



**Diseño y construcción de una plataforma móvil diferencial para la implementación de mapeo  
semántico**

Cañar Ortiz, Edison Fabricio

Departamento de Ciencias de la Energía y Mecánica

Carrera de Ingeniería en Mecatrónica

Trabajo de titulación, previo a la obtención del título de Ingeniero en Mecatrónica

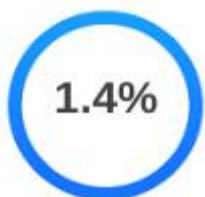
Ing. Loza Matovelle, David César, MSc.

18 de agosto del 2022

# COPYLEAKS

TESIS\_CAÑAR\_EDISSON.pdf

Scanned on: 2:34 August 15, 2022 UTC



Overall Similarity Score



Results Found



Total Words in Text

Identical Words	137
Words with Minor Changes	34
Paraphrased Words	58
Omitted Words	0



DAVID CESAR  
LOZA  
MATOVELLE



**Departamento de Ciencias de la Energía y Mecánica**

**Carrera de Ingeniería en Mecatrónica**

### **Certificación**

Certifico que el trabajo de titulación: **"Diseño y construcción de una plataforma móvil diferencial para la implementación de mapeo semántico"** fue realizado por el señor **Cañar Ortiz, Edison Fabricio**; el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

**Sangolquí, 16 de agosto de 2022**

Firma:



.....  
**Loza Matovelle, David César**

C. C.: 1708661549



Departamento de Ciencias de la Energía y Mecánica

Carrera de Ingeniería en Mecatrónica

**Responsabilidad de Autoría**

Yo, **Cañar Ortiz, Edison Fabricio**, con cédula de ciudadanía n°1805237334, declaro que el contenido, ideas y criterios del trabajo de titulación: **Diseño y construcción de una plataforma móvil diferencial para la implementación de mapeo semántico** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

**Sangolquí, 16 de agosto de 2022**

Firma

**Cañar Ortiz, Edison Fabricio**

C.C.: 1805237334



Departamento de Ciencias de la Energía y Mecánica

Carrera de Ingeniería en Mecatrónica

**Autorización de Publicación**

Yo **Cañar Ortiz, Edison Fabricio**, con cédula de ciudadanía n°1805237334, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Diseño y construcción de una plataforma móvil diferencial para la implementación de mapeo semántico** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

**Sangolquí, 16 de agosto de 2022**

Firma

**Cañar Ortiz, Edison Fabricio**

C.C.: 1805237334

### **Dedicatoria**

Dedico el proyecto de titulación a mi madre Gloria América Ortiz por su apoyo incondicional, quien me ha guiado y dado fuerza para continuar a lo largo de mi camino. Sus enseñanzas son el reflejo del amor y orgullo, los cuales siempre me ha brindado. A mi hermana Jessica, por creer en mí. A mi tía, por ser una segunda madre. Finalmente, a mis amigos, compañeros y profesores quienes han sido parte de mi formación durante la carrera y mi vida.

### **Agradecimiento**

Ante todo, quiero agradecer a Dios por otorgarme salud, inteligencia y herramientas para encarar los retos presentados a lo largo de mi vida. Doy gracias mi madre, Gloria Ortiz por enseñarme la responsabilidad y la perseverancia desde pequeño, así como el cariño, amor y respeto reflejado mientras realizaba mis estudios superiores. A mi tía, Nelly Ortiz por haber sido parte de mi desarrollo emocional e intelectual, además, ser una fuente de inspiración al haber superado una de las más difíciles pruebas y declararse victoriosa. A mi hermana Jessica Cañar por confiar en mis capacidades y su apoyo incondicional mientras crecíamos. Finalmente, agradezco mi padre, Nelson Cañar por su paciencia y fortaleza.

Agradezco a mi tutor Ing. David Loza por la paciencia al guiarme a lo largo del proceso de titulación, así como el valor de la responsabilidad y perseverancia durante los momentos más críticos. De igual manera doy gracias a los docentes de la Universidad de las Fuerza Armadas quienes han sido parte de mi formación académica durante y fuera de las horas de clases.

Finalmente, quiero agradecer mis amigos Alexander, Walter, Edwin, Gabriel y Andrés por haber sido mi familia en nuestra querida institución, con quienes comparto experiencias que llevare

**Edisson Fabricio Cañar Ortiz**

## Índice de contenido

Certificado del director .....	3
Responsabilidad de autoría.....	4
Autorización de publicación.....	5
Dedicatoria .....	6
Agradecimiento .....	7
Índice de contenido .....	8
Índice de Tablas .....	12
Índice de Figuras .....	14
Resumen .....	18
Abstract .....	19
Capítulo I.....	20
Conceptos Básicos .....	20
Antecedentes en Mapeo Semántico .....	20
Antecedentes Internacionales .....	21
Antecedentes en Latinoamérica .....	25
Justificación e importancia .....	27
Objetivos.....	27
Objetivo principal.....	27
Objetivos específicos .....	27
Área de Influencia .....	28



Descripción del Proyecto .....	28
Estructura del Documento .....	29
Capítulo II Estado del arte .....	31
Robótica Móvil .....	31
Locomoción.....	33
Navegación .....	36
Percepción .....	37
Clasificación de Sensores .....	38
Localización.....	39
Localización Relativa .....	39
Localización Global.....	39
Odometría.....	40
Cognición .....	41
Mapeo.....	41
Mapa Métrico .....	42
Mapa Topológico .....	43
Mapa Híbrido .....	43
SLAM.....	43
Arquitectura de SLAM .....	44
Métodos de SLAM.....	45

	10
Detección de objetos .....	46
Ontología .....	46
Mapeo Semántico .....	48
Proceso para la generación de mapa semántico.....	49
Taxonomía del Mapa Semántico.....	49
Resumen del Capítulo .....	50
Capítulo III Metodología y Diseño .....	51
Definición de necesidades .....	51
Definición de Especificaciones .....	51
Definición de Estructura Funcional .....	53
Estructura y movilidad .....	56
Modelo Cinemático.....	56
Modelamiento Dinámico .....	59
Dinámica Inversa.....	62
Potencia y tracción.....	66
Percepción del entorno.....	73
Procesamiento central de la plataforma.....	79
Software del sistema.....	82
Diseño de la base inferior de la plataforma .....	83
Análisis estático de la base.....	83

	11
Diseño de Piezas de Unión y Soporte.....	86
Diseño para la base intermedia .....	88
Diseño para el soporte de la cámara Kinect.....	89
Diseño para la cubierta de la plataforma móvil .....	91
Diseño del controlador para los actuadores .....	93
Diseño de la conexión electrónica de la plataforma. ....	95
Diseño de la placa PCB .....	97
Especificaciones Finales del Diseño .....	97
Resumen .....	100
Capítulo IV Construcción y Pruebas .....	101
Construcción de la plataforma.....	101
Software de la plataforma móvil.....	111
Controlador de los motores .....	111
Pruebas y Resultados .....	114
Resumen .....	124
Capítulo V Conclusiones, Recomendaciones y Trabajos Futuros .....	126
Conclusiones y Recomendaciones .....	126
Trabajos Futuros .....	127
Bibliografía.....	129
Apéndices .....	136

## Índice de Tablas

Tabla 1 Similitudes entre POO y la ontología .....	48
Tabla 2 Necesidades para diseño de la plataforma móvil .....	52
Tabla 3 Especificaciones técnicas de la plataforma.....	53
Tabla 4 Subsistemas establecidos para el diseño de la plataforma móvil.....	54
Tabla 5 Matriz de ponderación con los criterios para el material base.....	64
Tabla 6 Matriz de ponderación con respecto al peso .....	64
Tabla 7 Matriz de Holmes para el material .....	65
Tabla 8 Propiedades mecánicas para el MDF.....	65
Tabla 9 Matriz de ponderación con los criterios para la fuente de energía .....	66
Tabla 10 Matriz de Holmes para la fuente de energía .....	67
Tabla 11 Matriz de ponderación con los criterios para la selección de actuadores .....	68
Tabla 12 Matriz de Holmes para el actuador .....	69
Tabla 13 Datos técnicos de la Teensy 3.5.....	70
Tabla 14 Especificaciones del motor DC Ocean 270:1 con 64CPR de CQRobot .....	70
Tabla 15 Matriz de ponderación con los criterios para la selección de driver para los motores.....	71
Tabla 16 Matriz de Holmes para el módulo controlador de motores .....	72
Tabla 17 Especificaciones del módulo L298N .....	72
Tabla 18 Matriz de ponderación con los criterios de la selección de sensores .....	73
Tabla 19 Matriz de Holmes para la selección de sensores .....	77
Tabla 20 Datos técnicos para el MPU6050.....	77
Tabla 21 Datos técnicos del sensor LIDAR A1M8 .....	78
Tabla 22 Datos técnicos para la Kinect v1 de la consola Xbox 360.....	78

Tabla 23 Matriz de ponderación con los criterios para la selección del CPU .....	79
Tabla 24 Matriz de Holmes para la selección del CPU.....	80
Tabla 25 Datos técnicos de la tarjeta Jetson Nano 4GB .....	81
Tabla 26 Consumo Energético de la plataforma .....	82
Tabla 27 Datos Técnicos de batería .....	82
Tabla 28 Lista para la masa de los elementos en la plataforma .....	84
Tabla 29 Propiedades Mecánicas del filamento PLA.....	87
Tabla 30 Sintonización de controladores mediante Ziegler-Nichols .....	94
Tabla 31 Módulos de la plataforma móvil.....	99
Tabla 32 Constantes PID para el motor derecho.....	112
Tabla 33 Constantes PID para el motor izquierdo.....	112
Tabla 34 Comprobación de velocidad de la plataforma a 0.15 (m/s).....	115
Tabla 35 Comprobación de velocidad de la plataforma a 0.2 (m/s).....	115
Tabla 36 Comprobación de velocidad de la plataforma a 0.25 (m/s).....	115
Tabla 37 Resultados de la prueba para odometría .....	117
Tabla 38 Resultados de la prueba para el mapa generado .....	120
Tabla 39 Resultados de la prueba para YOLO .....	122

## Índice de Figuras

Figura 1 Representación de jerarquías espacial y conceptual.....	21
Figura 2 Escenario experimental (Derecha). Robot Magellan Pro (Izquierda).....	22
Figura 3 Representación de mapeo semántico .....	23
Figura 4 Diagrama de Flujo de SpCoMapping .....	24
Figura 5 Visión General de mapeo semántico aumentado .....	25
Figura 6 Representación de un mapa aumentado .....	26
Figura 7 Robot AURORA (Universidad de Málaga) .....	31
Figura 8 Robots en aplicaciones comunes .....	32
Figura 9 Robots Estacionarios .....	33
Figura 10 Robots móviles terrestres .....	34
Figura 11 Tipos básicos de Ruedas.....	34
Figura 12 Robot Aéreo .....	35
Figura 13 Robot Acuático .....	35
Figura 14 Ciclo en Robots Móviles .....	36
Figura 15 Diagrama del proceso para Percepción.....	37
Figura 16 BIBA Robot, BlueBotics SA.....	38
Figura 17 Error de Odometría .....	40
Figura 18 Representación visual de un mapa métrico .....	42
Figura 19 Representación de la arquitectura de SLAM .....	44
Figura 20 Descripción de la detección de objetos.....	46
Figura 21 Visualización de una ontología.....	47
Figura 22 Metodologías para mapeo semántico.....	50
Figura 23 Diagrama de la funcionalidad del robot móvil .....	54

Figura 24 Diagrama de las funciones para la funcionalidad de la plataforma .....	55
Figura 25 Diagrama de los subsistemas para la plataforma .....	55
Figura 26 Robot móvil diferencial .....	56
Figura 27 Velocidades del robot diferencial.....	57
Figura 28 Representación giro con respecto a ICC.....	58
Figura 29 Energía sobre el sistema .....	60
Figura 30 Fuerzas actuantes en el sistema.....	61
Figura 31 Representación esquemática de la cámara Kinect .....	74
Figura 32 RpLidar A1-M8 .....	75
Figura 33 Diagrama del funcionamiento del sensor ultrasónico .....	75
Figura 34 Representación de la visión estereoscópica .....	76
Figura 35 Representación de la medición por parte del sensor inercial .....	76
Figura 36 Forma de la base inferior de la plataforma .....	83
Figura 37 Resultado del análisis estático (Von Mises).....	85
Figura 38 Deformación resultante de la base .....	86
Figura 39 Eje para unir el motor con la rueda.....	86
Figura 40 Piezas adicionales para sujeción y soporte.....	87
Figura 41 Representación del ensamblaje para el piso inferior del robot.....	88
Figura 42 Base intermedia de la plataforma móvil .....	89
Figura 43 Soporte para el tubo base .....	89
Figura 44 Elementos para la sujeción de la cámara al tubo .....	90
Figura 45 Repisa superior de la plataforma móvil.....	90
Figura 46 Representación del ensamblaje para el piso intermedio de la plataforma .....	91
Figura 47 Recubrimientos para el piso primario de la plataforma .....	92

Figura 48 Diseño mecánico final para la plataforma móvil .....	92
Figura 49 Diagrama de bloque para controlador del motor.....	93
Figura 50 Diagrama eléctrico del sistema .....	95
Figura 51 Suministro de energía del sistema .....	95
Figura 52 Conversión de voltaje de 5V .....	96
Figura 53 Control de los motores.....	96
Figura 54 Conexión de los dispositivos USB al CPU.....	97
Figura 55 Diseño de la placa PCB para circuito de potencia y control.....	98
Figura 56 Piezas para la base de la plataforma diferencial .....	102
Figura 57 Soporte para la batería.....	102
Figura 58 Instalación parte motriz y energética a la base .....	103
Figura 59 Sistema de locomoción del móvil.....	103
Figura 60 Montaje de los componentes electrónicos .....	104
Figura 61 Conexión interna entre los componentes electrónicos.....	105
Figura 62 Conjunto de soporte (150 mm) .....	106
Figura 63 Pieza frontal derecha del contorno .....	106
Figura 64 Ensamblaje de la plataforma móvil .....	107
Figura 65 Forma para el corte de la cubierta de la plataforma móvil .....	108
Figura 66 Montaje del sensor IMU y voltímetros.....	108
Figura 67 Instalación de los dispositivos A1M8 y Jetson Nano 4GB.....	109
Figura 68 Soporte para la cámara Kinect .....	109
Figura 69 Distribución del cableado del sistema .....	110
Figura 70 Conexión USB de los dispositivos a la tarjeta Jetson .....	110
Figura 71 Ensamblaje final del robot móvil.....	111



Figura 72 Respuesta del controlador del motor derecho .....	113
Figura 73 Respuesta del controlador del motor izquierdo.....	114
Figura 74 Prueba: Cuadrado de 1.5x1.5.....	116
Figura 75 Nodos para la prueba de odometría .....	116
Figura 76 Visualización de Odometría en RViz .....	117
Figura 77 Ejecución de SLAM .....	118
Figura 78 Nodos relevantes para el Gmapping .....	119
Figura 79 Visualización del mapa generado mediante SLAM-Gmapping .....	119
Figura 80 Prueba del mapa generado sobre un cuarto .....	120
Figura 81 Nodo /darknet_ros.....	121
Figura 82 Detección de objetos con YOLO .....	121
Figura 83 Nodo para publicar las locaciones de interés .....	122
Figura 84 Cuartos referenciados dentro del mapa.....	123
Figura 85 Cuartos referenciados dentro del mapa.....	123
Figura 86 Visualización de la ontología en Protégé.....	124

## Resumen

Un robot móvil es un dispositivo que tiene la capacidad para movilizarse en el mundo real al utilizando sistemas de locomoción, mientras emplea sensores para percibir el ambiente y tomar acciones que permitan estimar su localización. Lo que ha permitido su aplicación en diversos campos como agricultura, medicina, educación, industria, militar, vigilancia, seguridad, entre otros. Debido a ello, el entendimiento entre la máquina y el usuario es fundamental en el desarrollo de plataformas móviles que tengan la habilidad para recibir y transmitir información al ser humano de forma sencilla en un lenguaje común.

En el presente trabajo se efectúa el estudio y obtención de mapeo semántico por parte de una plataforma móvil. Comúnmente se utiliza métodos de SLAM para la construcción de mapas, lo cual requiere la aplicación de odometría para calcular la posición del robot, y nubes de puntos que realizan una reconstrucción probabilística de los espacios ocupados dentro del entorno. La detección de objetos es un proceso el cual implica dotar de razonamiento a la máquina debido a que es capaz de diferenciar los objetos entre sí y presentar dicha información. Al utilizar algoritmos ontológicos para asignar información entre los objetos y el lugar le permite al dispositivo construir un mapa provisto de información que el usuario puede entender.

Para complementar la investigación, se presenta el resultado mediante un archivo tipo OWL, el cual es leído por el software Protégé.

*Palabras claves:* mapeo semántico, odometría, SLAM, detección de objetos, ontología.

### **Abstract**

A mobile robot is a device that has the ability to move in the real-world using locomotion systems, while using sensors to perceive the environment and take actions to estimate its location. Which has allowed its application in various fields such as agriculture, medicine, education, industry, military, surveillance, security, among others. Because of this, the understanding between the machine and the user is fundamental in the development of mobile platforms that have the ability to receive and transmit information to the human being in a simple way in a common language.

In the present work, the study and obtaining of semantic mapping by a mobile platform is carried out. SLAM methods are commonly used for the construction of maps, which requires the application of odometry to calculate the position of the robot, and point clouds that perform a probabilistic reconstruction of the spaces occupied within the environment. Object detection is a process that involves providing reasoning to the machine because it is able to differentiate objects from each other and present such information. By using ontological algorithms to assign information between objects and the place allows the device to build a map provided with information that the user can understand. To complement the research, the result is presented using an OWL-type file, which is read by the Protégé software.

*Key words:* semantic mapping, odometry, SLAM, object detection, ontology.

## Capítulo I

### INTRODUCCIÓN

El primer capítulo expone la necesidad del estudio en el área de mapeo semántico en plataformas móviles, para optimizar la interacción entre humano y máquina. El cual se constituye de una breve revisión de conceptos e investigaciones previas relacionadas al tema permitiendo presentar las generalidades del proyecto que conllevan el planteamiento del problema e importancia en su resolución, objetivos y el alcance del mismo. Finalmente se da paso a como se constituye la estructura del documento en los próximos capítulos.

#### Conceptos Básicos

A continuación, se presentan ciertas definiciones de interés:

- Mapeo: En robótica móvil, es modelar el mapa del medio donde se desempeña el robot y permitir la navegación dentro del mismo, por lo cual requiere la información proveniente de sensores como cámaras, láser y giroscopios, etc. (Thrun, 2002; Guzmán, 2016)
- Localización: Es el proceso para establecer la posición y orientación del robot dentro del marco de referencia mediante sensores y odometría. (Saputra, 2015)
- Navegación: Es la capacidad de los robots para trasladarse de una locación a otra en diversos entornos. (Rubio, Valero, & Llopies-Albert, 2019)
- SLAM: Se denomina al procedimiento donde el robot aplica mapeo y localización de forma simultánea a medida que se desplaza en su entorno. (N, T, & B, 2015)

#### Antecedentes en Mapeo Semántico

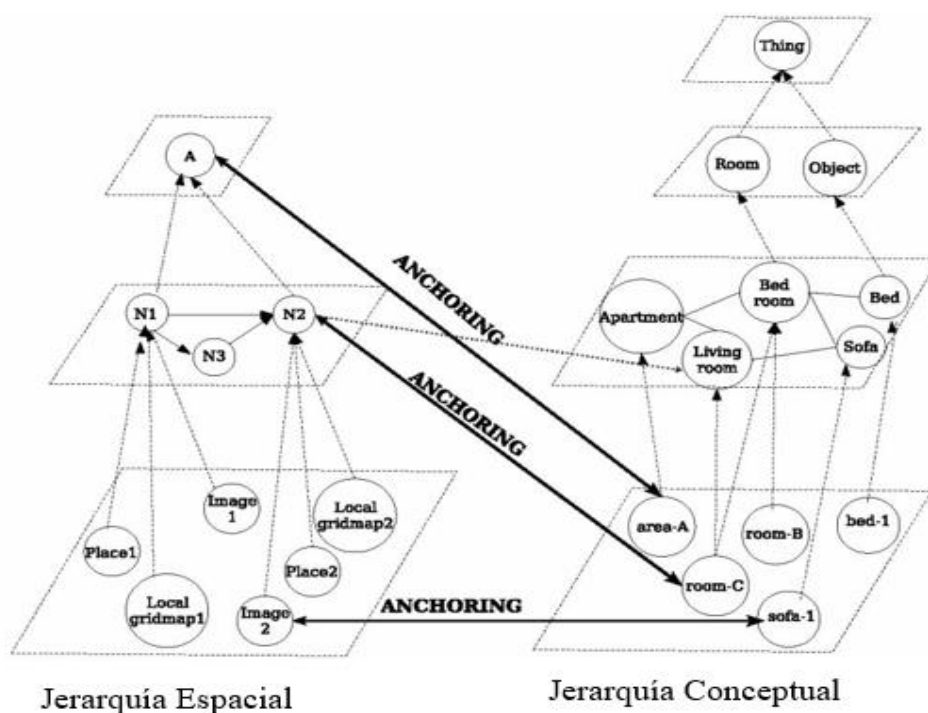
Se indaga los trabajos más representativos para la temática, por lo cual se realiza un estudio de investigaciones tanto internacional como local.

### Antecedentes Internacionales

En (Galindo, Saffiotti, Coradeschi, & Buschka, 2005) se presenta un mapa multijerárquico para robots móviles compuesto por información métrica y semántica. Su representación está formada por una perspectiva espacial, la cual permite realizar planificación junto con la realización de tareas mientras el componente semántico realiza el razonamiento simbólico. Dichas perspectivas se enlazan mediante un “enlace” que crea una correspondencia entre simbología y datos de sensores referidos a objetos físicos como se observa la Figura 1.

**Figura 1**

*Representación de jerarquías espacial y conceptual*

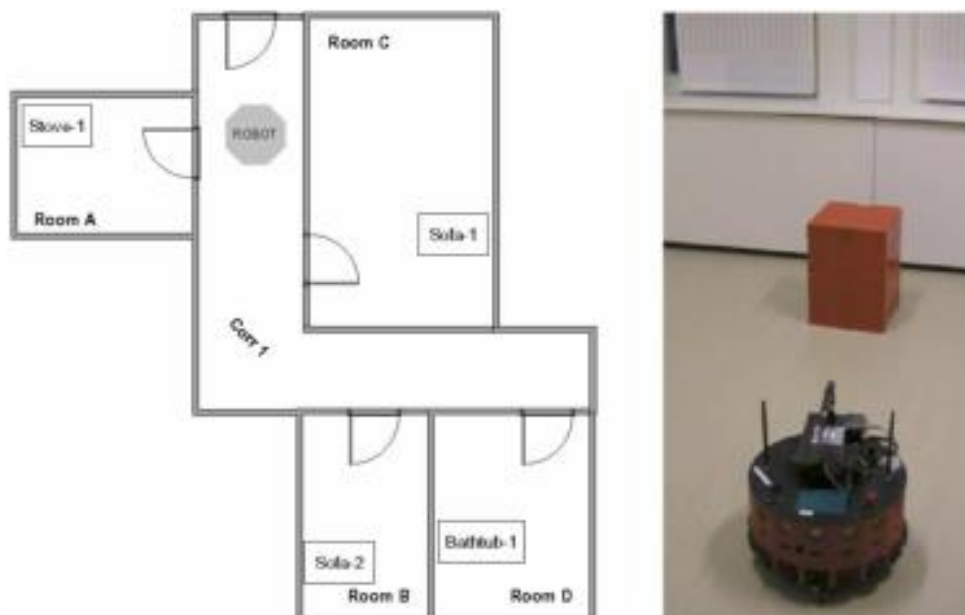


Nota: La figura muestra las jerarquías espaciales y semánticas, a la izquierda se encuentra la información espacial suministrada por los sensores del robot, a la derecha esta la información *semántica* que conlleva los conceptos de dominio y relaciones entre ellas, mediante el anclaje se crean las relaciones entre las jerarquías. (Galindo, Saffiotti, Coradeschi, & Buschka, 2005)

El robot Magellan Pro de la Figura 2, permitió la aplicación del enfoque anterior en tareas en modelamiento, navegación y localización; las habilidades resultantes fueron la interfaz mediante un conjunto de conceptos, clasificación del cuarto, etc. (Galindo, Saffiotti, Coradeschi, & Buschka, 2005)

## Figura 2

*Escenario experimental (Derecha). Robot Magellan Pro (Izquierda)*



Nota: Diagrama del interior de una casa, seguido por el robot Magellan Pro ubicado al frente de un bloque color naranja para representar un sofá. (Galindo, Saffiotti, Coradeschi, & Buschka, 2005)

En (Wolf & Sukhatme, 2008) aborda el desarrollo de mapeo semántico con la aplicación de dos técnicas. El primer enfoque está basado en modelos de Márkov y el segundo se denomina como vectores de soporte. En la técnica de Márkov cada secuencia de datos se considera a la vez, mientras que al aplicar vectores de soporte cada punto es clasificado de forma individual.

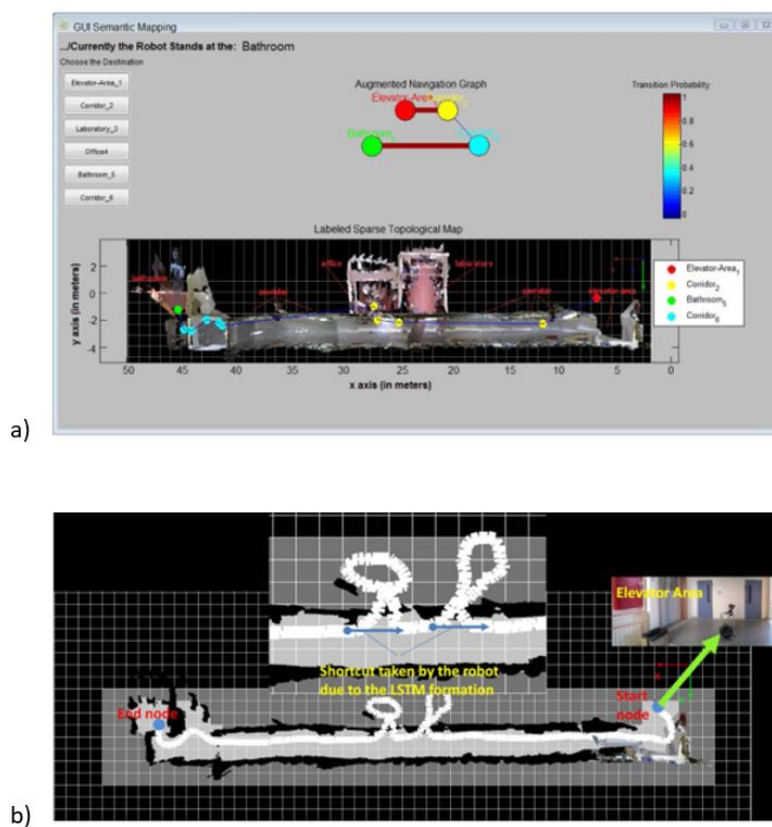
En la investigación de (Douillard, Fox, Ramos, & Durrant-Whyte, 2011) se presenta la clasificación de objetos y mapeo semántico de ambientes urbanos para lo cual se propone un modelo basado en campo aleatorio condicional (Conditional Random Field o CRF en inglés) que permite representar las correlaciones espaciales y temporales entre la información que regresa de los sensores. El resultado es

una imagen probabilística en 2D que permite el reconocimiento de objetos mediante múltiples sensores para obtener características relevantes y formar correlaciones mediante CRF.

La unión tanto de la navegación junto con mapas semánticos es fundamental en el avance de este campo tal como se detalla en (Kostavelis, Charalampous, Gasteratos, & Tsotsos, 2015)

### Figura 3

*Representación de mapeo semántico*



Nota: a) La interfaz gráfica para el usuario que conceptualiza el camino, b) la localización representada por una nube de puntos en 3D. (Kostavelis, Charalampous, Gasteratos, & Tsotsos, 2015)

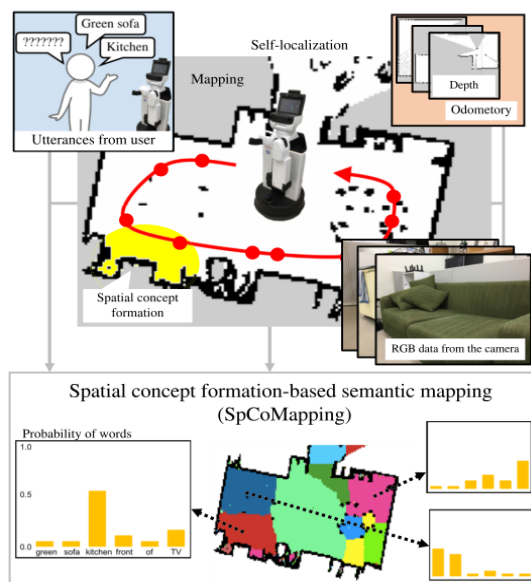
Las investigación utiliza muestras sintéticas para capturar la semántica de su entorno al usar de medio relaciones codificadas por el ser humano disminuyendo procesamiento de datos sensoriales (Ruiz-Sarmiento, Galindo, & Gonzalez-Jimenez, Exploiting semantic knowledge for robot object recognition, 2015); mientras tanto en el trabajo de (Ruiz-Sarmiento, Galindo, & Gonzalez-Jimenez, Robot@Home, a

robotic dataset for semantic mapping of home environments, 2017) se expone los datos Robot@Home, es una colección de información reunida por robots móviles en entornos domésticos que sea utilizado como un banco de pruebas para algoritmos de mapeo semántico, a través de la categorización para objetos y/o cuartos esto se ha logrado al implementar secuencias de láser RGB-D, información topológica de la conexión entre los cuartos, mapas geométricos 2D y 3D.

En (Katsumata, Taniguchi, Hagiwara, & Taniguchi, 2019) se plantea el método para mapeo semántico llamado SpCoMapping, que extendió el concepto de adquisición espacial mediante campo aleatoria de Markov, ver la Figura 4.

**Figura 4**

*Diagrama de Flujo de SpCoMapping*



Nota: El diagrama refleja como el robot obtiene histogramas de palabras e imágenes que al unirse dan como resultado la auto localización y la categorización multimodal.

El resultado de la investigación muestra que los mapas semánticos generados mediante dicha técnica, los cuales presentaban una mejoría en comparación a los métodos existentes desde puntos de vista en coincidencia de lugar y comunicación humano-máquina. (Katsumata, Taniguchi, Hagiwara, & Taniguchi, 2019)



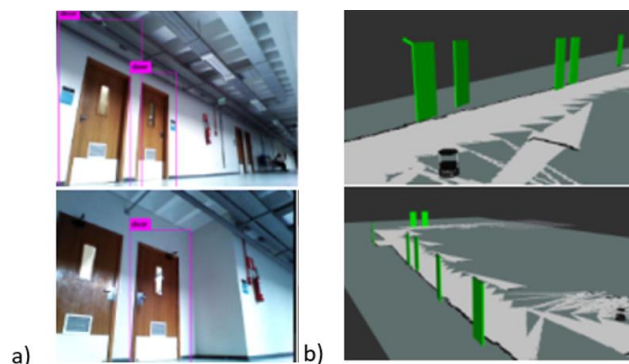
### ***Antecedentes en Latinoamérica***

La investigación (Silva, 2016) propone un modelo de relaciones semánticas para ejecutar segmentación semántica mediante un sensor RGB-D, al finalizar el estudio se determina que el enfoque gráfico probabilístico, mejorando los resultados de clasificación al modelar la información contextual y utilizarla para optimizar los resultados en un modelo que no posee dicha información.

El estudio realizado por (Bersan, Martins, Campor, & Nascimento, Semantic Map Augmentation for Robot Navigation: A Learning Approach based on Visual and Depth Data, 2018) presenta una metodología en tres secciones: un módulo de SLAM, para crear mapas 2D mediante sensores en el robot; el siguiente es el módulo para detección de objetos, el cual ejecuta una red neuronal convolucional que detecta clases pre entrenadas de objetos en tiempo real; y finalmente se establece un módulo para localizar y posicionarlos objetos previamente detectados utilizando la nube de puntos en la imagen RGB-D, donde los resultados del estudio se presentan en la Figura 5.

