Cleanbot 3000

Joshua Garner, Haibert Hatamian, Menua Aghajanian, Arnold Ortega, George Flores, Jesus Alvarez, Justin Ngheim, John Bedikian, Arin Barouni, Freddy Nunez, Vahe Boghosian, Harut Ketsoyan, Brian Somundjian, Cristian Miramontes, Waheedah Akberzie, and Shiban Radhanpuri
Advisor: Professor James Flynn
California State University Northridge
April 22, 2020

***Abstract* — This paper presents the considerations, progress, and research in the design of an autonomous sanitization robot, the Cleanbot 3000. The Cleanbot 3000 has the primary objective of safely navigating and sanitizing cleanrooms while adhering to design requirements which take into account ease of use, efficiency, and standards placed by regulatory bodies. The team charged with the development of the Cleanbot were split into four different groups. The navigation team focused on researching and testing potential navigation algorithms which the robot could use to navigate the cleanrooms. The trident reflector team was a group which supplemented navigation as the team sought to develop a system through the use of artificial landmarks which would provide the Cleanbot with a method of finding its absolute position within any room regardless of day to day changes. The sensor team looked through various potential sensors which the Cleanbot would use and made decisions on which was the best based on their specifications. Finally, the sanitization and power team focused on researching the efficiency and use of UVC LEDs which the Cleanbot would use to sanitize the rooms. Additionally, the team researched measures which would be taken in order to meet the power requirements of the Cleanbot's numerous possible components.**

Keywords — navigation, guidance reflectors, sanitization, sensors.

1. Introduction

The Cleanbot 3000, or the Cleanbot as it will be further referred to, is tasked with sanitizing cleanrooms at the Jet Propulsion Laboratory (JPL) through the use of UVC LEDs. The primary task is to deactivate microbial life which could cause potential environmental harm should these organisms latch onto any spacecraft and be taken into the upper atmosphere and beyond. The concern of contamination and the survival of microorganisms through space transit is valid and has been proven to be a possibility [1], thus an autonomous sanitization system such as the Cleanbot which would deactivate these organisms should be considered.

This paper will discuss the progress made by the Cleanbot team's four sub-groups: Navigation, Trident Reflectors, Sensors, and Sanitization/Power.

The primary aim of these sub-groups is to primarily lay a foundation through research that can be used in decision-making on what would be the most optimal design choices for the Cleanbot moving forward as well as to conduct tests which could validate ideas or simulate scenarios which the Cleanbot will eventually find itself in.

2. Navigation

2.1 SLAM Mapping Algorithm

Simultaneous Localization and Mapping (SLAM) is a process in which a mobile robot can build a map of its environment and at the same time be able to realize its own location.SLAM has been in use since the 1980's, but has not found success in robotics as much as it has as of late. In SLAM, both the trajectory of the platform and the location of all landmarks are estimated on-line without the need of prior knowledge of the location. Just like humans, a robot cannot always rely on a GPS system. A GPS system's degree of certainty is vastly lowered once indoors and cannot be as reliable as it is outside because precision within a few inches is required to move about safely. This is why a system such as SLAM is needed; the basis of SLAM is that it is possible for an autonomous vehicle to start in an unknown location in an unknown environment and, using relative observations only, incrementally build a perfect map of the world and to compute simultaneously a bounded estimate of vehicle location [2].

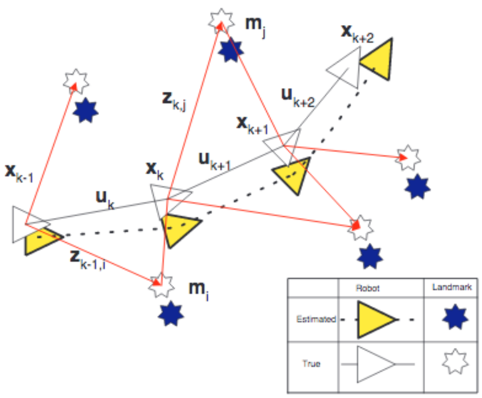

Figure 2.1: Basic Outline of a Robots tracking with estimation

With that in mind, the robot would be able to build its map on the go and keep track of its position by aligning the sensor data it collects with whatever sensor data it has already collected in order to build out a map.SLAM is a multi-stage process that includes alignment of sensor data using a variety of algorithms well suited for the parallel processing capabilities of GPUs.

