



*"Responsabilidad con pensamiento positivo"*

**UNIVERSIDAD TECNOLÓGICA ISRAEL**

**TRABAJO DE TITULACIÓN EN OPCIÓN AL GRADO DE:  
INGENIERO EN ELECTRÓNICA DIGITAL Y  
TELECOMUNICACIONES**

**TEMA:**

**DESARROLLO DE ALGORITMO PARA VISIÓN ARTIFICIAL APLICADA A  
ROBOTS COOPERATIVOS EN PROCESO DE CLASIFICACIÓN Y  
DISTRIBUCIÓN DE SUSTANCIAS QUÍMICAS.**

**AUTOR:**

**LUIS EDUARDO ESCOBAR CARRIÓN**

**TUTOR:**

**MG. FRANCISCO XAVIER JURADO**

**QUITO, ECUADOR**

**2020**

# UNIVERSIDAD TECNOLÓGICA ISRAEL

## APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de titulación certifico:

Que el trabajo de titulación “**DESARROLLO DE ALGORITMO PARA VISIÓN ARTIFICIAL APLICADA A ROBOTS COOPERATIVOS EN PROCESO DE CLASIFICACIÓN Y DISTRIBUCIÓN DE SUSTANCIAS QUÍMICAS.**”, presentado por el Sr. Luis Eduardo Escobar Carrión, estudiante de la carrera de Electrónica Digital y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito, 14 de Febrero del 2020

TUTOR

.....

Mg. Francisco Xavier Jurado

# UNIVERSIDAD TECNOLÓGICA ISRAEL

## APROBACIÓN DEL TUTOR

En mi calidad de tutor del componente práctico certifico:

Que el trabajo de titulación “**DESARROLLO DE ALGORITMO PARA VISIÓN ARTIFICIAL APLICADA A ROBOTS COOPERATIVOS EN PROCESO DE CLASIFICACIÓN Y DISTRIBUCIÓN DE SUSTANCIAS QUÍMICAS.**”, presentado por el Sr. Luis Eduardo Escobar Carrión, estudiante de la carrera de Electrónica Digital y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito, 14 de Febrero del 2020

TUTOR

.....

Mg. Francisco Xavier Jurado

## **AGRADECIMIENTO**

En estos cortos párrafos quiero agradecer en primer lugar a Dios y a la Virgencita María, por todas sus bendiciones que me han ayudado en este camino. Agradecer de todo corazón a todas esas personas que siempre fueron mi apoyo incondicional, para alcanzar esta gran meta y que de alguna manera estuvieron conmigo en los momentos más difíciles, alegres y tristes de mí vida.

A mis amados padres por todo su apoyo, cariño y comprensión más que nadie ellos han sido fieles testigos de todos los momentos buenos y malos de mi vida. A ustedes quienes con su humildad y valores, han hecho de mí un verdadero ser humano que día a día, lucha por superarse para ser el orgullo de sus esfuerzos. Estas pequeñas frases no son suficientes para agradecerles las innumerables ocasiones en que me brindaron su apoyo y consejo, en cada decisión que he tomado a lo largo de mi vida, unas buenas, otras malas y claro otras completamente locas. Gracias por enseñarme que un buen ser humano forja su futuro, con esfuerzo, responsabilidad y perseverancia, demostrando cada uno de los valores que desde muy pequeño me enseñaron a vivir.

A mis dos hermanos Andrea, Jhon agradecerles por llenarme de tantas alegrías, ustedes que me llenaron de grandes experiencias que me ayudaron en el camino a mejorar mi forma de pensar y vivir.

Luis Eduardo Escobar Carrión

## **DEDICATORIA**

El presente trabajo lo dedico principalmente a Dios y a la Virgen María, por haberme dado la vida y permitirme el haber llegado hasta este momento tan importante de mi formación profesional y personal.

A mi madre, por ser el pilar más importante de mi vida y por demostrarme siempre su cariño y apoyo incondicional sin importar nuestras diferencias de opiniones.

A mi padre, por enseñarme que con esfuerzo y trabajo todo se puede lograr, sin olvidar la humildad y respeto que se debe de hacer las cosas.

Luis Eduardo Escobar Carrión

## TABLA DE CONTENIDO

RESUMEN (ABSTRACT).....	11
INTRODUCCIÓN.....	13
Antecedentes .....	13
Planteamiento.....	17
Justificación e importancia .....	18
Objetivo General.....	20
Objetivos Específicos .....	20
Hipótesis.....	21
Alcance.....	22
Descripción de los capítulos .....	24
1. Introducción .....	24
2. Capítulo I. Fundamentación Teórica.....	24
3. Capítulo II. Marco Metodológico .....	24
4. Capítulo III. Propuesta.....	24
5. Capítulo IV. Implementación .....	24
6. Conclusiones y Recomendaciones.....	25
CAPÍTULO I.....	26
1. FUNDAMENTACIÓN TEÓRICA.....	26
1.1. Introducción .....	26
1.2. Robótica Cooperativa .....	27
1.2.1 Tipos de Control de los Sistemas de Robótica Cooperativa.....	29
1.2.1.1 Control Centralizado .....	29
1.2.1.2 Control Descentralizado .....	29
1.3. Visión Artificial .....	30
1.3.1 Cámara.....	31
1.3.1.1 Proceso de formación de imágenes .....	32
1.3.2 Modelo de color.....	33
1.3.2.1 RGB .....	33
1.3.2.2 HSI.....	34
1.3.3 Iluminación .....	36
1.3.3.1 Iluminación de fondo.....	36
1.3.3.2 Iluminación direccional .....	36
1.3.3.3 Iluminación difusa .....	37

1.3.4	Procesamiento de imágenes.....	37
1.3.4.1	Acondicionamiento.....	39
1.3.4.2	Adquisición de la imagen .....	39
1.3.4.3	Preprocesamiento.....	39
1.3.4.4	Segmentación .....	40
1.3.4.5	Extracción de características .....	41
1.3.4.6	Reconocimiento e interpretación .....	41
1.4.	Planificación de Trayectoria .....	41
CAPÍTULO II.....		43
2.	MARCO METODOLÓGICO .....	43
2.1.	Diseño de la Investigación .....	43
2.2.	Enfoque de la Investigación.....	43
2.3.	Instrumentos de investigación .....	43
2.4.	Etapas del proyecto .....	44
2.5.	Población .....	45
2.6.	Técnicas de recolección de datos.....	45
2.7.	Técnicas de procesamiento de datos .....	45
2.8.	Herramientas para el procesamiento de datos.....	45
CAPÍTULO III .....		46
3.	PROPUESTA .....	46
3.1.	Introducción .....	46
3.2.	Hardware del Sistema .....	47
3.2.1.	Robots móviles.....	48
3.2.2.	Motores DC .....	49
3.2.3.	Objetos de color .....	50
3.2.4.	Cámara IP.....	51
3.2.5.	Driver para motores DC .....	52
3.2.6.	Brazo electromecánico .....	54
3.2.7.	Servomotor .....	54
3.2.8.	Tarjeta de desarrollo.....	55
3.2.8.1.	Raspberry Pi .....	56
3.2.8.2.	NodeMCU .....	58
3.2.8.3.	Arduino Mega R3 con ESP8266 .....	59
3.2.9.	Batería LiPo.....	61
3.3.	Software .....	62
3.3.1.	Etapas del software del sistema.....	63

3.3.1.1.	Algoritmo RRT.....	67
3.3.1.2.	Algoritmo Pure Pursuit.....	68
3.3.2.	Python 3.....	70
3.4.	Análisis de costo .....	71
3.5.	Análisis de tiempo .....	74
3.6.	Ventajas del proyecto.....	77
CAPÍTULO IV .....		79
4.	RESULTADOS .....	79
4.1.	Introducción .....	79
4.2.	Implementación de hardware.....	79
4.3.	Desarrollo del sistema de adquisición de datos .....	86
4.3.1.	Configuración de la estación central de control .....	86
4.3.2.	Configuración de la tarjeta nodeMCU .....	88
4.3.3.	Configuración de la cámara IP .....	89
4.4.	Desarrollo del sistema de procesamiento.....	91
4.4.1.	Control centralizado .....	92
4.4.2.	Tareas de cooperación y coordinación entre los robots móviles.....	93
4.5.	Desarrollo de algoritmos del sistema.....	93
4.5.1.	Algoritmo de inicialización y conexión del sistema .....	94
4.5.2.	Algoritmo de localización de objetos de color y marcas .....	95
4.5.3.	Algoritmo de generación de trayectorias .....	97
4.5.4.	Algoritmo de seguimiento de trayectoria .....	99
4.6.	Implementación de algoritmos.....	100
4.6.1.	Algoritmo de detección y localización de objetos de color y marcas .....	101
4.6.2.	Algoritmo de generación de trayectorias .....	104
4.6.3.	Algoritmo de seguimiento de trayectoria .....	105
4.7.	Pruebas de funcionamiento y análisis de resultados .....	106
4.7.1.	Reconocimiento de objetos de color .....	106
4.7.2.	Generación y seguimiento de trayectorias .....	112
4.7.2.1.	Primera configuración de distribución .....	112
4.7.2.2.	Segunda configuración de distribución .....	114
4.7.2.3.	Tercera configuración de distribución.....	117
5.	CONCLUSIONES .....	120
6.	RECOMENDACIONES .....	122
7.	REFERENCIAS BIBLIOGRÁFICAS .....	123



## ÍNDICE DE TABLAS

<b>Tabla 1.</b> Código de colores para clasificación de material químico basado en (Carrascal Quintero, 2010). .....	19
<b>Tabla 3.1</b> Análisis comparativo entre tarjetas de desarrollo.....	60
<b>Tabla 3.2</b> Análisis de consumo de energía para la selección de la batería. ....	61
<b>Tabla 3.3</b> Dispositivos electrónicos.....	71
<b>Tabla 3.4</b> Herramientas eléctricas y materiales varios. ....	72
<b>Tabla 3.5</b> Mano de obra.....	73
<b>Tabla 3.6</b> Presupuesto para el desarrollo del proyecto. ....	73
<b>Tabla 4.1</b> Hardware implementado en cada robot móvil del sistema.....	79
<b>Tabla 4.2</b> Distribución de las marcas aruco de identificación.....	82
<b>Tabla 4.3</b> IP estáticas para los robots móviles.....	89
<b>Tabla 4.4</b> Rango de valores HSV para los objetos de color. ....	102
<b>Tabla 4.5</b> Resultados de las pruebas con iluminación adecuada. ....	108
<b>Tabla 4.6</b> Resultados de las pruebas con iluminación adecuada. ....	109
<b>Tabla 4.7</b> Resultados obtenidos para 5 pruebas realizadas en la primera configuración. ....	114
<b>Tabla 4.8</b> Resultados obtenidos para 5 pruebas realizadas en la segunda configuración.....	117
<b>Tabla 4.9</b> Resultados obtenidos para 5 pruebas realizadas en la tercera configuración. ....	119