### **Figura 5**

#### *Visión General de mapeo semántico aumentado*

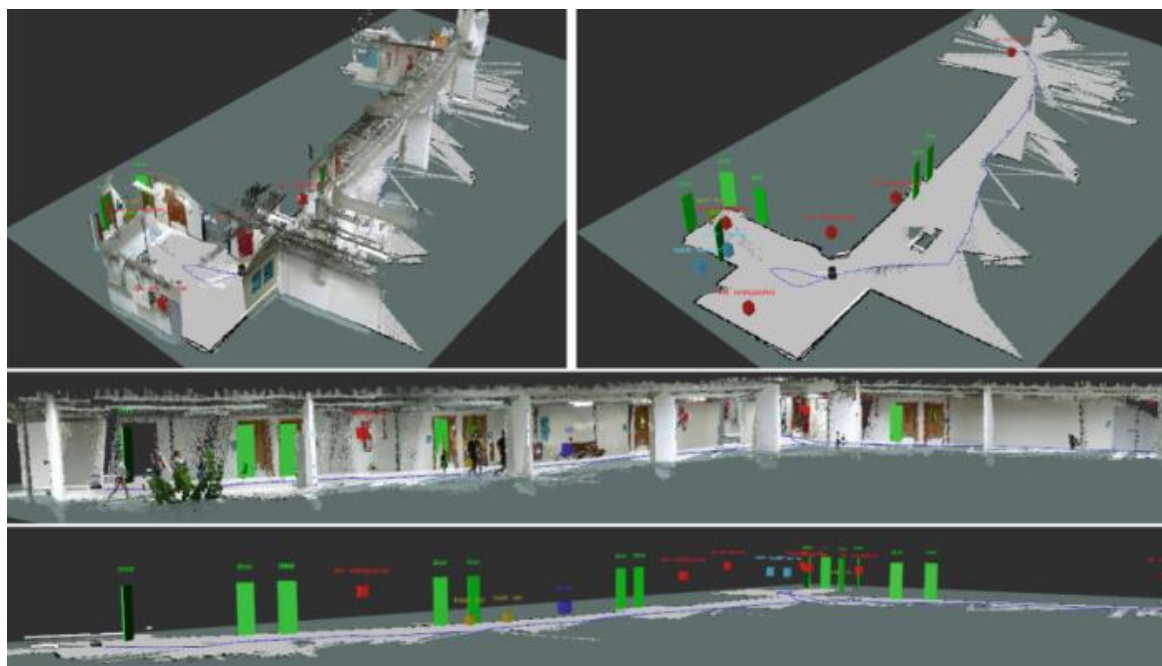


Nota: a) Ejemplo de la detección de objetos como puertas, b) es una visualización de la salida del mapa semántico aumentado ya que las puertas han adquirido un valor diferente dentro de la escena del mapeo común. (Bersan, Martins, Campor, & Nascimento, Semantic Map Augmentation for Robot Navigation: A Learning Approach based on Visual and Depth Data, 2018)

Más adelante en (Bersan, Martins, Campor, & Nascimento, Extending Maps with Semantic and Contextual Object Information for Robot Navigation: A Learning-Based Framework Using Visual and Depth Cues, 2020) implementa su metodología basada en cuatro etapas, que son: la primera etapa contempla la detección y categorización visual de objetos mediante segmentación semántica; seguido por la etapa del entrenamiento y agrupación datos donde la red neuronal es entrenada mediante imágenes pre etiquetadas; en la tercera etapa se aplica la localización y navegación con mapas generados en 2D aplicando sensores y finalmente la etapa del modelo de emparejamiento y posicionamiento; el proceso realiza un seguimiento de objetos para mejorar el mapa como se observa en la Figura 6.

### Figura 6

*Representación de un mapa aumentado*



Nota: En la imagen se visualiza los objetivos geométricos primitivos en verde, rojo y azul que representan las puertas, fuentes de agua y extintores de incendio respectivamente. (Bersan, Martins, Campor, & Nascimento, Extending Maps with Semantic and Contextual Object Information for Robot Navigation: A Learning-Based Framework Using Visual and Depth Cues, 2020)

## **Justificación e importancia**

La investigación en el campo de la robótica ha permitido un avance significativo al desarrollo de dispositivos, al ejecutar órdenes programadas previamente permiten realizar tareas complicadas y/o repetitivas para mejorar la condición de vida del ser humano. Sin embargo, tanto acciones como los entornos donde se devuelven los robots están en un constante cambio por lo que interacción entre humano-máquina es un requisito para su desempeño en áreas de salud, educación, industria, etc.; implicando que el robot sea capaz de socializar al entablar relaciones interactivas mientras desarrolla un cierto grado de inteligencia al decidir por sí mismo.

Lo cual ha motivado que el proyecto se oriente al desarrollo de mapeo semántico por parte de una plataforma móvil equipada con sensores los cuales permiten obtener un mapa bidimensional con la finalidad de relacionar información conceptual; donde un ser humano sea capaz de entender mientras el robot interpreta las instrucciones del usuario disminuyendo la modificación en la programación interna, obteniendo un alto nivel de interacción humano-máquina mediante representación mejorada del entorno.

## **Objetivos**

### ***Objetivo principal***

Diseñar y construir una plataforma móvil diferencial para la implementación de mapeo semántico.

### ***Objetivos específicos***

- Diseñar una plataforma móvil en configuración diferencial.
- Dimensionar y seleccionar actuadores para producir una velocidad aproximada de 0.25 m/s.
- Seleccionar e implementar sensores que permitan ejecutar odometría en la construcción de mapas mediante SLAM.
- Utilizar algoritmos que permita relacionar los objetos detectados y las ubicaciones en un mapa bidimensional.

- Visualizar el resultado del mapa semántico creado por ontología.

### **Área de Influencia**

El desarrollo del proyecto se orienta al avance en la investigación de plataformas móviles para realizar tareas con capacidad de razonamiento al discernir entre lugares representados en un mapa obtenido mediante métodos de localización (SLAM), conjuntamente con la adición de información significativa como definición de objetos, áreas y las relaciones entre ellas proporcionando un alto nivel de interacción con el usuario.

De forma aplicativa el dispositivo está destinado al Laboratorio Sistemas Dinámicos de Mecatrónica ubicado en la Universidad de las Fuerzas Armadas -ESPE con un futuro aplicativo como medio de investigación en el ámbito de mapeo y detección de objetos, además de la posibilidad para transportar objetos entre diferentes lugares a medida que elabora un mapa dotado de información conceptual. Sin embargo, se busca aplicaciones se extienden al ámbito social de las áreas de salud, educación, militar, transporte, entre otras dado que la interacción es fundamental.

### **Descripción del Proyecto**

El proyecto consiste en la aplicación de localización y mapeo en plataformas móviles con mayor grado de interacción con el usuario por lo cual se ejecutó los siguientes puntos:

Se diseñará y construirá una plataforma móvil con la configuración diferencial planteado mediante el modelamiento cinemático que permita el uso de dos ruedas motrices y una de tipo castor para el desplazamiento. Los actuadores del dispositivo se determinarán mediante un análisis dinámico con la finalidad que puedan desplazar un peso combinado de 15 kg, consecuentemente la selección de actuadores contempla motores con torque alto con encoder integrado para la aplicación de odometría. Finalmente, la fuente de alimentación abastecerá a la plataforma completa de forma que su funcionamiento sea inalámbrico.

La plataforma estará provista de un sensor LIDAR y/o cámara KINECT las cuales permitan generar nubes de puntos y aplicar técnicas de SLAM en la obtención de representación geométricas. Con referencia al sistema electrónico se manejaría mediante un procesador computacional; el cual tendrá un sistema operativo compatible con plataformas robóticas, en conjunto con una placa para la comunicación de sensores y actuadores para reducir el uso de recursos computacionales. El control de la máquina y algoritmos se efectuarán con software libre mediante lenguajes de programación como C++ y/o Python para establecer una conexión en paralelo entre diferentes programas. La plataforma establecerá una relación entre puntos geométricos ubicados en un mapa bidimensional e información conceptual del área mediante técnicas ontológicas, con la posibilidad de realizar navegación objetiva hacia un lugar específico mientras suministra una información que el usuario sea capaz de interpretar fácilmente y desarrollar alto nivel de interacción entre estos elementos.

### **Estructura del Documento**

El trabajo de titulación se compone de cinco capítulos donde se describirá la información necesaria para el diseño y construcción de un robot móvil para mapeo semántico.

El capítulo uno presenta las generalidades y antecedentes en investigaciones que fundamentan el tema asociado al proyecto, también se destaca la importancia, el área de influencia y objetivo principal a cumplir.

El capítulo dos es una conceptualización teórica de los temas aplicados al proyecto entre los más destacados dentro del capítulo son: robótica móvil, SLAM, mapeo semántico.

El capítulo tres muestra el diseño del sistema de la plataforma al aplicar metodología mecatrónica al desarrollar los subsistemas los cuales darán paso a: la constitución de la plataforma, selección de actuadores, mapeo y conceptualización, y la integración de los mismos al probar los conceptos.

El capítulo cuatro se realiza la construcción del robot junto con la implementación de conceptualización al mapa generado para ejecutar entrenamiento; y describir las pruebas al comprobar el funcionamiento junto con los resultados de las mismas.

Finalmente, el capítulo cinco expone las conclusiones obtenidas de la ejecución del proyecto, así como recomendaciones para posibles trabajos a futuro.

## Capítulo II

### Estado del arte

El capítulo expone conceptos concernientes a la robótica móvil, en conjunto se analiza los principios de navegación además del método para localización y mapeo simultáneo conocido como SLAM. De igual forma se implementará el desarrollo de mapeo semántico como parte de la solución al planteamiento de la investigación.

#### Robótica Móvil

Un robot móvil es un sistema electromecánico el cuál posee la facultad para trasladarse autónomamente en ambientes no controlados, donde mediante un control programado mientras cumple tareas específicas. (Aguilera, Bautista, & Iruegas, 2007; Xia, 2015) Dentro de los robots móviles la autonomía juega un papel importante en su desempeño, debido a que conlleva realizar un proceso de razonamiento propio al tomar decisiones de acuerdo a datos recolectados por sensores, elige una ruta y cumple el objetivo evadiendo los obstáculos, es decir, desarrolla cierto grado de inteligencia sin intervención del ser humano. (Bambino, 2008)

#### Figura 7

*Robot AURORA (Universidad de Málaga)*

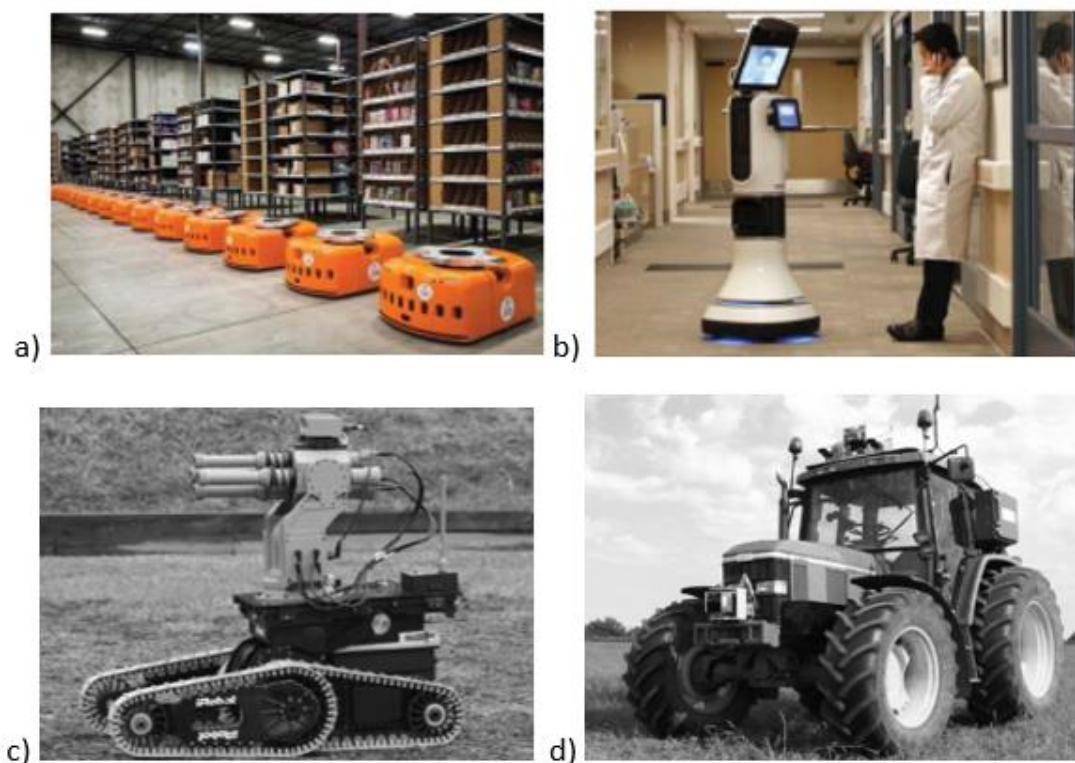


Nota: La imagen presenta al robot Aurora utilizado en invernaderos para la fumigación, desarrollado por la Universidad de Málaga (UMA). Sistema de sensores basado en ultrasónicos y cámara CCD (Dispositivo de Carga Acoplada) para control reactivo y tele operación. (Bambino, 2008)

La robótica móvil se destaca por sus aplicaciones en áreas referentes a doméstica y de servicio, agricultura, medicina, educación, exploración, industria, militar, vigilancia, seguridad, entre otras (Ortigoza R, 2012; Bambino, 2008). Sin embargo, la principal finalidad de los robots móviles es evitar riesgos para los seres humanos, en particular el acceso a lugares peligrosos; ejemplo de esto son el traslado y acopio de materiales pesado, mantenimiento en reactores nucleares, exploración subterránea, manejo de explosivos, etc. (Bambino, 2008; Xia, 2015)

### Figura 8

#### *Robots en aplicaciones comunes*



Nota: La figura presentan aplicativos para los robots: a) los Kiva, pertenecientes a la compañía Amazon en asistencia para sus empleados, b) RP-VITA permite realizar el control de los pacientes mediante dispositivos remotos, c) como aplicativo militar se observa a Warrior X700 y finalmente en d) el tractor 6410 de John Deere convertido en vehículo autónomo por la NREC. (Xia, 2015; Bambino, 2008)



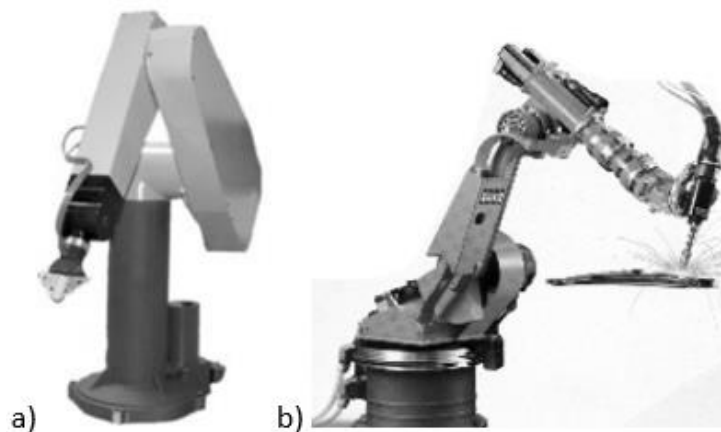
## Locomoción

La locomoción en robots móviles es vital para moverse libremente de acuerdo al entorno donde se desempeña (Siegwart & Nourbakhsh, 2011), por su movilidad se clasifican en:

- **Estacionario:** Son robots sujetos a una base fija, con frecuencia al final de la cadena cinemática están provistos de pinzas para la manipulación de objetos, o herramientas que permiten la ejecución en operaciones de soldadura, pintura, ensamblaje y maquinado entre otras. Un ejemplo de esto tipo se puede ver en la Figura 9.

### Figura 9

*Robots Estacionarios*



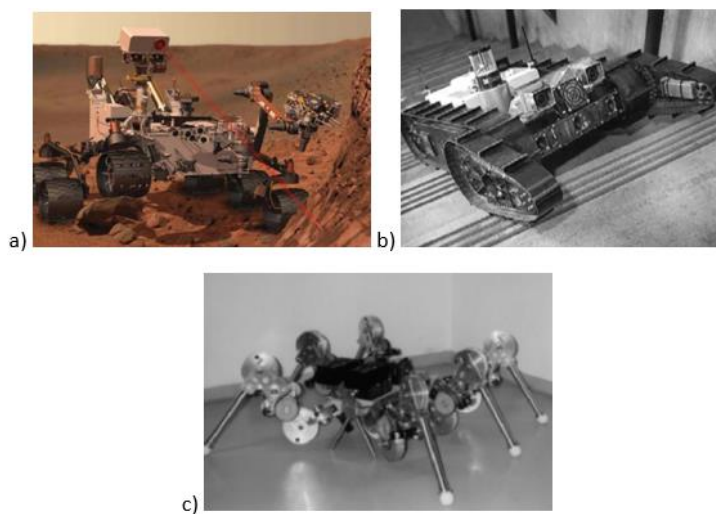
Nota: a) Robot industrial PUMA (Unimation) (Bambino, 2008) y en b) robot KUKA soldando (Siegwart & Nourbakhsh, 2011).

- **Terrestre:** Se trasladan por contacto directo con superficies, su aplicación principal es para transporte o exploración. En la Figura 10 se encuentra:
  - a) Un modelo provisto de ruedas las cuales presentan ventajas en la implementación, mínimo control de estabilidad y mayor eficiencia energética a velocidades altas.
  - b) El modelo oruga evita el deslizamiento al aumentar el área de contacto mejorando la maniobrabilidad.

- c) Las piernas buscan asemejar movimientos de carácter biológico sin embargo su costo elevado al implementarse se presenta como desventaja. (Rubio, Valero, & Llopies-Albert, 2019)

## Figura 10

Robots móviles terrestres

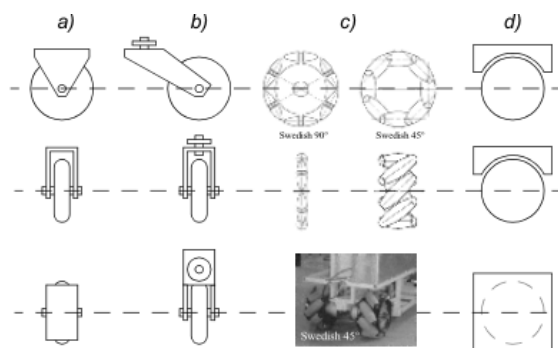


Nota: Ejemplos de robots móviles terrestres: a) Curiosity NASA (Xia, 2015) representa a los dispositivos que utilizan llantas, b) Scour USF (Bambino, 2008) utiliza orugas para subir escaleras debido su alta tracción y c) Lauron II (Siegwart & Nourbakhsh, 2011) es un hexápodo (seis patas) desarrollado por la universidad de Karlsruhe.

En las configuraciones para ruedas existen diferentes opciones indicadas en la Figura 11.

## Figura 11

*Tipos básicos de Ruedas*



Nota: a) Rueda convencional, b) rueda castor, c) omnirueda y d) esférica (Siegwart & Nourbakhsh, 2011)

- **Aéreo:** Es un robot provisto de hélices y autonomía energética para movilizarse en el espacio aéreo. Actualmente posee aplicativos en medicina, comercial, agricultura, recreación, vigilancia, correo o distribución de productos, un diseño se ve en la Figura 12. (Rubio, Valero, & Llopies-Albert, 2019)

**Figura 12**

*Robot Aéreo*

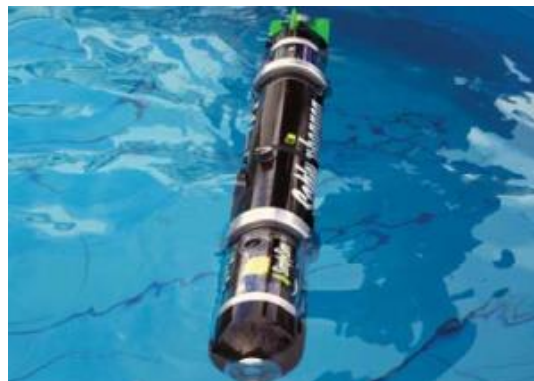


Nota: Robot aéreo ofrecido por la compañía FLYTREX, diseñado para ambientes suburbanos con capacidad de carga de 6,6 libras por una distancia de 3.5 millas de ida y vuelta. (FLYTREX, 2020)

- **Acuático:** estos dispositivos se diseñan con el propósito de operar sobre o debajo del agua sin control externo como el caso de la Figura 13.

**Figura 13**

*Robot Acuático*



Nota: El submarino Blackghost diseñado para actuar de forma autónoma. (Xia, 2015)

## Navegación

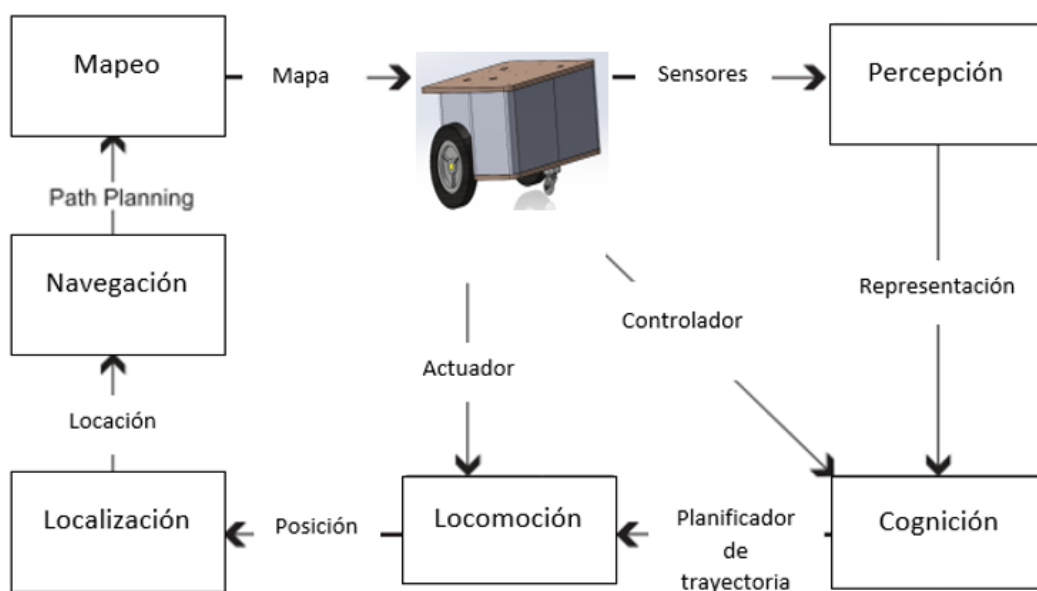
Para robots móviles la navegación es un proceso la cual busca situarse en un lugar determinado con precisión utilizando los valores de sensores, planificadores de trayectoria y su respectiva actualización mientras genera mapas en entornos desconocidos, y evitando colisiones al aplicar control sobre los actuadores. La navegación móvil en la robótica debe cumplir las siguientes tareas:

- Generar un modelo del mundo en forma de mapa.
- Programar una trayectoria sin colisiones partiendo desde un punto inicial hasta la posición objetivo.
- Desplazarse a lo largo de la trayectoria, evitando colisiones y obstáculos.

El éxito al cumplir las tareas antes mencionadas comprende desarrollar cuatro etapas que son: percepción, localización, cognición y control de movimiento. (Berry, 2012; Rubio, Valero, & Llopies-Albert, 2019)

**Figura 14**

*Ciclo en Robots Móviles*



Nota: Etapas que involucran el movimiento en robots móviles. (Berry, 2012)

## Percepción

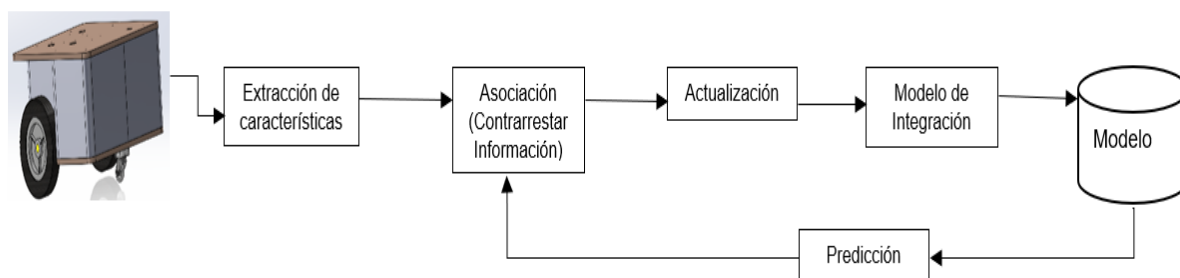
La autonomía en robots móviles es esencial al interactuar con el medio y desarrollar conocimiento del mismo por lo cual se implementan dispositivos para obtener información del mundo, seguido de la interpretación y organización de los datos recopilados lo que permite a las máquinas desenvolverse ante posibles cambios.

El uso de sensores y transductores es indispensable al recolectar información que será útil al ejecutar acciones por parte del robot para moverse libremente en su entorno y desarrollar aplicaciones las cuales involucran localización como son: vehículos autónomos, detección de obstáculos y humanos, reconocimiento de objetos, representación de entornos, etc.

La percepción en una plataforma móvil utiliza datos obtenidos mediante sensores y un modelo parcial del entorno, inicialmente los datos son preprocesados con la intención de reducir el ruido de transductores y eliminar errores, consecuentemente son comparados con información existente del ambiente, una vez que la información ha sido contrarrestada es actualizada para crear un modelo dinámico que permite la predicción de cambios en el entorno hasta la siguiente adquisición de datos. (Siciliano & Khatib, Handbook of Robotics, 2016)

**Figura 15**

*Diagrama del proceso para Percepción*



Nota: Figura realizada por el autor

### Clasificación de Sensores

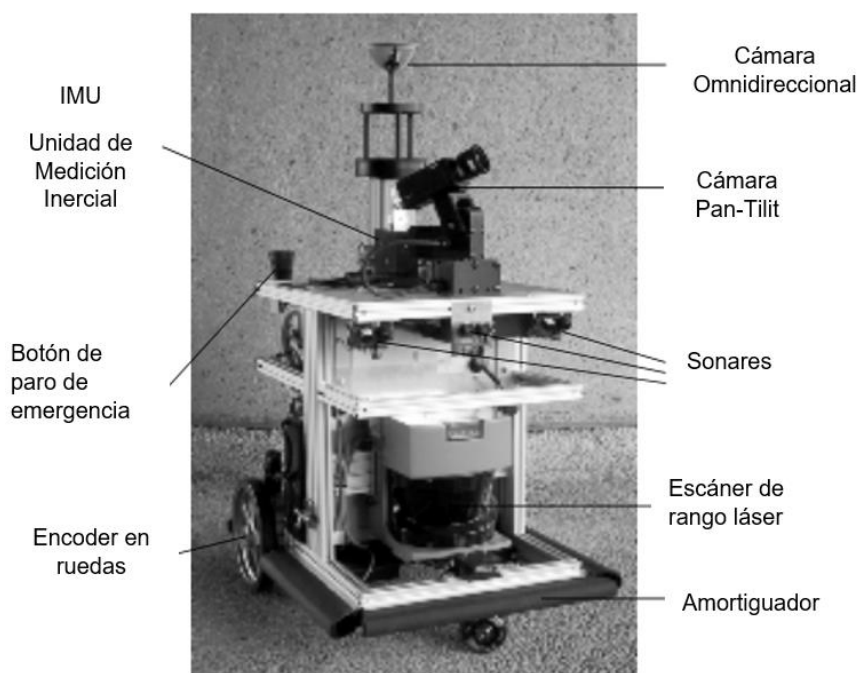
Existe una gran variedad de sensores que determinan el estado interno del robot mientras que otros permiten conocer la información externa, se clasifican en propioceptivo-exteroceptivo y pasivo-activo.

Los sensores propioceptivos miden valores internos en el robot como la velocidad, aceleración, torque, ángulos en las uniones, voltaje, etc. por otro lado, los exteroceptivos adquieren la información alrededor del dispositivo como distancia, amplitud de sonido, intensidad luminosa, etc.

Los sensores pasivos miden la energía del ambiente entrante al sensor como micrófonos, sondas de temperatura, cámaras, mientras que los dispositivos de tipo activo emiten energía hacia el ambiente y miden las propiedades de la reacción.

**Figura 16**

*BIBA Robot, BlueBotics SA.*



Nota: El robot BIBA está provisto de diferentes sensores con la finalidad de cumplir su objetivo, tal como se observa está provisto de diferentes tipos de sensores. (Siegwart & Nourbakhsh, 2011)

## **Localización**

Es una habilidad en los robots móviles donde permite determinar su posición y orientación dentro del mundo, facilitando su desplazamiento de forma autónoma en ambientes no controlados por lo que utiliza datos provenientes de sensores junto con técnicas las cuales involucran odometría e inercia. El primer método (odometría) mide la velocidad y distancia recorrida por el dispositivo, mientras tanto, el inercial determina la velocidad angular y aceleración con respecto a los tres ejes locales (x, y, z). (Permana Saputra, 2015)

La localización es relativa a los puntos de referencia dando origen al problema de estimación de estado debido a que las mediciones de sensores podrían contener ruido o medidas erróneas, por lo que es necesario una referencia con respecto a un modelo del mundo para la planificación de trayectorias. (Berry, 2012)

Existen tipos básicos del proceso las cuales son la localización relativa y la localización absoluta.

### ***Localización Relativa***

La localización relativa utiliza mediciones de velocidad, rango angular, distancia recorrida, trayectoria y tiempo de ejecución. La técnica por estimación o “dead reckoning” es la más utilizada dentro del proceso porque se basa en medidas de odometría y unidades inerciales disminuyendo recursos de procesamiento y facilitando su implementación, sin embargo, debido a las continuas iteraciones en medición el error se acumula y amplifica durante el cálculo por lo que no es recomendable en procesos continuos de larga duración. (Permana Saputra, 2015)

### ***Localización Global***

La localización global depende de la información provista por los sensores con respecto al marco referencial global, es decir, adquiere puntos de referencia eliminando el error acumulativo. Típicamente los sensores extraen información de objetos conocidos que poseen una posición a nivel global, los sensores utilizados para esta localización son: GPS (Sistema de Posicionamiento Global), LIDAR (Detección

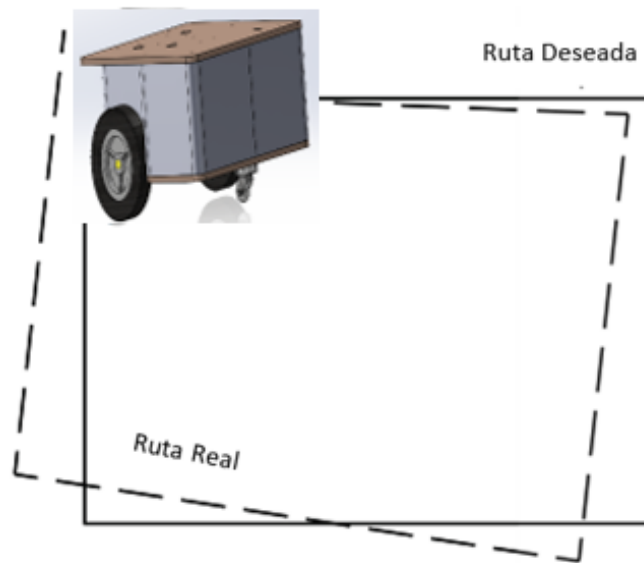
y rango de imágenes láser), RADAR (Detección y rango de radio) conjuntamente con sensores visuales y sonar. (Permana Saputra, 2015)

### **Odometría**

Es una técnica que permite la navegación al determinar la posición relativa en función de la medida anterior conjuntamente con la trayectoria y la velocidad, matemáticamente consiste en desarrollar un modelo del movimiento de ruedas y articulaciones que mueven al robot para integrar las acciones y desarrollar un modelo de la posición del vehículo en función del tiempo, se conoce como navegación de estima o deductiva. (Berry, 2012; Siegwart & Nourbakhsh, 2011) Presenta ventajas como autonomía, estimación de la posición y encontrarla en cualquier lugar siguiendo las trayectorias curvas, sin embargo, su desventaja es presentar error de posición acumulativo por lo que requiere una medición precisa de las velocidades en las ruedas mientras está en movimiento.

### **Figura 17**

#### *Error de Odometría*



Nota: Es una representación gráfica del error en odometría donde el camino recorrido podría divergir de la trayectoria establecida. (Berry, 2012)



Existen diferentes tipos de errores en odometría que involucran el diámetro y/o desalineación de ruedas, resolución finita del codificador y la frecuencia de muestreo del codificador. También hay errores influenciados por el entorno como la irregularidad del piso o deslizantes y objetos inesperados. Por último, se pueden presentar errores por mediciones imprecisas, modelos de control inexactos y características físicas lo que provoca que el robot ejecute una trayectoria similar al visto en la Figura 17.

### **Cognición**

En la cognición se analizan los datos entrantes de los sensores y ejecuta acciones correspondientes lo cual está a cargo del sistema de control. El proceso se lleva a cabo al determinar el entorno, destino y propósito de la máquina, permitiendo que la arquitectura cognitiva desarrolle la planificación de rutas a seguir por parte del robot para cumplir sus asignaciones; el resultado del nivel cognitivo es proveer la capacidad para decidir y ejecutar en función del desempeño esperado del robot. (Rubio, Valero, & Llopies-Albert, 2019)

Además, el sistema de control coordina el conjunto de datos entrantes y planificar el movimiento acorde al robot lo cual requiere que el dispositivo tenga provisto un modelo cognitivo encargado de representar a la máquina, el ambiente y como interactúan entre sí; entre estos se encuentran: la visión computarizada y reconocimiento de patrones en rastreo de objetos, algoritmos de mapeo para construcción de mapas junto con planificadores de movimiento u otras inteligencias artificiales. Ejemplo de lo mencionado anteriormente es el planificador, el cual determina como alcanzar su objetivo evitando la colisión con objetos, caídas, etc. (Rubio, Valero, & Llopies-Albert, 2019)

### **Mapeo**

El mapeo es un proceso el cual crea un mapa al unificar el movimiento del robot y los datos extraídos de los sensores, lo que requiere que la máquina asimile el medio mientras incorpora la nueva información al modelo. Esta representación del entorno incluye información concerniente a la posición del robot (localización) y mediante la cognición puede ejecutar tareas como la navegación y la

planificación de trayectoria, considerándose fundamental en ambientes los cuales poseen características tanto conocidas como imprevistas por lo que es necesario para cumplir los siguientes criterios: (Berry, 2012)

- El nivel del mapa debe ser el apropiado para que el robot cumpla sus objetivos.
- La precisión del mapa posea compatibilidad con los sensores del robot.
- La complejidad de cómputo del mapa concuerde con la complejidad de razonamiento.