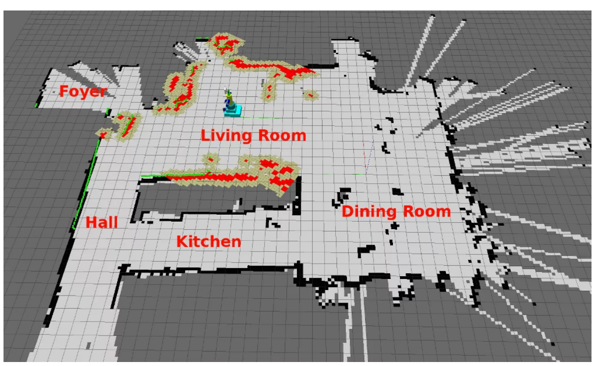

Figure 2.2: Full Mapping of a portion of a house using SLAM

There are many different versions of SLAM available, but the ones which will be discussed are Hector SLAM, and Cartographer ROS. They can all be accessed through a Robotic Operating System (ROS) library which is the main source for the use of SLAM. The most basic of the three is Hector SLAM; it is a SLAM approach that can be used without odometry, It can also be used on platforms that exhibit a roll/pitch motion. It supports the high update rate of modern LIDARs and provides 2D orientation estimates at the scan rate of the sensors.

Although Hector SLAM does not provide loop closing abilities, it is more than accurate enough to be used in real world scenarios such as the Cleanbot..

Figure 2.3: Hector SLAM mapping

However, Cartographer ROS is the more advanced system compared to the Hector SLAM because it is a system that is able to provide real time SLAM in a 2D and 3D across multiple platforms and sensors. Its main advantage over Hector SLAM is its use of local and global mapping.

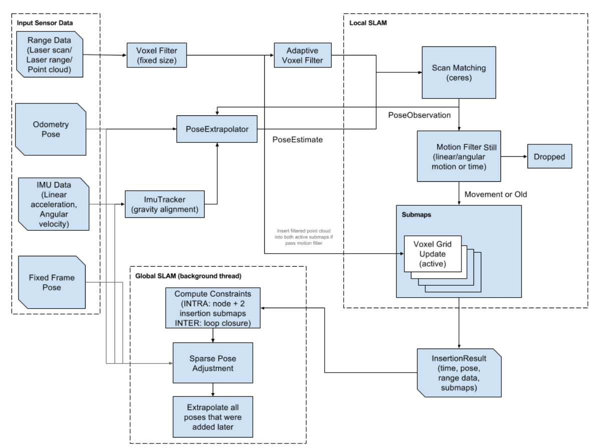

Figure 2.4: Visual Diagram of how the two Sub-systems operate in Cartographer

The use of Local SLAM(local mapping) is the same basic principle as regular SLAM, it uses the raw laser data from the sensor and begins mapping. For Local SLAM(also referred to as trajectory builder), the main objective is to create a succession of submaps for the robot to use.

Each submap is supposed to be locally consistent and will eventually drift over time. This is where Global SLAM(Global mapping) comes in. As Local SLAM is operating, Global SLAM operates in the background of the system and rearranges these submaps together to get a complete cohesive global map. It also utilizes other sensor data in order to get a high level view and identify the most accurate global map.

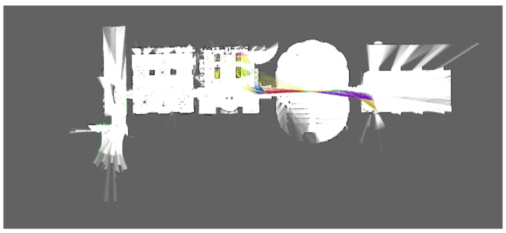

Figure 2.5: 2D Mapping using Cartographer

Figure 2.6 3D/FPOV visualization of the map

Although both of these versions of SLAM would be more than satisfactory for use in the finished product, there is a clear winner. As stated, Hector SLAM has been used in various unmanned robotic projects and can be easily used. When comparing it to Cartographer It seems more useful as a stepping stone in preparing for the final configurations of the project. With the combination of the 3D mapping and two-part mapping system Cartographer is the more promising system to be used in the long run.