## ÍNDICE DE FIGURAS

<b>Figura 1.1.</b>	Principales características de los robots cooperativos.	28
<b>Figura 1.2.</b>	Robots en trabajo cooperativo para levantar columnas.	28
<b>Figura 1.3.</b>	Formación triángulo de robots terrestres	29
<b>Figura 1.4.</b>	Sistema de fútbol robótico vista de la cámara cenital.	30
<b>Figura 1.5.</b>	Conexión en red de una cámara IP	31
<b>Figura 1.6.</b>	Cámara IP HikVision.	32
<b>Figura 1.7.</b>	Modelo Pin-Hole	33
<b>Figura 1.8.</b>	Modelo de color RGB.	33
<b>Figura 1.9.</b>	Modelo de color (a) HSV, (b) HSL.	34
<b>Figura 1.10.</b>	Saturación en el modelo de color (a) HSL, (b) HSV.	35
<b>Figura 1.11.</b>	Iluminación de fondo.	36
<b>Figura 1.12.</b>	Iluminación direccional.	37
<b>Figura 1.13.</b>	Iluminación difusa.	37
<b>Figura 1.14.</b>	Imagen Binaria.	38
<b>Figura 1.15.</b>	Imagen RGB.	38
<b>Figura 1.16.</b>	Procesamiento digital de imágenes.	39
<b>Figura 1.17.</b>	(a) Imagen en escala de grises, (b) histograma, (c) imagen binaria.	41
<b>Figura 3.1.</b>	Diagrama de las etapas del proyecto.	46
<b>Figura 3.2.</b>	Diagrama de bloques del hardware del sistema.	47
<b>Figura 3.3.</b>	Estructura de los robots móviles.	48
<b>Figura 3.4.</b>	Dimensiones del chasis de los robots móviles.	49
<b>Figura 3.5.</b>	Motorreductor DC.	49
<b>Figura 3.6.</b>	Objetos de color.	50
<b>Figura 3.7.</b>	Cámara IP marca Hikvision.	51
<b>Figura 3.8.</b>	Driver L298N.	53
<b>Figura 3.9.</b>	Driver TB6612FNG.	53
<b>Figura 3.10.</b>	Brazo electromecánico.	54
<b>Figura 3.11.</b>	Servomotor MG90.	55
<b>Figura 3.12.</b>	Tarjeta Raspberry Pi.	56
<b>Figura 3.13.</b>	Puertos GPIO de la tarjeta Raspberry Pi.	57
<b>Figura 3.14.</b>	Esquema pinout de la tarjeta NodeMCU.	58
<b>Figura 3.15.</b>	Esquema pinout de la tarjeta arduino mega R3 con WiFi.	59
<b>Figura 3.16.</b>	Batería LiPo 7.4V y 2200mAh	61
<b>Figura 3.17.</b>	Diagrama de bloques del software de la estación central.	62
<b>Figura 3.18.</b>	Diagrama de flujo de las etapas de software del sistema.	63
<b>Figura 3.19.</b>	Diagrama de flujo de la conexión inalámbrica del sistema.	64
<b>Figura 3.20.</b>	Diagrama de flujo de la estimación de posición de objetos y marcas.	65
<b>Figura 3.21.</b>	Diagrama de flujo del algoritmo de generación y seguimiento de trayectoria.	66
<b>Figura 3.22.</b>	Diagrama de flujo del algoritmo de control para los robots.	67
<b>Figura 3.23.</b>	Diagrama de flujo del procedimiento del algoritmo RRT.	68
<b>Figura 3.24.</b>	Diagrama de flujo del procedimiento del algoritmo Pure Pursuit.	69
<b>Figura 3.25.</b>	Software Python.	70
<b>Figura 3.26.</b>	Software IDE PyCharm.	70
<b>Figura 3.27.</b>	Planificación de tareas del proyecto.	74
<b>Figura 3.28.</b>	Diagrama de grant etapa 1.	75
<b>Figura 3.29.</b>	Diagrama de grant etapa 2.	75
<b>Figura 3.30.</b>	Diagrama de grant etapa 3.	76

<b>Figura 3.31.</b> Diagrama de grant etapa 4. ....	76
<b>Figura 3.32.</b> Diagrama de grant etapa 5 y 6. ....	77
<b>Figura 4.1.</b> Diagrama de conexión de los robots móviles. ....	80
<b>Figura 4.2.</b> Robots Móviles vista superior. ....	81
<b>Figura 4.3.</b> Robots Móviles vista lateral. ....	81
<b>Figura 4.4.</b> Marcas aruco de identificación. ....	82
<b>Figura 4.5.</b> Estructura para la cámara cenital. ....	83
<b>Figura 4.6.</b> Colocación de la cámara IP a una altura de 1.60 metros. ....	84
<b>Figura 4.7.</b> Colocación de la cámara IP a una altura de 1.5 metros. ....	84
<b>Figura 4.8.</b> Colocación de la cámara IP a una altura de 1.40 metros. ....	85
<b>Figura 4.9.</b> Entorno de pruebas. ....	85
<b>Figura 4.10.</b> Diagrama Cliente/Servidor del sistema. ....	86
<b>Figura 4.11.</b> Diagrama de flujo de la comunicación Server del sistema. ....	87
<b>Figura 4.12.</b> Diagrama de flujo del proceso de comunicación de los clientes. ....	88
<b>Figura 4.13.</b> Visualización de la cámara IP en el software SADP. ....	89
<b>Figura 4.14.</b> Configuración de la IP de la cámara en el software SADP. ....	90
<b>Figura 4.15.</b> Visualización de la cámara IP en el navegador. ....	90
<b>Figura 4.16.</b> Arquitectura del control del sistema cooperativo. ....	92
<b>Figura 4.17.</b> Diagrama de flujo del software del sistema de desarrollar. ....	94
<b>Figura 4.18.</b> Diagrama de flujo de la rutina de inicialización del sistema. ....	95
<b>Figura 4.19.</b> Diagrama de flujo de la rutina de actualización de posiciones. ....	96
<b>Figura 4.20.</b> Diagrama de flujo de la rutina de generación de trayectoria. ....	97
<b>Figura 4.21.</b> Diagrama de flujo de la evaluación de distancia entre robots y objetos. ....	98
<b>Figura 4.22.</b> Diagrama de flujo de la generación de trayectoria con RRT. ....	98
<b>Figura 4.23.</b> Diagrama de flujo del algoritmo de seguimiento de trayectoria. ....	99
<b>Figura 4.24.</b> Diagrama de flujo de la actualización de los estados. ....	100
<b>Figura 4.25.</b> Obtención del rango de valores HSV para el color verde. ....	102
<b>Figura 4.26.</b> Detección de marcas Aruco vista aérea. ....	103
<b>Figura 4.27.</b> Generación de trayectoria con el algoritmo RRT para puntos fijos. ....	104
<b>Figura 4.28.</b> Generación de trayectoria mediante el algoritmo RRT en el escenario. ....	105
<b>Figura 4.29.</b> Seguimiento de trayectoria aplicando el algoritmo pure pursuit. ....	105
<b>Figura 4.30.</b> Detección de objetos con iluminación artificial adecuada. ....	106
<b>Figura 4.31.</b> Detección de objetos con iluminación artificial inadecuada. ....	107
<b>Figura 4.32.</b> Detección de objetos con iluminación natural adecuada. ....	107
<b>Figura 4.33.</b> Detección de objetos con iluminación natural inadecuada. ....	107
<b>Figura 4.34.</b> Eficacia de las distintas configuraciones de los objetos en el escenario. ....	110
<b>Figura 4.35.</b> Eficacia según el color del objeto. ....	110
<b>Figura 4.36.</b> Eficacia según la iluminación. ....	111
<b>Figura 4.37.</b> Eficacia para iluminación adecuada. ....	112
<b>Figura 4.38.</b> Primera distribución de objetos en el escenario. ....	113
<b>Figura 4.39.</b> Secuencia de imágenes del seguimiento de trayectoria de la primera configuración. ....	114
<b>Figura 4.40.</b> Generación de trayectoria para el trayecto B. ....	115
<b>Figura 4.41.</b> Segunda distribución de objetos en el escenario. ....	115
<b>Figura 4.42.</b> Secuencia de imágenes del seguimiento de la trayectoria de la segunda configuración. ....	116
<b>Figura 4.43.</b> Tercera configuración de distribución de objetos en el escenario. ....	118
<b>Figura 4.44.</b> Secuencia de imágenes del seguimiento de la trayectoria de la tercera configuración. ....	119

## RESUMEN (ABSTRACT)

El uso de sistemas de robots cooperativos en ambientes industriales permite una respuesta rápida ante la detección de riesgos en el transporte, clasificación y almacenamiento de sustancias, y así evitar impactos sobre la seguridad física del personal y equipos, impactos económicos por paros en las líneas de producción o impactos sociales que afectan a la imagen de la empresa en el mercado. El método SAF-T-DATA, es usado como medida de seguridad en el transporte, clasificación y almacenamiento de productos o sustancias químicas. Este sistema organiza los productos químicos de acuerdo con sus propiedades que pueden ser: inflamables, reactivas, corrosivas, tóxicas y no peligrosas, asignándoles un color para poder almacenarlos. Es importante almacenar de manera adecuada estas sustancias puesto que pueden generar reacciones peligrosas como la emisión de gases tóxicos, corrosivos, inflamables, reacciones explosivas, exotérmicas, entre otros; comprometiendo la seguridad del personal y el establecimiento.

**Palabras Claves:** Cooperativos, exotérmicas, corrosivos, sustancias, detección.

The use of cooperative robot systems in industrial environments allows a rapid response to the detection of risks in the transport, classification and storage of substances, and thus avoid impacts on the physical safety of personnel and equipment, economic impacts from stoppages in the lines of production or social impacts that affect the image of the company in the market. The SAF-T-DATA method is used as a safety measure in the transport, classification and storage of chemical products or substances. This system organizes chemical products according to their properties that can be: flammable, reactive, corrosive, toxic and non-dangerous, assigning them a color to store them. It is important to properly store these substances since they can generate dangerous reactions

such as the emission of toxic, corrosive, flammable gases, explosive, exothermic reactions, among others; compromising the safety of the staff and the establishment.

**Keywords:** Cooperative, exothermic, corrosive, substances, detection.

# INTRODUCCIÓN

## Antecedentes

Debido al constante desarrollo de la robótica actualmente se cuenta con robots autónomos utilizados en aplicaciones de: investigación, militar e industria; estos robots poseen la capacidad de percepción del entorno y navegación autónoma, siendo aptos para seguir una trayectoria y llegar a una posición específica dentro de un entorno. Los robots autónomos son diseñados para un propósito determinado y están imposibilitados a la realización de multitareas (Inglett & Rodríguez-Seda, 2017).

Con la constante evolución de la sociedad tecnológica se desarrollan aplicaciones cada vez más complejas que necesitan ser resueltas en menor tiempo, es ahí cuando aparece la robótica cooperativa. Un sistema cooperativo está constituido por robots de diseño flexible y adaptable a diferentes entornos, que trabajan juntos para cumplir un objetivo en común; es decir, la robótica cooperativa ofrece robots que realizan tareas más allá del propósito para el cual fueron desarrollados individualmente (Illanes Tibanlombo, 2018; Lara, Urrea, & Barbosa, 2015).

La robótica cooperativa está presente en aplicaciones de exploración (Illanes Tibanlombo, 2018), vigilancia (Murray, n.d.), transporte de objetos (Inglett & Rodríguez-Seda, 2017; Murray, n.d.). En la industria Química el transporte de material es de suma importancia debido a los riesgos presentes en el ambiente donde se manipulan estas sustancias químicas.

En (Carrascal Quintero, 2010; Gonzalo Rafael Rojo Aguirre, 2014) se analizan diversas normas internacionales para el transporte, clasificación y almacenamiento de material químico. El sistema SAF-T-DATA de J.T. Baker sobresale debido a su método

codificado en colores para la organización de sustancias químicas (Carrascal Quintero, 2010; Colmenares, 2014).

El método SAF-T-DATA es usado tanto en Universidades como en la Industria. En Ecuador la Universidad de las Fuerzas Armadas - ESPE aplica el método en laboratorios de biotecnología (ESPE, 2014), en Colombia la universidad de Antioquia desarrollo un manual de técnicas para el manejo seguro de químicos en el laboratorio (Osorio Giraldo, 2009), la Comunidad Andina (Bolivia, Colombia, Ecuador y Perú) aplica el método de SAF-T-DATA en el programa anti-drogas ilícitas (CENAPRED, Dirección de análisis y gestión de riesgos, & Subdirección de riesgos sanitarios y toxicología, 2016) con el fin de fortalecer el conocimiento de técnicos y operadores sobre las sustancias químicas, los sistemas de clasificación y el uso ilícito de las mismas mejorando las acciones de control y fiscalización, también la empresa global Avantor (Avantor Performance Materials, 2014) proveedor de material químico a instituciones académicas e industrias químicas, ofrece en sus productos información de seguridad a través de etiquetación utilizando el método SAF-T-DATA.

Para que los robots móviles transporten objetos desde una posición inicial hasta otra final en un entorno, es indispensable que estos robots tengan capacidad de navegar autónomamente, evitando colisiones con obstáculos y otros robots como se ve en (Diego & Fernando, 2012; Pérez Checa, 2016).

Para navegar autónomamente los robots necesitan percibir el entorno en el que se encuentran, y obtener información relevante. Una de las técnicas más empleadas para este tipo de navegación es la utilización de cámaras junto con algoritmos de visión artificial, debido a que un solo sensor (cámara) provee de suficiente información al robot para la toma de decisiones, incrementando su autonomía (Efrain Ernesto, Arturo, & Juan, 2015; Etxeberria, 2010; Pérez Checa, 2016).

Un sistema de visión artificial capta información real del entorno mediante la captura de imágenes, analizándolas para discriminar las características del entorno y poder tomar las decisiones pertinentes a su aplicación (Efrain Ernesto et al., 2015; Pérez Checa, 2016). A través de técnicas de visión artificial se detectan objetos tanto por la forma como por el color, a este proceso se le llama “Segmentación” (Conlago Guatemal & Yunda Sangoluisa, 2016; Efrain Ernesto et al., 2015).