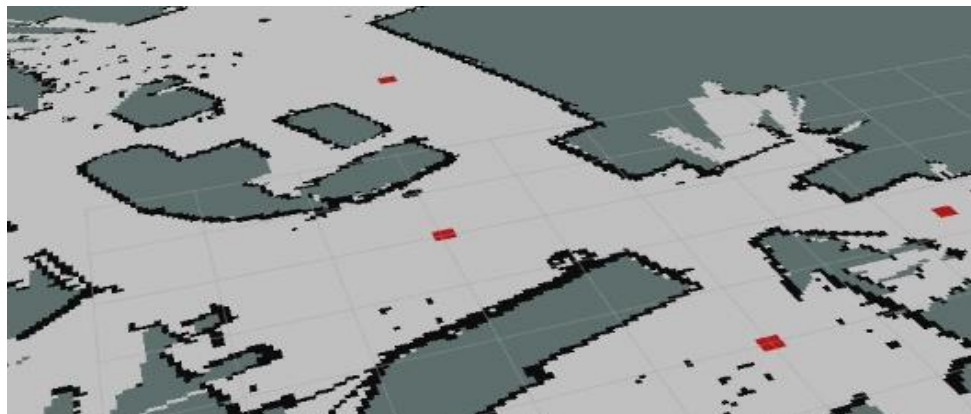
El lazo cerrado es una parte esencial dentro del mapeo porque permite al robot diferenciar un lugar visitado previamente con lo que optimiza su posicionamiento, reduciendo errores de odometría juntos con cambios bruscos. (Taderi & Xia, 2020)

### ***Mapa Métrico***

Los mapas métricos describen su medio en función de distancias donde la distancia en el mapa corresponde a una medida encontrada en un entorno real mapeado. Es decir, un robot móvil es posible calcular la distancia hacia un muro o una puerta, los mapas métricos son utilizados cuando es necesaria la precisión para auto localización y planificador de trayectoria (Buschka, 2005) .

### **Figura 18**

*Representación visual de un mapa métrico*



Nota: Representación bidimensional donde los espacios grises no están ocupados mientras tanto el negro delimita.

Realizado por el autor.

El mapa métrico es representado por rejillas, el entorno es representado mediante una matriz de celdas donde cada celda es una pequeña parte del mundo. Las celdas poseen uno o más valores representando el estado de la celda, comúnmente cuando la celda se encuentra ocupada es por un obstáculo.

### ***Mapa Topológico***

Un mapa topológico describe como los lugares u objetos en el mundo están conectados o relacionados entre sí, se representa gráficamente donde lugares son nodos y las conexiones son los bordes, de esta manera describe la estructura del entorno que lo rodea. Principalmente buscan determinar la relación si el robot puede navegar directamente de un área a otra. (Buschka, 2005)

### ***Mapa Híbrido***

El concepto de mapa híbrido implica la combinación de diferentes tipos de mapas con la finalidad de crear una representación más sólida del medio, por lo general se emplean de carácter topológico y métrico. Las ventajas en la aplicación son menor complejidad, localización mejorada, mayor razonamiento y planificación, mientras disminuye las exigencias métricas. (Buschka, 2005; Landsiedel, 2018)

### ***SLAM***

El mapeo y localización simultaneo (SLAM en inglés) es un proceso en el cual el robot es capaz de construir un mapa sin conocimiento preliminar del entorno y a su vez es utilizado para determinar su propia posición actualizando su estado continuamente permitiendo a los robots operar de forma autónoma. (Kumar, 2015; Permana Saputra, 2015; Taderi & Xia, 2020)

SLAM está formado por dos paradigmas presentados anteriormente, la localización y el mapeo. En sus inicios, las investigaciones plantearon el estudio por separado de estos conceptos, sin embargo, más adelante se establece una dependencia entre si dado que el mapa es necesario para determinar la localización, mientras la localización es esencial en el mapeo, por lo cual las técnicas de SLAM consideran un paralelismo en el cumplimiento de ambos conceptos. La odometría provee una estimación del

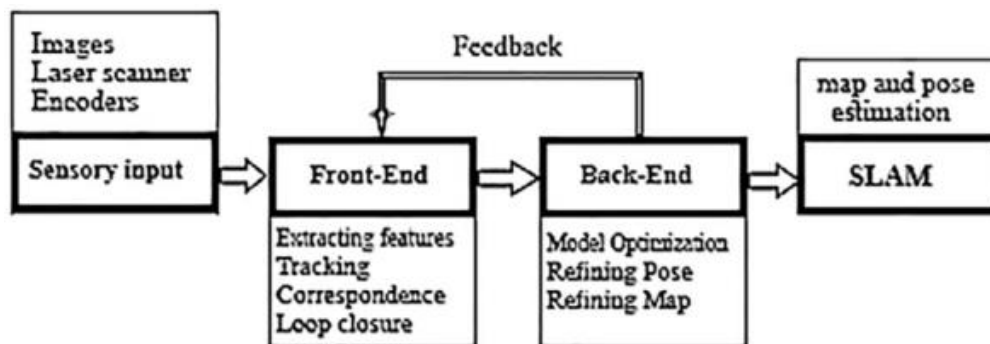
movimiento del robot mediante el sentido, varios tipos de odometría se han desarrollado en base a sensores tales como encoders, ultrasónicos, láseres, cámaras RGB; estos últimos han avanzado en la aplicación de odometría visual (VO), al precisar esta técnica se puede eliminar la incertidumbre en la navegación mejorando el proceso de mapeo.

### **Arquitectura de SLAM**

El sistema SLAM establece las siguientes etapas: la entrada de sensores, procesamiento frontal y procesamiento final ver la Figura 19. La primera sección se enfoca en la adquisición sensorial de información, la cual proviene de imágenes, encoders, laser de escáner, entre otros; en el procesamiento frontal se ingresan los datos para extraer características, asociar de datos a corto y largo plazo para transformar la información geométrica en modelos matemáticos y enviarlos al procesamiento final; donde se retroalimenta a la sección anterior para optimizar el modelo gráfico, depurar la posición y el mapa. Finalmente, al completar este proceso se obtiene el una localización y mapeo simultáneos permite obtener el mapa y estimar la posición. (Taderi & Xia, 2020)

**Figura 19**

*Representación de la arquitectura de SLAM*



Nota: Se observa los componentes que conlleva la arquitectura en un sistema de mapeo y localización simultánea donde interviene de forma secuencial la adquisición y el procesamiento de datos, optimización de mapa y posición para finalmente presentar sus resultados, obtenida de (Taderi & Xia, 2020).

## Métodos de SLAM

Existen una variedad de métodos para la localización y mapeo simultáneo que permiten obtener representaciones bidimensionales (2D) y tridimensionales (3D) del espacio. A continuación, se mencionan algunos de ellos:

- **HectorSLAM:** Es una técnica bidimensional, la cual utiliza principalmente información proveniente de sensores láser para obtener una nube de partículas sin la necesidad de odometría. La estimación de posición se basa en alinear los puntos finales de la nube con el mapa que sigue generándose de manera que la proyección sobre el mismo permite estimar las probabilidades de ocupación. (Santos, Portugal, & Rocha, 2013)
- **Gmapping:** Es un algoritmo de remuestreo adaptativo basado en filtros de partículas que toma en consideración tanto el movimiento del robot y las recientes detecciones, reduciendo la incertidumbre en la predicción de la posición del móvil. (Grisetti, Stachniss, & Burgard, OpenSLAM, s.f.)
- **KartoSLAM:** Es un enfoque gráfico desarrollado por SRI International's Karto Robotics, el cual establece una posición del móvil a lo largo de la trayectoria y su conjunto de mediciones sensoriales como nodos, los cuales se conectan por arcos que representan el movimiento entre ubicaciones continuas. La construcción del mapa procede al encontrar la configuración espacial de los nodos que son consistentes con las restricciones de los arcos. (Santos, Portugal, & Rocha, 2013)
- **ORB-SLAM:** Es un tipo de vSLAM, el cual emplea sensores de imagen como cámaras de tipo monocular, estereotípicas y/o RGB-D, para calcular su posición y orientación en tiempo real con respecto al medio donde se desplaza mientras construye una representación 3D de lo que se observa. (Mur-Artal, Tardós, Montiel, & Gálvez-López, 2016; The MathWorks, Inc, 2022)

## Detección de objetos

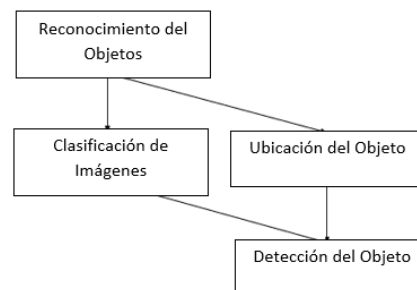
Es una técnica de visión artificial que busca determinar la ubicación individual del objeto y la clase a la cual pertenece. (Zhao, Zheng, Xu, & Wu, 2019)

La acción contempla las siguientes etapas:

- **Selección de región:** Debido a las diferentes características de los objetos en las imágenes se utiliza una ventana deslizante de escala múltiple, la cual permite encontrar todas posibles posiciones de los objetos, sus deficiencias también son obvias, sin embargo, su costo computacional es bastante alto.
- **Extracción de características:** Busca las características visuales que puedan proporcionar una representación semántica, con lo cual busca emular el comportamiento del cerebro humano.
- **Clasificación:** Es necesario un clasificador para diferenciar un objetivo de las demás categorías.

### Figura 20

*Descripción de la detección de objetos*



Nota: Figura realizada por el autor para representar el procedimiento de detección de objetos.

## Ontología

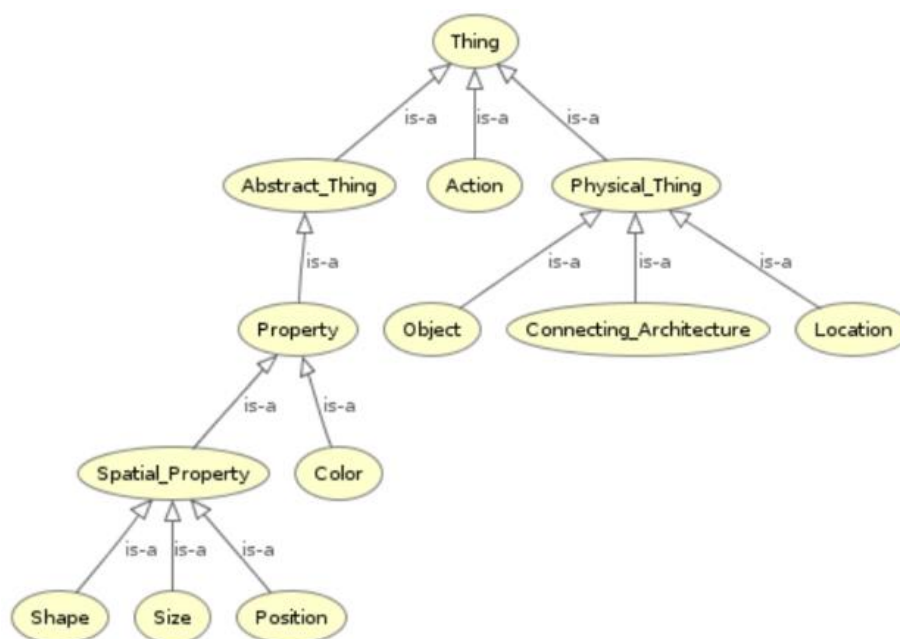
Es el método para describir y representar el conocimiento de las entidades pertenecientes a un dominio y las relaciones que existen entre ellas, con lo cual busca cumplir con dos objetivos (Jean-Baptiste, 2021):

- Ejecutar razonamiento automático: Las ontologías permiten realizar deducciones lógicas, usando el denominado razonador.
- Enlazar conocimiento de diferentes fuentes: Lo realiza al aplicar direcciones de internet denominadas IRI (Internationalized Resource Identifier) para identificar diferentes entidades u objetos.

Para la aplicación de las ontologías se programan utilizando lenguajes de ontología. OWL (Web Ontology Language), es el lenguaje para construir ontologías, reconocido como estándar a nivel mundial para codificar y compartir ontologías y está diseñado para soportar la Web Semántica. Una ontología OWL dispone de clases, relaciones, atributos, axiomas formales e instancias, con lo cual permite la información puede ser procesada de forma automática e incluso verificar su consistencia. (W3C, 2004; No Magic, Inc., 2022) Entre sus ventajas destaca las similitudes que comparte con la programación orientada a objetos (POO) como se muestra en la Tabla 1.

**Figura 21**

*Visualización de una ontología*



Nota: Representación gráfica de las relaciones ontológicas realizado por (Capobianco, y otros, 2015).

**Tabla 1***Similitudes entre POO y la ontología*

<b>POO</b>	<b>Ontología</b>
Objeto	Entidad
Módulo	Ontología
Clase	Clase
Herencia de Clase	Llamada relación "is-a"
-	Herencia de propiedad
Instancia	Individual
Atributo	Propiedad
Valor de un atributo para una instancia	Relación
Nombre de clase	IRI
Tipos de datos	Tipos de datos
Método	-
-	Constructor Lógico
	Restricción
	Disjoint

Nota: En la tabla se observa la homologación entre las entidades para la programación orientada a objetos y ontologías.

### **Mapeo Semántico**

En la actualidad los robots son capaces de generar mapas para determinar su localización y navegar, sin embargo, los móviles requieren de una habilidad adicional para poder extraer información y aprender del medio tal como lo haría a un ser humano.



Un mapa semántico para un robot móvil es un mapa extendido que además de contener las características espaciales consta con información sobre cada tipo de lugar mediante lo cual le permite un alto grado de representación conceptual del entorno e incluso objetos dentro del mismo dotando de un alto nivel de interacción entre el usuario y la máquina (Nüchter Andreas, 2008)

### ***Proceso para la generación de mapa semántico***

El proceso para la construcción de un mapa semántico se compone esencialmente de cuatro etapas:

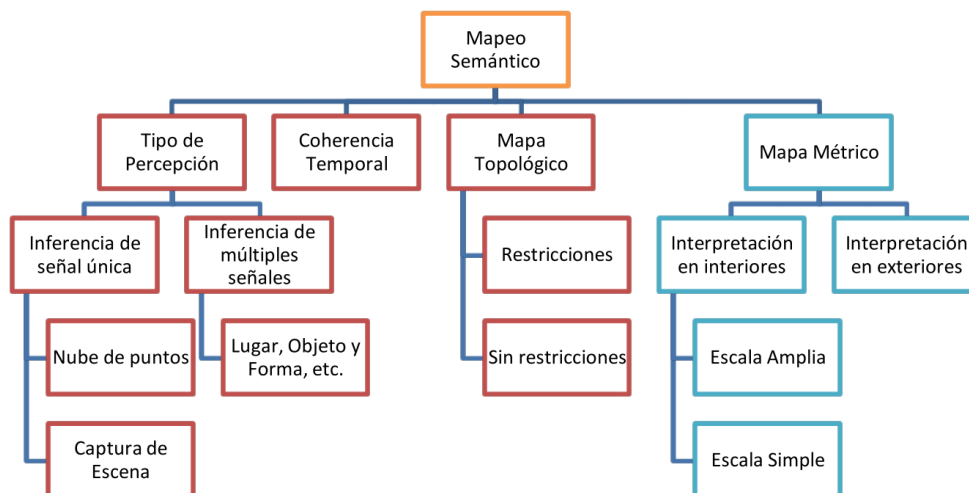
- Percepción: Adquiere la información del entorno mediante sensores.
- Representación: Se construyen mapas del entorno aplicando técnicas de SLAM, los cuales contienen información métrica y/o topológica.
- Conceptualización: Proceso para detección de objetos.
- Mapa Semántico: Relacionar las entidades con el mapa.

### ***Taxonomía del Mapa Semántico***

De forma ilustrativa se representa la taxonomía con los enfoques más conocidos para mapeo semántico como se muestra en la Figura 22. Destacan el método de señal única que proporciona la inferencia del lugar, a diferencia de las señales múltiples con respecto a objetos, lugares y formas lo que proporciona un mayor conocimiento.

La forma topológica presenta una abstracción del entorno en términos de gráficos, mientras tanto la utilización de mapas métricos es un componente complementario que sirve para fortalecer los métodos anteriores. (Kostavelis & Gasteratos, Robotics and Autonomous Systems, 2014)

Figura 22

*Metodologías para mapeo semántico*

Nota: Representación de los enfoques en mapeo semántico realizada por el autor.

### Resumen del Capítulo

La robótica móvil ha desarrollado dispositivos provistos con la capacidad para desplazarse en diferentes entornos al estar provisto de sensores, con lo cual desarrollan una percepción de sí mismos lo que les permite realizar tareas que involucran de riesgo para el ser humano.

Para estimar su posición utilizan odometría que involucra el cálculo de la distancia recorrida por la máquina, así como información de carácter inercial, lo que permite que utilice esta información para determinar su localización dentro de entornos desconocidos, un ejemplo es la aplicación de métodos de SLAM, con lo cual es capaz de generar mapas que doten de información al usuario y su aplicación.

La aplicación de los robots móviles es extensa, sin embargo, el enfoque tomado radica sobre su aplicación para detectar objetos y relacionar con mapas dotados con información métrica, lo cual resulta en la generación de los denominados mapas semánticos, que son una representación ampliada del entorno al proveer información del lugar donde se desempeña de manera que el ser humano sea capaz de interpretar dicha información.

## Capítulo III

### Metodología y Diseño

El capítulo indica el proceso para la aplicación de las metodologías mecatrónicas existentes para el desarrollo del producto, con la cuál, mediante la creación de subsistemas y la implementación del diseño concurrente se procede a tomar la mejor decisión para la constitución a nivel del hardware (parte física) y software (programación) de tal forma tener un diseño optimó para la interacción humano - máquina.

#### **Definición de necesidades**

Definir las necesidades del sistema es fundamental para cumplir los objetivos propuestos de la manera más optima debido a la influencia directa en la selección de materiales, el diseño mecánico y electrónico, programación, etc. Por lo tanto, la información se analiza e interpreta con el fin de establecer las características técnicas que justifiquen el producto final.

En la Tabla 2, son enunciadas las necesidades para la plataforma y el concepto asociado a cada una respectivamente, de igual manera muestra que la plataforma tenga la capacidad para ejecutar SLAM mientras reconoce imágenes con una cámara RGB, consecuentemente se enfatiza en aspectos mecánicos los cuales influyen en el desempeño de la máquina.

#### **Definición de Especificaciones**

Las necesidades expuestas en la sección anterior son interpretadas a lenguaje técnico con la finalidad de crear métricas que involucran una o más necesidades permitiendo obtener las especificaciones técnicas del sistema y verificar su cumplimiento.

En la Tabla 3 se muestran las especificaciones provistas de valores ideales a conseguir por la plataforma y sus unidades correspondientes.

**Tabla 2***Necesidades para diseño de la plataforma móvil*

<b>N</b>	<b>Necesidad</b>	<b>Concepto</b>
<b>1</b>	Fabricación sencilla	Mecánico
<b>2</b>	Resistente	Mecánico
<b>3</b>	Peso Liviano	Mecánico
<b>4</b>	Tamaño compacto	Mecánico
<b>5</b>	Aspecto Amigable	Mecánico
<b>6</b>	Robot diferencial	Función
<b>7</b>	Autónomo	Función
<b>8</b>	Velocidad ajustable	Función
<b>9</b>	Aplicación de SLAM	Función
<b>10</b>	Control de motores	Control
<b>11</b>	Software Libre	Control
<b>12</b>	Procesamiento de imágenes	Control
<b>13</b>	Utilización cámara RGB	Control
<b>14</b>	Interacción entre datos de sensores	Control
<b>15</b>	Interactivo con usuario	Comunicación/Monitoreo
<b>16</b>	Inalámbrico	Comunicación/Monitoreo
<b>17</b>	Visualización de Resultados	Comunicación/Monitoreo
<b>18</b>	Costo relativamente bajo	Costo

Nota: La tabla muestra los requerimientos planteados por el usuario que ha de cumplir la plataforma.

**Tabla 3***Especificaciones técnicas de la plataforma.*

N. de Métrica	N. de Necesidad	Métrica	Magnitud	Unidad
1	1, 2, 3, 4	Masa Total del robot	< 15	kg
2	1, 4	Altura del robot	< 1.25	cm
3	4, 5, 6	Diámetro del robot	< 45	cm
4	1, 3, 4	Tiempo de ensamblaje	< 15	min
5	3, 8, 10	Velocidad de desplazamiento	<=0.25	m/s
6	3,10	Cantidad de Motores	2	unidades
7	1, 3, 7, 8, 13, 16	Duración de Batería	> 30	min
8	1, 2, 3, 4, 7	Deformación del material	< 1	mm
9	1, 9, 11, 14	Tipo de código	Abierto	-----
10	12, 13	Procesamiento de Imágenes	>400	GFLOPS
11	1, 3, 4, 16	Capacidad de baterías	>2	Ah
		mínimo		
12	18	Costo	< 700	\$

Nota: Presenta las métricas establecidas partiendo necesidades en lenguaje técnico.

### **Definición de Estructura Funcional**

Para generar conceptos es esencial analizar al sistema separándolo mediante subsistemas, donde permite entender de forma más simple su respectiva función y desempeño. (Ulrich & Eppinger, Diseño y

desarrollo de productos, 2013) Se inicia tomando al conjunto como una caja negra provista de las entradas y salidas correspondientes a los resultados deseados, para el funcionamiento correcto de la plataforma robótica móvil, véase la Tabla 4. En la Figura 23, son presentadas las funciones comprenden al funcionamiento interno de la plataforma, así como la sinergia que existen entre las mismas.

La Figura 24 muestra la agrupación de las funciones con la finalidad de formar subsistemas de manera que facilite el desarrollo del prototipo. En la Figura 25 se presenta los subsistemas obtenidos para el desarrollo del robot junto con la función a cumplir respectivamente.

**Tabla 4**

*Subsistemas establecidos para el diseño de la plataforma móvil*

Número	Subsistema	Función
1	Estructura	Diseño y Material
2	Potencia y Tracción	Energización y Actuadores
3	Percepción	Odometría y Mapeo
4	Procesamiento y Detección	Algoritmos para detectar objetos
5	Mapeo y Ontologías	Algoritmos para relacionar mapa y los objetos

Nota: Subsistemas para el desarrollo del prototipo

**Figura 23**

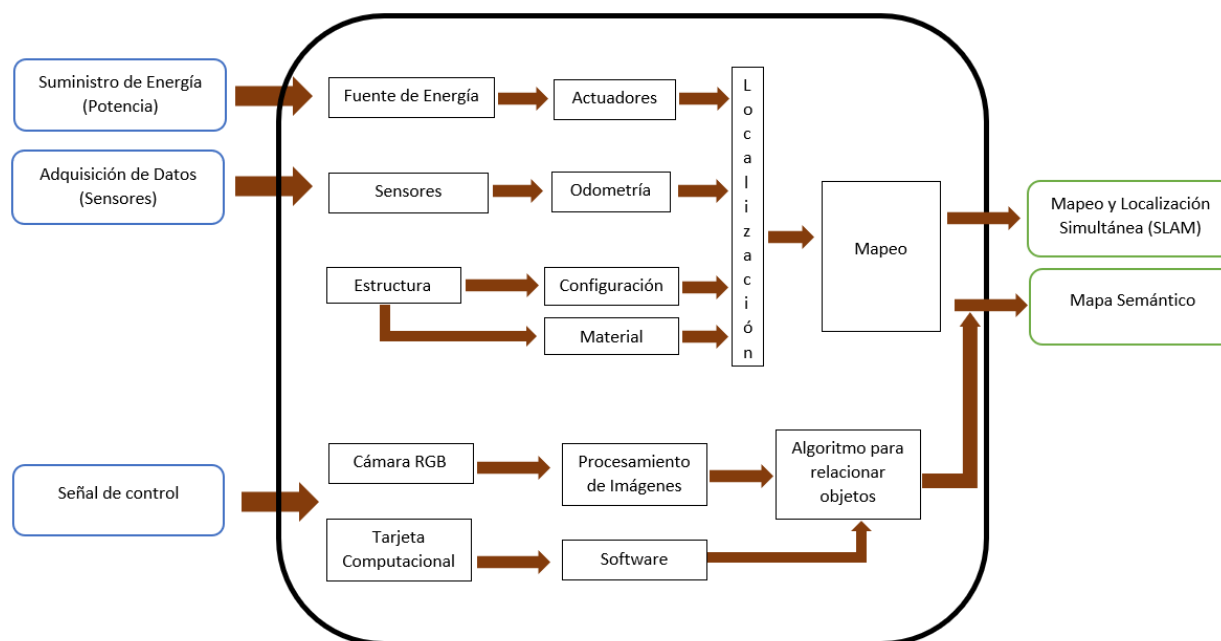
*Diagrama de la funcionalidad del robot móvil*



Nota: Sistema visto como caja negra realizada por autor.

Figura 24

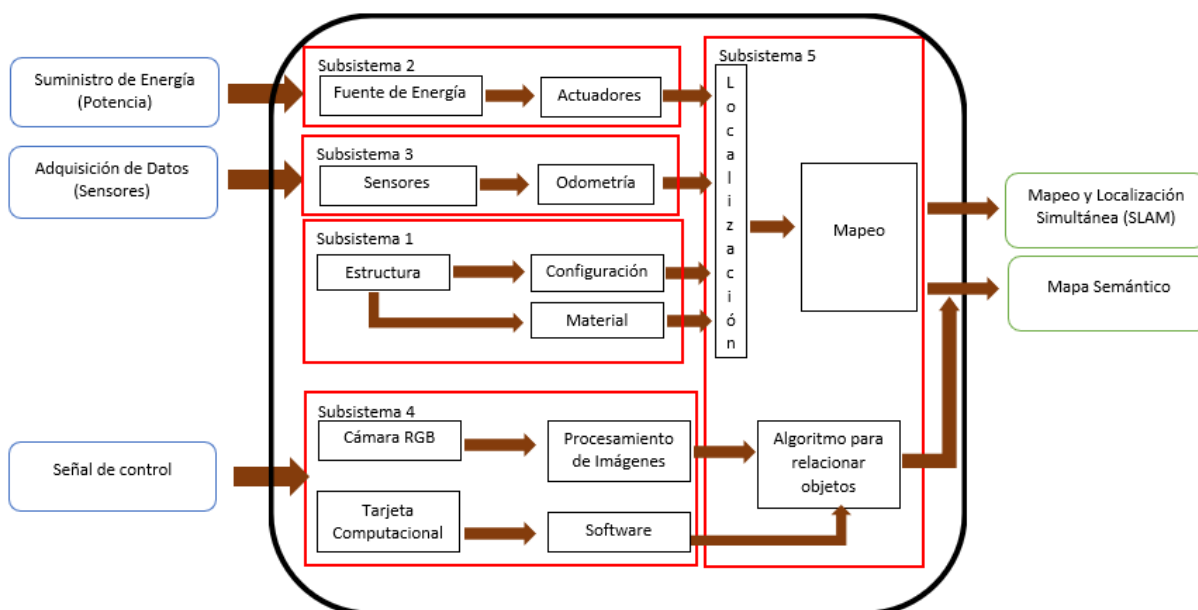
Diagrama de las funciones para la funcionalidad de la plataforma



Nota: Desarrollo del sistema. Realizada por autor.

Figura 25

Diagrama de los subsistemas para la plataforma



Nota: Realizada por autor.

El planteamiento en (Ulrich & Eppinger, Diseño y desarrollo de productos, 2013), establece dos etapas en el desarrollo del diseño, la primera etapa comprende proporcionar alternativas que serán cotejadas en una matriz de priorización mientras que la segunda fase comprende establecer las especificaciones finales con las opciones mejor valoradas en el paso anterior.

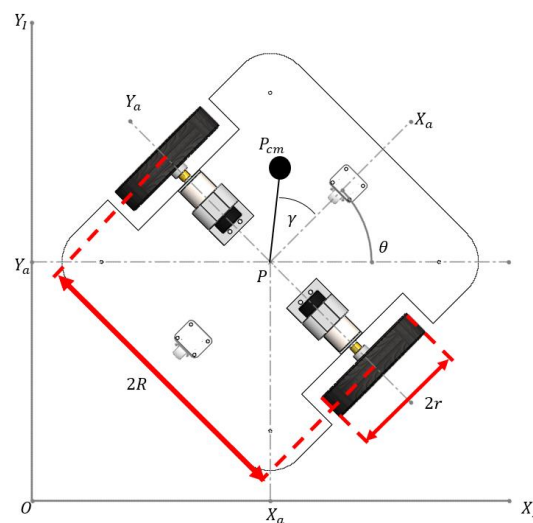
### Estructura y movilidad

Para el diseño de la movilidad se optó por una configuración diferencial combinado con una rueda tipo castor, su modelo cinemático permite un equilibrio entre el control sobre los actuadores y el desplazamiento del dispositivo. El sistema diferencial consiste de dos ruedas motrices montadas a lo largo de un eje común donde cada rueda puede moverse independientemente en dirección de avance o reversa.

### Modelo Cinemático

**Figura 26**

*Robot móvil diferencial*



Nota: Realizada por autor.

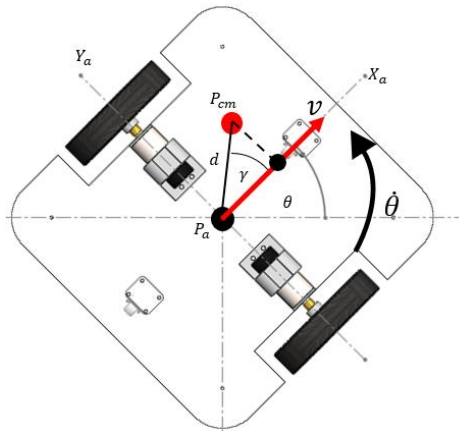
El modelo cinemático de un robot diferencial nos permite relacionar la posición dada por las componentes para traslación (componentes en  $x$ ,  $y$ ) y rotación (ángulo de giro  $\theta$ ) dado el valor de las



velocidades en las ruedas; para lo cual se toma en cuenta el sistema de coordenadas en un marco local, el cual está fijado al plano donde se mueve la plataforma ( $Xl, Yl$ ) y la referencia de coordenadas del robot que comprende el marco referencial sujeto a la plataforma y se mueve junto con ella ( $Xa, Ya$ ), ver la Figura 26.

**Figura 27**

*Velocidades del robot diferencial*



Nota: Realizada por autor.