2.2 Extracting Raw Sensor Data

For the navigation group to obtain a better understanding as well as hands on experience when it comes to building and navigating a robot, the team decided to start working with two raspberry pi cars. The Pi Car-S is a smart car which can operate with Raspberry Pi model 4B, 3B+, 3B, 2B+, and 2B. It works with three sensor modules which include ultrasonic obstacle avoidance, light follower, and line follower. Python code is provided for the car, and it can also be programmed and debugged with Dragit, a Snap-based graphical interface through dragging and dropping code blocks. Through this system, mapping algorithms and sensors can be tested as well.

In order to lay a foundation for a mapping algorithm for the Cleanbot, a bridge between the sensor and the code must be built. To achieve this, a livestream of the raw laser range and angle data must be extracted as it is required for SLAM. This acquired data must then be fed into the Raspberry Pi 3B+ as well as onto the operating system, Debian Buster Raspbian. Multiple approaches were taken in order to achieve this task.

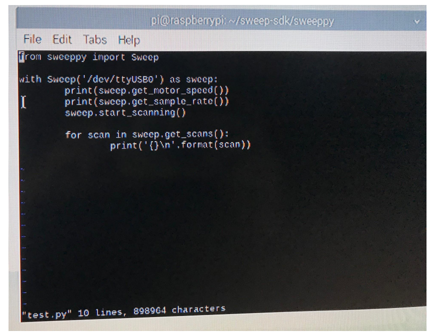

Figure 2.7: Code for retrieving raw sensor data. After executing our python program we were able to stream live sensor data onto the terminal of the Raspberry Pi. The data consists of distance measurements (in cm) and angle measurements (1/1000 of a degree).

First, in order to use the Scanse Sweep LIDAR, the Sweep SDK repository was installed from Github onto the Raspberry Pi. CMake was then installed onto a Linux device in order to build an SDK repository. Once this was completed we navigated through our SDK repository and built the repository using CMake in the “libsweep” directory. We then setup and installed python 3 with our SDK repository through the “sweeppy” directory. Once this was completed we connected the Scanse LIDAR to the Raspberry Pi and wrote a python program to retrieve the live data as shown in **Figure 2.1**. Raw data which was achieved through this extraction process is shown in **Figure 2.2.** A visual representation of this raw data and tests conducted specifically with the Scanse Sweep LIDAR alone is discussed in the “Sensors” section.

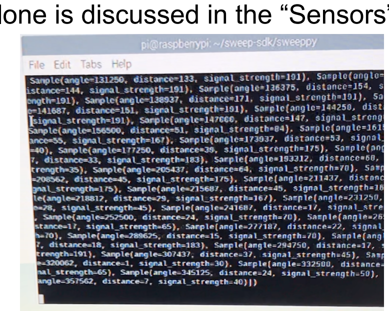

Figure 2.8: Raw sensor data achieved through the process.

3. Trident Reflectors

The Trident Reflectors, as they have been named, is a navigation system formulated in order to achieve localization within the indoor environment. Due to the ever-changing environment of a cleanroom, it is difficult for the Cleanbot to determine its positioning through the use of landmarks naturally found within the room. Due to this, it is necessary to employ the use of artificial landmarks, reflective prongs, which will be attached onto the walls of the cleanrooms that the robot can use as reference points. In combination with a camera, an image of the reflectors can be analyzed in a way in which the position of the viewer can be determined.

3.1 Proof of Concept

Proofs of concept were created to serve as foundations for the idea of the Trident Reflectors; that is, the use of pixel information on the 2D world, the image plane, and it's transformation into inferences, and eventual estimation of position in the 3D world. It is also to be used to test ideas relating to the trident system.

Figure 4.1 shows how the 2D triangle formed by the prongs transforms based on the movement of the viewer, allowing for an inference of the direction of movement relative to a reference point. This is an example of one of many uses of the trident system.

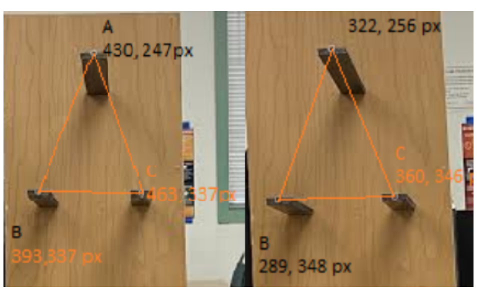

Figure 3.1: An example of a test using the physical proof of concept. Reference photo (left) is compared to photo after movement (right).

The guidance reflector demos further illustrate the transformation of a 2D image into the 3D world. While the plots are currently created using selected points in the 2D plane and 3D plane, it can be modified with equations from the pinhole camera model in order to simulate a real world camera and determine positioning. Figures 4.2a and 4,2b show the plots of the demos.