La realización de este proyecto de grado se respalda en trabajos como:

Normativas y métodos para el manejo seguro de material químico que son analizados: en (Carrascal Quintero, 2010) quien presenta una guía para la clasificación y almacenamiento de sustancias químicas según el código de colores SAF-T-DATA, según su pictograma, entre otros; en (Colmenares, 2014) se plantean las bases para la manipulación de sustancias químicas, alternativas y tratamientos para minimizar los desechos peligrosos, y en (Gonzalo Rafael Rojo Aguirre, 2014) se analiza las normativas de seguridad básicas a implementarse en una planta química, así como el control y prevención de riesgos en las actividades realizadas por los operarios.

Trasporte cooperativo de objetos que es desarrollado en (Inglett & Rodríguez-Seda, 2017) donde utilizan dos robots diferenciales de configuración simple para transportar un objeto demasiado pesado para un solo robot, en (Apolo Peñaloza & Velasco Barrera, 2013) trabajo realizado en el país se utiliza tres robots para transportar un balón de un punto inicial a un final a través de pases entre los elementos robots del sistema, emulando el principio básico del fútbol.

Técnicas de detección de objetos basadas en visión artificial como en: (Lin, Wang, Yang, & Lai, 2015) quienes presentan un modelo reconfigurable And-Or para reconocer formas de objetos en imágenes mediante la extracción de bordes, en (Chen et al., 2016)



se analiza la utilización de una cámara monocular para la detección de objetos 3D en conducción autónoma, en (Possegger & Mauthner, 2015) abordan el problema de seguimiento de objetos de interés en movimiento basado en el color con un enfoque favorable en precisión y robustez en comparación con los algoritmos de seguimiento anteriores, en (Conlago Guatemal & Yunda Sangoluisa, 2016) se desarrolla un sistema de reconocimiento y detección automático de señales de tránsito con el objetivo de hacer más seguro conducir en Ecuador, y en (Constante Prócel & Gordón Garcés, 2015) presenta una técnica de visión artificial combinada con redes neuronales artificiales para la detección de características complejas de tres diferentes frutas ecuatorianas y clasificarlas mediante la detección de bordes y color.

Planificación de trayectoria analizado en: (Diego & Fernando, 2012) quienes estudian diversos algoritmos de planificación enfocándose en el algoritmo RRT, tanto en (Pérez Checa, 2016) como en (Pérez Álvarez, 2017) utilizan una cámara cenital para obtener una imagen de todo el escenario de pruebas y mediante procesamiento de imágenes obtener la posición del robot a cada instante en el escenario, así como información para controlar la trayectoria que deben seguir los robots; estos trabajos fueron desarrollados en el país.

Tomando en cuenta estos antecedentes, y con el objetivo de superar las dificultades expuestas y los desafíos de la clasificación de objetos de color, utilizando técnicas de visión artificial y seguimiento de trayectoria, se propone este proyecto titulado “Sistema de transporte y clasificación de objetos de color mediante visión artificial y robótica cooperativa”.

## **Planteamiento**

La industria química a nivel mundial está constituida de numerosas organizaciones públicas y privadas enfocadas en el descubrimiento, desarrollo y fabricación de: medicamentos tanto para humanos como para animales, polímeros utilizados en la elaboración de piezas de automóviles y frenos, plaguicidas que mejoren las cosechas, entre otros.

Los principios activos utilizados en la elaboración de productos químicos son peligrosos para la salud de los trabajadores, por lo que se debe establecer medidas de control adecuadas para proteger a los trabajadores de los productos químicos industriales y los principios activos durante las operaciones de fabricación (Tait, 2015).

El transporte, clasificación y almacenamiento de sustancias químicas es de suma importancia en la industria química, debido a que estas sustancias pueden quemar, explotar, contaminar, disolver o producir reacciones perjudiciales causando daño al personal o equipos, es importante organizar las sustancias químicas de acuerdo a sus propiedades y características ya que no todas las sustancias son compatibles entre sí.

La aplicación incorrecta de normativas para el manejo seguro de sustancias químicas como el método SAF-T-DATA de J.T. Baker, al momento de transportar, clasificar y almacenar sustancias químicas, puede desencadenar accidentes debido a errores humanos.

Utilizar robots cooperativos en espacios químicos industriales puede aportar seguridad tanto al personal como a los equipos, debido a que los robots operan en condiciones hostiles reduciendo la posibilidad de lesiones al personal o daños al equipo. Además, el uso de sistemas cooperativos dentro de las actividades de fabricación química es una

alternativa eficiente para resolver problemas de automatización e incrementar la producción nacional.

Todo lo anterior mencionado lleva a la necesidad de desarrollar algoritmos de visión artificial aplicados a robots cooperativos en el proceso de clasificación y distribución de sustancias químicas, sistema que podría ser implementado en laboratorios farmacéuticos, elaboración de pinturas y barnices, elaboración de plásticos, entre otras. Y así aportar en la seguridad del personal e influir en el incremento de la producción de estos.

### **Justificación e importancia**

Los sistemas de múltiples robots cooperativos solucionan problemas que son física y computacionalmente complejos, donde no basta la intervención de un solo robot de propósito específico. Estos problemas se resuelven subdividiendo una tarea compleja en tareas sencillas que son ejecutadas por robots de diseño y control simple. Un sistema cooperativo presenta mayor tolerancia a fallos por el número de robots que lo componen, es decir, si un robot deja de funcionar otro robot puede sustituirlo y realizar la tarea que le fue asignada.

El uso de sistemas de robots cooperativos en ambientes industriales permite una respuesta rápida ante la detección de riesgos en el transporte, clasificación y almacenamiento de sustancias, y así evitar impactos sobre la seguridad física del personal y equipos, impactos económicos por paros en las líneas de producción o impactos sociales que afectan a la imagen de la empresa en el mercado (Giacomett, 2017).

El método SAF-T-DATA de J.T. Baker desarrollado en 1982, es usado como medida de seguridad en el transporte, clasificación y almacenamiento de productos o sustancias químicas. Este sistema organiza los productos químicos de acuerdo con sus propiedades

que pueden ser: inflamables, reactivas, corrosivas, tóxicas y no peligrosas, asignándoles un color para poder almacenarlos.

Es importante almacenar de manera adecuada estas sustancias puesto que pueden generar reacciones peligrosas como la emisión de gases tóxicos, corrosivos, inflamables, reacciones explosivas, exotérmicas, entre otros; comprometiendo la seguridad del personal y el establecimiento.

Para el presente proyecto de grado se utiliza los colores azul, verde y amarillo que pertenecen a las propiedades tóxicas, no peligrosas y reactivas u oxidantes respectivamente como se muestra en la Tabla 1. Las sustancias Tóxicas definidas con el color azul conllevan efectos nocivos al penetrar el organismo con procesos como ingestión, inhalación o absorción. Las sustancias Oxidantes conllevan la posibilidad de incendios y propagación, finalmente las sustancias no peligrosas no ofrecen un riesgo para la salud pero deben ser almacenadas correctamente (Carrascal Quintero, 2010; Gonzalo Rafael Rojo Aguirre, 2014).

**Tabla 1.**

*Código de colores para clasificación de material químico basado en (Carrascal Quintero, 2010).*

<b>Color</b>	<b>Tipo de químico</b>	<b>Especificación</b>
Azul	Tóxico	Área de almacenamiento de químicos con riesgo para la salud.
Verde	No peligroso	Área de almacenamiento de sustancias químicas que no ofrecen un riesgo importante.
Amarillo	Oxidante	Área de almacenamiento de químicos con riesgo de oxidación y reactividad.

Fuente: Elaborado por el autor

La industria química está en la necesidad de trabajar con químicos con estas propiedades tóxicas y oxidantes donde su aplicación es sumamente importante como, por ejemplo:

Los químicos tóxicos son usados en aislantes térmicos, fabricación de frenos (amianto), marcos de ventana (cloruro de vinilo), fabricación de pintura (plomo), fabricación de plásticos, resinas y poliéster (tolueno), estos pueden causar somnolencia, irritación de piel y mucosa, dificultad respiratoria entre otras.

Los materiales oxidantes son utilizados principalmente para el tratamiento de aguas residuales, fabricación de baterías, desinfectantes y pesticidas, entre otros, pudiendo ocasionar incendios incontrolados cuando entran en contacto con sustancias inflamables, compuestos orgánicos o inorgánicos.

En Ecuador se han realizado varios trabajos basados en técnicas de visión artificial para la detección de objetos por color y/o forma como se analiza en (Conlago Guatemal & Yunda Sangoluisa, 2016; Constante Prócel & Gordón Garcés, 2015), pero no hay registros de trabajos realizados utilizando visión artificial y robótica cooperativa para clasificar objetos de color y trasladarlos de una posición inicial a otra final en un entorno libre de colisiones.

Por estas razones, se pretende realizar un sistema de transporte y clasificación de objetos de color utilizando técnicas de visión artificial y robótica cooperativa como solución para almacenar correctamente sustancias tóxicas, oxidantes y no peligrosas con base en el método SAF-T-DATA de J.T. Baker.

### **Objetivo General**

Desarrollar algoritmos de visión artificial y robótica cooperativa aplicados en el proceso de clasificación y distribución de sustancias químicas.

### **Objetivos Específicos**

- Elaborar algoritmos de visión artificial con OpenCV para detectar objetos de color, diferenciarlos del entorno y obtener su localización en un escenario de pruebas.

- Diseñar algoritmos de trabajo cooperativo en Python para la clasificación de los objetos de color utilizando un sistema de tres robots libres de colisión con base en los parámetros obtenidos por visión artificial.
- Implementar los algoritmos del sistema de clasificación de objetos de color en una estación de trabajo remota que permita procesar la información recibida por la cámara y enviar alertas.
- Ensamblar tres robots móviles con las características técnicas necesarias para realizar la clasificación de objetos de color de forma autónoma en un entorno controlado.
- Analizar el desempeño del sistema de robots cooperativos en tiempo real para evaluar la velocidad y precisión del sistema en la clasificación con diferentes escenarios y condiciones de iluminación.

### **Hipótesis**

En el presente proyecto de grado se plantean las siguientes hipótesis a alcanzar:

- El uso de robótica cooperativa para el transporte de sustancias tóxicas permite evitar la manipulación humana, dándole mayor seguridad al proceso y disminuyendo los accidentes por contaminación de sustancias tóxicas e incendios con víctimas humanas.
- La visión artificial como principal elemento para la detección y localización de sustancias químicas por el color de etiquetado permite prescindir de la interacción humana directa con los elementos tóxicos u oxidantes dándole paso a la robótica como elemento autónomo de clasificación y transporte.
- El sistema, haciendo uso de técnicas de visión artificial y robótica cooperativa, logrará reemplazar la intervención humana en el proceso de transportar y clasificar sustancias químicas de una posición inicial a otra final dentro del escenario de

pruebas, dejándole únicamente el trabajo de supervisión al operador evitando accidentes por errores humanos.

### **Alcance**

Para defender la hipótesis se plantea realizar el desarrollo de algoritmos de robótica cooperativa en un sistema que incluye tres robots móviles utilizando técnicas de visión artificial para la clasificación de objetos de color, localización y seguimiento de trayectoria en un entorno de pruebas, con el fin de emular el proceso de trasladar, clasificar y almacenar a un área segura sustancias químicas utilizadas en la industria.

Al tratarse de detección de objetos de color se centra en tres colores (azul, verde y amarillo) que representan sustancias tóxicas, no peligrosas y oxidantes o reactivas del método SAF-T-DATA para clasificación de material químico.

El sistema de robots cooperativos está compuesto de tres robots terrestres de dos ruedas de configuración diferencial, cada robot constituye un sistema embebido que tiene en general: sensores de navegación, microcontroladores y módulos WiFi para la comunicación entre los robots y un computador de procesamiento que será la unidad de control central.

La selección del modelo de los robots, el microcontrolador y los sensores que serán adquiridos e implementados en el sistema, se realizará de acuerdo a las características técnicas necesarias para el correcto funcionamiento del mismo como corresponde al presente proyecto de investigación.

El entorno de trabajo contiene al sistema de tres robots cooperativos y los objetos de color de manera aleatoria, éste debe ser observado en su totalidad por una cámara cenital con el fin de realizar la detección y realimentación de las posiciones.

La detección de objetos se realiza por medio de visión artificial a través de la cámara cenital, enfocándose en objetos de los colores: azul, verde y amarillo, los cuales se detallan en la Tabla 1 y marcas distintivas sobre los mismos. Para dicha detección se capturan imágenes en tiempo real, se realiza el procesamiento continuo de las mismas en un computador y de acuerdo con el color detectado se envía instrucciones para la secuencia de clasificación de los objetos y el seguimiento de trayectoria de manera remota por los robots.

Los robots deben recoger el objeto de color más cercano a su posición y trasladarlo a una posición determinada. Debido a la naturaleza cooperativa del sistema los obstáculos presentes en el entorno para cada robot serán: los objetos de diversos colores y los demás robots.

