set_rotation_matrix(R)  # set the rotation
pose.set_origin(org)        # update the translation property accordingly...
```

Once 3D models are created in OpenSFM, the next step is to visualize and interact with these models in Apple Vision Pro. The 3D reconstruction will be converted to a format the VisionOS system can recognize (such as USDZ, converting via Reality Converter or USDZ Tools). We then will import the resulting USDZ file into Xcode, and import the Vision

framework to write code to load the USDZ model. Finally, we will utilize the Vision framework's functions to reproduce and interact with the 3D model.

Requirements	Verification
OpenSFM completes the model in less than 20 seconds. <ul style="list-style-type: none">- In a real world scenario, the surgeon needs immediate access to the models	We will measure completion time via software flags.
Compete with other 3D modeling approaches in low reconstruction error (~1mm of reconstruction error as compared to benchmark)	We will utilize OpenSFM's commands <code>compute_statistics</code> and <code>export_report</code> to generate a JSON statistics file and a PDF report, containing all relevant information
Utilize the IMU roll, pitch, and yaw to manually provide camera pose information. This must improve reconstruction error as compared to the standard approach.	We will compare the output of the above script to determine improvements in reconstruction error and camera drift.

2.5 Vision Module: Part 2 - Apple Vision Pro Projection Subsystem

This subsystem is responsible for precise projection of the 3D reconstructed tumors and overlaying them in augmented reality. This will help guide the surgeon in discovering tumorous regions, which will assist in This will be done using Apple’s proprietary VisionOS platform, designed for the Apple Vision Pro mixed reality head-mounted display (HMD). Specifically, we will be utilizing the Unity3D engine as well as Apple’s XCode and ARKit APIs to take advantage of their seamless plane estimation, image anchoring, and world tracking tools. Every instance the Jetson Nano produces an updated 3D reconstruction (every T time), the asset will be loaded into Unity to be projected in AR overlaying the exact location of the patient the doctor is operating on. The user will be able to interact with the 3D object to more closely analyze the size and nature of the tumor. Metrics will also be visible that display the accuracy of the photogrammetric reconstruction and location. In order to accomplish this in semi-real time, a MacBook is necessary to both collect data from the Jetson Nano and generate the mixed reality scene. To properly ensure that this projection subsystem is fully fulfilling its responsibilities for successful loading of 3D objects, accurate mapping/overlay of objects on patient, and upholding reliable data transmission speeds, see the requirements and verification table below.

Requirements	Verification
<ul style="list-style-type: none">• The overlay/reconstruction must allow the user to have a completely unobstructed view of any part of the patient not classified as a tumor.	<ul style="list-style-type: none">• Ensure that during any instance where image anchoring/plane estimation fails, the model is not directly obstructing user’s view. This may involve temporarily removing the model in certain circumstances.• Any model of a tumor that has less than 80% accuracy from the Jetson

	<p>should be discarded (or simply not displayed). The new iteration of the 3D model would be rendered when the next model information is received. Until then, previous acceptable model would be displayed.</p> <ul style="list-style-type: none"> •
<ul style="list-style-type: none"> • Scene tracking and plane estimation must be reliable under expected lighting conditions. 	<ul style="list-style-type: none"> • Ensure when spot LEDs are on, the VisionOS app is still able to track important room features. This includes verification that the plane estimation/image anchoring is not experiencing any abnormal glitches or inconsistent draws. • Then, turn LEDs off and turn the NIR lights on. Ensure that plane estimation, room tracking, and image anchoring are not negatively affected in any way. Lighting should not interfere with the Vision Pro's sensors or plane estimation capabilities. • Then, turn both LEDs and NIR lights on and test to make sure all tracking capabilities as stated above are also fully functional under these conditions. Note: this is the final testing setup as this is what would be used in a normal operating room scenario.
<ul style="list-style-type: none"> • Must receive/process 3D object data from Jetson Nano with a latency of 2s +/- 500ms. 	<ul style="list-style-type: none"> • The maximum CPU frequency of the Jetson Nano is approximately 1.43Ghz and the GPU max frequency is 921MHz. This coupled with the 472 GFLOPS AI performance should be more than enough to generate 3d models at this speed depending on the size and resolution of the images. • To verify that we can transmit 3D models from the Jetson to the Mac device, we will provide two metrics alongside the object file: accuracy and latency.
<ul style="list-style-type: none"> • The error rate for successful 3D 	<ul style="list-style-type: none"> • Here, we must define what is

<p>reconstruction must be <85% in any given user session.</p>	<p>categorized as a successful 3D reconstruction: Latency must be adhering to the requirement as stated above. A 3D object must be composed of a collection of stable, clear, and well-positioned images (accuracy metric will be provided).</p> <ul style="list-style-type: none">● A running average metric will be provided for both latency and accuracy (using structure for motion). Whenever either latency or accuracy fails requirements, the model is labeled as a fail and discarded.● If at any point after 10 models this running average falls below 85%, this requirement has not been satisfied.
--	---

2.6 Tolerance Analysis

The IMU is the main component which is subject to failure, given we are creating a custom PCB and not relying on standard MIPI protocol such as for the NIR/CMOS Sensors, or open source libraries like OpenSFM.