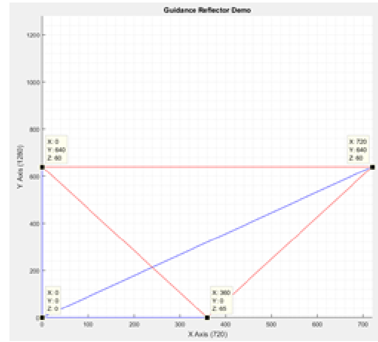

Figure 3.2a: 2D plot of the guidance reflector demo.

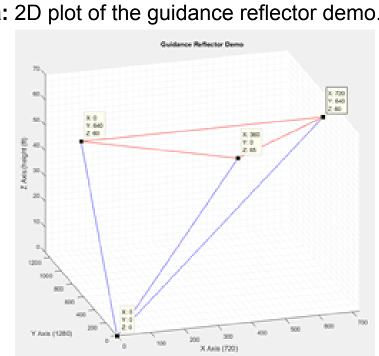

Figure3.2b: 3D plot of the guidance reflector demo.

3.2 The Pinhole Model

In order to better understand the relationship between a given point in 3D space and its equivalent projection on a 2D plane, the Pinhole model, a type of Perspective Projection, was observed. The Pinhole Model is a mathematical model that relates a specific point (P) within a 3D space to its projected point (p') on a 2D plane. In this case, the 2D plane considered is the image plane of the camera. In its simplest form, it can be denoted as such:

$$x / f = X / d \quad [3.1]$$

Where:

x = size (in pixels squared, px^2) of the object on the camera sensor

f = Focal Length of the camera sensor

X = actual size (in meters cubed, m^3) of the object

d = Distance from the camera nodal point to the object

Unfortunately, the simplicity of this model does not take into account much of the camera's parameters, possible image distortions, and other errors that can result in incorrect calculations. Thus, a more functional and practical model is required to derive the relationship between a point in 3D space and its 2D projection on the image plane. Further research led to the Camera Pinhole Matrix.

The Camera Pinhole Matrix is a mathematical model that more accurately models the relationship expressed between a point in 3D space and its projection onto the camera image plane.

To better understand the Camera Pinhole Matrix, a brief introduction to the camera and its basic operation will be detailed below.

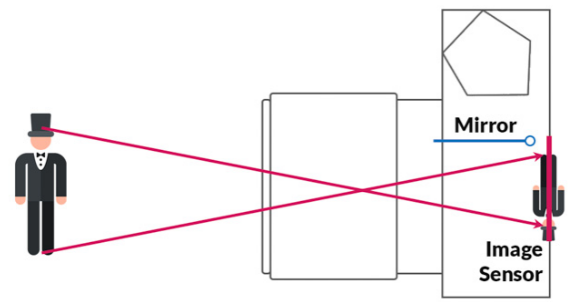

Figure 3.3: Simple diagram of a camera capturing an image

A camera captures an image by collecting rays of light that enter through its lens and converge to a single point. This point can be noted by the intersection of the two red lines in Figure 3.3. This point is known as the Center of Projection. The light rays are inverted through the lens and pass onto the Image Sensor. The "upside down" image is inverted using a mirror and the pentaprism reflector within the camera (Also seen in Figure 3.3). The distance between the center of projection and the image sensor is known as the Focal Length. The focal length is usually measured in millimeters (mm) [6]. The focal length is crucial in determining the relationship between the image plane and the 3D space. Next, an illustration of the physical relationship between the camera image plane and an object in 3D space is observed [8].

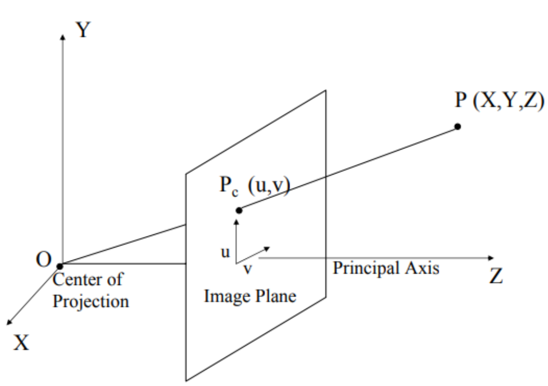

Figure 3.4 : Perspective Projection of a point (P, in a 3D space) onto the image plane (P_C, on a 2D plane).