La trayectoria a ser alcanzada por los robots será una línea recta, cuando el robot localice un obstáculo debe rodearlo siguiendo un algoritmo RRT (Rapidly Exploring Random Trees) de planificación de trayectoria hasta retornar a la trayectoria lineal, el robot recurre a la información del entorno proporcionada por la cámara cenital. La transferencia y adquisición de datos se realizará mediante comunicación WiFi.

El sistema clasificador de objetos de color será implementado en un entorno de pruebas con condiciones de iluminación artificial, para la realización de pruebas se variará la iluminación, la disposición de los robots y los objetos en el escenario.

Se realizarán pruebas de funcionalidad a los algoritmos para la validación del desempeño del sistema, en estas pruebas se busca medir la precisión en la clasificación, la velocidad de respuesta del sistema ante la presencia de obstáculos, el nivel de precisión referente a la localización y distancia entre robots. Dichos resultados serán analizados y de acuerdo con su relevancia podrían ser presentados en un documento científico.



## **Descripción de los capítulos**

En el presente documento se presentan los capítulos descritos a continuación:

### **1. Introducción**

En este capítulo se presenta la descripción de la problemática relacionada con el desarrollo de algoritmos de visión artificial para clasificación de objetos de color trabajando en conjunto con algoritmos de robótica cooperativa para la distribución de dichos objetos. Se presenta también una descripción general del proyecto, en donde se analiza los trabajos previos realizados a esta investigación.

### **2. Capítulo I. Fundamentación Teórica**

El capítulo I presenta el estudio realizado a investigaciones enmarcadas tanto en la detección de objetos de color mediante técnicas de visión artificial, como investigaciones de robótica móvil cooperativa presentes en aplicaciones que involucre el transporte de objetos.

### **3. Capítulo II. Marco Metodológico**

El capítulo II contiene la explicación de los mecanismos aplicados para el análisis de la problemática de este proyecto. Está basado en la aplicación sistemática y lógica de lo expuesto en la fundamentación teórica.

### **4. Capítulo III. Propuesta**

El capítulo III describe la propuesta del proyecto detallando el desarrollo de algoritmos, el diseño de los robots del sistema, los componentes electrónicos y mecánicos a utilizar y los diferentes colores de los objetos.

### **5. Capítulo IV. Implementación**

El capítulo VI presenta la implementación del proyecto en su totalidad, el hardware y software desarrollado para el sistema, además, de las pruebas de funcionamiento y el análisis de estas.

## **6. Conclusiones y Recomendaciones**

En este apartado se presenta las conclusiones obtenidas como resultado del proyecto, las recomendaciones a problemas encontrados tanto en el desarrollo como en la implementación de los algoritmos de visión artificial y robótica cooperativa.

# CAPÍTULO I

## 1. FUNDAMENTACIÓN TEÓRICA

### 1.1. Introducción

Con el paso del tiempo los procesos productivos han ido tomando caminos altamente eficaces, con su inicio desde la creación de la máquina de vapor, para pasar por los diferentes niveles de automatización del proceso de producción llegando finalmente al concepto de los “**Robots**” como herramienta para la escalabilidad en la producción y reducción del riesgo para el operario (Soria, Carelli, Kelly, & Zannatha, 2004).

En primera instancia se trataba de realizar robots móviles que pudieran solventar necesidades y procesos de alta complejidad, dando como resultado un fuerte procesamiento individual (Soria et al., 2004). Esto aumentaba el costo de desarrollo e implementación debido a que la adición de sofisticadas capacidades individuales convierte una inversión mayoritaria con poca capacidad de crecimiento, escalabilidad y flexibilidad (Inglett & Rodríguez-Seda, 2017).

La opción tomada para solventar los inconvenientes antes mencionados es la implementación de sistemas basados en la cooperación de múltiples robots, debido a que permite desarrollar aplicaciones de mayor complejidad, reduciendo tiempo de ejecución y costo de implementación, mediante el uso de varios robots de diseño sencillo que trabajan juntos para ejecutar una tarea (Illanes Tibanlombo, 2018; Lara et al., 2015; Pérez Checa, 2016).

Una de las aplicaciones destacadas en el campo de la robótica es el reemplazo de un operario en la manipulación de sustancias que podrían resultar perjudiciales para la integridad del mismo (Apolo Peñaloza & Velasco Barrera, 2013), debido a las

condiciones de riesgo presentes en ambientes que manejan sustancias químicas o de otras índoles.

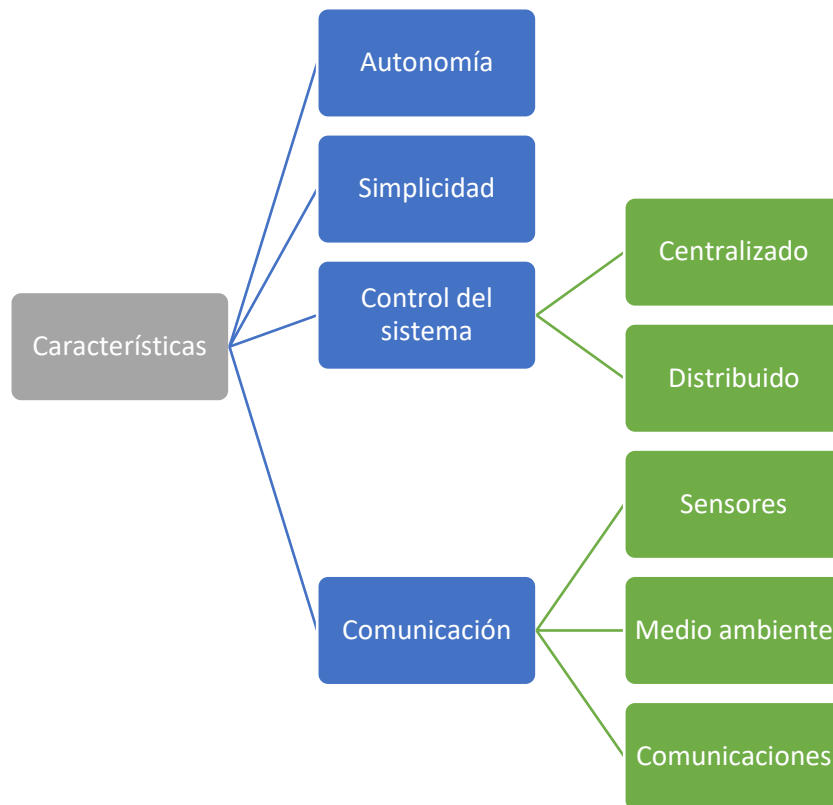
Para evitar complicaciones en dichos procesos, se desarrollan normas internacionales para el orden, limpieza, señalización, clasificación y almacenamiento de sustancias químicas (Gonzalo Rafael Rojo Aguirre, 2014). Un método de interés para el presente trabajo es el sistema SAF-T-DATA de J.T. Baker que maneja un método codificado en colores para organizar las sustancias químicas de acuerdo a sus propiedades (Colmenares, 2014).

La segmentación del entorno y objetos por medio de colores es un campo que ha sido explorado con gran afán en trabajos como: (Chen et al., 2016; Lin et al., 2015; Possegger & Mauthner, 2015) donde se presentan métodos de detección de objetos basados en el color y/o forma, en los mismos la utilización de cámaras permite al robot adquirir percepción del ambiente obteniendo así información para la detección de objetos (Possegger & Mauthner, 2015).

## **1.2. Robótica Cooperativa**

La robótica cooperativa tiene como fundamento la sencillez en el diseño y control, teniendo una comunicación entre robots que permita tomar decisiones de grupo y no solo como individuo, dándole así versatilidad a la funcionalidad y permitiéndoles adaptarse a diversos entornos (Illanes Tibanlombo, 2018; Inglett & Rodríguez-Seda, 2017).

Otra de las características que definen a los sistemas de robótica cooperativa es que pueden como no colaborar entre sí (Murray, n.d.), no obstante, el objetivo de su utilización es agilizar el proceso así que al aumentar el número de individuos realizando una acción mejora el rendimiento y tiempo de operación. En la Figura 1.1. Se muestran las principales características de los robots cooperativos.



**Figura 1.1.** Principales características de los robots cooperativos.  
Fuente: (Solís Ortega, 2016)

La robótica cooperativa para resolver una tarea se basa en el concepto de inteligencia cooperativa, es decir al existir distintos robots se pueden dividir o compartir las tareas logrando reducir el tiempo de ejecución del sistema como se observa en la Figura 1.2, y en algunos sistemas cooperativos en caso de defecto de uno otro robot toma su posición logrando un sistema robusto y confiable (Apolo Peñaloza & Velasco Barrera, 2013; Inglett & Rodríguez-Seda, 2017).



**Figura 1.2.** Robots en trabajo cooperativo para levantar columnas.  
Fuente: (Illanes Tibanlombo, 2018)

### 1.2.1 Tipos de Control de los Sistemas de Robótica Cooperativa

Estos sistemas necesitan ser controlados de manera total por un individuo en el sistema o bien parcialmente; por tanto, los sistemas de robótica cooperativa pueden tener una topología de control centralizada o distribuida.

#### 1.2.1.1 Control Centralizado

En el control centralizado una unidad central se encarga de las funciones de decisión y planificación, brindando las soluciones de grupo y manteniendo al sistema integrado (Apolo Peñaloza & Velasco Barrera, 2013; De La Cruz, Carelli, & Gava, 2006; Soria et al., 2004). En este caso se necesita gran complejidad en un individuo de manera que se puedan implementar otros robots sencillos que solo sigan órdenes, ver Figura 1.3.

Al manejar toda la información la unidad central de control, puede generar retardos en acciones de control, y si sufre algún daño se comprometería el funcionamiento de todo el sistema.



*Figura 1.3.* Formación triángulo de robots terrestres  
Fuente: (Soria et al., 2004)

#### 1.2.1.2 Control Descentralizado

Por otra parte, el control distribuido mantiene una topología en estrella donde cada individuo puede tener control parcial de sus acciones y del grupo, dándole una carga menor a su unidad de control, un sistema de fútbol robótico es un claro ejemplo de control distribuido, ver la Figura 1.4. Los robots del sistema están provistos de una red de

sensores para la adquisición de datos, un controlador propio que le permite resolver problemas basándose en los datos adquiridos del medio, y tecnologías de comunicación para el intercambio de datos entre los robots del sistema (De La Cruz et al., 2006; Soria et al., 2004).



**Figura 1.4.** Sistema de fútbol robótico vista de la cámara cenital  
Fuente: (Chicala, Jacho, Atencia, & Vintimilla, 2006)

### 1.3. Visión Artificial

Desde la aparición del procesamiento de imágenes, la visión artificial se ha convertido en un sistema confiable y de bajo costo para tareas como la detección de objetos, estimación de la posición espacial, interpretación de rasgos, detección de sucesos, etc. (Efrain Ernesto et al., 2015) .

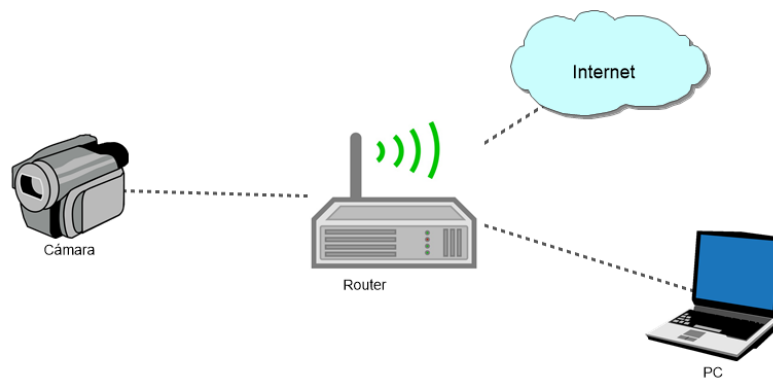
El uso de visión artificial potencia las aplicaciones de robótica móvil a un bajo costo y ofreciendo una amplia funcionalidad, debido a que un solo sensor (cámara) puede proveer de niveles elevados de autonomía al robot y dotarlo de información relevante del entorno para la toma de decisiones (Etxeberria, 2010; Lara et al., 2015).

Un sistema de visión artificial también comprende el reconocimiento y la determinación de la posición de determinados objetos en una imagen 2D, para ello capta

la información del entorno 3D y la extrae para su posterior análisis (Efrain Ernesto et al., 2015).