Donde  $L$  es la distancia entre los centros de las ruedas;  $V_r$ ,  $V_l$  corresponden a las velocidades lineales de las ruedas derecha e izquierda respectivamente:

$$V_r = \omega \left( R + \frac{L}{2} \right) \quad (1)$$

$$V_l = \omega \left( R - \frac{L}{2} \right) \quad (2)$$

$$\omega = \frac{V_r - V_l}{L} = \frac{V_r + V_l}{2R} \quad (3)$$

La velocidad lineal de la plataforma en la referencia del robot es el promedio de las velocidades lineales de sus ruedas motrices.

$$V = \frac{V_r + V_l}{2} = r \frac{\omega_r + \omega_l}{2} = r \frac{\omega_r - \omega_l}{L} \quad (4)$$

$$R = \frac{L V_l + V_r}{2 V_r - V_l} \quad (5)$$

$$\dot{X}_a^r = \frac{r}{2} (\omega_r + \omega_l) \quad (6)$$

$$\dot{Y}_a^r = 0 \quad (7)$$

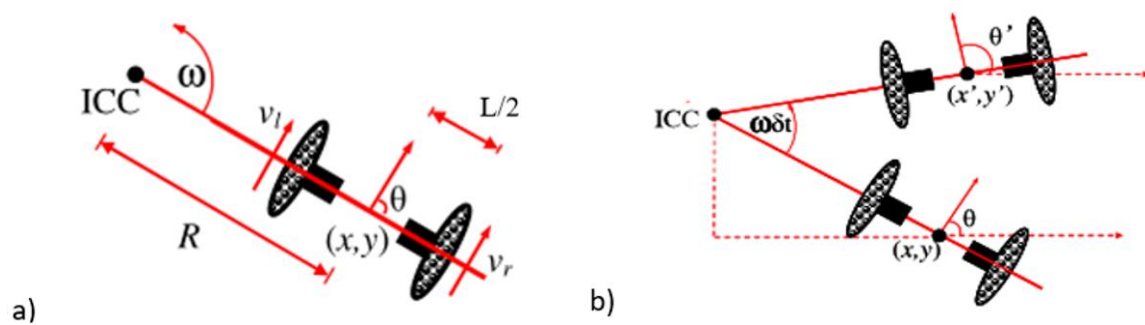
$$\dot{\theta}_a^r = \frac{r}{L} (\omega_r - \omega_l) \quad (8)$$

$$\begin{bmatrix} \dot{X}(t) \\ \dot{Y}(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ 0 & 0 \\ \frac{r}{L} & -\frac{r}{L} \end{bmatrix} \cdot \begin{bmatrix} \omega_r(t) \\ \omega_l(t) \end{bmatrix} \quad (9)$$

Para que el robot se mueva variamos las velocidades de las ruedas mientras que al rotar lo hace alrededor de un punto que se encuentra entre la intersección de cada línea imaginaria desde el eje común de las ruedas, este punto de rotación se determina como Centro de Curvatura Instantánea (ICC). (Siegwart & Nourbakhsh, 2011)

**Figura 28**

*Representación giro con respecto a ICC*



Nota: Representación del centro de curvatura instantánea para un robot diferencial ( (Siegwart & Nourbakhsh, 2011)).

$$ICR = [ICR_x, ICR_y] = [x - R \sin \theta, y + R \cos \theta] \quad (10)$$

Matriz  $R(\theta)$  es la matriz ortogonal de rotación:

$$R(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (11)$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \omega \delta t & -\sin \omega \delta t \\ \sin \omega \delta t & \cos \omega \delta t \end{bmatrix} \begin{bmatrix} x - ICR_x \\ y - ICR_y \end{bmatrix} + \begin{bmatrix} ICR_x \\ ICR_y \end{bmatrix} \quad (12)$$

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} \cos \omega \delta t & -\sin \omega \delta t & 0 \\ \sin \omega \delta t & \cos \omega \delta t & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x - ICR_x \\ y - ICR_y \\ \theta \end{bmatrix} + \begin{bmatrix} ICR_x \\ ICR_y \\ \omega \delta t \end{bmatrix} \quad (13)$$

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} \cos \omega \delta t & -\sin \omega \delta t & 0 \\ \sin \omega \delta t & \cos \omega \delta t & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x - x + R \sin \theta \\ y - y - R \cos \theta \\ \theta \end{bmatrix} + \begin{bmatrix} x - R \sin \theta \\ y + R \cos \theta \\ \omega \delta t \end{bmatrix} \quad (14)$$

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} \cos \omega \delta t & -\sin \omega \delta t & 0 \\ \sin \omega \delta t & \cos \omega \delta t & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R \sin \theta \\ R \cos \theta \\ \theta \end{bmatrix} + \begin{bmatrix} x - R \sin \theta \\ y + R \cos \theta \\ \omega \delta t \end{bmatrix} \quad (15)$$

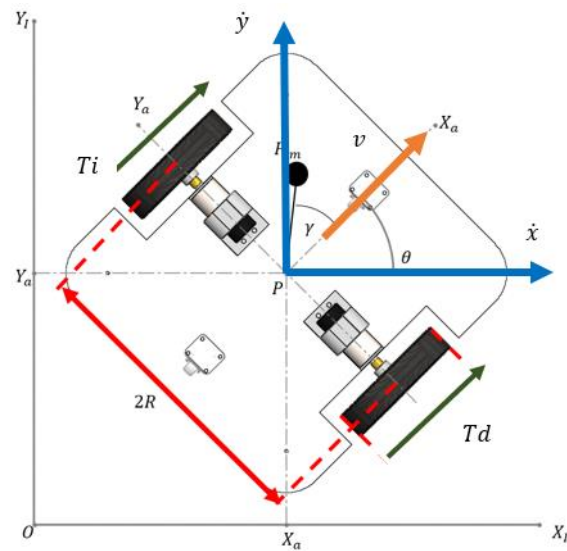
$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} \cos \omega \delta t \cdot R \sin \theta - \sin \omega \delta t \cdot R \cos \theta \\ \sin \omega \delta t \cdot R \sin \theta + \cos \omega \delta t \cdot R \cos \theta \\ \theta \end{bmatrix} + \begin{bmatrix} x - R \sin \theta \\ y + R \cos \theta \\ \omega \delta t \end{bmatrix} \quad (16)$$

### **Modelamiento Dinámico**

En el modelamiento dinámico se toma en cuenta las fuerza, energía, masa y la inercia. La representación matemática se realiza mediante la formulación de Lagrange, debido a que la energía potencial es constante  $U = 0$ , además se desprecia el valor de inercia para los neumáticos dejando solo la masa del sistema. Se analiza el sistema con respecto a la referencia global.

Figura 29

Energía sobre el sistema.



Nota: En la figura se observan las energías actuantes sobre el sistema, figura realiza por el autor.

$$L = \frac{1}{2}m \dot{x}^2 + \frac{1}{2}m \dot{y}^2 + \frac{1}{2}J\dot{\theta}^2 \quad (17)$$

Donde:

$T$  = Suma de la energía cinética del sistema

$U$  = Suma de la energía potencial del sistema

$m$  = Masa del sistema

$J$  = Inercia del sistema

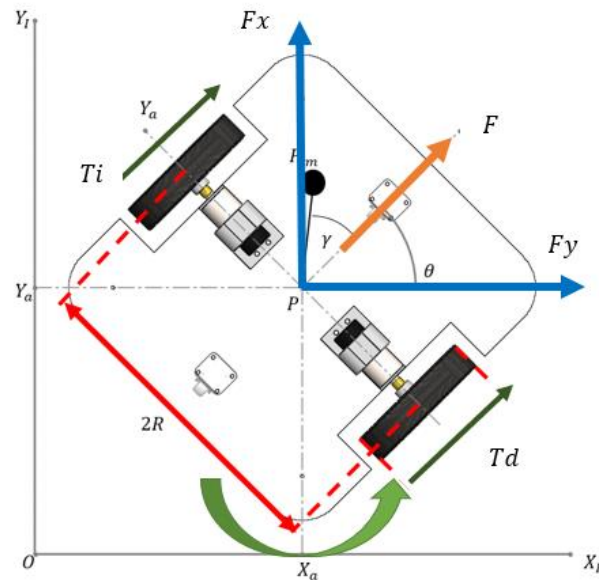
$\dot{x}$  = Velocidad lineal en el eje x global

$\dot{y}$  = Velocidad lineal en el eje y global

$\dot{\theta}$  = Velocidad angular del sistema

**Figura 30**

Fuerzas actuantes en el sistema



Notas: En la figura se muestran las figuras sobre el sistema que actúan sobre el sistema. Figura realiza por el autor.

$$f_k = Fx + Fy + M$$

Donde:

$f_k$  = Sumatoria de las fuerzas que actúan sobre el sistema.

$F$  = Fuerza resultante del robot.

$M$  = Momento del robot.

La Figura 31 representan las fuerzas que afectan a la plataforma móvil, la fuerza proviene del torque entregado por los motores, mientras que el momento tiene su origen en el torque que presenta el cuerpo, por lo cual se obtiene la siguiente ecuación:

$$Fx = \frac{1}{r} (Td + Ti) \cos(\theta) \quad (18)$$

$$Fy = \frac{1}{r} (Td + Ti) \sin(\theta) \quad (19)$$

$$M = \frac{R}{r} (Td - Ti) \quad (20)$$

Donde:

$L$  = Distancia entre el centro del robot y la rueda.

$r$  = Radio de la rueda.

$Td$  = Torque del motor derecho.

$Ti$  = Torque del motor izquierdo.

Para el planteamiento se desprecian la fricción y el amortiguamiento, se procede a analizar cada estado del sistema de la siguiente manera:

$$m \ddot{x} - \frac{1}{r}(Td + Ti) \cos(\theta) - \lambda \sin(\theta) = 0 \quad (21)$$

$$m \ddot{x} - \frac{1}{r}(Td + Ti) \sin(\theta) - \lambda \cos(\theta) = 0 \quad (22)$$

$$J\ddot{\theta} - \frac{L}{2r}(Td + Ti) = 0 \quad (23)$$

De forma matricial, el modelamiento se presenta así:

$$M(q)\ddot{q} + V(q, \dot{q}) + F(\dot{q}) + G(q) + T = E(q)u - A^T(q)\lambda \quad (24)$$

### ***Dinámica Inversa***

El modelo para el robot se plantea en las siguientes matrices:

$$\tilde{M} = S^T M S \quad (25)$$

$$\tilde{M} = \begin{bmatrix} m & 0 \\ 0 & J \end{bmatrix} \quad (26)$$

$$\tilde{V} = S^T M \dot{S} v + S^T (V + F - G) \quad (27)$$

$$\tilde{E} = \frac{1}{r} \begin{bmatrix} 1 & 1 \\ L & -L \\ 2 & -2 \end{bmatrix} \quad (28)$$

Se utilizan espacios de estados de la siguiente manera:

$$\dot{x} = \begin{bmatrix} S v \\ -\tilde{M}^{-1} \tilde{V} \end{bmatrix} + \begin{bmatrix} O_{n \times r} \\ \tilde{M}^{-1} \tilde{E} \end{bmatrix} u \quad (29)$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{v} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} v \cos(\theta) \\ v \sin(\theta) \\ \omega \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{Td + Ti}{mr} \\ \frac{L(Td + Ti)}{2Jr} \end{bmatrix} \quad (30)$$

Los torques se plantean así:

$$Td = \frac{\dot{v}mr}{2} + \frac{\dot{\omega}Jr}{2R} \quad (25)$$

$$Ti = \frac{\dot{v}mr}{2} + \frac{\dot{\omega}Jr}{2R} \quad (26)$$

Con las ecuaciones obtenidas anteriormente se determina que el torque a soportar por la plataforma es de 2.64 Nm.

El material a elegir será usado en la construcción del bastidor para la plataforma de ahí que su elección es fundamental debido a que soportará el peso de los componentes y facilidad para ser maquinado, entre el aspecto se toma en cuenta los siguientes:

- Peso
- Disponibilidad de compra
- Resistencia
- Costo

El factor principal que influye entre los criterios propuestos es el costo con 50% en su ponderación seguidos con por la resistencia y disponibilidad en el mercado como se observa en la Tabla 5. Entre las alternativas se encuentra: el acero que posee resistencia y peso altos, el filamento para impresión 3D posee una alta resistencia y bajo peso, sin embargo, su adquisición es bastante elevado; el material compuesto MDF posee menor costo y alta resistencia, aunque requiere de corte láser y el acrílico que presenta beneficios similares al MDF, pero puede resultar frágiles antes fuerzas altas.

**Tabla 5**

*Matriz de ponderación con los criterios para el material base*

Criterio	Peso	Disponibilidad	Resistencia	Costo	Suma+1	Ponderación
<b>Peso</b>	-	0.2	1	0.1	2.3	0.07
<b>Disponibilidad</b>	5	-	0.2	1	7.2	0.21
<b>Resistencia</b>	1	5	-	0.2	7.2	0.21
<b>Costo</b>	10	1	5	-	17	0.50
<b>Total</b>					33.7	1

Nota: La matriz indica como se pondera las opciones.

Las alternativas propuestas son evaluadas con respecto a los criterios propuestos anteriormente.

La Tabla 6 es un ejemplo de la evaluación de las opciones en base al criterio (peso), sucesivamente se realiza el proceso con el resto de criterios.

**Tabla 6**

*Matriz de ponderación con respecto al peso*

Peso	Acero	PLA-3D	MDF	Acrílico	Suma+1	Ponderación
<b>Acero</b>	0	0.1	0.1	0.1	1.3	0.03
<b>PLA-3D</b>	10	0	5	5	21	0.42
<b>MDF</b>	10	0.2	0	5	16.2	0.32
<b>Acrílico</b>	10	0.2	0.2	0	11.4	0.23
<b>Total</b>					49.9	1

Nota: La matriz indica como se pondera el peso de los materiales.



**Tabla 7***Matriz de Holmes para el material*

	<b>Peso</b>	<b>Disponibilidad</b>	<b>Durabilidad</b>	<b>Costo</b>	<b>Suma</b>	<b>Ponderación</b>	<b>Prioridad</b>
<b>Acero</b>	0.002	0.008	0.142	0.064	0.22	21.51	3
<b>PLA-3D</b>	0.029	0.038	0.007	0.015	0.09	8.84	4
<b>MDF</b>	0.022	0.110	0.032	0.313	0.48	47.77	1
<b>Acrílico</b>	0.016	0.059	0.032	0.112	0.22	21.88	2
				<b>Total</b>	1	100	

Nota: La matriz se observa el orden de prioridad

La matriz de Holmes de la Tabla 7 presenta los resultados de la ponderación al evaluar los valores de los criterios con respecto de cada alternativa de solución, tal como se observa la prioridad la obtiene el material denominado MDF con 47.7% del valor total, por lo cual será usado como base en el diseño. Algunas de las propiedades mecánicas para el material elegido se muestran en Tabla 8.

**Tabla 8***Propiedades mecánicas para el MDF*

<b>Especificación</b>	<b>Magnitud</b>
Resistencia a la compresión	10 MPa
Alargamiento a la rotura	0.5%
Módulo de Young	4 GPa
Coefficiente de Poisson	0.25
Módulo de corte	2.5 GPa
Resistencia a la tracción	18 MPa

Nota: Denotamos la resistencia del material para el análisis estructural.

De forma similar el proceso de elección es aplicado para el resto de subsistemas, por tal motivo se enunciará los criterios a considerar, las opciones de solución y la matriz de Holmes con los resultados de prioridad.

### Potencia y tracción

El subsistema de potencia lo conforman la fuente de energía y los actuadores por lo cual se tomarán alternativas de solución para los dos casos. La fuente de energía es esencial en el desempeño del dispositivo, su función comprende el abastecimiento al sistema con énfasis en las áreas de autonomía, movimiento y computo. En la selección se toman en cuenta los siguientes criterios:

- Voltaje nominal
- Amperaje
- Costo
- Peso

**Tabla 9**

*Matriz de ponderación con los criterios para la fuente de energía*

Criterio	Voltaje	Amperaje	Costo	Peso	Suma+1	Ponderación
	Nominal					
<b>Voltaje</b>	-	1	0.2	5	7.2	0.25
<b>Nominal</b>						
<b>Amperaje</b>	1	-	0.2	5	7.2	0.25
<b>Costo</b>	5	5	-	1	12	0.42
<b>Peso</b>	0.2	0.2	1	-	2.4	0.08
				<b>Total</b>	28.8	1

Nota: Muestra la matriz formada por los criterios correspondientes al voltaje, amperaje, costo y peso; es utiliza la columna "Suma+1" para compensar el valor anulado por la comparación del criterio consigo mismo

En la Tabla 9 se observa un valor del 42% correspondiente al costo lo que implica su predominancia en la posterior selección. Debido que es requerido el voltaje DC (corriente continua) se presentan las opciones a considerar:

- **Batería LiPo:** Es un tipo de batería recargable compuesta por polímero de litio, posee ventajas tales como una alta capacidad de almacenamiento y descarga además de poco peso. La batería está constituida por la agrupación de celdas cuyos valores se acercan a los 4.2 v en cada, dando resultado voltajes nominales entre 7 y 12 v. (BLUBBER, 2022)
- **Batería de ácido sellada:** La batería de ácido-plomo presenta una gran variedad en voltaje (2v a 48v), alta tasa de amperio hora (hasta 100 Ah), destaca su capacidad recargable y no requerir mantenimiento, sin embargo, su peso es una desventaja. (Interstate Batteries, 2021)

**Tabla 10**

*Matriz de Holmes para la fuente de energía*

	<b>Voltaje</b>	<b>Amperaje</b>	<b>Costo</b>	<b>Peso</b>	<b>Suma</b>	<b>Ponderación</b>	<b>Prioridad</b>
	<b>Nominal</b>						
<b>Sellada de Ácido</b>	0.13	0.21	0.21	0.02	0.57	56.60	1
<b>Polímero de Litio</b>	0.13	0.04	0.21	0.06	0.43	43.40	2
				<b>Total</b>	1.00	100.00	

Nota: Los valores se obtienen del producto de la ponderación con cada uno de los valores en cada tabla.

La batería sellada obtiene la prioridad por un margen superior del 7 % a diferencia de la batería de polímero de litio tomando en cuenta que el amperaje es su principal cualidad en la valoración de acuerdo a las ponderaciones en la Tabla 10.

Los actuadores se encuentran como parte del subsistema de potencia porque permiten a la plataforma moverse de forma traslacional o rotacional al convertir energía eléctrica en mecánica por lo cual para seleccionar un actuador adecuado se toman en cuenta los siguientes criterios:

- Costo
- Torque
- Disponibilidad de compra
- Voltaje
- Control

**Tabla 11**

*Matriz de ponderación con los criterios para la selección de actuadores*

Criterio	Costo	Torque	Disponibilidad	Voltaje	Control	Suma+1	Ponderación
<b>Costo</b>	-	0.2	1	5	0.1	7.3	0.127
<b>Torque</b>	10	-	5	5	1	22	0.383
<b>Disponibilidad</b>	1	0.2	-	1	0.2	3.4	0.059
<b>Voltaje</b>	0.2	0.2	1	-	0.2	2.6	0.045
<b>Control</b>	10	1	5	5	0.2	22.2	0.386
<b>Total</b>						57.5	1

Nota: Se observa cómo se realiza el análisis de los actuadores.

Al evaluar los criterios establecidos previamente se muestra que el más relevante por una ínfima diferencia es el control, ver la Tabla 11. A continuación, se presentan las opciones para los actuadores que son esencialmente motores eléctricos.

- **Motor DC:** Es un motor que utiliza corriente directa, por lo general poseen cajas reductoras de engranes para disminuir la velocidad a la que gira eje (rpm) mientras aumenta el torque de tracción.
- **Servomotor:** Es un motor eléctrico de tipo lineal o rotatorio que consiste de un motor acoplado a un sensor para retroalimentar la información y controlar la posición angular, aceleración y velocidad. (Electrical4U, 2020)

- **Motor a pasos:** Es un motor DC que puede dividir su rotación en pasos utilizando operación magnética para que el eje del motor gire un valor preciso cuando es suministrado un pulso eléctrico, por lo cual su control no requiere un mecanismo de retroalimentación.

**Tabla 12**

*Matriz de Holmes para el actuador*

	Costo	Torque	Disponibilidad	Voltaje	Control	Suma	Ponderación	Prioridad
<b>DC Motor</b>	0.086	0.226	0.033	0.004	0.184	0.534	53.399	1
<b>Servomotor</b>	0.007	0.025	0.023	0.034	0.018	0.107	10.722	3
<b>Motor a Pasos</b>	0.033	0.132	0.003	0.007	0.184	0.359	35.879	2
					<b>Total</b>	1	100	

Nota: La tabla evalúa la importancia de los criterios con respecto a cada actuador.

La matriz de Holmes para el actuador muestra al motor de corriente continua con prioridad primaria con valor aproximado al 54%, el control para el motor de corriente continua es sencillo debido a que existe una variedad de técnicas que permiten realizar dicha tarea, un ejemplo es la utilización del puente H.

Para el envío y recepción de datos en los actuadores se utilizará la tarjeta de desarrollo Teensy 3.5 cuya programación es compatible con la interfaz de Arduino, entre sus especificaciones posee 20 salidas PWM, las cuales son esenciales en el control del motor, tal como se observa en la Tabla 13.

El actuador escogido es un motor de corriente continua a 12V con encoder integrado, sin embargo, su preferencia radica en el valor del torque que genera, de acuerdo a la Tabla 14 es un valor de 70 kg.cm equivalente a 6 N.m cumpliendo con el valor determinado en el análisis dinámico. Además, la velocidad de salida en el eje es 40 RPM, lo cual nos permitirá alcanzar una velocidad aproximada a 25 m/s al utilizar un neumático que posea un diámetro superior a 12 cm. La relación de transmisión combinada con la resolución del encoder tipo nos da como resultado 17280 detecciones por revolución.

**Tabla 13***Datos técnicos de la Teensy 3.5*

<b>Especificación</b>	<b>Magnitud</b>
Procesador	<ul style="list-style-type: none"> <li>ARM Cortex-M4 at 120 MHz -Unidad matemática de punto flotante, 32 bits</li> </ul>
Memoria	<ul style="list-style-type: none"> <li>512K Flash, 256K RAM, 4K EEPROM</li> </ul>
Conexión	<ul style="list-style-type: none"> <li>USB 12 Mbit/segundo</li> </ul>
Pines	<ul style="list-style-type: none"> <li>64 entrada o salida / 20 salidas PWM/27 entradas analógicas/2 salidas analógicas</li> </ul>

Nota: La matriz indica las propiedades ofrecidas por el fabricante.

**Tabla 14***Especificaciones del motor DC Ocean 270:1 con 64CPR de CQRobot*

<b>Especificación</b>	<b>Magnitud</b>
Relación de transmisión	270:1
Máxima Potencia	6V: 3W, 12V: 6W
Velocidad sin carga	6V: 20 rpm, 12V: 40rpm
Corriente sin carga	6V: 0.15 A, 12V: 0.2 A
Torque de parada	6V: 40 kg.cm, 12V: 70 kg.cm
Corriente de parada	6V: 20 2.8 A, 12V: 5.5 A
Voltaje del encoder	3.3V a 24V
Tipo de encoder	Hall (Tipo incremental)
Resolución del encoder	64CPR

Nota: La matriz indica las propiedades ofrecidas por el fabricante (CQRobot, 2019).

En el control se requieren dispositivos que permitan la interacción entre los actuadores y los códigos de programación, y así conseguir los valores deseados de acuerdo a las variables. Los criterios que se analizan son:

- Tamaño
- Costo
- Implementación

A continuación, se describen las alternativas para los dispositivos de control de los actuadores:

- **L298N:** El módulo es un puente dual que permite el manejo de motores DC hasta un 2 A, permite controlar velocidad y giro siendo compatible con tarjetas como arduino o Raspberry. Posee un regulador de voltaje de 5V para la alimentación de la sección lógica.
- **BTS7960:** El componente permite el control de giro de 1 motor DC a la vez evitando el calentamiento debido a su disipador integrado. Funciona mediante un voltaje de 5 a 27 V y soportar corrientes hasta 43 A.
- **VNH2SP30:** Es capaz de soportar una corriente de hasta 30 A con un voltaje máximo de 16 V, necesita la adición de un disipador. Es compatible con diferentes tarjetas de control como Arduino.

**Tabla 15**

*Matriz de ponderación con los criterios para la selección de driver para los motores*

<b>Criterio</b>	<b>Tamaño</b>	<b>Costo</b>	<b>Implementación</b>	<b>Suma +1</b>	<b>Ponderación</b>
<b>Tamaño</b>	-	5	1	7	0.45
<b>Costo</b>	0.2	-	0.2	1.4	0.1
<b>Implementación</b>	1	5	-	7	0.45
			<b>Total</b>	15.4	1

Nota: La tabla muestra la ponderación de los criterios en tamaño, costo e implementación.

**Tabla 16**

*Matriz de Holmes para el módulo controlador de motores*

	<b>Tamaño</b>	<b>Costo</b>	<b>Implementación</b>	<b>Suma</b>	<b>Ponderación</b>	<b>Prioridad</b>
<b>L298N</b>	0.063	0.068	0.265	0.396	39.6	1
<b>BTS7960</b>	0.063	0.006	0.148	0.217	21.7	3
<b>VNH2SP30</b>	0.324	0.026	0.036	0.386	38.7	2
			<b>Total</b>	1	100	

Nota: Analiza los diferentes puentes H con posibilidad para implementarse.

El primer controlador presenta una ligera predominancia correspondiente al 39.6%, su elección radica en la su fácil implementación, ver la Tabla 16. En Tabla 17 se encuentran los datos de la opción elegida.

**Tabla 17**

*Especificaciones del módulo L298N*

<b>Características</b>	<b>Magnitud</b>
Controlador	L298N
Canales	2
Voltaje de Control	5V
Voltaje de Potencia	5V-35V
Corriente de control	36 mA
Corriente máxima	2A
Dimensiones	43*43*27 mm

Nota: La matriz indica las propiedades ofrecidas por el fabricante.



## Percepción del entorno

La odometría forma parte fundamental en la percepción del robot, es decir, los sensores incorporados a la plataforma. Por lo cual se toma particular interés en los criterios a ser aplicados para la elección los cuales son:

- Tamaño
- Costo
- Implementación
- Facilidad de uso (programación)
- Peso
- Rango

**Tabla 18**

*Matriz de ponderación con los criterios de la selección de sensores*

<b>Criterio</b>	<b>Tamaño</b>	<b>Costo</b>	<b>Implementación</b>	<b>Facilidad de Uso</b>	<b>Peso</b>	<b>Rango</b>	<b>Suma+1</b>	<b>Ponderación</b>
<b>Tamaño</b>	-	0.2	0.1	1	1	0.2	3.5	0.051
<b>Costo</b>	5	-	1	5	5	0.2	17.2	0.250
<b>Implementación</b>	5	5	-	1	5	1	18	0.262
<b>Facilidad de uso</b>	1	0.2	1	-	5	1	9.2	0.134
<b>Peso</b>	1	0.2	0.2	0.2	-	5	7.6	0.111
<b>Rango</b>	5	5	1	1	0.2	-	13.2	0.192
<b>Total</b>							<b>68.7</b>	<b>1</b>

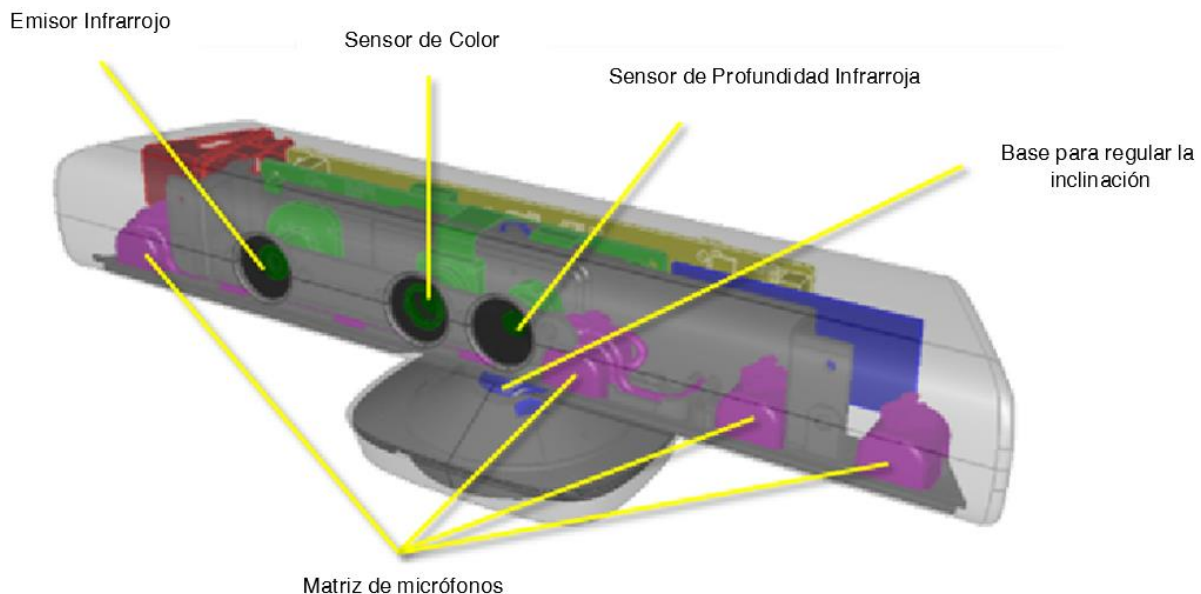
Nota: Se denota la evaluación de los criterios.

Los criterios con mayor relevancia son el costo y la implementación con valores similares entre si lo que es apreciado en la Tabla 18. Entre las opciones a ser consideradas son excluidos los encoder debido a que el motor elegido tiene incorporado el sensor, las opciones a considerarse son las siguientes:

- **Cámara Kinect:** Fue implementado por la compañía Microsoft para permitir a sus usuarios controlar las funciones de las consolas mediante acciones corporales o comando de voz. El dispositivo esta provisto de una cámara RGB, un micrófono de matriz múltiple, sensor infrarrojo y de profundidad; la combinación de los componentes permite la captura de movimiento tridimensional, reconocimiento facial y de voz. (Techopedia Inc, 2022)
- **RPLIDAR:** Es un escáner láser de bajo costo que permite detención de 360° dentro un rango de 6m, su función consiste en generar una nube de puntos bidimensional diseñado para mapeo, localización y modelaje de objetos en interiores. (Shanghai Slamtec Co., s.f.)

**Figura 31**

*Representación esquemática de la cámara Kinect*



Nota: Figura basado en (Nag & Deshmukh, 2015)

**Figura 32**

RpLidar A1-M8

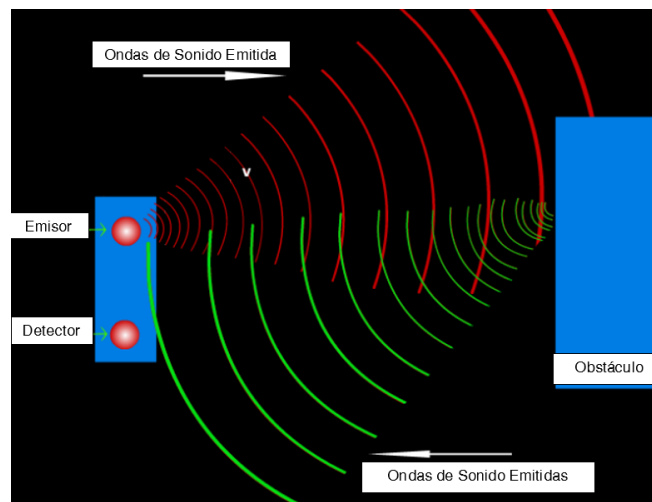


Nota: Figura realizada por (Shanghai Slamtec Co., s.f.)

- **Ultrasónico:** Es un sensor que mide la distancia mediante ondas ultrasónicas, el dispositivo envía las ondas de sonido y recibe el reflejo proveniente del objetivo, esta información la convierte en una señal eléctrica para medir tiempo entre la emisión y recepción. Su funcionamiento se visualiza de mejor manera en la Figura 33. (Questex, 2022)

**Figura 33**

*Diagrama del funcionamiento del sensor ultrasónico*

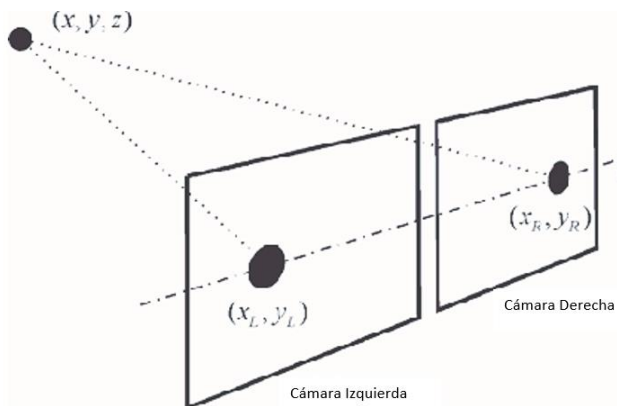


Nota: Figura realizada por (Questex, 2022).

- **Cámara Estéreo:** Es una cámara que utiliza el principio de fotografía en 3D para percibir la profundidad de una imagen similar al del ser humano, para lo cual utiliza dos cámaras sincronizadas separadas a una por una distancia. (Karunanayake & Gnanasekera, 2016)

**Figura 34**

*Representación de la visión estereoscópica*

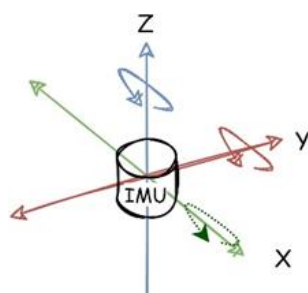


Nota: Figura realizada por (Karunanayake & Gnanasekera, 2016).

- **Unidad de medición inercial:** Conocido comúnmente como IMU por su abreviación en inglés de Inertial Measurement Unit. Es un dispositivo electrónico provisto de giroscopios, acelerómetros y/o magnetómetros que permite obtener la aceleración y orientación angulares además de la fuerza gravitacional del objeto donde sea instalado con respecto de los ejes (X, Y, Z). (Or, 2021)

**Figura 35**

*Representación de la medición por parte del sensor inercial*



Nota: Figura obtenida de (Or, 2021).

**Tabla 19***Matriz de Holmes para la selección de sensores*

	Tamaño	Costo	Implementación	Facilidad de Uso	Peso	Rango	Suma	%	Prioridad
<b>Kinect</b>	0.002	0.059	0.049	0.027	0.003	0.019	0.160	15.976	4
<b>RP Lidar</b>	0.010	0.024	0.049	0.027	0.015	0.042	0.167	16.733	3
<b>Ultrasónico</b>	0.014	0.081	0.015	0.009	0.044	0.004	0.168	16.764	2
<b>Estéreo</b>	0.002	0.006	0.049	0.014	0.003	0.019	0.093	9.299	5
<b>IMU</b>	0.023	0.081	0.101	0.057	0.044	0.107	0.412	41.228	1
						<b>Total</b>	1	100	

Nota: Las opciones de los criterios llevan a definir los sensores a implementar.