In Figure 3.4:

- (X,Y,Z)-axis = 3D Coordinate system
- O = Center of Projection, or the center of the camera lense as mentioned in the section above.
- Image Plane = The projected image of the 3D space.
- (u,v)-axis = Image Plane Coordinate system.
- P = (X,Y,Z) . Point/Object in 3D space
- P_C = (u,v) . Projection of point 'P' onto the image plane.

Observing this diagram through the YZ Plane will produce the following:

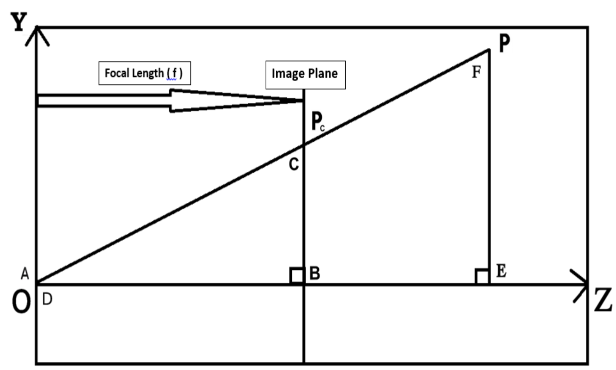

Figure 3.5: YZ Plane of the perspective projection. Observing the similar triangles.

The YZ Plane brings about two similar triangles, ΔABC and ΔDEF. This allows for the relationship between the similar sides to be defined as:

$$AB / DE = BC / EF \qquad [3.2]$$

Where:

AB = f (Focal Length)

From Figure 3.4, it was noted that Point 'P' was located in a 3D space at the location (X,Y,Z), while the projection of 'P', 'P_C', is located at (u,v) on the image plane. Using the knowledge from Figure 3.4 and Figure 3.5, the following similar sides can be derived [7]:

$$f / Z = u / X = v / Y \qquad [3.3]$$

The following can be re-written in matrix form using the camera-focal-length (f) and object-distance-from- camera (Z) relationship:

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \qquad [3.4]$$

Where the point P_C = (u,v,w) = ([fX/Z], [fY/Z], 1). w is just a "dummy-variable" used to ensure the proper matrix multiplication can be completed [8]. In most cases, the Z-axis will not directly intersect the image plane at the principal point 'o'. For this, a translation of 'u' and 'v' must be accounted for. In addition to this translation, the pixel aspect ratio (PAR) must also be considered, as camera resolutions are not always defined by square pixels. This results in what is known as the Intrinsic (Camera) Parameter Matrix.

$$\begin{pmatrix} \alpha_x & s & u_o \\ 0 & \alpha_y & v_o \\ 0 & 0 & 1 \end{pmatrix} \quad [3.5]$$

Where:

- α_x and α_y are the scaling along the U-axis and V-axis of the image plane respectively.
- s is the camera Skew (s = 0 in most cases)
- u_o and v_o are the coordinates of the principle point, the intersection between the principal axis and the image plane [8].

For the time being, this matrix will be referred to as '**K**'. Thus,

$$\mathbf{P_c \approx KP} \quad [3.6]$$

This equation will hold if the center of projection is at point (0,0,0). If this is not the case, then a rotation and translation is needed to adjust the camera coordinate system. This will require more research to acquire the proper formula for calculations.

As of this semester, this is the method deemed most likely to help achieve the desired results from the Trident reflectors and camera sensor. The goal will to test the validity

4. Sensors

Two types of potential sensors for the Cleanbot were researched: light detection and ranging (LIDAR), and ultrasonic. The current plan is to have a single LIDAR unit which will serve as the primary sensor. LIDAR was chosen above other sensors after extensive research on several different types of sensors and models. It was concluded as the most applicable sensor as it provides the necessary data outputs in order to satisfy both obstacle avoidance and mapping. It is also compatible with SLAM as was discussed in the section prior. Ultrasonic sensors, on the other hand, were chosen in order to serve the function of being a backup measure for obstacle detection should the LIDAR malfunction or run into any issues.