### 1.3.1 Cámara

Los sistemas de video-vigilancia evolucionan a la par con la tecnología, hoy en día la video-vigilancia se realiza de manera remota mediante el empleo de cámaras IP. Utilizando sistemas basados en el protocolo IP se puede obtener una mejor calidad de video a un costo bajo de implementación, debido a que las cámaras IP se conectan a la red mediante un puerto Ethernet, ver la Figura 1.5, en el caso de ser alámbricas evitando cableado coaxial adicional que necesitan las cámaras analógicas (Mejía, 2015; Revista Negocios de Seguridad, 2007).



**Figura 1.5.** Conexión en red de una cámara IP  
Fuente: (Revista Negocios de Seguridad, 2007)

Una cámara IP es la combinación de una cámara y un computador en un solo dispositivo, la cual transmite video, imagen y audio en vivo mediante una red IP a los usuarios autorizados, quienes pueden acceder a la información desde cualquier computador que disponga de acceso a Internet. Para esta función es necesario asignar una dirección IP a la cámara, la cual puede ser alámbrica o inalámbrica (Diestra & Guazhambo Ortiz, 2012; Revista Negocios de Seguridad, 2007).



Las cámaras IP que se utilizan en sistemas de video-vigilancia deben ser capaces de integrarse a sistemas más grandes y funcionar también como solución aislada, en el mercado existen diversos modelos y marcas de cámaras IP, de las cuales destaca la marca HikVision (Hikvision, 2019).

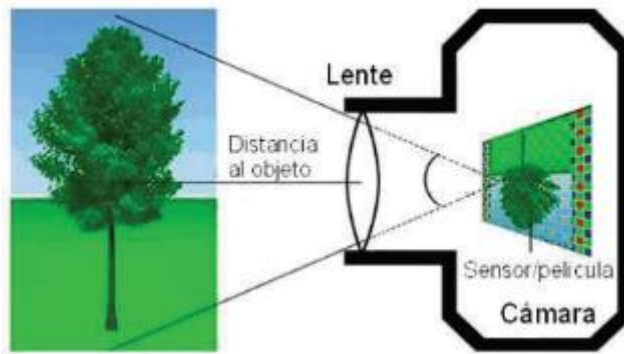
Sus productos son ideales para aplicaciones de vigilancia, aprendizaje e industria; las imágenes de las cámaras HikVision, ver la Figura 1.6, son de alta calidad en condiciones de baja iluminación.



**Figura 1.6.** Cámara IP HikVision  
Fuente: (Hikvision, 2019)

#### **1.3.1.1 Proceso de formación de imágenes**

El proceso de formación de imágenes inicia con la captura de la imagen a través de la cámara, para después ser digitalizada y entregada en forma de datos para su posterior procesamiento. El método de pin-hole describe un sistema para captar una imagen, se basa en una cámara ideal y expone la formación de una imagen como la proyección bidimensional de un escenario tridimensional como se observa en la Figura 1.7 (Pérez Checa, 2016).

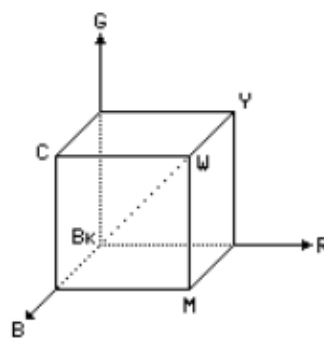


**Figura 1.7.** Modelo Pin-Hole  
Fuente: (Pérez Checa, 2016)

### 1.3.2 Modelo de color

El modelo de color representa matemáticamente las propiedades de un color para su uso digital. Un solo modelo de color no puede explicar todos los aspectos y características percibidas de un color, por lo que es necesario utilizar diferentes modelos que ayuden en la descripción (Kravtchenko, 1992). El modelo o espacio de color está definido por una base de  $n$  vectores, las combinaciones lineales generan todos los elementos del espacio; entre mayor sea  $n$  mayor será la cantidad de colores comprendidos (Rojas & Alonso Pérez, 2008). Los modelos de color más utilizados son HSI, RGB y YIQ.

#### 1.3.2.1 RGB



**Figura 1.8.** Modelo de color RGB  
Fuente: (Kravtchenko, 1992)

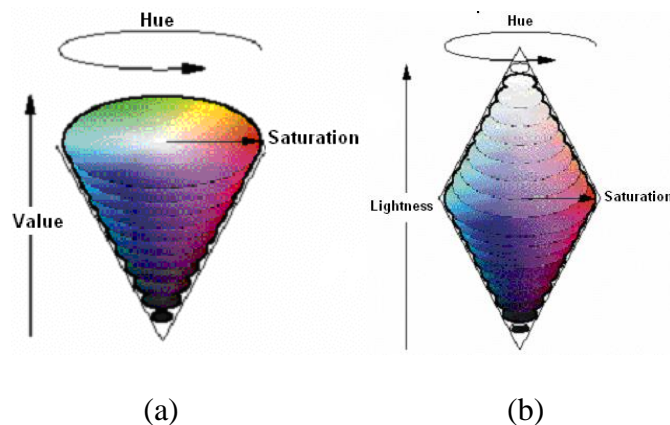
El modelo RGB (rojo, verde, azul) está basado en un cubo unitario en el sistema de coordenadas cartesiano como se observa en la Figura 1.8. Los colores R, G y B se

encuentran distribuidos en los ejes principales, por conveniencia los colores dentro del cubo están normalizados, es decir están en el intervalo de [0,1] (Rojas & Alonso Pérez, 2008). Para imágenes digitalizadas los valores de R, G y B son números enteros comprendidos en el intervalo [0, 255]. Todos los demás colores se generan por síntesis aditiva, es decir por la combinación de los colores primarios R, G y B (Etxeberria, 2010).

La diagonal principal representa el color gris conformado por cantidades iguales de rojo, verde y azul. El modelo de color RGB está orientado a hardware y es utilizado en dispositivos de captura, procesamiento y renderización de imágenes. Debido a la mezcla de datos de crominancia y luminancia no es un modelo de color aplicable para algoritmos de reconocimiento (Etxeberria, 2010; Gonzales & Woods, 2007; Kravtchenko, 1992).

### 1.3.2.2 HSI

El modelo HSI está basado en la percepción fisiológica del espacio de color del ojo humano describiendo el color con valores intuitivos (Vezhnevets, Sazonov, & Andreeva, 2003). En el modelo HSI y sus variantes, ver la Figura 1.9, el color se definen por tres características perceptuales: matiz (Hue), saturación (Saturation) y luminosidad o intensidad (Luminance) (Etxeberria, 2010).

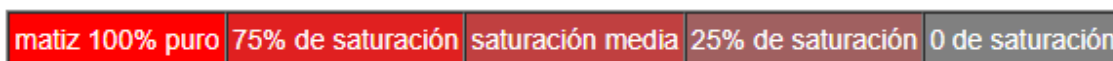


**Figura 1.9.** Modelo de color (a) HSV, (b) HSL.  
Fuente: (Gonzales & Woods, 2007)

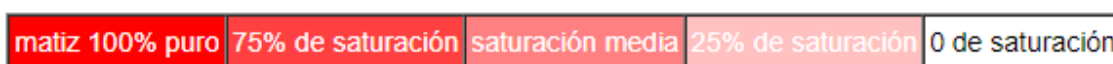
El matiz representa la percepción del color dominante; saturación corresponde a la cantidad de luz blanca presente en la imagen, los posibles valores que puede tomar van del 0 al 100%, siendo grisácea la imagen mientras menor cantidad de saturación presente la imagen. Y la intensidad representa la cantidad de iluminación que refleja un objeto (Rojas & Alonso Pérez, 2008).

Los modelos de color HSV (Hue, Saturation, Valor) y HSL (Hue, Saturation, Luminance) son variantes del espacio de color HSI, se diferencian entre sí por la característica de saturación expuesta en la Figura 1.10 (Etxeberria, 2010; Gonzales & Woods, 2007; Rojas & Alonso Pérez, 2008).

En el modelo HSL la saturación va desde el color puro hasta una tonalidad gris, mientras que en el modelo HSV la saturación va desde el color puro hasta el color blanco; siendo el modelo HSL el más intuitivo (Etxeberria, 2010; Gonzales & Woods, 2007; Rojas & Alonso Pérez, 2008).



(a)



(b)

**Figura 1.10.** Saturación en el modelo de color (a) HSL, (b) HSV.  
Fuente: (Etxeberria, 2010)

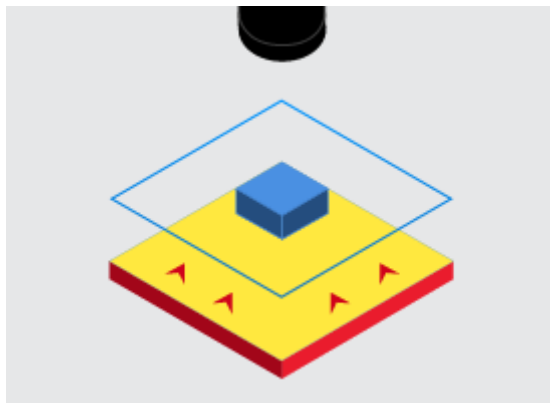
El modelo de color general HSI es más utilizado en aplicaciones de detección de colores y procesamiento de imágenes, debido a que con este espacio de color se puede reducir la influencia de la intensidad de luz del ambiente (Etxeberria, 2010).

### 1.3.3 Iluminación

La iluminación es un aspecto de suma importancia en los sistemas que utilicen visión artificial. Una adecuada iluminación permite controlar la forma en que la cámara observa al objeto de manera precisa y sin perder información. Existen diversas formas de iluminar un ambiente, de entre ellas se tiene: iluminación de fondo, iluminación direccional e iluminación difusa.

#### 1.3.3.1 Iluminación de fondo

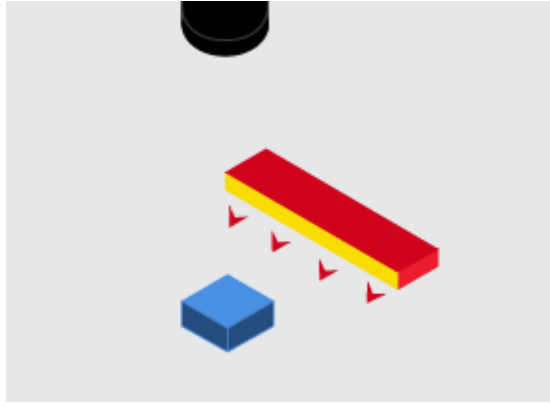
Esta técnica de iluminación proyecta uniformemente luz desde atrás del objeto, destacando su silueta, ver Figura 1.11. Se utiliza para comprobar la presencia/ausencia de orificios, detectar la forma del objetivo, entre otros.



**Figura 1.11.** Iluminación de fondo.  
Fuente: (Corporation, 2019)

#### 1.3.3.2 Iluminación direccional

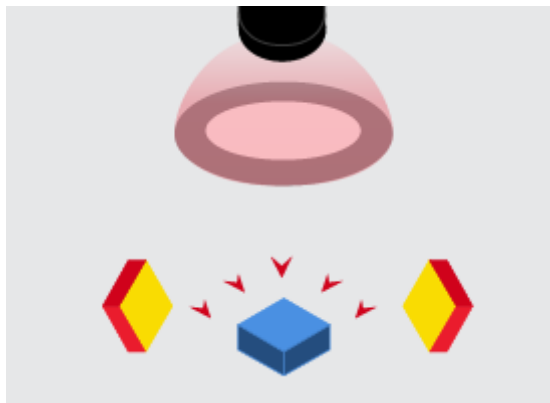
Este tipo de iluminación proporciona iluminación uniforme en un área localizada, ver Figura 1.12, según el ángulo la iluminación puede generar sombras en los objetos dificultando la aplicación de visión artificial.



**Figura 1.12.** Iluminación direccional.  
Fuente: (Corporation, 2019)

### 1.3.3.3 Iluminación difusa

Esta técnica de iluminación consiste en distribuir en el ambiente varias fuentes luminosas, ver Figura 1.13, para brindar una propagación de luminosidad uniforme, con el objetivo de eliminar las sombras en los objetos y disminuir el ruido de la iluminación (brillo).