El sensor inercial destaca como la opción principal, seguido por el sensor ultrasónico y el láser de rango, sin embargo, la opción dos se descarta debido a que usará el encoder de los motores. En cámaras se escoge la Kinect debido a su costo bajo con respecto al tipo estéreo. Las especificaciones técnicas de los sensores elegidos de muestras en las siguientes tablas:

**Tabla 20***Datos técnicos para el MPU6050*

Especificación	Magnitud
Voltaje de alimentación	3-5V
Interfaz digital	I2C digital
Sensor angular de 3 ejes	Giroscopio (X, Y, Z)
Acelerómetro	Precisión programable en rangos de $\pm 2g$ , $\pm 4g$ , $\pm 8g$ and $\pm 16g$
Dimensiones	14*21 mm

Nota: Especificaciones obtenidas de (NovatronicEc, 2020).

**Tabla 21***Datos técnicos del sensor LIDAR A1M8*

<b>Especificación</b>	<b>Magnitud</b>
Voltaje de Operación	5V DC $\pm$ 5%
Fuente de Luz	Diodo láser semiconductor ( $\lambda=785\text{nm}$ )
Consumo de Corriente	400mA o menor (Corriente Pico 1A)
Distancia de detección	120mm ~ 3,500mm
Rango de muestro	1.8kHz
Dimensiones	69.5(W) X 95.5(D) X 39.5(H) mm
Peso	Menos de 125g

Nota: Especificaciones obtenidas de (Shanghai Slamtec Co., s.f.).

**Tabla 22***Datos técnicos para la Kinect v1 de la consola Xbox 360*

<b>Especificación</b>	<b>Magnitud</b>
Voltaje de operación	12V
Resolución RGB	640 x 480, 30 fps
Resolución de la cámara infrarroja	320 x 240, 30 fps
Tipo de sensor de profundidad	Luz estructurada
Campo de visión para imagen RGB	62° x 48.6°
Campo de visión para profundidad	57° x 43°
Rango de medición	0.8 – 4 por defecto

Nota: Especificaciones obtenidas del fabricante.

### Procesamiento central de la plataforma

La unidad central de procesamiento es la responsable de recibir la información proveniente de los sensores, la cual actuará como valores de entrada en algoritmos de cálculo con el fin de obtener resultados de odometría y la detección de imágenes, entre otros. Esta pieza le permite al robot tomar decisiones y enviar una señal de control de acuerdo a los objetivos de funcionamiento con lo cual enlaza el hardware con el software, ejemplo de ello es la plataforma denominada turtlebot3 que puede realizar tareas de navegación, exploración y manipulación de objetos. (ROBOTIS, 2022)

Los aspectos a considerar son:

- Velocidad de procesamiento
- Disponibilidad de compra
- Tamaño
- Costo
- Soporte Técnico
- Implementación

**Tabla 23** Matriz de ponderación con los criterios para la selección del CPU

Criterio	Procesamiento	Disponibilidad	Tamaño	Costo	Soporte	Implementación	Suma+1	Ponderación
<b>Procesamiento</b>	0	0.2	1	0.1	1	5	2.3	0.04
<b>Disponibilidad</b>	5	0	0.2	1	0.2	1	7.2	0.11
<b>Tamaño</b>	1	5	0	0.2	0.1	0.2	7.2	0.11
<b>Costo</b>	10	1	5	0	5	0.2	17	0.27
<b>Soporte</b>	1	5	10	0.2	0	1	17.2	0.27
<b>Implementación</b>	0.2	1	5	5	1	0	12.2	0.19
<b>Total</b>							63.1	1

Nota: Las opciones de los criterios llevan a definir el procesador central el cual se podrá utilizar.

La Tabla 23 presenta una igualdad de ponderación entre los criterios soporte y costo con un valor correspondiente al 27% lo que correlaciona directamente el valor de adquisición junto con el soporte para el software. Actualmente existe una gran diversidad entre dispositivos que desempeñan el papel de procesador central por lo cual se han planteado las siguientes alternativas:

- **Jetson Nano:** Es una tarjeta compacta que permite el procesamiento de múltiples aplicaciones de redes neuronales en paralelo como clasificación de imágenes, detección de objetos, segmentación, etc.; posee una capacidad de procesamiento igual a 1.43 GHz tanto para la versión de 2 GB y 4 GB siendo compatible con Linux. (NVIDIA Corporation, 2022)
- **Raspberry Pi B:** Presenta ventajas en la velocidad de procesamiento, memoria y conectividad en comparación a modelos de la misma marca. Esta provisto de una procesador Cortex-A72@1.5 GHz de 64 bits cuya memoria depende modelo, actualmente puede alcanzar hasta 8GB. ( Raspberry Pi Foundation, 2022)

Además, se añaden como posibilidades un computador portátil y la unidad de cómputo denominada NUC.

**Tabla 24**

*Matriz de Holmes para la selección del CPU*

	Procesamiento	Disponibilidad	Tamaño	Costo	Soporte	Implementación	Suma	Ponderación	Prioridad
<b>Jetson 2GB</b>	0.00	0.00	0.04	0.07	0.04	0.02	0.18	17.59	4
<b>Raspberry Pi 4</b>	0.00	0.02	0.04	0.12	0.01	0.04	0.22	22.49	2
<b>Jetson 4GB</b>	0.00	0.01	0.04	0.06	0.09	0.04	0.23	23.49	1
<b>NUC</b>	0.01	0.04	0.00	0.01	0.07	0.01	0.14	14.26	5
<b>Portátil</b>	0.01	0.04	0.00	0.01	0.07	0.09	0.22	22.16	3
						<b>Total</b>	1	100	

Nota: Resultado de las evaluaciones de los criterios con las alternativas.



Tomando en cuenta la información de la Tabla 24 se opta por elegir la opción primaria correspondiente a la Jetson Nano 4Gb cuyas especificaciones se presentan en la Tabla 25. Entre las principales características se encuentran su tamaño y peso bajos, una velocidad de procesamiento 1.43 GHz y una memoria de 4 GB lo que permite realizar el procesamiento de imágenes; de igual manera se encuentra provisto por puestos USB, lo cual permite conectar dispositivos como cámaras, láseres, entre otros. Finalmente, esta provisto de una conexión de pines.

**Tabla 25**

*Datos técnicos de la tarjeta Jetson Nano 4GB*

<b>Especificación</b>	<b>Magnitud</b>
GPU	128-core Maxwell
CPU	Quad-core ARM A57 @ 1.43 GHz
Memoria	4 GB 64-bit LPDDR4 25.6 GB/s
Almacenamiento	Tarjeta SD
Conectividad	Gigabit Ethernet, M.2 Key E
USB	4x USB 3.0, USB 2.0 Micro-B
Dimensiones	69 mm x 45 mm, 260 pines

Nota: Especificaciones dadas por el fabricante.

Para determinar el consumo energético se toma en cuenta un valor de 12V como el máximo en la batería, el amperaje se obtiene al sumar el consumo de los componentes a implementarse.

El resultado de la Tabla 26 determina que se consume alrededor de 5.5, con la finalidad de precautelar el desempeño de la plataforma y sus componentes se opta por el uso de dos baterías de 5A de la marca FisrtPower cuyos datos técnicos están en la Tabla 27.

**Tabla 26***Consumo Energético de la plataforma*

<b>Dispositivos</b>	<b>Amperaje (mA)</b>
Jetson Nano 4GB	2000
L298n	2000
Encoder Derecho	20
Encoder izquierdo	20
Kinect	1500
<b>Total</b>	<b>5540</b>

Nota: La plataforma consumirá 5540 mA.

**Tabla 27***Datos Técnicos de batería*

<b>Especificación</b>	<b>Magnitud</b>
Voltaje	12V
Capacidad	5Ah
Dimensiones	90x70x101 mm
Peso	1.65 kg

Nota: Especificaciones dadas por ( Kamtex Industries Pte Ltd, 2021).

**Software del sistema**

Como parte fundamental del CPU se encuentra el sistema operativo, por lo cual se utilizará la distribución de Linux-Ubuntu 18.04, la cual es compatible con la tarjeta Jetson Nano 4Gb. Debido a las características de la tarjeta central en el procesamiento de imágenes y video se opta por implementar el algoritmo YOLO, el cual será utilizado en la detección de objetos y la relación de entre los datos

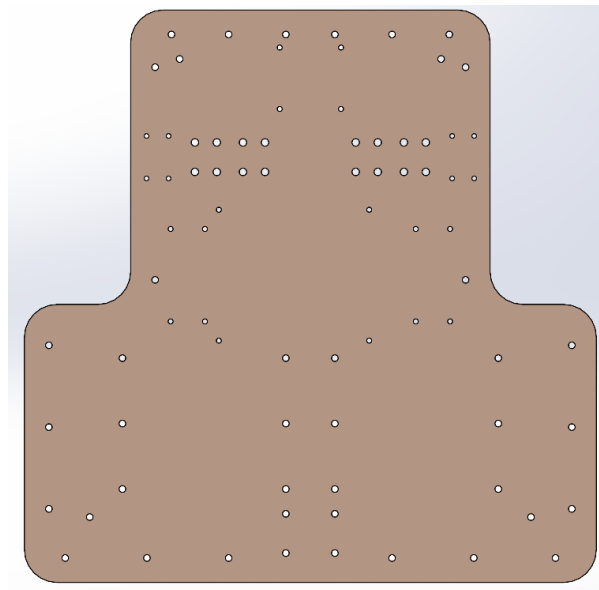
provenientes del mapa y las imágenes se desarrolla con algoritmos programados en el lenguaje Python utilizando el complemento owlready2 para generar la ontología.

### **Diseño de la base inferior de la plataforma**

El proceso comienza al diseñar la base inferior a causa que soportará el peso conjunto de la plataforma por lo cual elementos como motores, batería y elementos electrónicos serán ubicados sobre su superficie. En la Figura 36 se realizaron perforaciones circulares con la intención de utilizar tornillos para el ensamblaje, los motores se ubican a una distancia de 146mm partiendo de la mitad de la pieza cuya longitud total en la parte delantera es 350mm.

### **Figura 36**

*Forma de la base inferior de la plataforma*



Nota: Figura realizada por autor.

### ***Análisis estático de la base***

La base es sometida a una simulación en software CAE para analizar la resistencia en función del peso total obtenido al multiplicar la masa total por la gravedad, la cual es a igual a  $9.81 \text{ m/s}^2$ , debido a que los valores aproximados podrían producir incertidumbre en el diseño se toma un factor de seguridad de 1.5, el resultado obtenido es 204.53 N (Tabla 28).

Tabla 28

Lista para la masa de los elementos en la plataforma

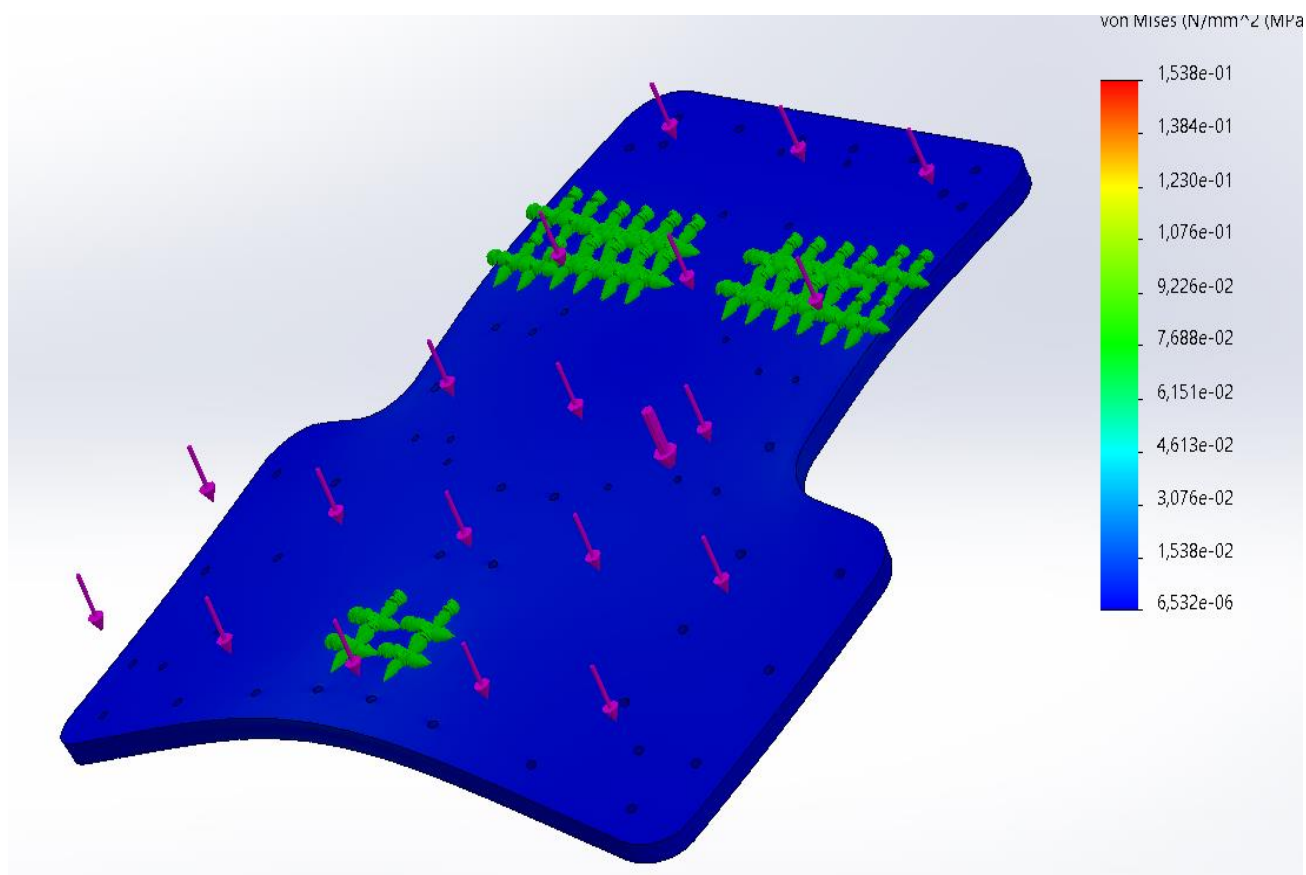
<b>Objeto</b>	<b>Cantidad</b>	<b>Masa (kg)</b>	<b>Masa Total</b>
<b>Kinect V1</b>	1	0.6	0.60
<b>Motor DC</b>	2	0.25	0.50
<b>Rueda</b>	2	0.05	0.10
<b>Base del Motor</b>	2	0.016	0.03
<b>Batería</b>	2	1.65	3.30
<b>Eje de Cobre (120mm)</b>	4	0.1	0.4
<b>Cubiertas para la base</b>	4	0.25	1.00
<b>Eje de Cobre (150mm)</b>	4	0.13	0.52
<b>Bases MDF</b>	3	0.8	2.4
<b>Soporte Tubo</b>	1	0.8	0.8
<b>Soporte Secundario</b>	2	0.2	0.4
<b>Soporte para Kinect</b>	1	0.2	0.2
<b>Seguro para Kinect</b>	2	0.02	0.04
<b>Rueda Castor</b>	4	0.035	0.14
<b>RPLidar</b>	1	0.37	0.37
<b>Flecha Rueda</b>	2	0.05	0.1
<b>Componentes verios</b>	1	3	3
			13.9
		<b>F.S.</b>	1.5
		<b>Total</b>	204.53 (N)

Nota: Masa total del sistema.

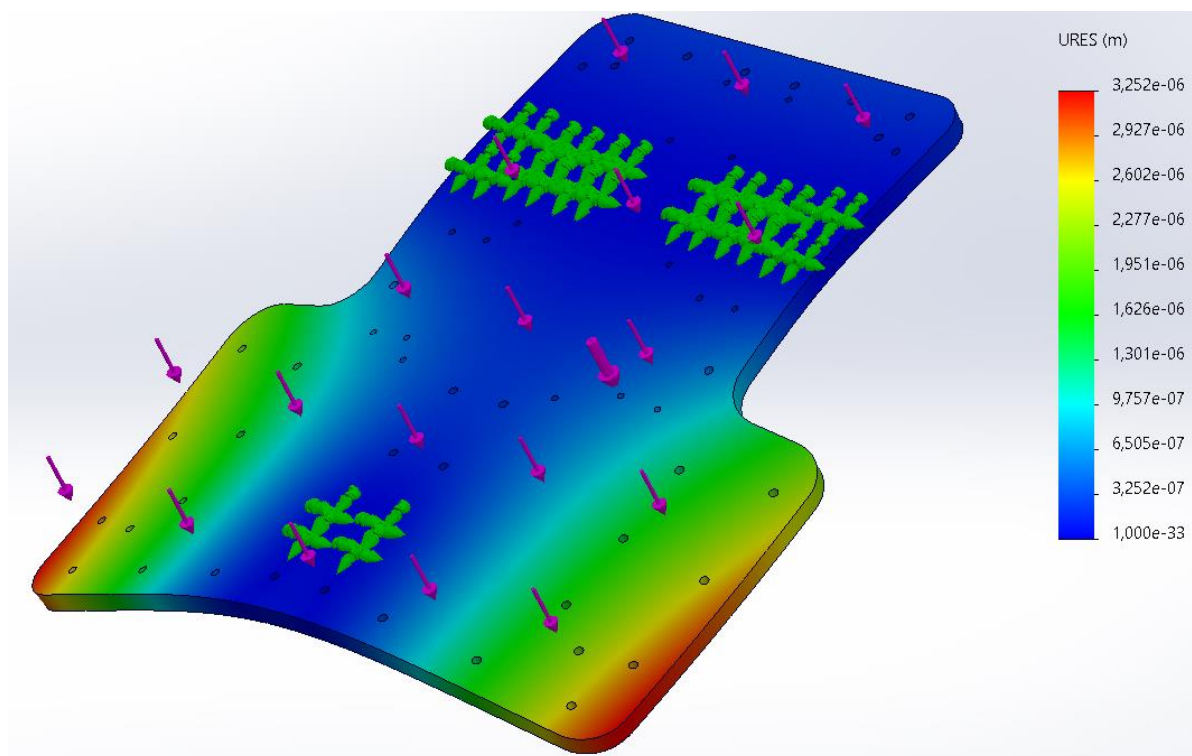
El material es configurado de acuerdo a los valores de la Tabla 8, el peso se ingresa como una carga distribuida sobre la cara superior de la base mientras los apoyos corresponden a la ubicación de los soportes para las ruedas. Los resultados corresponden a la teoría de Von Mises para los esfuerzos de distorsión, cuyo valor máximo es 0.153 MPa concentrados sobre la pieza como se ve en Figura 37 mientras su deformación máxima es  $3.25e-3$  mm de acuerdo a la Figura 38 por lo cual no existe riesgo de rotura en el material.

**Figura 37**

*Resultado del análisis estático (Von Mises)*



Nota: Figura realizada por autor

**Figura 38***Deformación resultante de la base**Nota: Figura realizada por autor***Diseño de Piezas de Unión y Soporte**

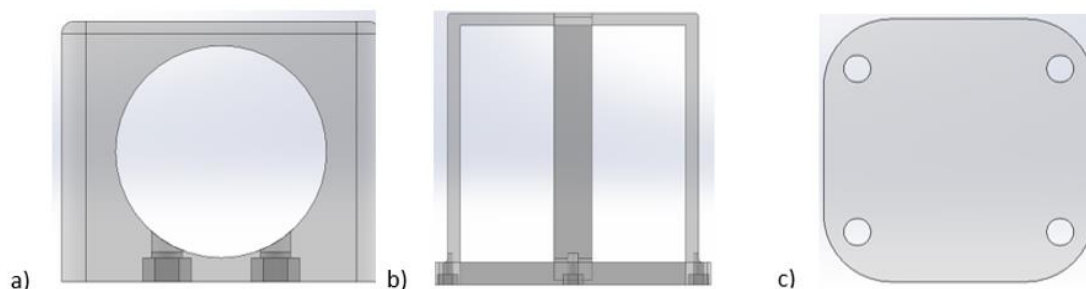
El motor seleccionado tiene un eje que posee las siguientes dimensiones, longitud de 16mm y diámetro igual a 6mm, mientras el cubo del neumático tiene una circunferencia interna de 12mm para ensamblar las piezas se diseña un eje de cobre, ver la Figura 39.

**Figura 39***Eje para unir el motor con la rueda**Nota: Figura realizada por autor*

Adicionalmente, es necesario diseñar piezas de soporte adicionales para el a) motor, b) la batería y c) la rueda loca como se ve en la Figura 40. Debido a al tamaño y precisión requerida se opta por el proceso de impresión 3D con filamento PLA, se observa las propiedades mecánicas en la Tabla 29. En consideración a los diseños planteados se observa un ensamblaje preliminar correspondiente a la base primaria que abarca la movilidad de la plataforma y contención de elementos tal como se observa en la Figura 36.

**Figura 40**

*Piezas adicionales para sujeción y soporte*



Nota: Figura realizada por autor

**Tabla 29**

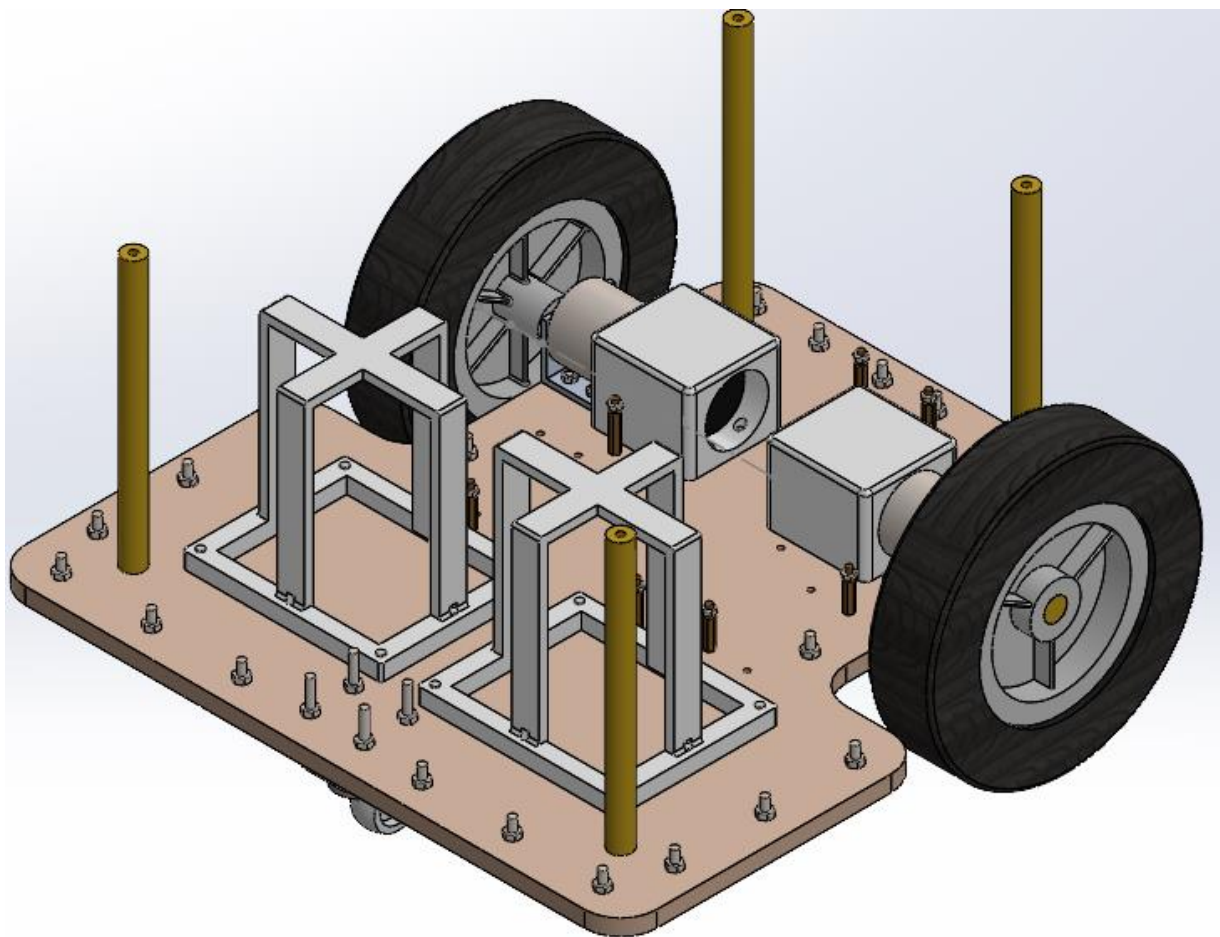
*Propiedades Mecánicas del filamento PLA*

Especificación	Magnitud
Módulo de elasticidad a la tracción	2.3 GPa
Esfuerzo de tracción a la deformación	49.5 MPa
Esfuerzo de tracción a la rotura	45.6 MPa
Alargamiento a la deformación	3.3%
Alargamiento a la rotura	5.2%
Resistencia a la flexión	103 MPa
Módulo de flexión	3.15 GPa

Nota: Especificaciones dadas por (Ultimaker, 2017).

**Figura 41**

*Representación del ensamblaje para el piso inferior del robot*

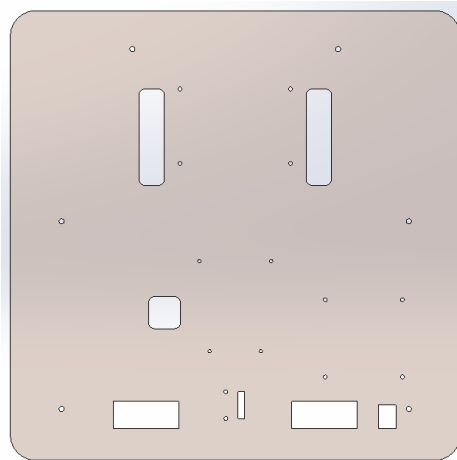


Nota: Figura realizada por autor

### **Diseño para la base intermedia**

Los componentes correspondientes al funcionamiento móvil de la plataforma se han ubicado en la base inferior, sin embargo, se requiere colocar elementos que posean estabilidad mientras la plataforma se desplaza. Los elementos pertenecientes a esta base son: la tarjeta de procesamiento Jetson, el sensor láser 1-M8 y la cámara Kinect. La base se muestra en la Figura 42. El peso acumulado por las piezas mencionadas no influye de manera significativa sobre la estructura por lo que se plantea su fabricación en MDF de 9 mm.

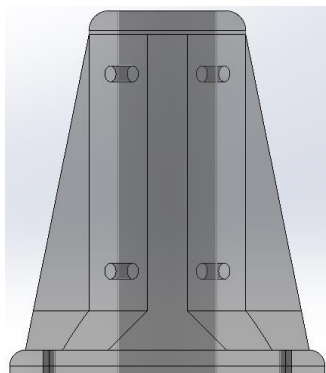


**Figura 42***Base intermedia de la plataforma móvil*

Nota: Figura realizada por autor

**Diseño para el soporte de la cámara Kinect**

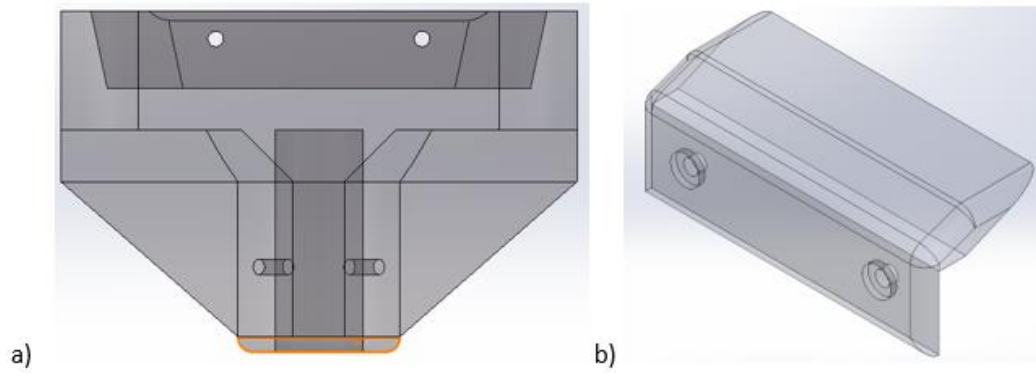
La cámara Kinect requiere un soporte conjunto que permita variar su altura para ello se plantea el uso de dos tubos cilíndricos. El tubo de diámetro mayor se fijará a la base anterior por medio del soporte en la Figura 43, el segundo tubo contempla en unos de los extremos los objetos de la Figura 44 los cuales serán usados para sostener la cámara. Debido al cableado de la Kinect el cual dificultaría el desempeño y aspecto de la plataforma se añade una repisa con grosor de 3mm para la sujeción de los cables.

**Figura 43***Soporte para el tubo base*

Nota: Figura realizada por autor

**Figura 44**

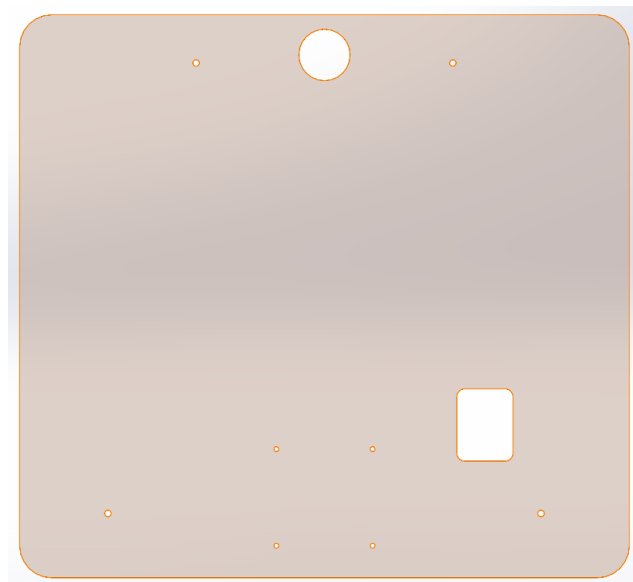
*Elementos para la sujeción de la cámara al tubo*



Nota: a) Base de soporte para la Kinect b) Seguro para la base de soporte realizado por el autor.

**Figura 45**

*Repisa superior de la plataforma móvil*

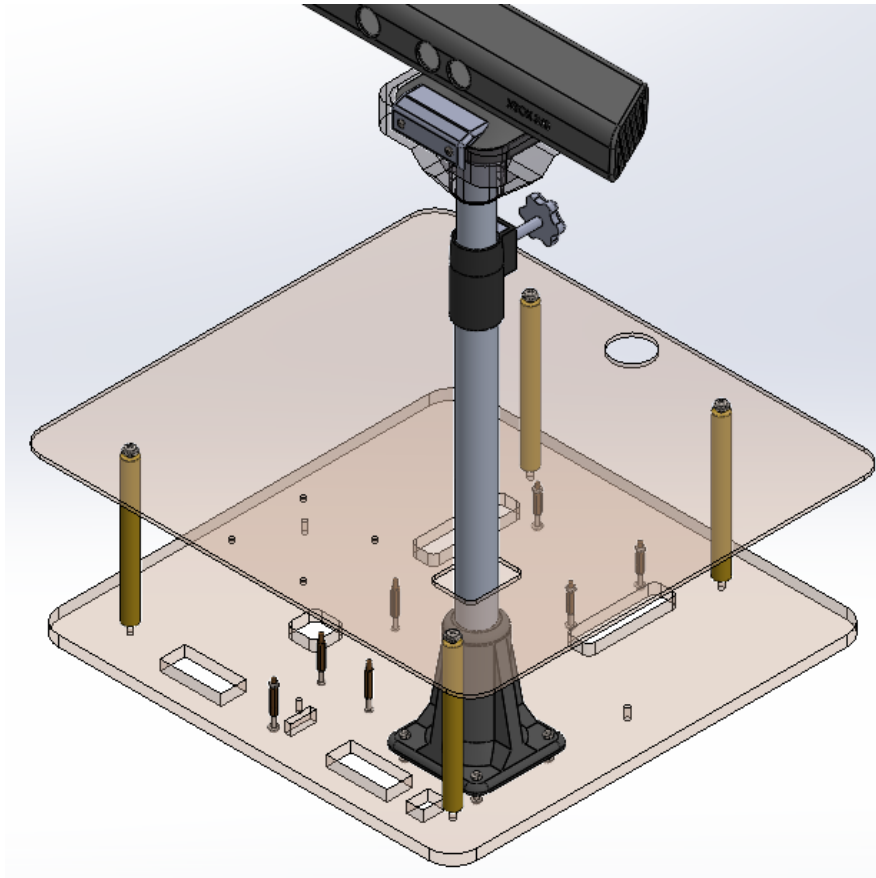


Nota: Figura realizada por autor

Durante el diseño se descarta el planteamiento al conjunto de los tubos debido que se establece la modificación de un pedestal para sistema de parlante. El ensamblaje resultante compuesto por las piezas mecánicas se observa en la Figura 46.