4.1 Sensor Models

The specific LIDAR model which was chosen is the OS - GEN1- 16 from Ouster. According to its specifications, the OS - GEN1 LIDAR is able to provide a 0.8-120m range with a 360° horizontal field of view and 33.2° vertical field of view. As staying one meter away from all flight hardware is one of the most important requirements for the Cleanbot, the 360° horizontal field of view and 33.2° vertical field of view specifications played a major role in the decision to select this specific model as these are considered to be features. The maximum range of the LIDAR is also 120m. While excessive this maximum range may be subjected to change due to the reflectivity of the objects that it is scanning. **Figure 4.1** below provides a specification matrix of the Ouster model and two other lidar sensor models we have considered. The Ouster model provides us with the best measurement range, vertical field of view, and reasonable weight. Monetary costs were also considered, as shown, but it is of less importance.

Sensor Model	Range (m)	Field of View	Power (W)	Weight (g)	Price
Ouster OS1	White object: 120 Black object: 60	Vertical: 33.2° Horizontal: 360°	14-20	396	$3,850
Velodyne Puck	100	Vertical: 30° Horizontal: 360°	8	830	~$3,000
RPLidar S1	White object: 40 Black object: 10	Vertical: N/A Horizontal: 360°	2.75	105	$319

Figure 4.1 : Lidar sensor matrix including multiple models.

In order to provide the Cleanbot with a back up system, the use of ultrasonic sensors were researched. With ultrasonic sensors, the Cleanbot would receive a shorter range of data compared to the LIDAR, but would be able to indicate darker objects which the LIDAR has difficulty doing. Also, the short range data provided from ultrasonic sensors would provide information of any immediate objects within the 0,8m range of the Cleanbot which the LIDAR may not detect. The MB1240 XL-MaxSonar-EZ4 manufactured by Maxboltix is the sensor which was chosen. This sensor will provide a range of 0.26m-7.76m with a 38° horizontal field of view. In order to encompass an entire 360° view, ten sensors will be used. This method of configuring the sensors in a multi-sensor system has been confirmed by the manufacturers to be possible. Similarly, it is confirmed that these sensors are compatible with Arduino and Raspberry Pi.

Sensor Model	MB1240 Max Sonar
Range (m)	0.25-7.65
Field of View	Horizontal: 38°
Power Consumption (W)	0.0187
Price	$32.95

Figure 4.2 : Specifications of the ultrasonic sensor.

4.2 Sensor Output Testing

The sensor team received and tested the Sweep v1.0 Scanse LIDAR mentioned in the Navigation section. Unfortunately, the LIDAR being tested had been discontinued by the company which manufactured it leading to difficulties in finding documentation and support. However, older versions of the software were recovered and the sensor was able to be used. The data files were also able to be converted into a (.py) extension which meant that it was usable for mapping.

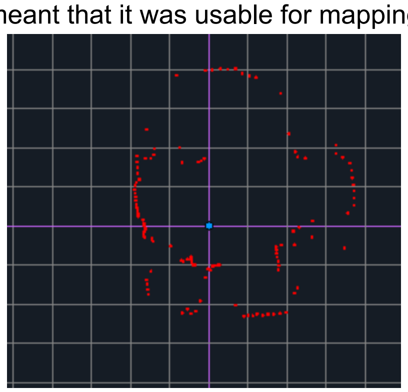

Figure 4.3 : Scanse lidar scanning a small room.

Figure 4.3 shows the data output from the software. It is displaying an outline of a small lab room. Each division resembles one meter and the blue dot indicates the position of the sensor. Near the top left and right of the image, it was discovered that not many data points were displayed due to the presence of black cabinets. Although the projected range of the sensor was 40m, it was unable to read dark objects from as little as 4m away. However, the Ouster model will be more efficient as it is able to read objects with 10% reflectivity compared to this Scanse LIDAR's 75%. Through testing the Scanse LIDAR, the sensor team was able to learn how to configure and output data from the sensor.

As for the ultrasonic sensors, simulations were ran

As indicated in the figure above, the data output from the software is displaying a small lab room's outline. Each division resembles one meter and the blue dot indicates the sensor. Near the top left and right of the image not many data points were displayed due to there being black cabinets. Although the projected range of the sensor was 40m, it was unable to read dark objects about four meters away. However, the Ouster model will be more efficient, as it may read objects with 10% reflectivity compared to the Scanse's 75%. By testing the Scanse model, we were able to learn how to configure and output data from the sensor.