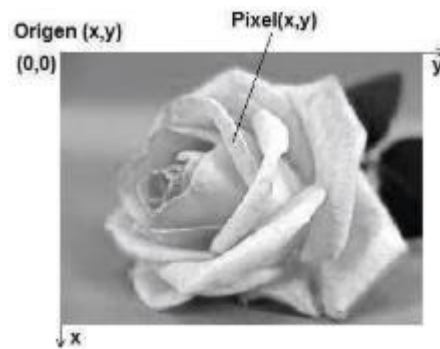


**Figura 1.13.** Iluminación difusa.  
Fuente: (Corporation, 2019)

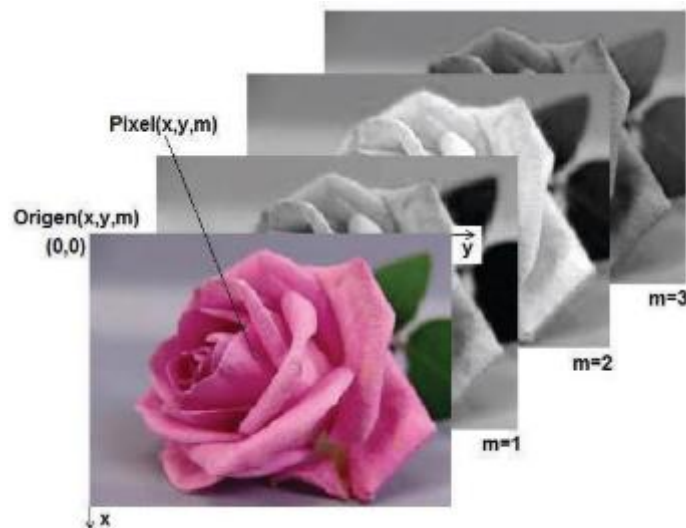
### 1.3.4 Procesamiento de imágenes

Una imagen digital se puede definir como función  $f(x,y)$ , donde  $x$  e  $y$  son coordenadas en un plano, ver Figura 1.14. Al digitalizar las coordenadas  $x$  e  $y$  representan un punto de la imagen y  $f$  representa el nivel de gris de ese punto, para las imágenes en

escala de grises o binarias. Para el caso de modelos de color RGB y HSI la función es  $f(x, y, m)$ , ver Figura 1.15, donde  $m$  representa las características de color (La Serna Palomino & Román Concha, 2009; Pérez Checa, 2016).

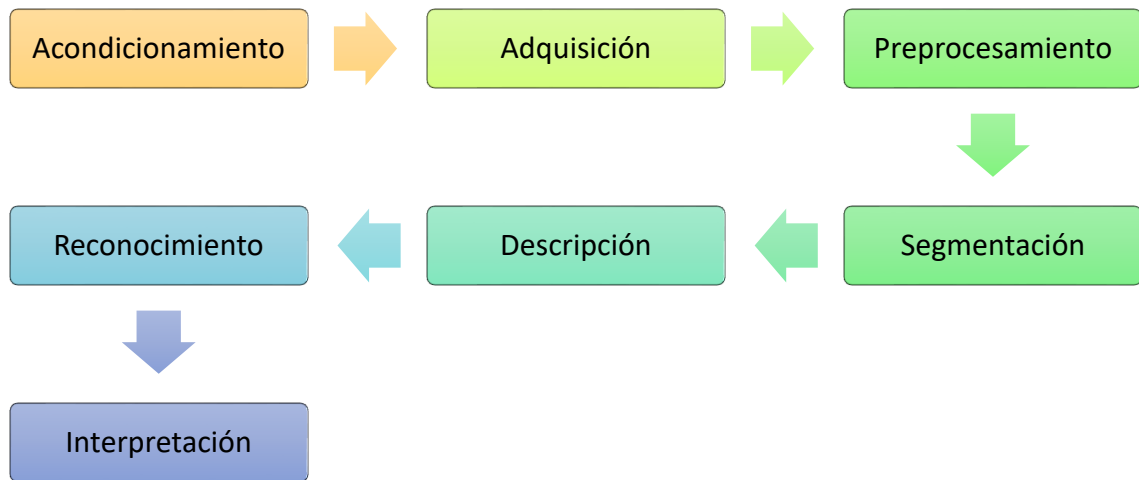


**Figura 1.14.** Imagen Binaria.  
Fuente: (Pérez Checa, 2016)



**Figura 1.15.** Imagen RGB.  
Fuente: (Pérez Checa, 2016)

Cada proyecto que aplique visión artificial puede tener sus propias características, sin embargo, el principio de funcionamiento del sistema es común a continuación, en la Figura 1.16 las principales etapas a seguir.



**Figura 1.16.** Procesamiento digital de imágenes.

Fuente: (Javier, Arias, Cesar, & Sotelo, n.d.; Pérez Álvarez, 2017)

#### **1.3.4.1 Acondicionamiento**

La etapa inicial de acondicionamiento hace referencia a las operaciones y cálculos necesarios que se pueden efectuar en el ambiente y en el dispositivo de captura previo a la adquisición de imágenes, tales como: iluminación, selección de la cámara, entre otros.

#### **1.3.4.2 Adquisición de la imagen**

En esta etapa se requiere de un sensor el cual transforma la señal eléctrica en función de la intensidad luminosa percibida en el ambiente, captando una imagen. Para obtener una imagen de alta calidad se debe considerar la resolución, profundidad del color y nitidez, estos aspectos evitan agregar errores al sistema (Javier et al., n.d.; La Serna Palomino & Román Concha, 2009).

#### **1.3.4.3 Preprocesamiento**

La etapa de preprocesamiento permite mejorar las imágenes originales detectando y eliminando características no deseadas o realzando partes de mayor interés. Las técnicas más utilizadas son: eliminación de ruido, mejora de contraste, conversión de los niveles de gris, transformaciones geométricas y restauración de la imagen (La Serna Palomino & Román Concha, 2009; Pérez Álvarez, 2017).



#### **1.3.4.4 Segmentación**

La etapa de segmentación constituye el proceso de subdividir una imagen en partes constituyentes que mantengan relación con la aplicación de interés (Gonzales & Woods, 2007). Un correcto proceso de segmentación simplificará el sistema de resolución del problema (Javier et al., n.d.). Los objetos de interés que aparecen en una imagen pueden ser aislados utilizando técnicas de procesamiento de imágenes basadas en detección de discontinuidades y similitudes (La Serna Palomino & Román Concha, 2009).

Los métodos de detección basados en discontinuidad se fundamentan en los cambios violentos de nivel de gris, detectando bordes, líneas y puntos en una imagen. Los métodos basados en similitudes detectan las partes de interés de una imagen basándose en características de homogeneidad como nivel de intensidad, posición, textura, entre otros (La Serna Palomino & Román Concha, 2009).

La umbralización es una técnica de suma importancia en la segmentación de imágenes, proporciona una forma de aislar los objetos del fondo convirtiendo una imagen con diversas tonalidades en una imagen binarizada, con la finalidad de que los píxeles que estén fuera del umbral tomen el color blanco y los que no tomen el color negro (Gonzales & Woods, 2007; Jiménez Camacho, 2009).

La umbralización se basa en el histograma de la imagen, seleccionando uno o más umbrales que permitan la agrupación de regiones de interés dentro de la imagen, como se observa en la Figura 1.17.



Para la autolocalización de los robots se utiliza la información del sistema sensorial implementado, que puede ser ubicado en el robot o en el escenario. Mediante este sistema se establece la posición del robot y los obstáculos, para esto se pueden aplicar grafos de visibilidad asignando peso a cada nodo o punto para identificarlo en el espacio de navegación (Pérez Checa, 2016).

Los métodos más conocidos y utilizados para planificación de trayectoria están basados en modelos geométricos como: *grafos de visibilidad* que establecen que dos puntos pueden ser unidos mediante una línea recta, *diagramas de Voronoi* que ubica puntos centrales, lugares geométricos en espacios libres de colisiones para generar trayectorias, *Roadmap (RPM)* que genera  $n$  trayectorias aleatorias libres de colisión, *Rapidly exploring random tree (RRT)* entre otros (Diego & Fernando, 2012; Pérez Checa, 2016).

## **CAPÍTULO II**

### **2. MARCO METODOLÓGICO**

#### **2.1. Diseño de la Investigación**

Dado que el objetivo del estudio será diseñar e implementar un sistema clasificador de objetos de color utilizando técnicas de visión artificial y robótica cooperativa para emular el proceso de trasladar, clasificar y almacenar sustancias químicas, se recurrirá a un diseño experimental que se aplicará de manera longitudinal.

#### **2.2. Enfoque de la Investigación**

De acuerdo al problema planteado y a los objetivos a alcanzar, el proyecto de investigación se considerará de tipo mixto sustentado en el libro de Roberto Sampieri Metodología de la Investigación, integrando métodos cualitativos y cuantitativos, puesto se adapta mejor a las características y necesidades del proyecto.

El enfoque cuantitativo utilizará la recolección y análisis de datos para evaluar las preguntas de la investigación y probar las hipótesis previamente establecidas. El enfoque cualitativo permite conformar una base de información a partir de documentos relacionados con los temas: visión artificial, robótica cooperativa y navegación autónoma.

#### **2.3. Instrumentos de investigación**

Para el presente proyecto de titulación se aplica la investigación enmarcada en procesos analíticos, descriptivos y experimentales.

El tema de esta investigación está enfocado en la tecnología, por lo que para el desarrollo de la fundamentación teórica del proyecto se recurre a utilizar el método deductivo junto con la técnica de revisión bibliográfica, obteniendo información relevante con respecto a robótica cooperativa, visión artificial, navegación autónoma, entre otros.

Para las fases de diseño de software y hardware se maneja el método de modelación, relacionando las características técnicas del sistema robótico con los objetivos a alcanzar.

En la fase de implementación del sistema robótico se emplea el método experimental, se realiza pruebas de funcionamiento y se corrige los errores encontrados en el sistema.

#### **2.4. Etapas del proyecto**

En el presente proyecto se aplica una metodología de siete etapas:

- Recopilación de información de fuentes confiables.
- Definición de los requisitos del sistema en base a los objetivos del proyecto; considerando la parte de hardware y software que conforman el sistema robótico.
- Diseñar el hardware y software del sistema mediante diagramas de flujo y diagramas de bloques. El diseño y su verificación es una etapa de suma importancia para el proyecto, ya que permite minimizar los errores en la etapa de implementación.
- Desarrollo de los algoritmos y configuración de los parámetros para emplear el software en el sistema.
- Integración de hardware con software, para esto se verifica la conectividad de todos los elementos que conforman el sistema.
- Realización de pruebas de desempeño del sistema, validando la lógica de programación, la comunicación entre los elementos electrónicos y la funcionalidad del sistema.

## **2.5. Población**

En este proyecto se tomará como población a tres robots de configuración diferencial, debido a que la población es de número reducido la muestra abarcará por completo el tamaño de la población.

## **2.6. Técnicas de recolección de datos**

Las técnicas de recolección de datos establecen los mecanismos, medios o recursos encaminados a recolectar, analizar y transmitir la información alcanzada una vez finalizada la investigación.

Se realizarán pruebas de desempeño de los algoritmos diseñados e implementados en el sistema cooperativo de múltiples robots midiendo robustez, velocidad y precisión de los mismos en diferentes condiciones de iluminación y cambios en el escenario, los datos de las diferentes pruebas serán almacenados en archivos de texto para su posterior análisis.

## **2.7. Técnicas de procesamiento de datos**

A través de la estadística se procesarán los datos obtenidos de las pruebas anteriormente realizadas para describir, organizar, analizar e interpretar de manera apropiada los resultados obtenidos.

La técnica que se utilizará en el procesamiento de datos será la estadística descriptiva, que consiste en presentar la información relevante adquirida en tablas, gráficos explicativos y/o medidas de resumen.

## **2.8. Herramientas para el procesamiento de datos**

Para llevar a cabo la tabulación de los datos se utilizará el programa Microsoft Office Excel y el software MATLAB para cálculo técnico y visualización gráfica de los datos obtenidos en las pruebas de funcionalidad realizadas.

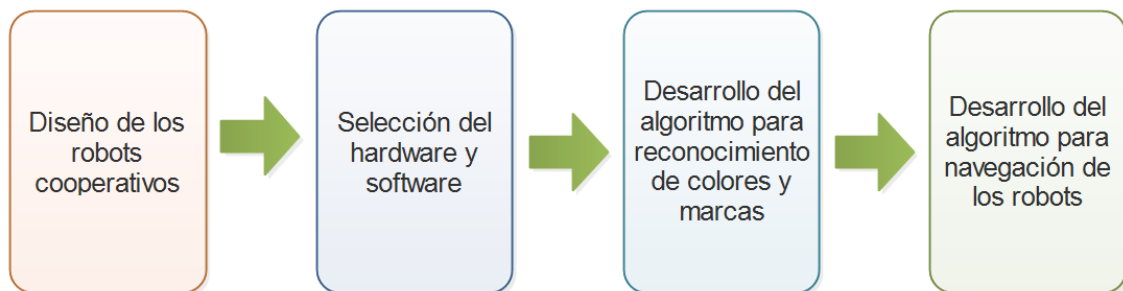
## CAPÍTULO III

### 3. PROPUESTA

#### 3.1. Introducción

El presente proyecto de titulación, busca realizar un aporte en el campo de la industria química, haciendo posible la utilización de un sistema de múltiples robots cooperativos para emular el proceso de clasificación y traslado de sustancias químicas en un entorno. Los sistemas de robots cooperativos trabajando en conjunto con sistemas de visión artificial en ambientes industriales, están orientados tanto a mejorar la seguridad de los operarios, como a optimizar el tiempo de trabajo.