**Figura 46**

*Representación del ensamblaje para el piso intermedio de la plataforma*



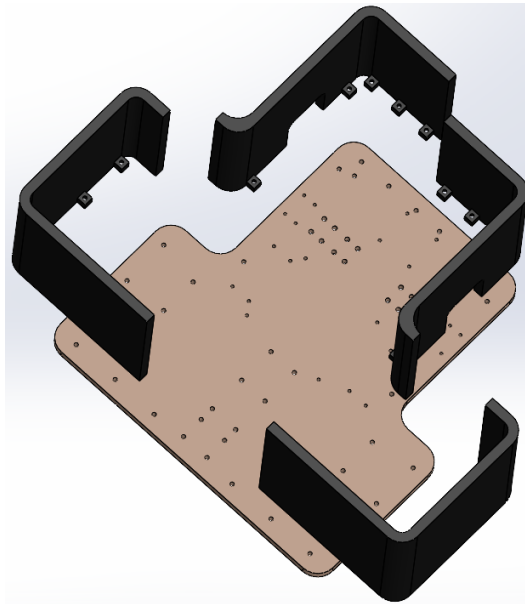
Nota: Figura realizada por autor

### **Diseño para la cubierta de la plataforma móvil**

La creación del recubrimiento consiste de cuatro piezas que asemejan el contorno de la base inferior como se ve en la Figura 47. Se establece un grosor de 10mm y una altura de 150mm para su diseño descartando una función de soporte no requiere un análisis de esfuerzos. Finalmente, en la Figura 48 se observa el ensamblaje final para el diseño mecánico de la plataforma.

**Figura 47**

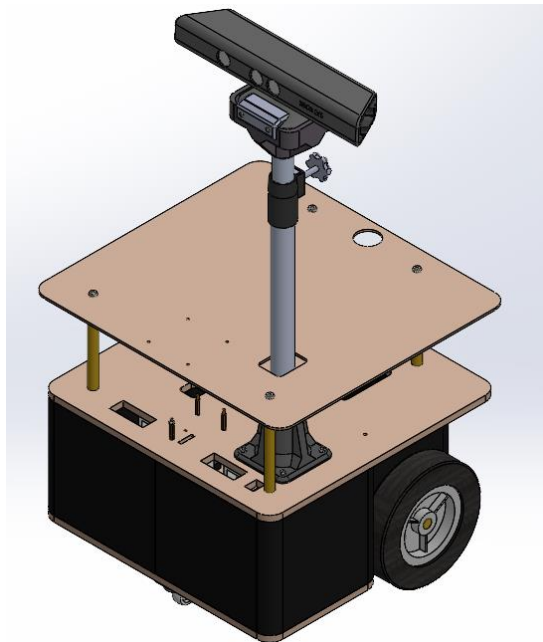
*Recubrimientos para el piso primario de la plataforma*



Nota: Figura realizada por autor

**Figura 48**

*Diseño mecánico final para la plataforma móvil*

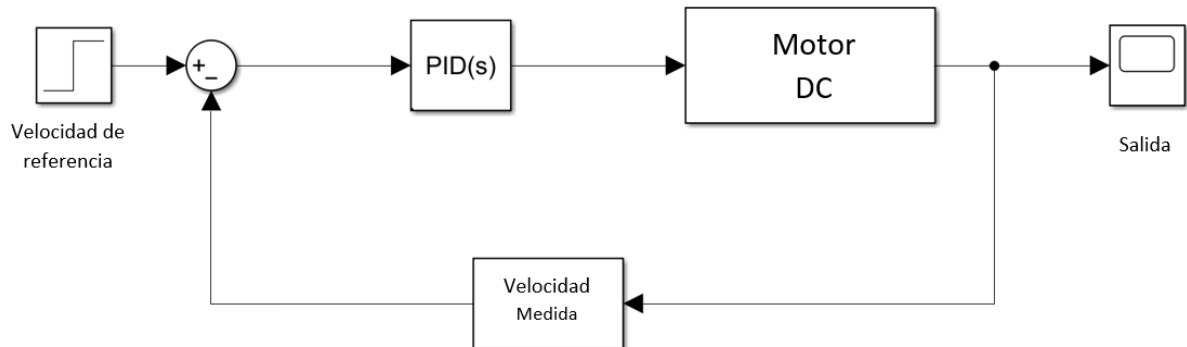


Nota: Figura realizada por autor

## Diseño del controlador para los actuadores

Figura 49

Diagrama de bloque para controlador del motor



Nota: Representación del proceso para controlar el motor.

Para el controlador del motor se aplica la lógica del diagrama en la Figura 49, por lo cual se obtendrá la función de transferencia mediante la identificación de cada motor. Se considera al actuador como una caja negra a la cual se aplica un voltaje de entrada para obtener la velocidad del eje del motor mientras se recolectan un conjunto de datos cada 100 milisegundos.

Los datos se ingresan al software de Matlab para realizar la identificación de la planta utilizando el comando "systemidentification", la interfaz del programa nos permite obtener las funciones de transferencia, así como el porcentaje de estimación. Posteriormente se aplicará el método de Ziegler-Nichols para calcular los valores correspondientes al controlador PID. Al establecer que la planta es de segundo orden y un lazo cerrado, se plantea de la siguiente manera:

$$G_Z = \frac{b_1 \cdot z + b_2}{z^2 + a_1 \cdot z + a_2} \quad (27)$$

$$\Delta(z) = z^2 + a_1 \cdot z + a_2 \cdot K_u \cdot (b_1 \cdot z + b_2) = 0 \quad (28)$$

Y al conocer que los polos están formados por una parte real y una imaginaria, como se observa en 29:

$$Z_{1,2} = \exp(\pm j\omega_0 \cdot T) = \cos(\omega_0 \cdot T) \pm j \cdot \sin(\omega_0 T) \quad (29)$$

Se igualan así:

$$\alpha = \cos(\omega_0 \cdot T) \quad (30)$$

$$\beta = \sin(\omega_0 \cdot T) \quad (31)$$

Se reemplazan las ecuaciones 28, 29, 30 y 31 en 27 para obtener lo siguiente:

$$\Delta(\alpha + j\beta) = (\alpha + j\beta)^2 + a_1 \cdot (\alpha + j\beta) + a_2 + Ku \cdot (b_1 \cdot (\alpha + j\beta) + b_2) \quad (32)$$

Los términos se igualan a cero tanto en la parte real como imaginaria:

$$\alpha^2 + \beta^2 + a_1 \cdot \alpha + Ku \cdot b_1 \cdot \alpha + Ku \cdot b_2 = 0 \quad (33)$$

$$2 \cdot \alpha \cdot \beta + a_1 \cdot \beta + Ku \cdot b_1 \cdot \beta = 0 \quad (34)$$

A continuación, se reemplaza 31 en 32 y se obtiene:

$$Ku = \frac{1 - a_2}{b_2} \quad (35)$$

Hallando el valor de  $\alpha$  de 33, se obtiene:

$$\alpha = -\frac{1}{2} \cdot (Ku \cdot b_1 + a_1) \quad (36)$$

Con el período crítico igual a:

$$Tu = \frac{2 \cdot \pi \cdot T}{\cos^{-1}(\alpha)} \quad (37)$$

Se encuentra la ganancia y el período críticos con las ecuaciones 35, 36 y 37 al reemplazar en conjunto con los valores de la respectiva función de transferencia. Finalmente, se aplica los valores de la Tabla 30 de acuerdo al método.

**Tabla 30**

*Sintonización de controladores mediante Ziegler-Nichols*

Controlador	$kp$	$Td$	$Ti$
PID	$0.6 Ku$	$0.5 Tu$	$0.125 Tu$

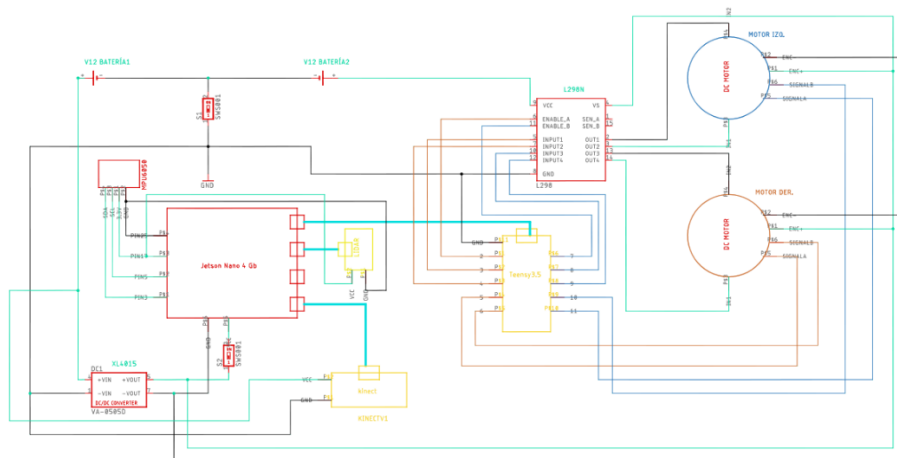
Nota: Obtenido de Ingeniería de control moderna, (Ogata, 2010).

## Diseño de la conexión electrónica de la plataforma.

En la Figura 50 se observa el esquema para la conexión electrónica del sistema, se conforma por el suministro de energía por parte de dos baterías como se muestra en la Figura 51, de igual manera se realiza una conversión de 12v a 5v para alimentar a los dispositivos a los encoders de los motores, al módulo L298n y a la tarjeta Jetson. En la Figura 53, las salidas PWM de la tarjeta Teensy al L298n, mientras las salidas del integrado se conectan a los motores respectivamente. Finalmente, la Figura 54 representa los dispositivos USB conectados como lo son: la cámara Kinect, tarjeta Teensy y el sensor LIDAR.

### Figura 50

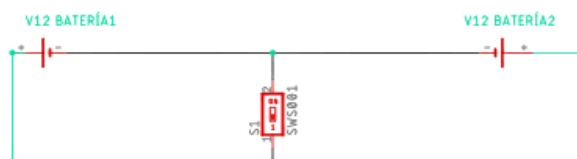
*Diagrama eléctrico del sistema*



Nota: Representación del sistema eléctrico. Figura realizada por el autor.

### Figura 51

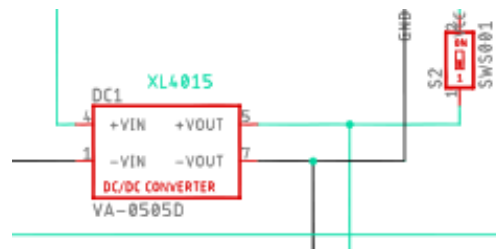
*Suministro de energía del sistema*



Nota: Sección de suministro de energía al sistema completo, figura realizada por el autor.

Figura 52

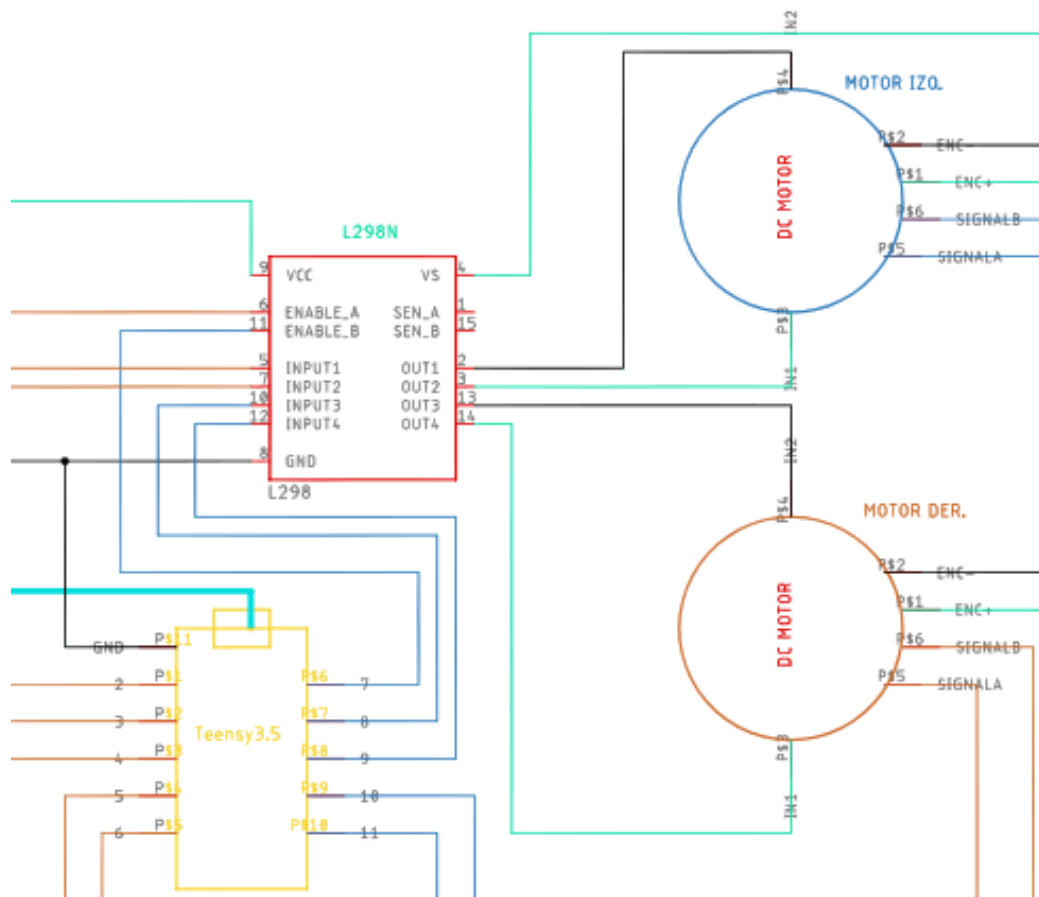
Conversión de voltaje de 5V



Nota: Conversión de 12V a 5V utilizando convertor dc-dc, figurada realizada por el autor.

Figura 53

Control de los motores

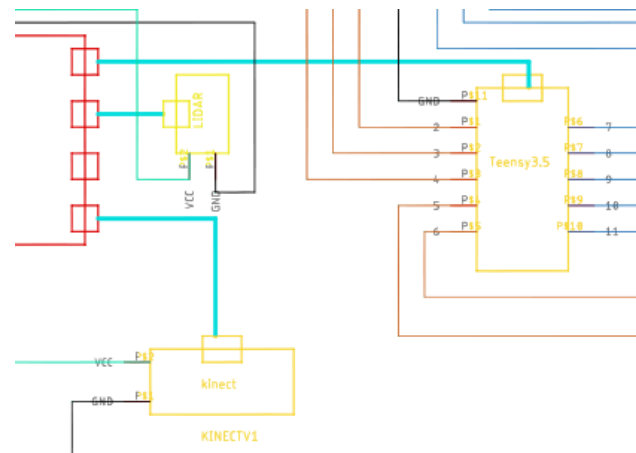


Nota: Conexión de los motores al microcontrolador Teensy 3.5, figura realizada por el autor.



**Figura 54**

*Conexión de los dispositivos USB al CPU.*



Nota: Conexión USB, figura realizada por el autor.

### **Diseño de la placa PCB**

La fuente de alimentación está formada por baterías selladas con valores nominales de 12V y 5Ah lo que permite abastecer a los dispositivos, por tal motivo se diseñó una placa PCB en el software EasyEDA en su versión gratuita.

El modelo de circuito PCB es accionado por un switch en la conexión común de los componentes a tierra (GND). Para la sección de control se agrega un juego de pines que permiten montar la Teensy evitando la soldadura entre la placa y el microcontrolador, en tanto, en la parte de potencia la mayoría de los componentes requieren un voltaje de 12V por lo que se distribuye tres salidas para cada fuente, sin embargo, la Jetson Nano y los encoder funcionan con 5V por lo que se agregó una sección correspondiente cuyo origen proviene del reductor de voltaje XL4015, ver la Figura 55.

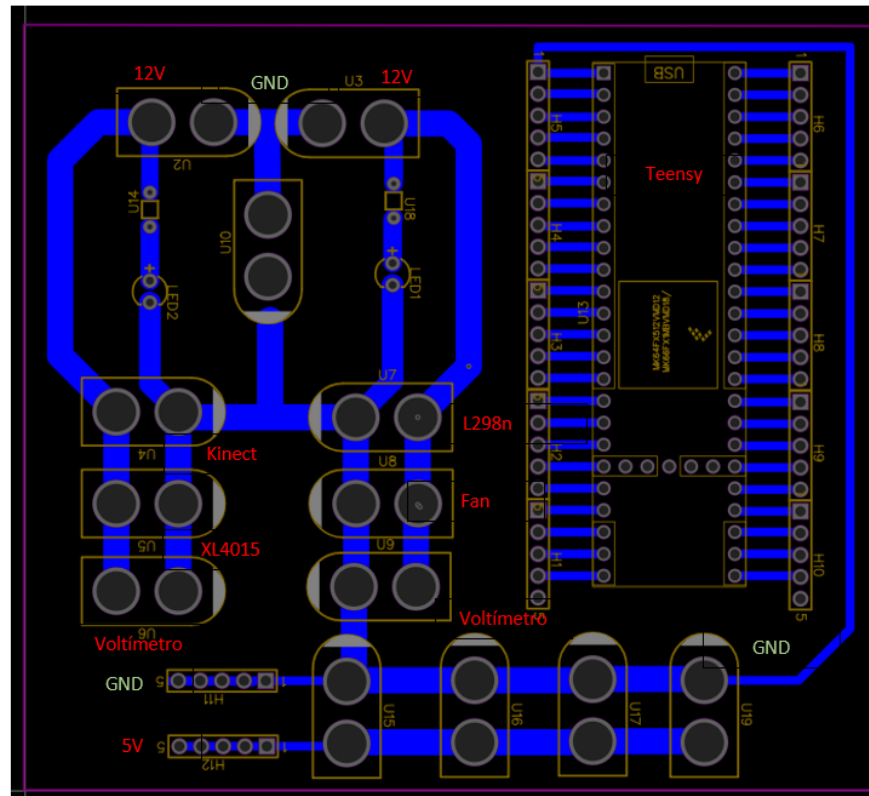
### **Especificaciones Finales del Diseño**

Al haber analizado las diferentes opciones de solución para cada aspecto destacado en los subsistemas seleccionamos los que presentan mayor valoración en sus ponderaciones como se indica en

la Tabla 31, con lo cual establecemos las condiciones finales para la construcción e implementación de la plataforma.

**Figura 55**

*Diseño de la placa PCB para circuito de potencia y control*



Nota: Placa desarrollada en EastEDA. Figura realizada por autor

Para el módulo de estructura, se analiza la base inferior de la plataforma tomando en cuenta el peso que soportara, en la misma se ubicaran los actuadores y elementos electrónicos que permiten su desplazamiento. En el módulo de la movilidad se analiza los actuadores con el modelamiento cinemático y dinámico. El funcionamiento de plataforma se enfoca en generar mapas 2D utilizando Gmapping y detectar objetos al implementar YOLO con la finalidad de relacionarlos utilizando algoritmos basados en lenguaje ontológico web.

**Tabla 31***Módulos de la plataforma móvil*

<b>Módulo</b>	<b>Subsistema</b>	<b>Alternativa de Solución</b>
Estructura de la plataforma móvil		La plataforma tendrá una distribución diferencial, dos ruedas motrices y una rueda castor.
	Estructura	La base del sistema se construirá en MDF cortador por láser. Los soportes estructurales serán de latón de cobre. Los soportes para piezas externas estarán hechos mediante impresión 3D con PLA.
Movilidad de la plataforma móvil	Potencia y Tracción	Para la alimentación se utilizarán dos baterías selladas de ácido con un voltaje nominal de 12V y 5 Ah. Los motores son de corriente continua con caja de reducción en relación 270:1 provisto de encoder tipos hall de interrupción 64CPR. El controlador a implementar es una tarjeta Teensy 3.5.
		Se utilizará un RPLIDAR A1M8.
	Percepción	El sensor inercial a instalarse será un MPU6050. La cámara RGB-D es una Kinectv1.
Funcionamiento de la plataforma móvil	Procesamiento y Detección	El procesador central será una Jetson Nano de 4GB con la instalación de Ubuntu 18.04. La detección de objetos será mediante el algoritmo YOLOv3. El método de SLAM a utilizar será Gmapping.
	Mapeo y Ontología	La relación de objetos con el mapa será mediante owlready2 para realizar algoritmos de ontologías en Python.

Nota: La tabla muestra los subsistemas agrupados por módulos con su respectiva solución de implementación.

## Resumen

La metodología y diseño inician a partir de los requerimientos del cliente o usuario para el producto, las cuales son convertidas a especificaciones técnicas que sea puedan medir, tomando el modelo de diseño de producto en (Ulrich & Eppinger, Diseño y desarrollo de productos, 2013) establecemos un diagrama funcional con las entradas y las salidas del sistema, subsecuentemente se dividen en funciones de manera progresiva hasta que simplificar el problema.

Las subfunciones obtenidas son asociadas de acuerdo a las similitudes que presentan formando subsistemas con su respectiva función. Por cada subsistema se presentan aspectos que requieren especificaciones de diseño por lo cual se escogen los criterios relevantes para ser evaluados al proponer alternativas de solución correspondientes, el resultado prioritario es obtenido al aplicar la matriz de Holmes lo que permite realizar dar solución a cada subsistema, obtenido las especificaciones finales para el diseño.

El diseño comienza con el análisis estático de la estructura base de material MDF para determinar los esfuerzos y deformación máximos que experimentaría al ser sometida al peso aproximando de sus componentes estructurales y de funcionamiento. Para la elección del motor DC, se analiza la plataforma de forma cinemática y dinámica para determinar el torque necesario que le permita moverse, determinamos los sensores y las tarjetas de control que permitan el funcionamiento eléctrico y software del sistema, de acuerdo al consumo energético total seleccionamos la fuente de energía que abarque a la plataforma completa. Finalmente, se implantan los algoritmos para mapear, detectar objetos y relacionarlos por ontología.

## Capítulo IV

### Construcción y Pruebas

El capítulo explica la implementación del robot que comprende la construcción mecánica de la plataforma diferencial y la implementación de algoritmos que cumplan con la generación de mapas, detección de objetos y permitan establecer las relaciones entre los mismos mediante ontologías.

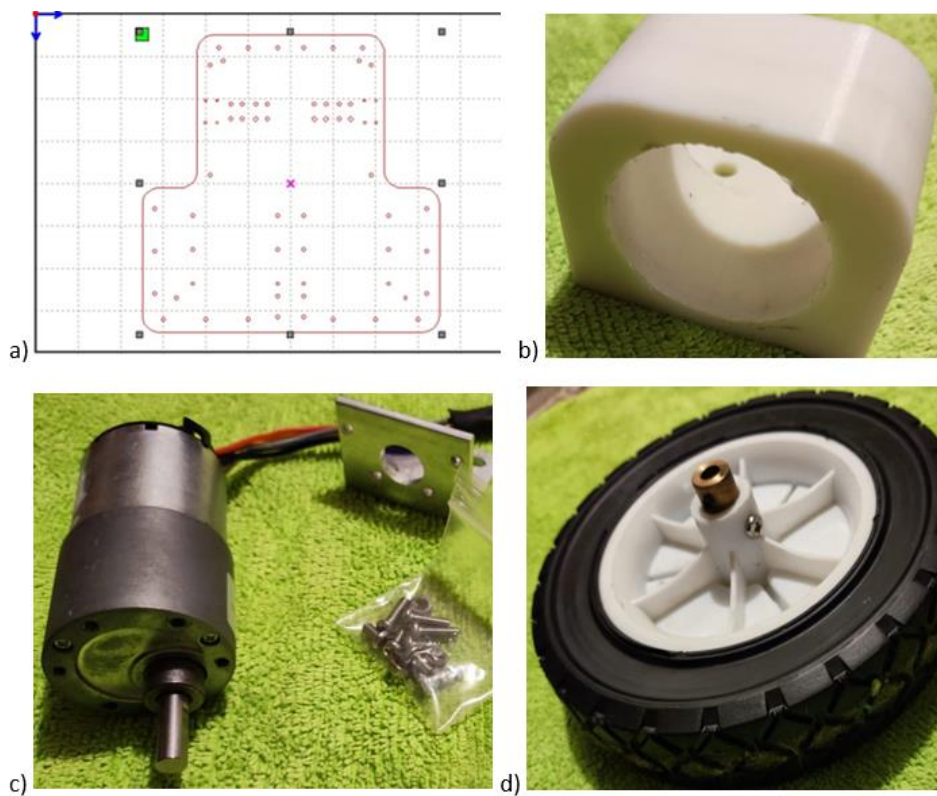
Al concluir la construcción del dispositivo se realizan un conjunto de pruebas que abarcan los diferentes aspectos en la funcionalidad de la plataforma.

#### Construcción de la plataforma

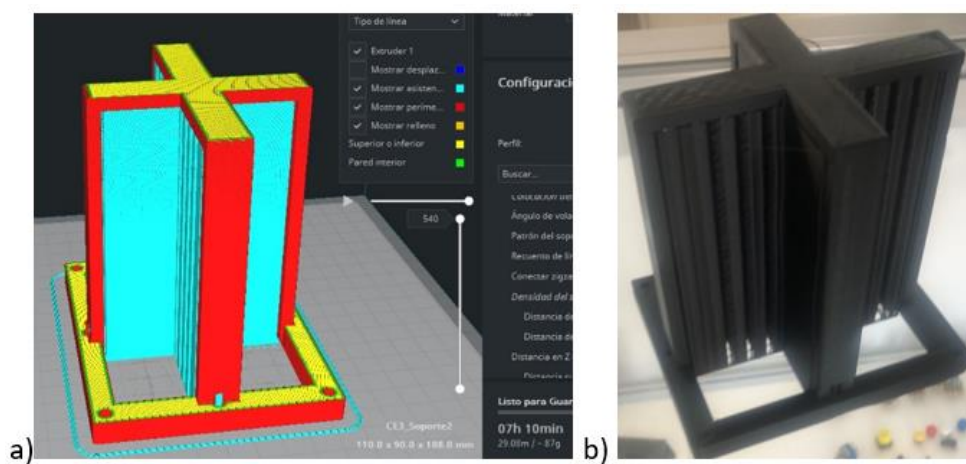
La estructura está distribuida en una configuración modular donde las diferentes piezas puedan ser montadas y viceversa de forma sencilla sin comprometer al diseño de forma conjunta. La base del móvil se realiza en corte láser en MDF de 9 mm, mientras los apoyos secundarios para los motores están realizados en impresión 3D con filamento de PLA, el resultado se visualiza en la Figura 50.

El motor en c) está provisto de 4 tornillos Ansis Metric M3x6 para sujetar al actuador al soporte y 2 pares de tornillos M3x12 para sujetar el conjunto a una base, en d) la rueda es de diámetro de 152.4mm provisto de un eje de latón de cobre que será la unión entre el actuador y el neumático. En la Figura 57 se presenta el tiempo de impresión para el soporte de la batería y su correspondiente resultado.

El conjunto de la rueda es acoplado con 8 pernos M5x12 al soporte secundario, mientras que en la batería son necesarios 3 pares de tornillos M4x20 con sus respectivas tuercas incorporadas dentro de las piezas impresas, ver Figura 58.

**Figura 56***Piezas para la base de la plataforma diferencial*

Nota: a) Diseño para corte láser b) acople secundario del motor c) motor DC d) neumático de la plataforma.

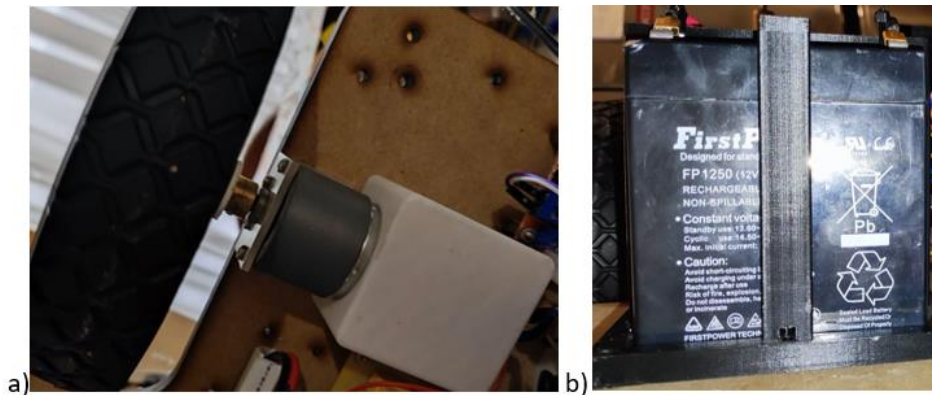
**Figura 57***Soporte para la batería*

Nota: a) Diseño CAD del soporte b) impresión 3D

Finalmente, la rueda tipo castor es acoplada a la base con 4 tornillos M4x40 junto con el acople diseñado en el capítulo 3 para compensar la diferencia de altura resultante por el diámetro de las ruedas con la configuración diferencial en la Figura 59 la plataforma es capaz de moverse de acuerdo al diseño planteado

### Figura 58

*Instalación parte motriz y energética a la base*



Nota: a) Acoplamiento de las ruedas b) batería

### Figura 59

*Sistema de locomoción del móvil*



Nota: Fijación del sistema de locomoción a la base de la plataforma.

En la base de MDF son montadas:

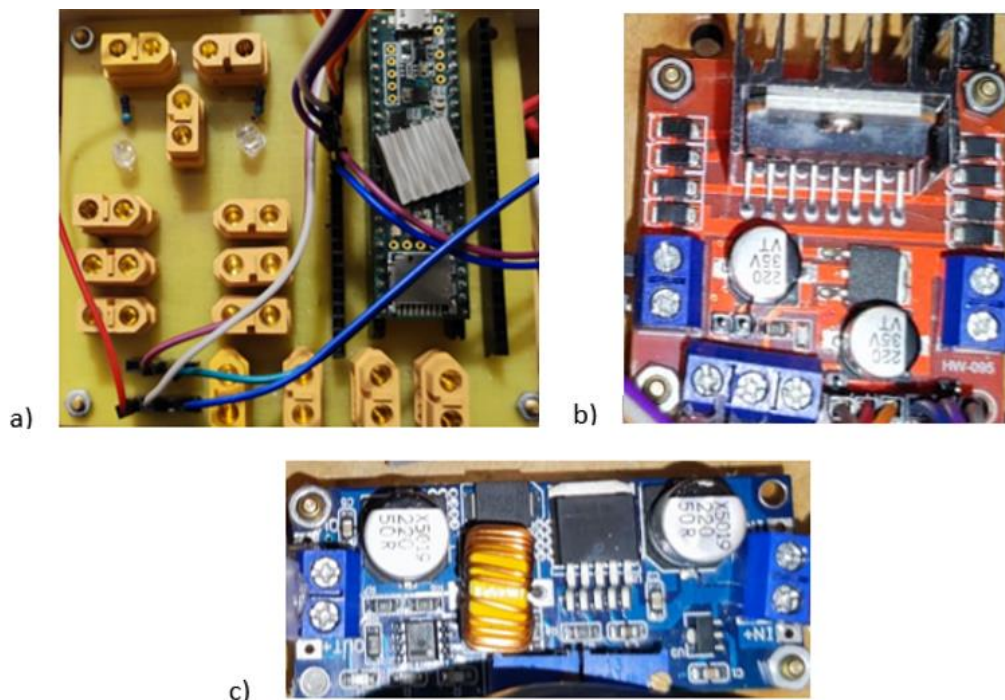
- a) La placa PCB integrado con el microcontrolador Teensy 3.5
- b) El módulo controlador de motores L298n
- c) El regulador XL4015

Mediante espaciadores hexagonales con longitud de 20 mm y tornillos M3x12 como se muestra en la Figura 60.

A la placa se suministra de energía mediante las baterías, de esta manera se conectan 12v al módulo l298n para energizar los motores y al regulador Buck para obtener una salida de 5v. La cual proveerá a los sensores en los motores y a la placa Jetson nano; subsecuentemente al puente H se conectan los pines del microcontrolador (2-11), véase la Figura 61.