The following formulas define how the acceleration and angular rate values that we will extract from the IMU are derived:

```

X_acceleration = ACCEL_XOUT/Accel_Sensitivity
Y_acceleration = ACCEL_YOUT/Accel_Sensitivity
Z_acceleration = ACCEL_ZOUT/Accel_Sensitivity
X_angular_rate = GYRO_XOUT/Gyro_Sensitivity
Y_angular_rate = GYRO_YOUT/Gyro_Sensitivity
Z_angular_rate = GYRO_ZOUT/Gyro_Sensitivity

```

According to the ICM-20948 datasheet, we can expect an initial tolerance of plus or minus 0.5% across all ACCEL_OUT values. It will be imperative to compare this error produced as a result of this tolerance to the reprojection error obtained without using the IMU sensor for OpenSFM.

3 Cost and Schedule

3.1 Cost Analysis

description, manufacturer, part #, quantity and cost

Note: N/A are parts we are given from Professor Gruev

Description	Manufacturer	Part # and Link	Quantity	Cost
CMOS Sensor		LI-IMX219-MIP I-FF-NANO-H9 0	1	\$29
NIR Sensor		LI-OV5640-MIP I-AF-NIR	1	\$45

Light Filter	N/A	N/A	1	N/A
NIR Filter	N/A	N/A	1	N/A
Beam Splitter	N/A	N/A	1	N/A
Lens		N/A	1	N/A
IMU		ICM-20948	1	~\$8.64
Casing	Proprietary	N/A	1	\$10-20 based on filament pricing

3.2 Schedule

Week	Task	Member(s) Assigned	Notes
1 (week of 1/15)	Web Board Post	Individual	
2 (week of 1/22)	Lab Safety/CAD	Individual	
3 (week of 1/29)	Project approval	All	
4 (week of 2/5)	Soldering	Individual	
5 (week of 2/12)	Meet with Prof. Gruev	All	1st discussion with Professor Gruev about scope and specific components of project.
6 (week of 2/19)	Complete design document/order parts	All	
7 (week of 2/26)	Design Review/PCB Cad/explore SFM	All	
8 (week of 3/4)	Get data collection subsystem prototype started/explore photogrammetry and OpenCV algorithms	Zach - CAD/SFM Amy - PCB/Photogrammetry TJ-Jetson OpenCV	
9 (SPRING BREAK)	SPRING BREAK	SPRING BREAK	SPRING BREAK

10 (week of 3/18)	Start introducing training and testing data for identification model	TJ - Jetson CUDA/OpenCV Zach-SFM Amy - OpenCV	
11 (week of 3/25)	Progress Reports/Start Projecting with VisionOS	VisionOS pipeline - TJ Amy - 3D Reconstruction Zach - 3D Reconstruction	
12 (week of 4/1)	Start streamlining full pipeline/improving model	All	
12 (week of 4/8)	Reliability and consistency testing	All	
14 (week of 4/15)	Final touches/preparing for final demo	All	
15 (week of 4/22)	Final Demo/Prepare for Final Presentation	All	
16 (week of 4/29)	Final Presentation/Final Papers	All	

4 Ethics and Safety

4.1 Potential Safety Concerns

Our ethical and safety framework prioritizes patient welfare, equitable access to innovation, and safety throughout the pen's development and deployment.

Patient Autonomy: Ensuring informed consent, patients understand the purpose, risks, and benefits of the tumor-detecting pen.

Privacy: We securely manage patient data, adhering to regulations like HIPAA.

Transparency: We maintain a lab notebook documenting the design process and progress, fostering trust by openly communicating the pen's capabilities, limitations, and risks.

Clinical Validation: Rigorous testing ensures the pen's safety, accuracy, and effectiveness before clinical use.

Regulatory Compliance: Adherence to medical device regulations, obtaining FDA approval to ensure pen safety and efficacy.

User Training: Comprehensive training for surgeons and healthcare professionals minimizes errors during procedures.

Continuous Monitoring: Mechanisms for ongoing surveillance promptly address safety concerns during pen use in clinical practice.

4.2 Mitigating Concerns

Start with battery discharged to the point where it reads 3.4V. Apply 5V input via power supply to input of BMS. Monitor current delivered to battery w/ multimeter and monitor battery voltage. Verify that at the end of charge cycle, battery voltage is 4.2V and no current is being delivered to the battery. Reco

5 Citations

- [1] B. Mondal, S., Gao, S., Zhu, N. et al. *Binocular Goggle Augmented Imaging and Navigation System provides real-time fluorescence image guidance for tumor resection and sentinel lymph node mapping*. *Sci Rep* 5, 12117 (2015). <https://doi.org/10.1038/srep12117>
- [2] Jetson modules, support, ecosystem, and lineup. *NVIDIA Jetson Developer Modules* [online]. [Accessed 22 February 2024]. Available from: <https://developer.nvidia.com/embedded/jetson-modules>
- [3] Jetson User Guide. *NVIDIA Developer User Guide* [online]. [Accessed 22 February 2024]. Available from: <https://developer.nvidia.com/embedded/downloads#?search=Jetson%20Nano%20Developer%20Kit%20User%20Guide>
- [4] VisionOS Documentation. *Apple Vision Pro Developer Documentation* [online]. [Accessed 22 February 2024]. Available from: <https://developer.apple.com/documentation/visionos>
- [5] *Apple Developer Documentation* [online]. [Accessed 22 February 2024]. Available from: <https://developer.apple.com/documentation/>
- [6] OpenSfM. *OpenSfM Documentation* [online]. [Accessed 22 February 2024]. Available from: <https://opensfm.org/docs/>