En la realización de este proyecto se han analizado distintos dispositivos electrónicos en busca de las características necesarias para cumplir con los objetivos planteados; en base a un detallado análisis se han seleccionado los siguientes elementos para hardware y software. El diagrama de bloques de la Figura 3.1 presenta las etapas del proyecto tanto de hardware como software.



**Figura 3.1.** Diagrama de las etapas del proyecto.

Fuente: Elaborado por el autor

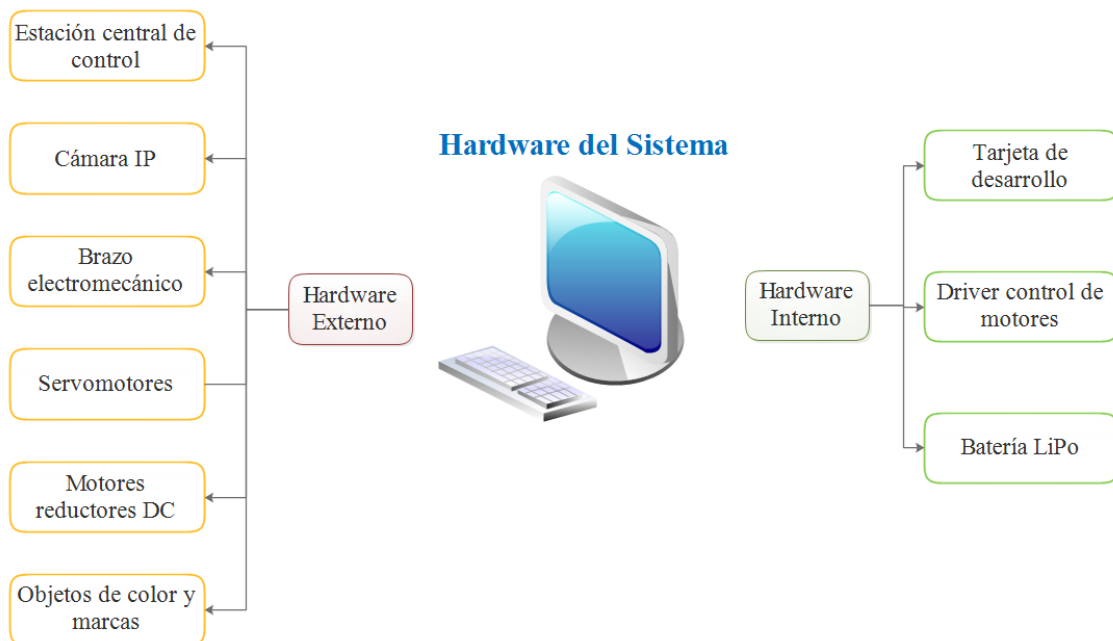
En la primera etapa se seleccionará los robots móviles a utilizar en el proyecto, con base a su configuración, su tamaño y su costo. En la segunda etapa se escogerá los dispositivos electrónicos y mecánicos que cumplan con los requerimientos para el

desarrollo del proyecto. En la tercera y cuarta etapa, se elegirá el lenguaje e interprete para el desarrollo de los algoritmos de visión artificial y los algoritmos de navegación de los robots.

### 3.2. Hardware del Sistema

El sistema cooperativo estará compuesto por tres robots de configuración diferencial dotados de un microcontrolador y actuadores para el control, navegación y comunicación con la estación central de procesamiento, y a su vez la estación central se conecta con la cámara IP que proporciona la información visual del entorno.

El sistema cooperativo presentará un control centralizado, en el cual la estación central de procesamiento se comporta como servidor enviando señales de control a los clientes (robots) para que puedan movilizarse en el entorno de prueba sin colisionar y trasladen los objetos de color a posiciones determinadas. En la Figura 3.2 se observa el diagrama de bloques del hardware requerido para el proyecto.



**Figura 3.2.** Diagrama de bloques del hardware del sistema.

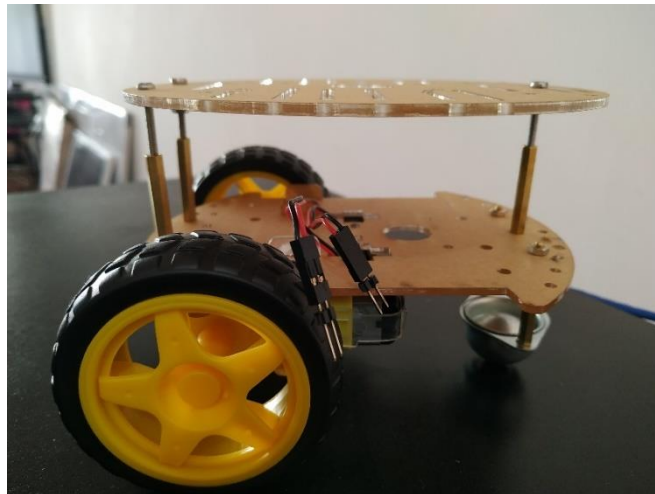
Fuente: Elaborado por el autor



### 3.2.1. Robots móviles

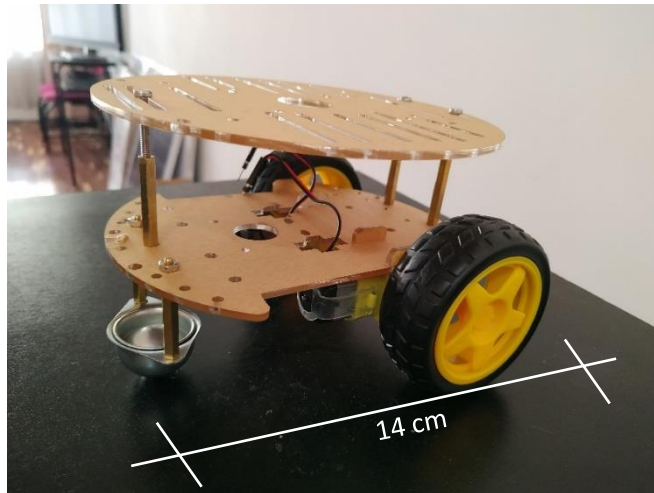
Para este proyecto se seleccionó robots móviles de configuración diferencial de dos plantas, cuya estructura consta de dos motores DC y una rueda loca para mantener la estabilidad. El material de que está fabricado el chasis de los robots es de acrílico transparente de 3mm de espesor, este material presenta alta resistencia a agentes químicos, baja conductividad y dureza.

Las capas del chasis se encuentran acopladas mediante tornillos y tuercas de 2.5mm de diámetro, como se puede observar en la Figura 3.3.



**Figura 3.3.** Estructura de los robots móviles.  
Fuente: Elaborado por el autor

El chasis de los robots mide 14 x 14 cm, ver la Figura 3.4, las dimensiones están acorde con los objetos de color y el ambiente en el que se van a desplazar los robots.



**Figura 3.4.** Dimensiones del chasis de los robots móviles.  
Fuente: Elaborado por el autor

### 3.2.2. Motores DC

Los motores DC permiten variar la velocidad de giro sin afectar el torque, debido a que transforman la energía eléctrica en energía mecánica. Son utilizados en aplicaciones educativas, de investigación y en la industria, dependiendo la escala de los mismos. Para este proyecto se utilizarán motorreductores, que son motores DC de baja escala con cajas reductoras, ver Figura 3.5.



**Figura 3.5.** Motorreductor DC.  
Fuente: Elaborado por el autor

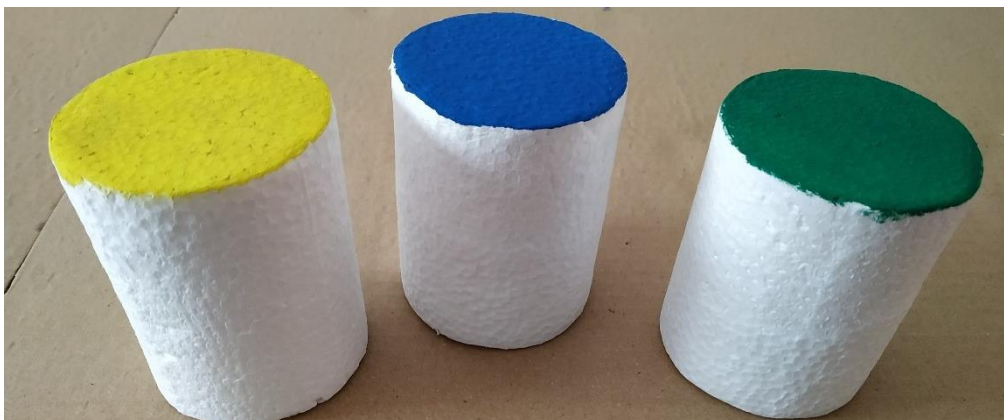
Los motorreductores seleccionados para este proyecto presentan las siguientes características:

- Torque: 0.8Kg.cm – 1.1Kg.cm.

- Velocidad sin carga: 125RPM - 230 RPM.
- Velocidad con carga: 95RPM – 175 RPM.
- Consumo de corriente sin carga: 60mA – 100mA.
- Consumo de corriente con carga: 200mA - 670mA.
- Voltaje de alimentación: 3V – 12V DC, recomendado 3V – 6V DC.
- Parámetros de reducción: 1:48.
- Dimensiones: 71mm x 27mm x 23 mm aproximadamente.
- Peso: 50g.
- Ruido: < 65dB.
- Material de fabricación: plástico resistente y no tóxico.

### 3.2.3. Objetos de color

En el escenario en el que desplazaran los robots se encuentran ubicados aleatoriamente objetos de color seleccionados de acuerdo al método SAF-T-DATA de J.T Baker (Colmenares, 2014). Los objetos deben ser livianos para que puedan ser transportados por los robots móviles seleccionados en el apartado anterior, por lo que se eligió trabajar con objetos cilíndricos de espuma flex de 8cm x 10cm. El tamaño de los cilindros permite que sean reconocidos por la cámara IP a una altura de dos metros desde el piso.



**Figura 3.6.** *Objetos de color.*  
Fuente: Elaborado por el autor

### 3.2.4. Cámara IP

Para la selección de la cámara IP se realizó un análisis de las características requeridas para lograr una detección de objetos de color y marcas a una altura máxima de dos metros.

- La cámara será usada en interiores, no es necesario protección para exterior.
- La cámara deberá tener una resolución comprendida en el rango de 0.8 (1024x768) – 2.3 (1920x1200) megapíxeles, ya que será utilizada para reconocimiento de objetos de color y marcas.
- La transmisión de video será en tiempo real, por lo que no es necesario que la cámara cuente con tarjeta de memoria.
- La calidad de la imagen debe ser alta en condiciones de baja iluminación.

Con base al análisis de los requerimientos planteados se optó por una cámara IP de marca Hikvision modelo DS-2CD1021-I, la misma que se observa en la Figura 3.7. Esta cámara cumple con los requerimientos descritos anteriormente necesarios para la detección de objetos de color y marcas bajo condiciones de alta y baja luminosidad.



**Figura 3.7.** Cámara IP marca Hikvision.  
Fuente: Elaborado por el autor

A continuación, se describen las principales características técnicas de la cámara seleccionada para este proyecto (Hikvision, 2019):

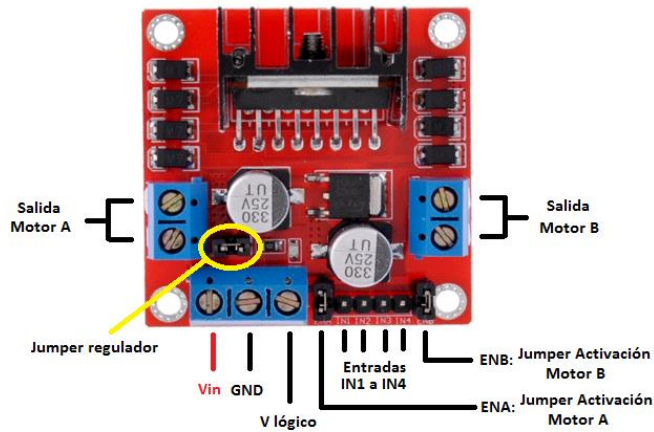
- Sensor de imagen CMOS.
- Visión nocturna IR.
- Resolución máxima de la imagen 1920 x 1080.
- Interfaz de comunicación: puerto ethernet auto adaptativo RJ45.
- Fuente de alimentación 12 VDC, 0.4A.
- Protección IP67.
- Dimensiones: 70 x 172 mm.
- Peso aproximado: 270 gramos.