**Figura 60**

*Montaje de los componentes electrónicos*

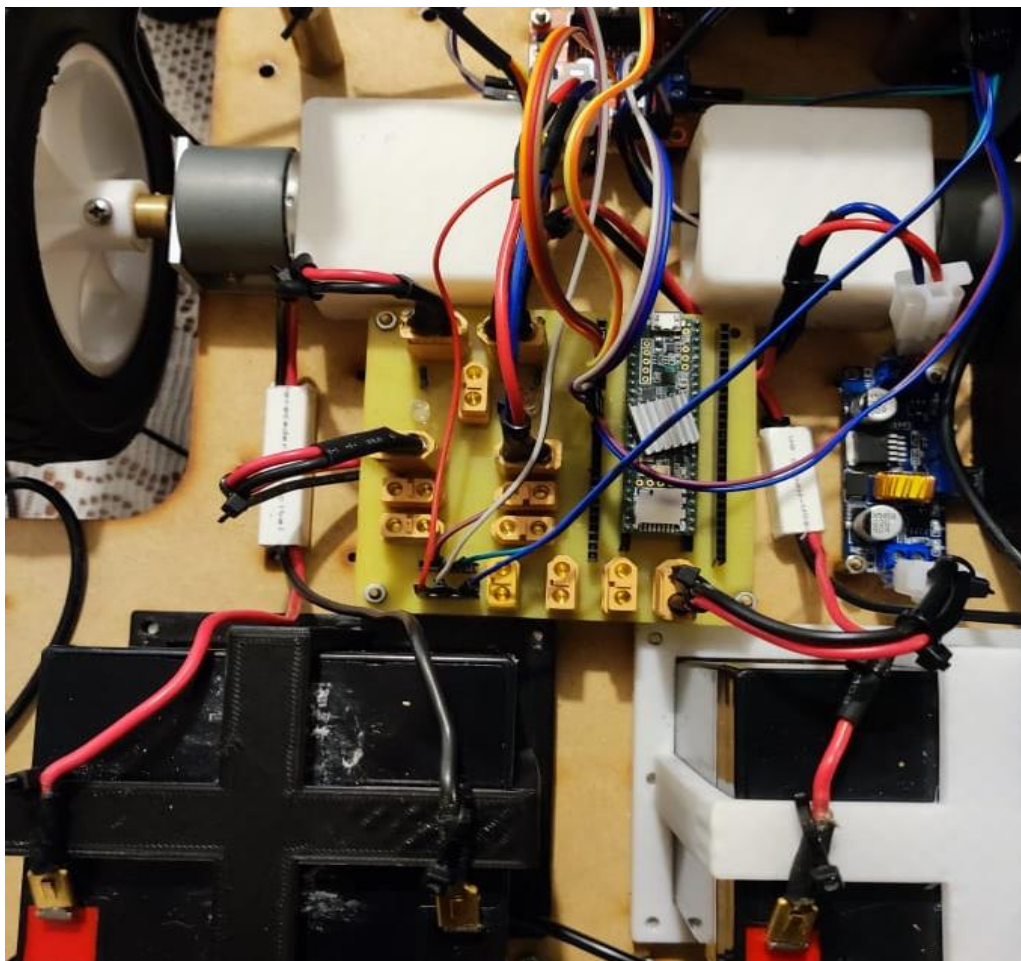


Nota: a) Placa PCB integrado con el microcontrolador Teensy 3.5 b) Módulo L298n c) El regulador XL4015



**Figura 61**

*Conexión interna entre los componentes electrónicos*



Nota: Integración de la parte de control y potencia a la plataforma.

Se fabrican dos conjuntos de soportes con dimensiones de 150 mm y 120 mm, ejemplo de ello se observa en la Figura 62. Para asegurar la integridad de los componentes internos al igual que su apariencia se utiliza cuatro piezas impresas a una altura de 150 mm correspondiente con la forma de la base, adicionalmente se usan los ejes en la Figura 63 para el ensamblaje que comprende las piezas indicadas se utilizan veintiséis tornillos M4, cuatro de 12 mm y veintidós de 20 mm parte del proceso se visualiza en la Figura 64.

**Figura 62**

*Conjunto de soporte (150 mm)*



Nota: Soportes de apoyo para la plataforma.

**Figura 63**

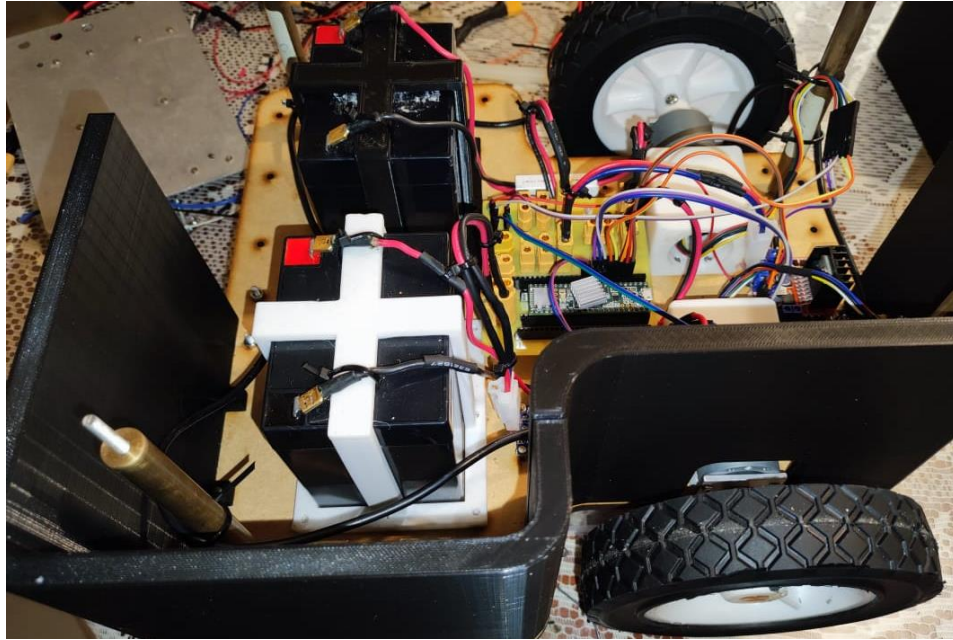
*Pieza frontal derecha del contorno*



Nota: Unión base de la plataforma a los componentes superiores.

**Figura 64**

*Ensamblaje de la plataforma móvil*



Nota: Resultado del ensamblaje.

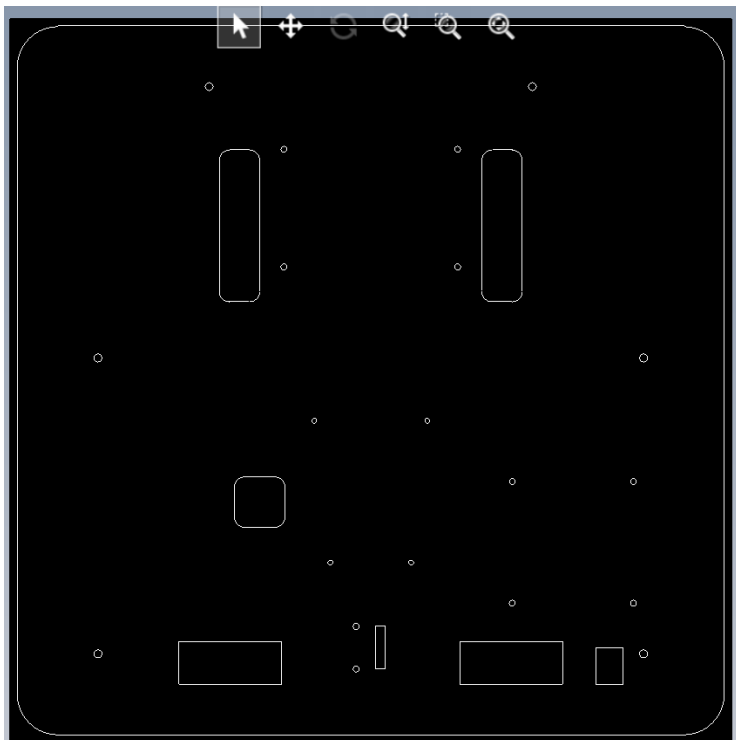
Para la cubierta de la plataforma se utiliza un corte realizado en MDF de 9 mm con la forma de la Figura 65, en la sección delantera de la pieza se ubican voltímetros para monitorear el estado de las baterías y de igual manera es montado el sensor IMU utilizando espaciadores de 20 mm de la Figura 66

Continuamos con el acoplamiento de la tarjeta embebida Jetson Nano y el sensor LiDAR A1M8, en los dos casos se empleó espaciadores de 10 mm y 12 mm respectivamente, además en la tarjeta se conectan al sensor inercial, ver la Figura 67. El soporte de la cámara RGB (Kinect del Xbox 360) de la Figura 68 está constituido por:

- Un asiento hecho con impresión 3D que soporta una sección de tubo cilíndrica con diámetro igual a 26 mm y longitud correspondiente de 330 mm
- Unida mediante un acople a un tubo que posee un diámetro de 22 mm y una longitud de 390 mm
- La sección superior se encuentra ubicada con la base para la cámara RGB-D.

**Figura 65**

*Forma para el corte de la cubierta de la plataforma móvil*



Nota: Boceto de la pieza a cortar en corte laser.

**Figura 66**

*Montaje del sensor IMU y voltímetros*

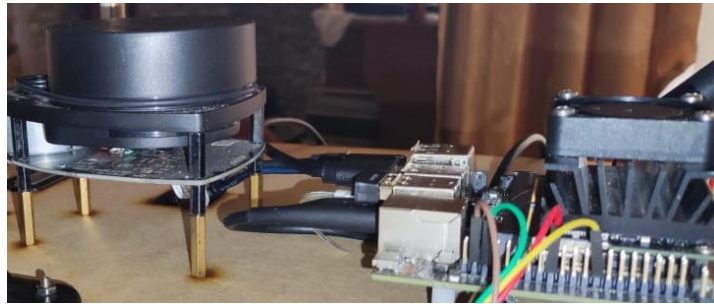


Nota: En la figura se observa cómo se montó los sensores.



**Figura 67**

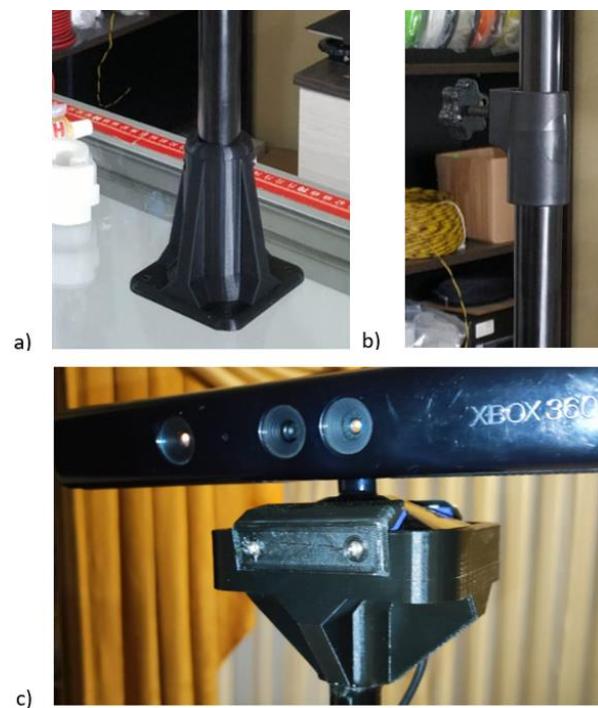
*Instalación de los dispositivos A1M8 y Jetson Nano 4GB*



Nota: Conexión de la Jetson Nano con los diferentes dispositivos.

**Figura 68**

*Soporte para la cámara Kinect*



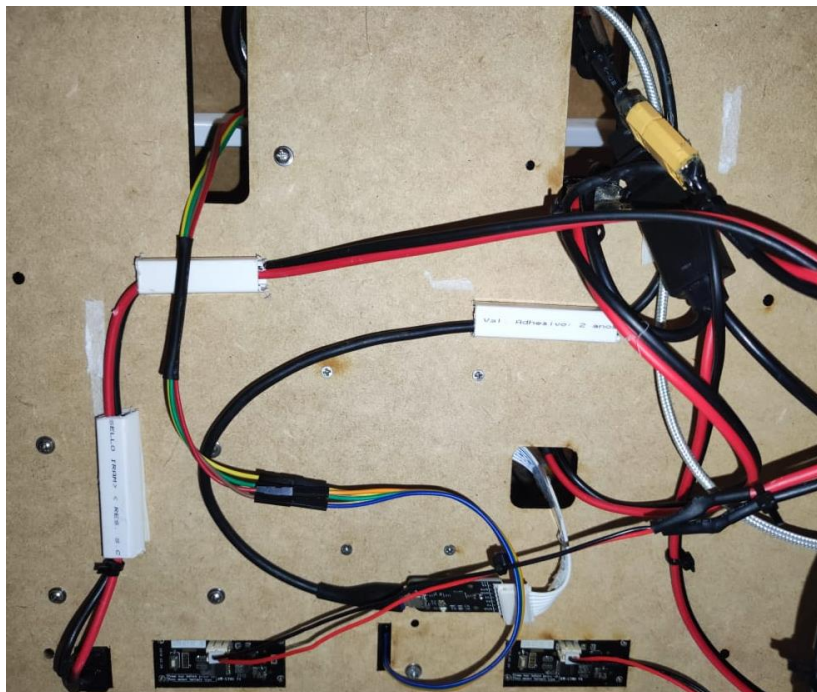
Nota: a) Asiento de la Kinect b) tubo de unión c) cámara Kinect.

El cableado es distribuido utilizando canaletas acopladas por debajo de la cubierta como se muestra en la Figura 69, seguidamente a la tarjeta se conectan los dispositivos USB que son: el adaptador Wi-Fi, el microcontrolador, el LiDAR y la cámara Kinect como se observa en la Figura 70. Finalmente, en la

plataforma son acoplados los soportes de 120 mm mencionados previamente para colocar el piso final y asegurarlo con cuatro tornillos M4x10, la plataforma ensamblada en su totalidad con respecta al hardware se muestra en la Figura 71.

### Figura 69

*Distribución del cableado del sistema*



Nota: La figura denota la disposición del cableado interno de la plataforma.

### Figura 70

*Conexión USB de los dispositivos a la tarjeta Jetson*



Nota: En la figura se puede apreciar el switch de encendido de la parte de control.

**Figura 71***Ensamblaje final del robot móvil*

Nota: Se aprecia el resultado final de la plataforma.

### **Software de la plataforma móvil**

El software de la plataforma implica la programación de algoritmos que ejecuten odometría, generación de mapas mediante mapeo y localización simultánea, detección de objetos y relacionamiento ontológico.

### **Controlador de los motores**

Los datos se ingresan al software Matlab para identificar la planta con el comando “systemidentification” y utilizando un tiempo de muestra igual 0.1 segundo.

El motor derecho presenta una aproximación del 92.68% a continuación, se presenta su función de transferencia:

$$Gz1 = \frac{0.3171z^{-1} - 0.37z^{-2}}{1 - 1.77z^{-1} - 0.532z^{-2} + 0.7133z^{-3}} \quad (17)$$

El motor izquierdo presenta una aproximación del 93.67%, su función de transferencia es la siguiente:

$$G_{z2} = \frac{0.3549z^{-1} - 0.3531z^{-2}}{1 - 1.187z^{-1} - 0.5112z^{-2} + 0.7029z^{-3}} \quad (17)$$

La toma de datos se efectuó excluyendo el contacto de las ruedas con a la superficie. Se calcula un control PID para los motores, las constantes se observan en la Tabla 32 y en la Tabla 33.

**Tabla 32**

*Constantes PID para el motor derecho*

<b>Coeficientes</b>	<b>PID</b>
Ganancia proporcional (Kp)	1.71
Tiempo Integral (ki)	0.30
Tiempo derivativo (kd)	0.03

Nota: La tabla contiene los valores para el control del motor derecho

**Tabla 33**

*Constantes PID para el motor izquierdo*

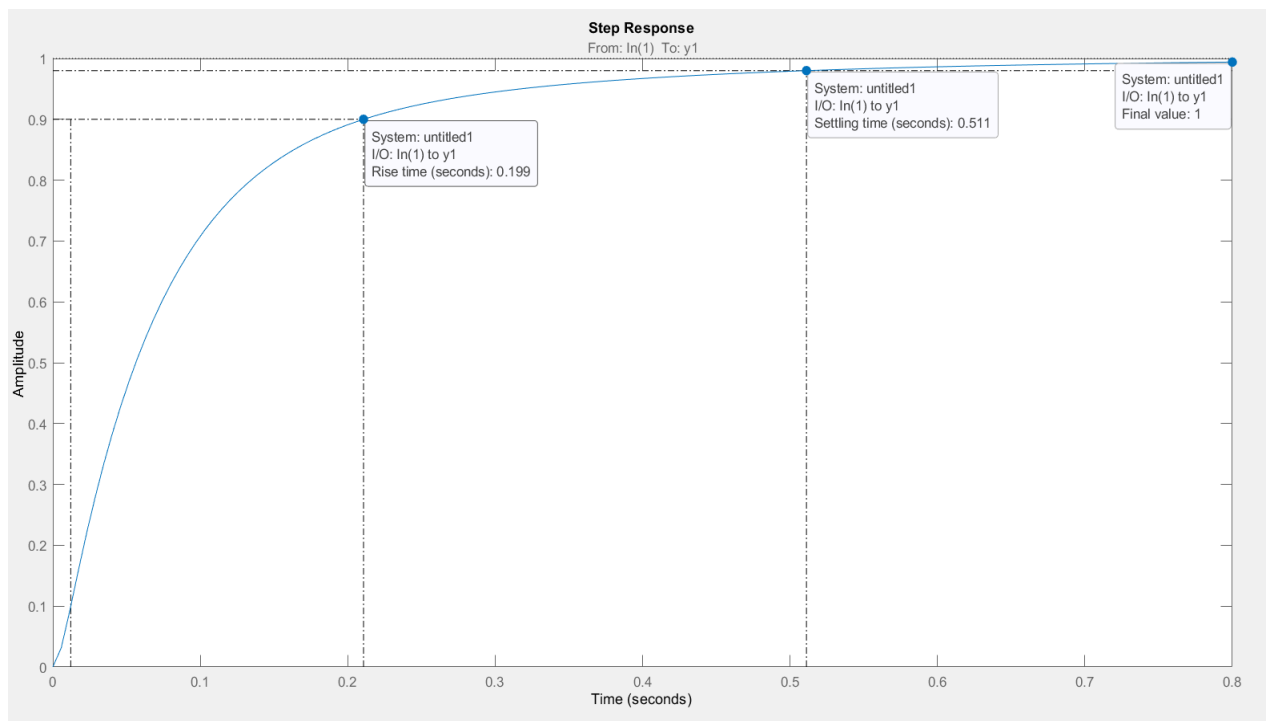
<b>Coeficientes</b>	<b>PID</b>
Ganancia proporcional (Kp)	1.71
Tiempo Integral (ki)	0.30
Tiempo derivativo (kd)	0.03

Nota: La tabla contiene los valores para el control del motor izquierdo.



**Figura 72**

*Respuesta del controlador del motor derecho*

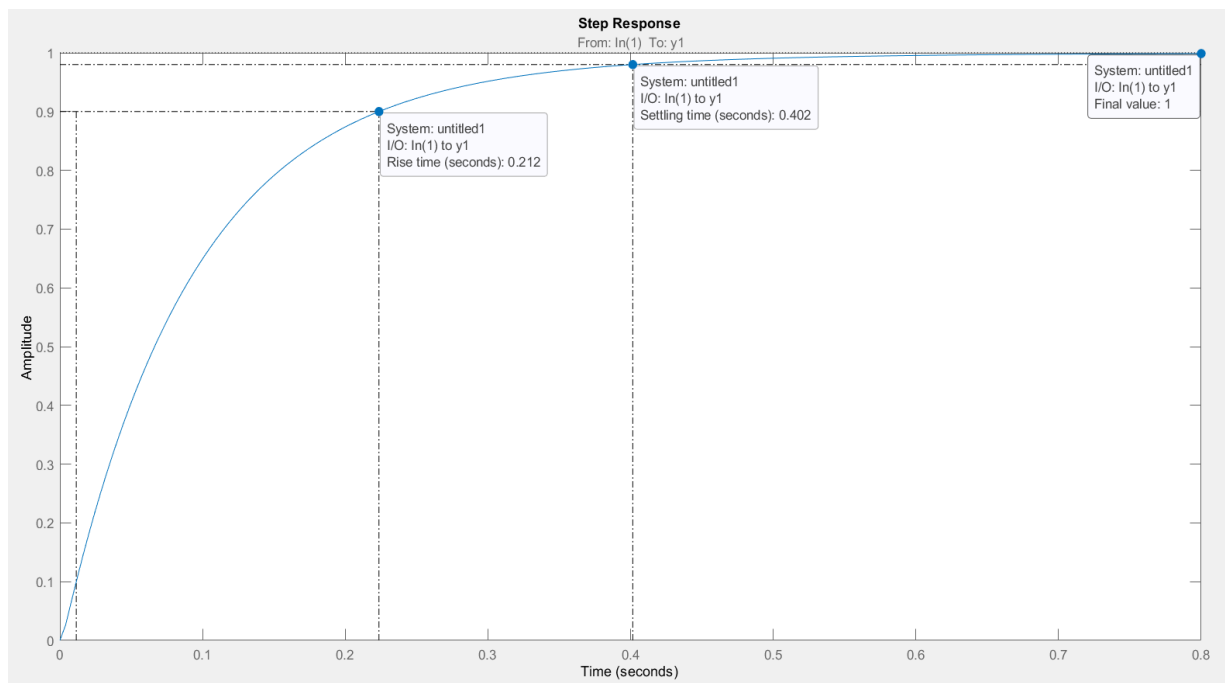


Nota: Valores del controlador PID para el motor derecho para una respuesta en escalón unitario, figura realizada por el auto utilizando MATLAB.

En la Figura 72 se observa que el tiempo de levantamiento (rise time) es de 0.199 segundos, el tiempo de asentamiento es de 0.511 segundos y el valor final alcanzado por el controlador es igual a 1, lo cual muestra que el controlador es bastante efectivo.

En la Figura 73, el tiempo que le toma al controlador del motor izquierdo alcanzar del 10% al 90% de su valor final es 0.212 segundos, para estabilizarse le toma un tiempo de 0.402 segundos, de igual manera que el controlador del motor derecho alcanza el valor correspondiente a 1.

A pesar que los controladores alcanzan la respuesta esperada, el controlador derecho le toma 0.1 segundos más que al controlador izquierdo en estabilizarse, lo cual representará un error durante el cálculo de la odometría.

**Figura 73***Respuesta del controlador del motor izquierdo*

Nota: Valores del controlador PID para el motor derecho para una respuesta en escalón unitario, figura realizada por el auto utilizando MATLAB.

Los valores obtenidos son utilizados en el código de los anexos el cual fue desarrollado en Arduino debido a la compatibilidad que presenta la tarjeta Teensy con la interfaz del programa.

Mientras tanto para el procesador central se utiliza ROS-Melodic el cual es un sistema de código gratuito que está provista por una vasta variedad de librerías y herramientas informáticas compartidas por los usuarios del software.

A continuación, se indica el funcionamiento de la plataforma móvil mediante nodos, dónde se aprecia la relación de datos con entradas y salidas de los distintos tópicos.

### Pruebas y Resultados

La primera prueba corresponde a la velocidad, donde la plataforma avanza tres distancias: 1 m, 1.5 m y 2 m por cada velocidad: 0.15, 0.2 y 0.25 m/s sucesivamente, de manera que se calcula el tiempo de desplazamiento del dispositivo y se compara con un el tiempo cronometrado por el robot. Los valores

tabulados se observan en la **Tabla 34**, **Tabla 35** y **Tabla 36**; con lo cual es posible determinar el error por cada velocidad.

**Tabla 34**

*Comprobación de velocidad de la plataforma a 0.15 (m/s)*

Distancia (m)	Tiempo Real (sg)	Tiempo Medido	Error (%)
1	6.67	6.72	0.5
1.5	10	10.12	1.2
2	13.33	13.49	1.6

Nota: Valores correspondientes a los tiempos calculados y medidos seteados a la velocidad de 0.15 m/s

**Tabla 35**

*Comprobación de velocidad de la plataforma a 0.20 (m/s)*

Distancia (m)	Tiempo Real (sg)	Tiempo Medido	Error (%)
1	5	5.18	0.8
1.5	7.5	7.66	1.6
2	10	10.23	2.3

Nota: Valores correspondientes a los tiempos calculados y medidos seteados a la velocidad de 0.2 m/s

**Tabla 36**

*Comprobación de velocidad de la plataforma a 0.25 (m/s)*

Distancia (m)	Tiempo Real (sg)	Tiempo Medido	Error (%)
1	4	4,16	1,6
1.5	6	6,29	2,9
2	8	8,37	3,7

Nota: Valores correspondientes a los tiempos calculados y medidos seteados a la velocidad de 0.25 m/s

La siguiente prueba es para la odometría, donde la plataforma ejecutará una trayectoria en forma cuadrada de 1.5m, ver la Figura 74, donde el /serial\_node que corresponde a la tarjeta Teensy se subscribe

al tópicico `/cmd_vel`, el cual envía los valores para la velocidad lineal y angular los cuales utilizan para obtener calcular la posición de acuerdo a la cinemática explorada en el capítulo 3, el nodo `/odom` publica la información como un mensaje en cada de ser requerido.

#### Figura 74

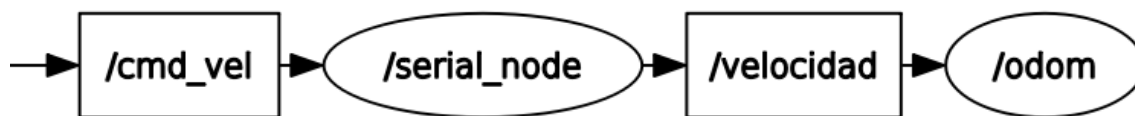
*Prueba: Cuadrado de 1.5x1.5*



Nota: En la imagen se observa un cuadrado

#### Figura 75

*Nodos para la prueba de odometría*



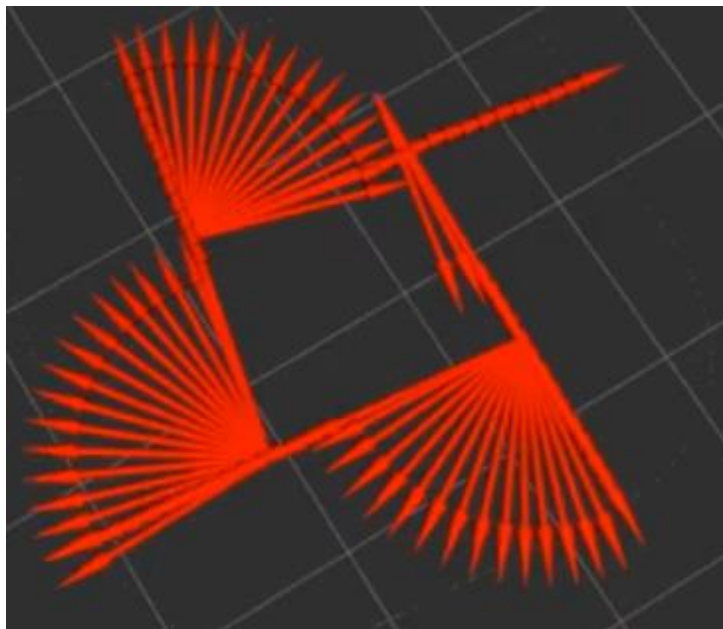
Nota: La imagen muestra los nodos y tópicos utilizados durante la prueba.

De forma adicional se utiliza RViz, la herramienta de visualización de ROS para observar comportamiento de la odometría. De acuerdo a la Figura 76 existe una desviación a medida que la plataforma avanza y realiza los giros lo que corresponde al error acumulativo por la medición de la

distancia por medio de los encoder. En la Tabla 37 se presentan los resultados, el error máximo para X es de 10% y para Y es 22%.

**Figura 76**

*Visualización de Odometría en RViz*



Nota: Las flechas indican el avance de la plataforma para representar la odometría.

**Tabla 37**

*Resultados de la prueba para odometría*

<b>Puntos</b>	<b>P1</b>	<b>P2</b>	<b>P3</b>	<b>P4</b>
X (m)	1.483	1.396	-0.037	0.055
Y (m)	0.005	1.521	1.474	-0.224
X deseado	1.5	1.5	0	0
Y deseado	0	1.5	1.5	0
Error en X (%)	1.7	10.4	3.7	5.5
Error en Y (%)	0.5	2.1	2.6	22.4

Nota: Resultados de los puntos que forman la trayectoria.

Para la obtención del mapa se utiliza SLAM, la muestra el diagrama de conexión entre nodos y tópicos para la tarea, para lo cual utiliza el nodo `/slam_gmapping` que se suscribe a los tópicos `/scan` para construir la nube de puntos y a `/tf` que proporciona la posición de la plataforma a través de `/odom` como se muestra en la Figura 78. La plataforma recorre el entorno utilizando el nodo `/joy_node` para enviar las velocidades seteadas en 0.25 m/s para el avance y 0.3 para rotación, el resultado se presenta en la Figura 79.

Aplicamos el algoritmo YOLO utilizando nodo el `/darknet_ros` el cual se suscribe al tópico proveniente del nodo `/camera` de la Kinect, ver la Figura 81. El resultado de la detección se observa en la Figura 82 de forma visual. Debido a que la plataforma es utilizada dentro de un ambiente hogareño se presenta algunas detecciones de objetos pertenecientes al entorno como se muestra en la Tabla 39.

**Figura 77**

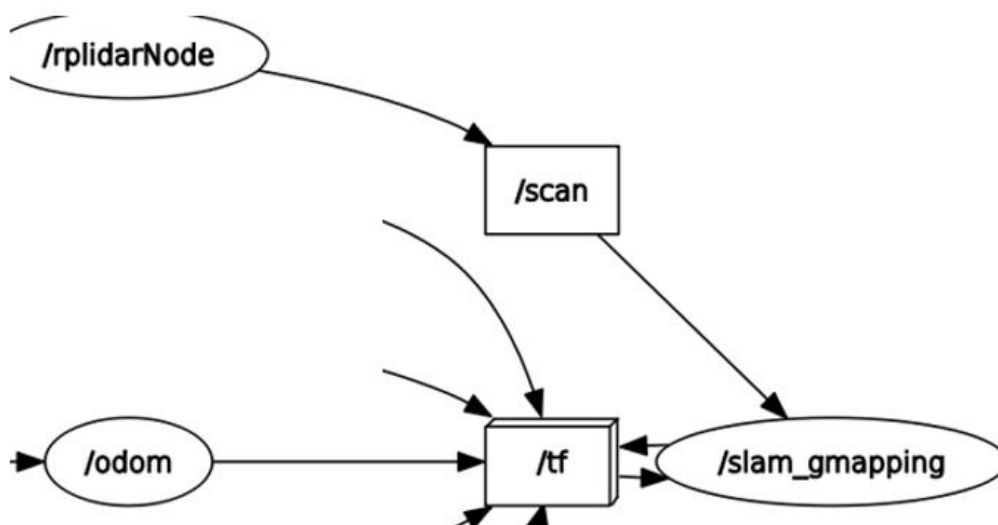
*Ejecución de SLAM*



Nota: Nodos y tópicos que intervienen en la obtención del mapa 2D.

**Figura 78**

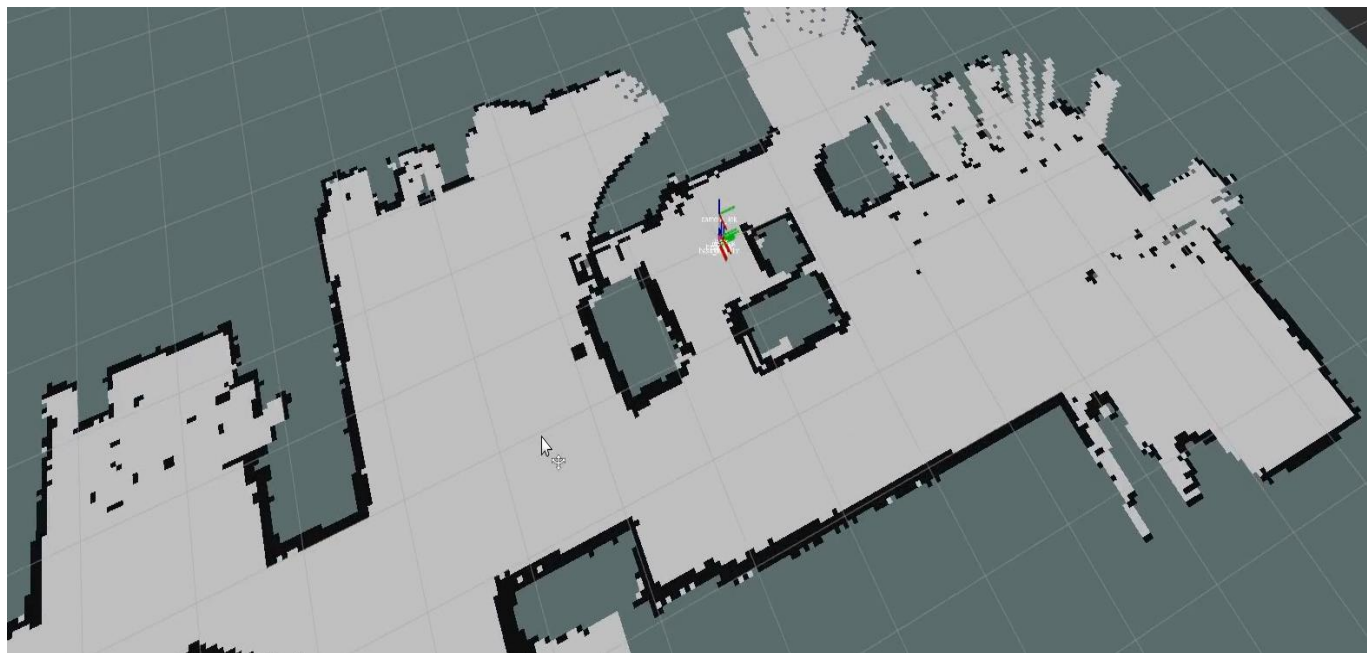
*Nodos relevantes para el Gmapping*



Nota: El nodo para la ejecución de Gmapping y los tópicos a los que se suscribe.