5. Sanitization & Power

To clean the cleanrooms that meet ISO standards involves destroying microscopic organisms from bacteria to fungi. Given the power constraints, our best method is relying on UVC LEDs with light that ranges around 260-275nm. The goal would be to apply a dose that would kill the majority of microorganisms as cleanrooms start to become cleaner and JPL becomes more worried about planetary protection in their pursuit to discover life. The advantage is, this cleaning process is done by an autonomous machine that has little to no outgassing and humans are the greatest threat to cleanrooms even under protective gear.

In the previous semesters, a model was created with values that could easily be changed. After several months of JPL's testing we have received the information desired to fill these models with actual values.

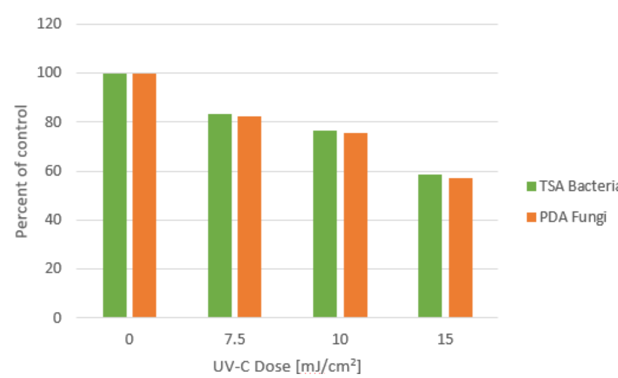

Figure 5.1: Chart of different dosages and their effect provided to us by JPL

The original hope, now outdated, was to clean 90% of microorganisms with a dosage of

3mJ/cm^2. The update was definitely a setback, that had us pursue new LEDs, and much higher capacity batteries. After contacting Klaran, the LED company, we were informed that the new LED, the KL265-50U-SM-WD, would provide an output of 60-70mW which is roughly double the output power of their previous LEDs while the input power only increased about 40%. This boost makes our current LED selection 1.5% efficient meaning the remaining 98.5% of the energy is turned into heat. Given that we will have 20-40LEDs, we will need a very large battery capacity and a capable cooling system.

Characteristic	Unit	Typical	Max
Viewing angle[1]	degrees	130	
Forward voltage at 500 mA	V	6.5	9
Thermal resistance, junction-to-case	°C/W	7.0	
Power dissipation at 500 mA	W	4.4	5

Figure 5.2: Table of the LED characteristics from Klaran's data sheet provided

Our LEDs generate a lot of heat. To safely dissipate the heat produced by said LEDs, Klaran recommended soldering the LEDs only onto Metal-Core PCBs (MCPCB) along with decent heatsinks and ventilation. MCPCBs generally come with an aluminum core, because aluminum spreads heat well enough and is cost efficient. Heatsinks are also recommended because the MCPCBs alone cannot dissipate all the heat produced by the LEDs. As we further develop our LED layout and PCB design, we will decide whether we will build a custom heatsink/s or purchase a heat sink/s. We used KiCAD to design a possible PCB layout almost ready to be printed on a MCPCB for our LEDs and we will develop this further after we find what our total budget will be for the coming semester and how much of it we can use to print MCPCBs.

Last semester we considered a 60Wh battery, but we have since moved to the NH2054HD34 lithium ion smart battery by Inspired Energy. The choice was based on a similar battery that JPL uses in their particle counters. The final decision was made because this is a sealed battery with low outgassing. The Inspired Energy battery has 14.4V, 8A, and has a capacity of 98Wh. To last at least 3 hours we would have at minimum 8 of these batteries connected in parallel.

As we progressed, we needed to calculate a power budget. After deciding to have eight 98Wh batteries in parallel, to increase our output current, we researched DC motors that were capable of moving Cleanbot. We found and tested two brushed DC gear motors on Amazon by "uxcell." A 10 RPM and a higher torque 5 RPM DC motor both had an operating voltage of 12V. We conducted two tests for our motors. The first was dragging a 10 lbs weight and the second was pulling a skateboard with 25 lbs or 35 lbs. We found that the 5 RPM motor was able to pull more weight than the 10 RPM while using less power, even when the 10 RPM motor was running at 5 RPM. Given the time it takes for the UV-C LEDs to sanitize a given surface, Cleanbot doesn't need to move faster than 5 RPM. One 5 RPM motor was able to pull a skateboard with 35 lbs while only consuming 0.38W. With a weight limit of 25 lbs and using four 5 RPM motors, Cleanbot should have no issues going over any cables or cable ramps while only consuming a maximum of 2W.

In order to give each component the correct voltage and current required to operate safely. Various DC-DC converters as well as a motor driver/controller that would be able to power and control the four motors with a Raspberry Pi or an Ardunio were also found. The efficiency of our chosen DC-DC converters were around 89%-91%. We found one brandless motor driver compatible with Arduinos and Raspberry Pis on *Amazon* that requires a 15A fuse but can only control 2 motors at a time. Below we can see a table of the required voltages and currents for each of Cleanbot 3000's subsystems. The current shown for the Lidar Sensor is actually for the Lidar Sensor's interface box, which is a DC-DC converter. The actual power that goes into the Lidar Sensor is 14W to 20W with an initial burst power consumption of 22W, with a max current of less than 1A. In the table, the currents shown are the subsystems average current and the voltages are either the range or average voltage of each subsystem, except for the Lidar Sensor.

Subsystem	Voltage (V)	Current (A)
Lidar Sensor	24	1.5
Ultrasonic Sensors	3 - 5.5	0.034
Motors (x1)	12.55	0.250
UV-C LEDs (x1)	6.5 - 10V	0.5

Table 5.1: Voltages and Currents of Cleanbot 3000's Major Subsystems

6. Conclusion

Much of the theoretical research has been completed and steps towards procuring and incorporating components are now a large part of the future plans. Each subteam has tasks that are looking to be accomplished in the following semesters.

The navigation team has made progress on the pi-car using both LIDAR and ultrasonic sensors. This team would like to continue using LIDAR sensors and begin incorporating SLAM.

The trident reflectors team has created a physical proof of concept and developed a skeleton for a model on MATLAB. The next steps involve applying the pinhole projection model onto the MATLAB model and attempting to create accurate predictions. Furthermore, the trident team seeks to choose a camera and begin testing with the pi-car.

The sensors team has chosen the LIDAR model that will be used, and extracted live data from a temporary test LIDAR sensor with the help of the navigation team. The sensor team from this point onward will now combine with and work with the navigation team in order to further bridge the gap between the navigation algorithm and the sensors.

The sanitization and power team have chosen various parts from LEDs to the DC to DC converters and developed a power budget which will continue to be updated. With the low efficiency of the LEDs and the high demand of power from all the components, a thermal budget is looking to be created in the future in order to avoid potential damage due to heat.

Overall, the Cleanbot is still in the early stages of development. Through further research, testing, and formulation of new ideas, it will continue to be built upon until completion. If completed, the Cleanbot could pave the way to higher standards for cleanrooms through its obstacle avoidance, intelligent navigation, and high-powered LEDs; furthermore, it achieves all of this without the need for a change of filters similar to vacuums, or the use of harsh chemicals that have negative impacts on the environment. Cleanrooms of all types, whether it be JPL's aerospace research and development cleanrooms, medical cleanrooms, or hospitals, can benefit from the Cleanbot's ability to achieve its task with independence and self-reliance.

References

[1] Smith, David. (2014). A Balloon-Based Payload for Exposing Microorganisms in the Stratosphere (E-MIST). Gravitational and Space Research. 2. 70-80.

[2] Dissanayake, M.W.M.G. *A Solution to the Simultaneous Localisation and Map Building (SLAM)ProblemM*.2006,www.cs.cmu.edu/afs/cs/Web/People/16899/notes/slam_ieee.pdf.

[3] Keeping, Steven. "Calculating LED Junction Temperature in Lighting Applications." *DigiKey*, Electronic Products, 7 June 2011, www.digikey.com/en/articles/techzone/2011/jun/calculating-led-junction-temperature-in-lighting-applications.

[4] Riisgaard, Soren, and Morten Rufus Blas. *SLAM for Dummies*. www-inst.eecs.berkeley.edu/~ee290t/fa18/readings/Slam-for-dummies-mit-tutorial.pdf.

[5]Slamtec "*RPLIDAR A1 Low Cost 360 Degree Laser Range Scanner*" Model:A1M8 -July. 2016 [Revised April. 2016].

[6] Hull, Craig. The Complete Guide to Camera Lenses: Parts, Functions, and Types. Expert Photography, Camera Lenses Guide, 2019.

[7] Bebis, George. *Image Processing and Interpretation.CSE*, Reno, NV: University of Nevada, Reno, 2019.

[8] Majumder, Aditi. *The Pinhole Camera*, Irvine, CA: University of California, Irvine, 2017.