**Figura 79**

*Visualización del mapa generado mediante SLAM-Gmapping*

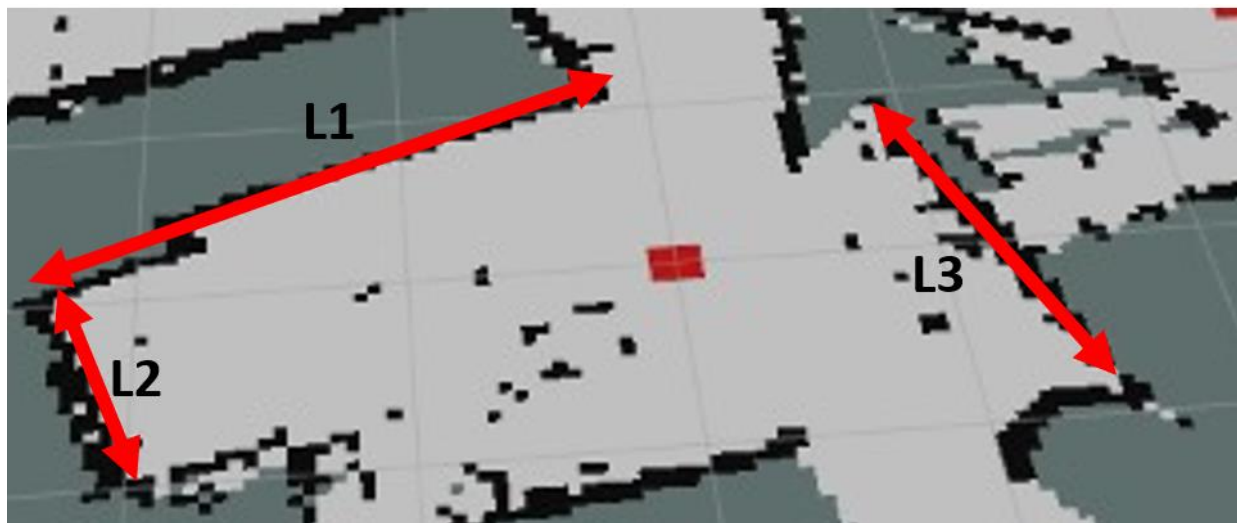


Nota: Es un mapa generado por probabilidad de ocupación y presentado en RViz.

En la Figura 80 se observa un cuarto mapeado sobre el cual se colocan flechas para representar las distancias a medir, dentro del mapa bidimensional cada cuadro corresponde a 1 m, lo cual facilita la prueba.

**Figura 80**

*Prueba del mapa generado sobre un cuarto*



Nota: Un cuarto mapeado utilizando Gmapping, se colocan flechas para indicar las distancias medidas durante la prueba.

Los resultados de la prueba a la cual se somete el mapa generado por SLAM, son presentados en la Tabla 38, el error entre las distancias no supera el 5.45%.

**Tabla 38**

*Resultados de la prueba para el mapa generado*

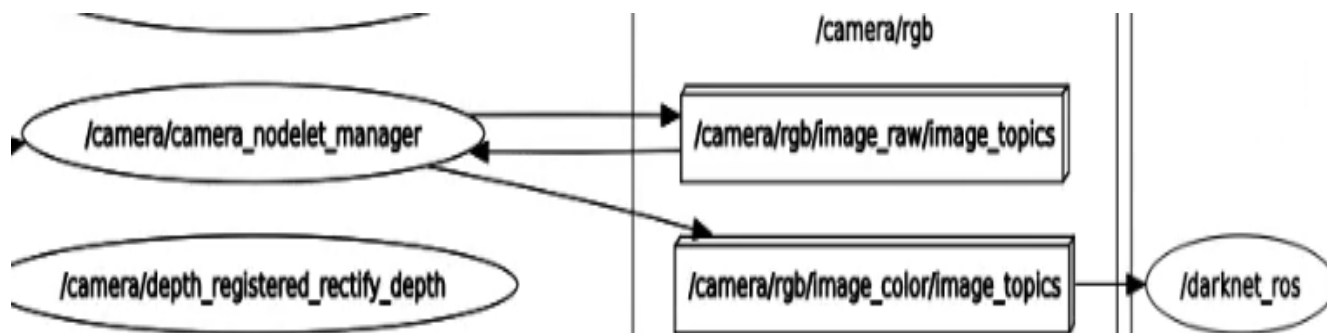
Lado	Distancia		Error (%)
	Real (m)	SLAM ( )	
L1	2.48	2.51	1.21
L2	1.10	1.04	5.45
L3	1.98	2.06	4.04

Nota: Resultados por la medición de las distancias en el mapa.



Figura 81

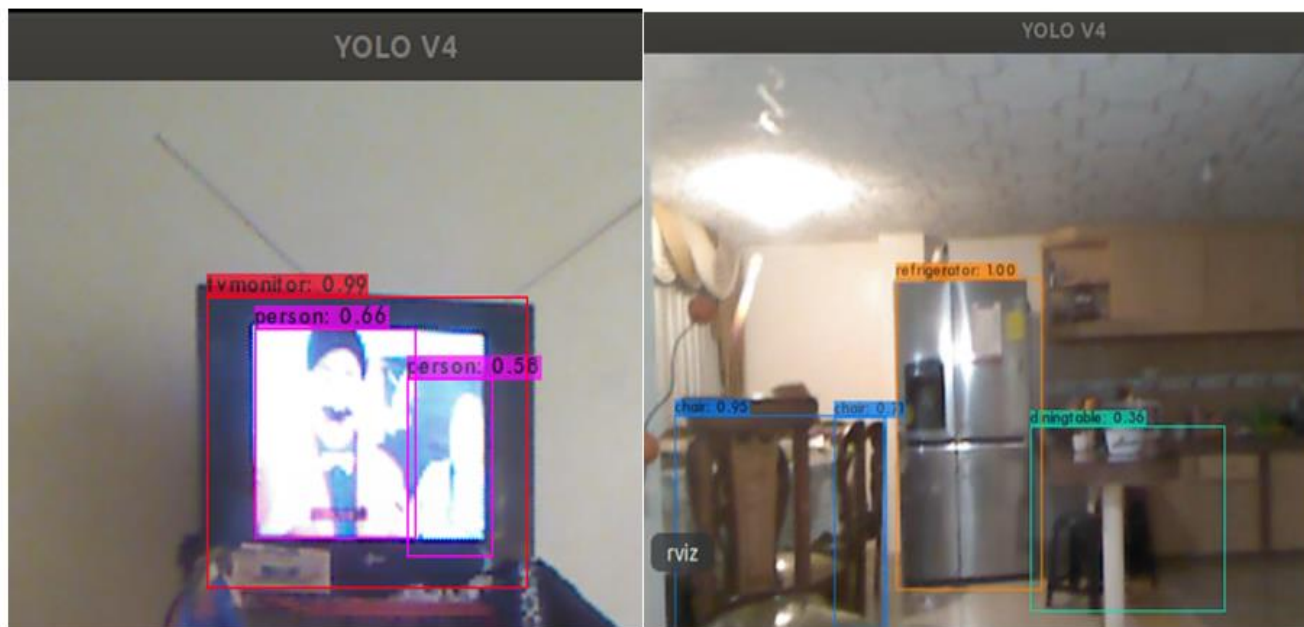
Nodo `/darknet_ros`



Nota: El nodo `/darknet_ros` se suscribe a `/image_color` para obtener la imagen de la Kinect.

Figura 82

Detección de objetos con YOLO



Nota: Se observa la aplicación del algoritmo de YOLO.

**Tabla 39***Resultados de la prueba para YOLO*

<b>Objeto Real</b>	<b>Clase</b>	<b>Probabilidad (%)</b>
Televisión	tvmonitor	99
Persona	person	58
Cama	bed	93
Silla	chair	95
Mochila	suitcase	39
Armario	refrigerator	57
Sofá	bed	42

Nota: La tabla presenta algunos resultados en la detección de objetos.

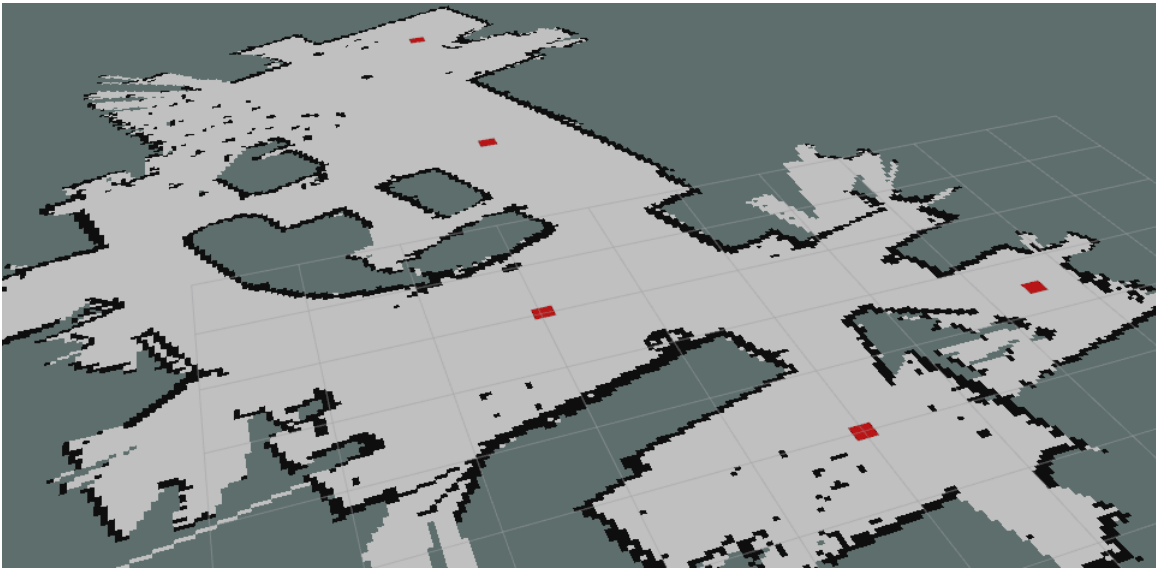
El mapa obtenido representa un área conformada por cuartos, por lo cual se utiliza un nodo para representar referencias dentro de cada uno, la Figura 83 muestra el nodo encargado de publicar una representación gráfica sobre el mapa de acuerdo a coordenadas pertenecientes al mapa. En RViz se puede verificar la aplicación, ver la Figura 84.

**Figura 83***Nodo para publicar las locaciones de interés*

Nota: Visualización del nodo que grafica los puntos en el mapa.

**Figura 84**

*Cuartos referenciados dentro del mapa*



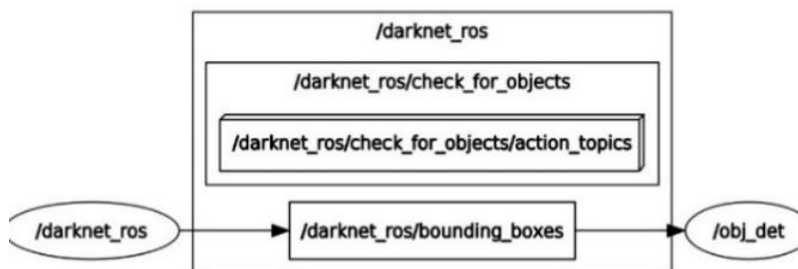
Nota: Los cuadrados rojos representan un punto ubicado dentro del respectivo cuarto.

Finalmente aplicamos el nodo `/obj_det` que requiere la información concerniente a la detección de objetos, asigna la clase de detección y la agrupa de acuerdo de acuerdo al cuarto más cercano a la plataforma.

El resultado final es un archivo tipo owl, el cual puede ser abierto en Protégé como se observa en la Figura 86.

**Figura 85**

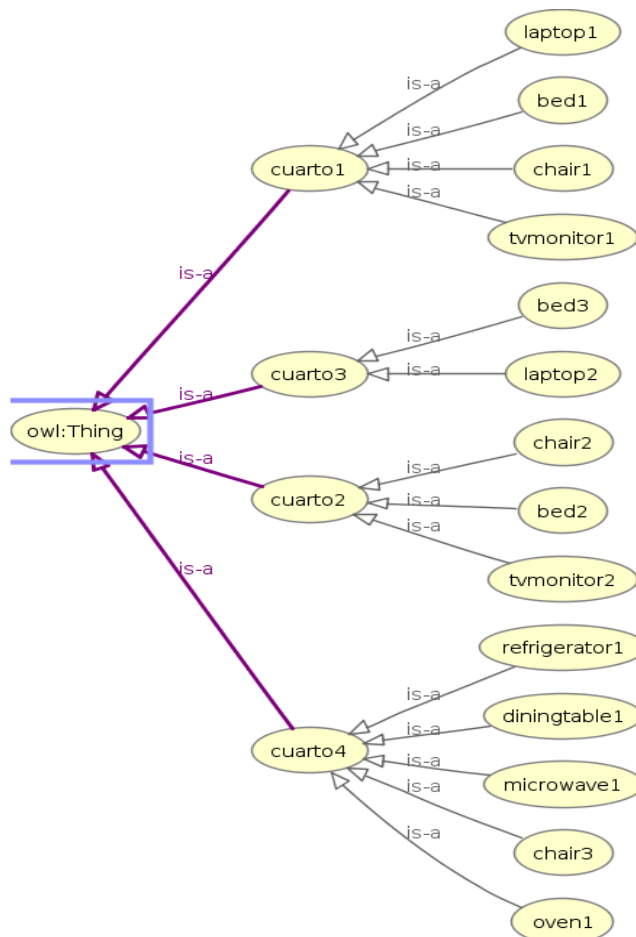
*Aplicación del nodo para ontología*



Nota: Suscripción del nodo `/obj_det` al tópico `/bounding_boxes` de YOLO en `darknet_ros`.

Figura 86

Visualización de la ontología en Protégé



Nota: Ontología creada por la plataforma que relaciona los objetos detectados de acuerdo al lugar donde se encuentra.

### Resumen

Los resultados obtenidos para los subsistemas en el capítulo 3 permiten la construcción de la plataforma. Las repisas del dispositivo están hechas de MDF de 9 mm para el primer y segundo piso mientras que para el tercer piso es de 3 mm, se le dio forma por corte láser de acuerdo a las medidas de diseño y detalladas por los planos en los anexos. Adicionalmente, se utilizó filamento de PLA en la impresión 3D de piezas que tenían la función de soporte y unión de baterías, motores y sensores, para el acoplamiento de los componentes se utilizaron tornillos Ansis Metric M4.

En la parte electrónica se diseñó y fabricó una placa PCB en fibra de vidrio que permita el suministro de energía a la plataforma completa y la conexión del microcontrolador Teensy, además se incluyen el módulo l298n para el control de los motores y el reductor de voltaje XL4015 que son fijados al bastidor de la plataforma utilizando separadores hexagonales y tornillos M3.

El sistema operativo a utilizar es Ubuntu 18.04, la cual es una distribución compatible y sugerida para ser instalada en la Jetson Nano donde se desarrollaron los algoritmos en lenguaje C y Python que permiten la conseguir movilidad y funcionalidad principal de la plataforma. Al terminar la implementación mecánica, electrónica y software de la plataforma se procede a realizar las pruebas.

Las pruebas inician para la odometría en la cual la plataforma se desplaza en una trayectoria cuadrada con la finalidad de estimar su posición con respecto de las medidas en un entorno real, las siguientes pruebas consisten en generar el mapa del área de prueba utilizando la técnica Gmapping y verificar la detección de objetos por parte de la Kinect utilizando YOLO, la última prueba consiste en generar un archivo que contiene la ontología, la cual relaciona los objetos y cuartos.

## Capítulo V

### Conclusiones, Recomendaciones y Trabajos Futuros

#### Conclusiones y Recomendaciones

Se diseñó y construyó una plataforma móvil utilizando la teoría de diseño en (Ulrich & Eppinger, 2013), la cual planea la definición de subsistemas que abarcan el funcionamiento total. El móvil está provisto por dos baterías de 12v y 5Ah de acuerdo al consumo energético de la plataforma. Para el desplazamiento de la plataforma se utiliza una configuración diferencial con dos ruedas y una rueda loca o castor. La percepción del entorno está provisto por: encoder tipo hall, un MPU6050, LIDAR A-M8 y una cámara Kinectv1.

El material seleccionado para la construcción fue MDF en grosores de 9 mm y 3 mm, el material posee alta resistencia y bajo peso lo que permite que su desplazamiento sea fluido sin comprometer los elementos internos de la plataforma. La pieza diseñada en función del material se sometió a un análisis estático en CAE, la cual dio como resultado una deflexión máxima  $3.25e-3$  mm, la cual se considera pequeña y no representa un riesgo para el diseño.

El algoritmo usado para el control de los motores fue desarrollado en la plataforma de Arduino debido a la compatibilidad del microcontrolador con el software. Se utiliza una tarjeta Teensy 3.5, la cual posee una capacidad de 120 MHz y velocidad de 12Mbit/sg en el puerto micro-USB mejorando el tiempo de envío y recepción de datos. El controlador PID del motor derecho, el cual posee los valores de:  $k_p= 1.71$ ,  $k_i= 0.3$  y  $k_d=0.03$  es más rápido en su respuesta al llegar al setpoint a diferencia en contraste con el motor izquierdo que presenta los valores de:  $k_p= 1.6$ ,  $k_i= 0.3$  y  $k_d= 0.3$ ; sin embargo, el desempeño en la creación del mapa no se presentan errores notables debido al filtro de partículas en el método Gmapping.

Las pruebas de odometría al desplazarse acorde a la trayectoria propuesta proporciona un error máximo en el eje X del 10% mientras que en Y presenta un valor del 22%, por lo cual se añade un sensor

inercial para corregir este inconveniente. Con respecto a la evaluación de YOLO se proporcionaron falsos positivos en la identificación de los objetos-prueba, tal como se muestra en la Tabla 39 los objetos detectados con bajo porcentaje de probabilidad, ejemplo de ellos son el 39% y 42%, lo que resulta en la asignación errónea de clases (nombres) diferentes. El nodo `/place_points` permite una representación visual de los lugares representativos dentro del área mapeada, lo cual da como resultado la comprensión entre el dispositivo y el usuario.

Las pruebas finales corresponden a la creación del mapa semántico utilizando ontologías para asignar objetos a los lugares donde se encuentran, el resultado del proceso es un archivo con extensión tipo OWL, el cual tiene la cualidad de actualizarse a medida que el dispositivo realiza el mapeo del entorno. Sin embargo, debido a los errores en la detección es posible que objetos que no se encuentran en el lugar sean asignados a dicho lugar.

La presencia de intermitencias en la conexión de red impide la actualización del estado de la plataforma por el error en la ejecución del control al retraso que existe al enviar datos a medida que la plataforma continúa moviéndose, lo cual puede provocar accidentes que inutilicen el dispositivo o entregue valores inexactos en las mediciones que involucran distancias.

### **Trabajos Futuros**

La plataforma actual ha sido diseñada para desempeñarse en lugares cerrados y poco extensos, por lo cual una implementación en exteriores requiere de cambios que involucran actuadores de mayor potencia, posicionamiento global y conexión de red estable. La visualización del mapa generado por SLAM, así como los resultados de la ontología requieren de una máquina externa para el monitoreo, por lo cual es conveniente la adición de una pantalla para otorgar completa autonomía a la plataforma.

La visualización de los cuartos dentro del mapa se representa utilizando marcadores gráficos de las librerías de ROS, sin embargo, el método planteado es poco amigable con el usuario debido por lo cual

se recomienda que las figuras sean intercambiadas por letras para un alto entendimiento o utilizar algoritmos que diferencien entre lugares al tomar referencias como la detección de puertas para generar trayectorias entre los cuartos como se plantea en (Gallart, 2013). Dentro de la detección de objetos se sugiere la utilización de segmentación de imágenes debido a que permite obtener información de interés como lo distancia al objeto de manera que se pueda estimar la posición dentro del mapa.

El resultado actual del sistema es un mapa semántico, el cual puede emplearse para dotar de razonamiento al robot al utilizar un razonador como lo sugiere (PRONOBIS, 2011), con lo cual se infiera el tipo de cuarto que es, además de ser posible que los objetos se representen sobre el mismo, un ejemplo es la investigación en (Zaenker, Verdoja, & Kyrki, 2020), sin embargo, requiere de un procesador que sea capaz de realizar la alta carga computacional.



## Bibliografía

- Kamtex Industries Pte Ltd. (2021). *KAMITEXSOLAR*. Obtenido de <https://solar.kamtexindustries.com/product/firstpower-12v-5ah-maintenance-free-lead-acid-solar-battery-fp1250>
- Raspberry Pi Foundation. (2022). *Raspberry Pi*. Obtenido de <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/>
- The MathWorks, Inc. (2022). *MathWorks*. Obtenido de SLAM (localización y mapeo simultáneos): <https://la.mathworks.com/discovery/slam.html>
- Aguilera, M., Bautista, M., & Iruegas, J. (2007). *Diseño y Control de Robots Móviles*. Nuevo Laredo, Tamaolipas, México.
- Arduino Mega 2560 Características, Especificaciones*. (23 de Agosto de 2018). Obtenido de Proyecto Arduino: <https://proyectoarduino.com/arduino-mega-2560/>
- Bambino, I. (2008). *Una Introducción a los Robots Móviles*.
- Berry, C. A. (2012). *Mobile Robotics for Multidisciplinary Study. Synthesis Lectures on Control and Mechatronics*.
- Bersan, D., Martins, R., Campor, M., & Nascimento, E. R. (2018). *Semantic Map Augmentation for Robot Navigation: A Learning Approach based on Visual and Depth Data*. Brazil: Brazilian Symposium on Robotics.
- Bersan, D., Martins, R., Campor, M., & Nascimento, E. R. (29 de Febrero de 2020). *Extending Maps with Semantic and Contextual Object Information for Robot Navigation: A Learning-Based Framework Using Visual and Depth Cues*. Belo Horizonte, Brazil.
- BLUBBER. (2022). *Mobus*. Obtenido de <https://mobus.es/blog/que-es-una-bateria-lipo/>
- Buschka, P. (2005). *An Investigation of Hybrid Maps for Mobile Robots*.

Capobianco, R., Serafin, J., Dichtl, J., Grisetti, G., Iocchi, L., & Nardi, D. (2015). *A Proposal for Semantic Map Representation and Evaluation*.

CQRobot. (2019). Obtenido de

[https://www.cqrobot.com/index.php?route=product/product&product\\_id=1138](https://www.cqrobot.com/index.php?route=product/product&product_id=1138)

Douillard, B., Fox, D., Ramos, F., & Durrant-Whyte, H. (2011). *Classification and Semantic Mapping of Urban Environments*. Sidney: Australian Centre for Field Robotics.

Electrical4U. (28 de Octubre de 2020). *Electrical4U*. Obtenido de <https://www.electrical4u.com/what-is-servo-motor/>

FLYTREX. (28 de Noviembre de 2020). Obtenido de <https://flytrex.com/>

Galindo, Saffiotti, Coradeschi, & Buschka. (2005). *Multi-Hierarchical Semantic Maps for Mobile Robots*. Málaga.

Gallart, X. (2013). *Semantic Mapping in ROS*. Suecia.

González, A. G. (02 de 12 de 2016). *Panama Hitek*. Obtenido de <http://panamahitek.com/que-es-y-como-funciona-un-servomotor/>

Grisetti, G., Kümmerle, R., Stachniss, C., & Burgard, W. (2010). *A Tutorial Graph-Based SLAM*. IEEE.

Grisetti, G., Stachniss, C., & Burgard, W. (s.f.). *OpenSLAM*. Obtenido de <https://openslam-org.github.io/gmapping.html>

Guzmán, D. T. (Septiembre de 2016). *Algoritmos de Mapeo para Robótica Móvil y Entorno de Simulación*. España. Recuperado el 22 de Julio de 2022

*Ingeniería Mecafenix*. (20 de Abril de 2017). Obtenido de <https://www.ingmecafenix.com/electricidad-industrial/motor-paso-a-paso/>

Interstate Batteries. (2021). *Interstate Batteries*. Obtenido de Baterías de ácido-plomo sellas: <https://es.interstatebatteries.com/sealed-lead-acid>

Jean-Baptiste, L. (2021). *Ontologies with python programming owl 2.0 ontologies with python and owlready2*.

Karunanayake, N., & Gnanasekera, M. (Abril de 2016). An Enhanced Computer Vision Based Hand Movement Capturing System with Stereo Vision. *An Enhanced Computer Vision Based Hand Movement Capturing System with Stereo Vision*. Sri Lanka Institute of Information Technology.

Katsumata, Y., Taniguchi, A., Hagiwara, Y., & Taniguchi, T. (28 de Mayo de 2019). *Semantic Mapping Based on Spatial Concepts for Grounding Words Related to Places in Daily Environments*. Japón.

KLANCAR, G., ZDEŠAR, A., BLAŽIC, S., & ŠKRJANC, I. (2016). *Wheeled Mobile Robotics*. Slovenia: Butterworth-Heinemann.

Kostavelis, I., & Gasteratos, A. (2014). Robotics and Autonomous Systems. *Semantic mapping for mobile robotics tasks: A survey*. Elsevier.

Kostavelis, I., Charalampous, K., Gasteratos, A., & Tsotsos, J. K. (23 de Enero de 2015). Engineering Applications of Artificial Intelligence. *Robot Navigation via spatial and temporal coherent maps*. Elsevier.

Kumar, M. (7 de Julio de 2015). Autonomous Robot Based on Robot Operating System (ROS) for Mapping and Navigation. India.

Künzel, G. (14 de 9 de 2005). *THE MECHATRONICS DESIGN PROCESS*.

Landsiedel, C. W. (23 de Abril de 2018). Semantic Mapping for Autonomus Robots in Urban Environmentes. Alemania.

*Mercado Libre Ecuador*. (2021). Obtenido de [https://articulo.mercadolibre.com.ec/MEC-431362118-mgsystem-monster-moto-shield-puente-h-30a-vnh2sp30-arduino-\\_JM#redirectedFromParent](https://articulo.mercadolibre.com.ec/MEC-431362118-mgsystem-monster-moto-shield-puente-h-30a-vnh2sp30-arduino-_JM#redirectedFromParent)

Mur-Artal, R., Tardós, J. D., Montiel, J. M., & Gálvez-López, D. (Octubre de 2016). *Robotis Universidad de Zaragoza*. Obtenido de <https://webdiis.unizar.es/~raulmur/orbslam/>

- N, M. K., T, S., & B, S. (7 de Julio de 2015). Autonomous Robot Based on Robot Operating System (ROS) for Mapping and Navigation. Mysuru, India. Recuperado el 25 de Julio de 2021
- Nag, A., & Deshmukh, S. (4 de Diciembre de 2015). Real Time Tracking System using 3D Vision. *Real Time Tracking System using 3D Vision*.
- No Magic, Inc. (2022). *The role of ontologies and reasoners*. Obtenido de <https://docs.nomagic.com/display/CCMP190/The+role+of+ontologies+and+reasoners>
- NovatronicEc. (2020). *Novatronic Ingeniería automatización y robotica*. Obtenido de <https://novatronicec.com/index.php/product/acelerometro-giroscopio-mpu6050/>
- Nüchter Andreas, H. J. (2008). Robotics and Autonomous Systems. *Towards semantic maps for mobile robots*. ELSEVIER.
- NVIDIA Corporation. (2022). *NVIDIA DEVELOPER*. Obtenido de Jetson Nano Developer Kit: <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>
- Ogata, K. (2010). *Ingeniería de control moderna*. Madrid: Pearson.
- Or, B. (31 de Julio de 2021). *Towards Data Science*. Obtenido de <https://towardsdatascience.com/what-is-imu-9565e55b44c>
- Ortigoza R, A. M. (Diciembre de 2012). *Wheeled Mobile Robots: A Review*.
- Permana Saputra, R. (26 de Octubre de 2015). *Implementation 2D EKF SLAM for Wheeled*. Australia.
- PRONOBIS, A. (2011). *Semantic Mapping with Mobile Robots*. Suecia.
- Questex. (2022). *fierce Electronics*. Obtenido de <https://www.fierceelectronics.com/>
- ROBOTIS. (2022). *Robotis e-Manual*. Obtenido de <https://emanual.robotis.com/docs/en/platform/turtlebot3/overview/>
- ROBOTIS e-Manual*. (2020). Obtenido de <https://emanual.robotis.com/docs/en/platform/turtlebot3/overview/>

- Rubio, F., Valero, F., & Llopies-Albert, C. (Abril de 2019). *A review of mobile robots: Concepts, methods, theoretical framework, and applications*. International Journal of Advanced Robotic Systems.
- Ruiz Sarmiento, J. R. (2016). Probabilistic techniques in semantic mapping for mobile robotics. UMA Editorial.
- Ruiz-Sarmiento, J. R., Galindo, C., & Gonzalez-Jimenez, J. (2015). Exploiting semantic knowledge for robot object recognition. Málaga.
- Ruiz-Sarmiento, J. R., Galindo, C., & Gonzalez-Jimenez, J. (2017). Robot@Home, a robotic dataset for semantic mapping of home environments. Málaga.
- Santos, J. M., Portugal, D., & Rocha, R. P. (Octubre de 2013). An Evaluation of 2D SLAM Techniques Available. *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics*.
- Saputra, R. P. (26 de Octubre de 2015). Implementation 2D EKF for Wheeled Mobile Robot. Australia. Recuperado el 15 de Mayo de 2022
- SERVOCITY. (2021). Obtenido de <https://www.servocity.com/5202-series-yellow-jacket-planetary-gear-motor-188-1-ratio-30-rpm-3-3-5v-encoder/>
- Shanghai Slamtec Co. (s.f.). Obtenido de 2022: <https://www.slamtec.com/en/Lidar/A1>
- Siciliano, B., & Khatib, O. (2016). *Handbook of Robotics*. Springer.
- Siciliano, B., & Khatib, O. (2016). Springer Handbook of Robotics. En B. Siciliano, & O. Khatib, *Springer Handbook of Robotics* (págs. 1153-1176). Stanford: Springer.
- Siegwart, R., & Nourbakhsh, I. (2011). Introducción a los Robots Autónomos Móviles. *Introduction to Autonomus Mobile Robots*. Londres, Inglaterra.
- Sigma Electronic. (2021). Obtenido de <https://www.sigmaelectronica.net/producto/esp-32/>
- Silva, C. (2016). Modelamiento Semántico del entorno de un robot utilizando informacion RGB-D. Santiago de Chile, Chile.
- Taderi, H., & Xia, Z. C. (19 de Octubre de 2020). SLAM; definition and evolution. China: Elseiver.

Techopedia Inc. (2022). Obtenido de <https://www.techopedia.com/definition/10695/kinect>

The Board of Trustees of the Leland Stanford Junior University. (s.f.). *Protégé*. Obtenido de <https://protege.stanford.edu/>

Thrun, S. (Febrero de 2002). *Robotic Mapping: A Survey*. Pittsburgh.

*Udacity*. (2021). Obtenido de Artificial Intelligence for Robotics - Udacity:

<https://classroom.udacity.com/courses/cs373/lessons/48646841/concepts/486468390923>

Ulrich, K. T., & Eppinger, S. D. (2013). *Diseño y desarrollo de productos*. México: Mc Graw Hill.

Ulrich, K. T., & Eppinger, S. D. (2013). *Diseño y desarrollo de productos*. México D.F., México: McGraw Hill. Recuperado el 17 de Diciembre de 2020

Ultimaker. (16 de Mayo de 2017). *Ficha de datos técnicos PLA*. Obtenido de <https://docs.rs-online.com/7ade/0900766b81698006.pdf>

Velásquez Hernández, C. A. (06 de Diciembre de 2017). *Desarrollo de algoritmo de mezclado de mapas por ocupación de celdas aplicado a la navegación y exploración colaborativa de entornos internos desconocidos*. Obtenido de <https://repositorio.unal.edu.co/handle/unal/62177?fbclid=IwAR3v4vRNCPyzjjSnxqsIPBixltvAVLG7p5HclqDgIiy25jmQuvNiHvfm774>

*Vorwerk International para España*. (14 de Mayo de 2018). Obtenido de

<https://support.vorwerk.com/hc/es/articles/360000896605--Con-qu%C3%A9-velocidad-se-desplaza-el-robot-aspirador->

W3C. (10 de Febrero de 2004). *Lenguaje de Ontologías Web (OWL)*. Obtenido de

<https://www.w3.org/2007/09/OWL-Overview-es.html>

Wolf, D. F., & Sukhatme, G. S. (2 de Abril de 2008). *Semantic Mapping Using Mobile Robots*. IEEE Transactions.

Xia, C. (24 de Noviembre de 2015). Intelligent Mobile Robot Learning in Autonomous. Ecole Centrale de Lille.

Zaenker, T., Verdoja, F., & Kyrki, V. (3 de Septiembre de 2020). *Hypermap Mapping Framework and its Application to Autonomous Semantic Exploration*.

Zhao, Z.-Q., Zheng, P., Xu, S.-T., & Wu, X. (Enero de 2019). IEEE Transactions on Neural Networks and Learning Systems. *Object Detection with Deep Learning: A Review*.

## Apéndices