### **3.2.5. Driver para motores DC**

Los robots móviles deben interpretar señales enviadas desde el centro de control para realizar el seguimiento de trayectoria dentro del medio de pruebas. Se deben emplear drivers de control de motores para cada uno de los robots del sistema, y así poder manipular las acciones de los robots. A continuación, se analizan dos módulos de control destacados en el mercado.

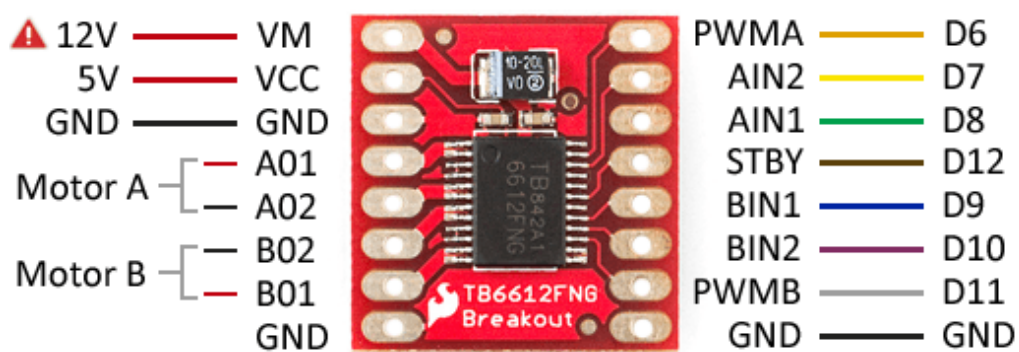
- Driver L298N
- Driver TB6612FNG

El módulo **L298N** es un puente H conformado por cuatro transistores BJT que permite controlar la velocidad y el sentido de giro de dos motores, ver Figura 3.8. Consume aproximadamente 3V de la alimentación, por lo que los motores reciben tres voltios menos de tensión. Trabaja en un rango de tensión medio que va desde 3V a 35V, y soporta una intensidad de corriente teórica de hasta 2A, en la práctica puede suministrar corriente hasta 1A debido a las pérdidas.



**Figura 3.8.** *Driver L298N.*  
Fuente: (STMicroelectronics, 2000)

El módulo **TB6612FNG** contiene dos puentes H conformados por transistores MOSFET, esto le permite tener mayor eficiencia en comparación con el driver L298N siendo capaz de suministrar 1.2A por canal, ver la Figura 3.9. Además, este módulo teóricamente no tiene caída de tensión y se comporta como una resistencia de 0.5 Ohm; también incorpora un modo que desactiva completamente al controlador, ahorrando energía. A diferencia del módulo L298N, no puede soportar tensiones superiores a 15V y presenta limitaciones en la protección contra corrientes inducidas.



**Figura 3.9.** *Driver TB6612FNG.*  
Fuente: (STMicroelectronics, 2000)

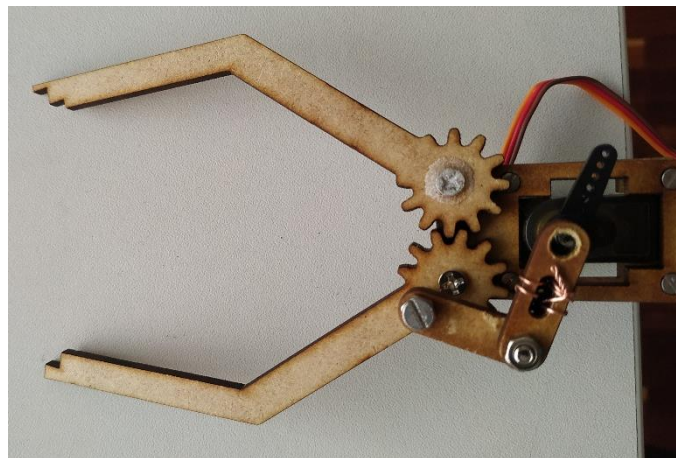
De los dos módulos analizados para este proyecto se trabajará con el módulo TB6612FNG, debido a su manejo eficiente de tensión, su reducido tamaño y las

prestaciones a nivel de hardware y software que ofrece en comparación con el módulo L298N.

### 3.2.6. Brazo electromecánico

Un brazo electromecánico está compuesto por una estructura mecánica y un control electrónico programable. También se los conoce como manipuladores y son utilizados para manipular cargas, ensamblar pieza, soldar entre otras actividades.

El número de grados de libertad del brazo está determinado por el número de articulaciones móviles, mismas que son determinadas por los requerimientos de movilidad de la actividad. El proceso de transporte de los objetos de color requiere de un brazo electromecánico de 1DOF, ver Figura 3.10, las señales de control serán interpretadas por un servomotor.



**Figura 3.10.** *Brazo electromecánico.*  
Fuente: (STMicroelectronics, 2000)

### 3.2.7. Servomotor

Un servomotor es un motor DC con la capacidad de control del desplazamiento angular, es decir, puede posicionar su eje en un determinado ángulo. Su rango de desplazamiento comprende de  $0^\circ$  a  $180^\circ$  siendo imposible completar una vuelta. Son



utilizados en aplicaciones que requieren el control del desplazamiento angular como: orientación de sensores, torretas, manipulación de objetos, entre otras.

Los modelos de servomotores se diferencian por el torque, siendo aconsejable elegir un servomotor con un torque mayor al requerido para no dañarlo. Este proyecto requiere de tres servomotores para el control de tres brazos electromecánicos, uno por cada robot móvil.

Se seleccionó el servomotor MG90, ver Figura 3.11, con base a las siguientes razones: tamaño reducido, bajo consumo de corriente, alto torque (2.2 Kg/cm) y precio accesible.



**Figura 3.11.** *Servomotor MG90.*  
Fuente: (STMicroelectronics, 2000)

### 3.2.8. Tarjeta de desarrollo

Una tarjeta desarrollo es un dispositivo electrónico que fusiona un microcontrolador, memoria, módulos de entradas/salidas y módulos de comunicación. Las tarjetas de desarrollo son utilizadas para aplicaciones de control, sistemas robóticos, adquisición y procesamiento de datos, entre otras. Para este proyecto se requiere de tarjetas de desarrollo que ofrezcan comunicación inalámbrica de datos, se analizó tres tarjetas.

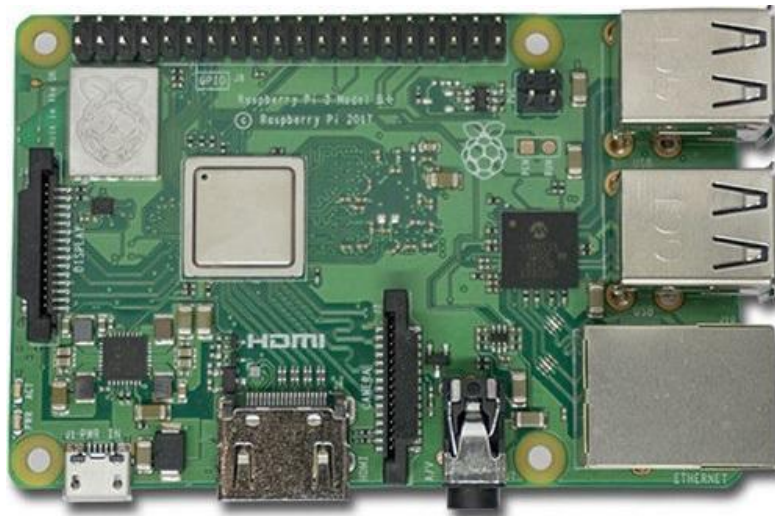
- Raspberry Pi
- NodeMCU



- Arduino Mega R3 con ESP8266

### 3.2.8.1. Raspberry Pi

La tarjeta raspberry pi, ver Figura 3.12, presenta todas las características de un computador con algunas limitaciones en el procesamiento y la memoria, por lo que puede ser utilizada para aplicaciones de procesamiento de video, audio y datos. Además, dispone de funciones de red local e inalámbrica (Raspberrypi.org, 2019).



**Figura 3.12.** Tarjeta Raspberry Pi.  
Fuente: (Raspberrypi.org, 2019)

Raspberry pi dispone de varios modelos de tarjetas, las características principales de los modelos son:

- CPU ARM Cortex o superior.
- Memoria RAM 1GB o superior.
- GPU que provee OpenGL.
- Conexión LAN.
- Conexión inalámbrica 802.11 b/g/n.
- Conexión bluetooth.
- 4 puertos USB.

- 40 pines GPIO de propósito general.
- Lector de tarjeta micro SD.
- Entrada micro USB.
- Puerto HDMI.
- Conector para cámara.

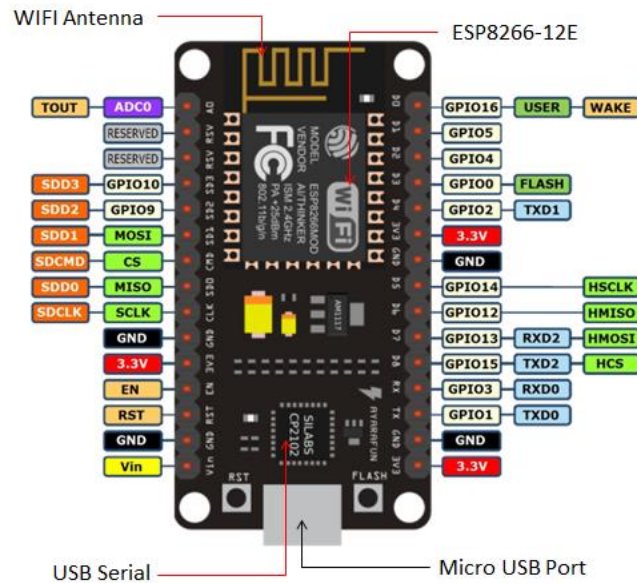
Los pines GPIO de la tarjeta raspberry pi pueden ser designados como entrada o salida mediante software, adicional pueden usarse para funciones como: PWM, SPI, I2C, serial, conexión con módulos, entre otras (Raspberrypi.org, 2019).

Raspberry Pi 3 Model B (J8 Header)					
GPIO#	NAME			NAME	GPIO#
	3.3 VDC Power	1			2
				5.0 VDC Power	
<b>8</b>	GPIO 8 SDA1 (I2C)	3		5.0 VDC Power	4
<b>9</b>	GPIO 9 SCL1 (I2C)	5		Ground	6
<b>7</b>	GPIO 7 GPCLK0	7		GPIO 15 TxD (UART)	<b>15</b>
	Ground	9		GPIO 16 RxD (UART)	<b>16</b>
<b>0</b>	GPIO 0	11		GPIO 1 PCM_CLK/PWM0	<b>1</b>
<b>2</b>	GPIO 2	13		Ground	14
<b>3</b>	GPIO 3	15		GPIO 4	<b>4</b>
	3.3 VDC Power	17		GPIO 5	<b>5</b>
<b>12</b>	GPIO 12 MOSI (SPI)	19		Ground	20
<b>13</b>	GPIO 13 MISO (SPI)	21		GPIO 6	<b>6</b>
<b>14</b>	GPIO 14 SCLK (SPI)	23		GPIO 10 CE0 (SPI)	<b>10</b>
	Ground	25		GPIO 11 CE1 (SPI)	<b>11</b>
<b>30</b>	SDA0 (I2C ID EEPROM)	27		SCL0 (I2C ID EEPROM)	<b>31</b>
<b>21</b>	GPIO 21 GPCLK1	29		Ground	30
<b>22</b>	GPIO 22 GPCLK2	31		GPIO 26 PWM0	<b>26</b>
<b>23</b>	GPIO 23 PWM1	33		Ground	34
<b>24</b>	GPIO 24 PCM_FS/PWM1	35		GPIO 27	<b>27</b>
<b>25</b>	GPIO 25	37		GPIO 28 PCM_DIN	<b>28</b>
	Ground	39		GPIO 29 PCM_DOUT	<b>29</b>

**Figura 3.13.** Puertos GPIO de la tarjeta Raspberry Pi.  
Fuente: (Raspberrypi.org, 2019)

### 3.2.8.2. NodeMCU

NodeMCU es una tarjeta de desarrollo orientada al internet de las cosas (IoT) y estaba basada en el módulo ESP8266 que soporta la comunicación inalámbrica mediante el estándar 802.11n, en Figura 3.14 se observa el esquema pinout de la tarjeta (nodeMCU, 2014).



**Figura 3.14.** Esquema pinout de la tarjeta NodeMCU.  
Fuente: (nodeMCU, 2014)

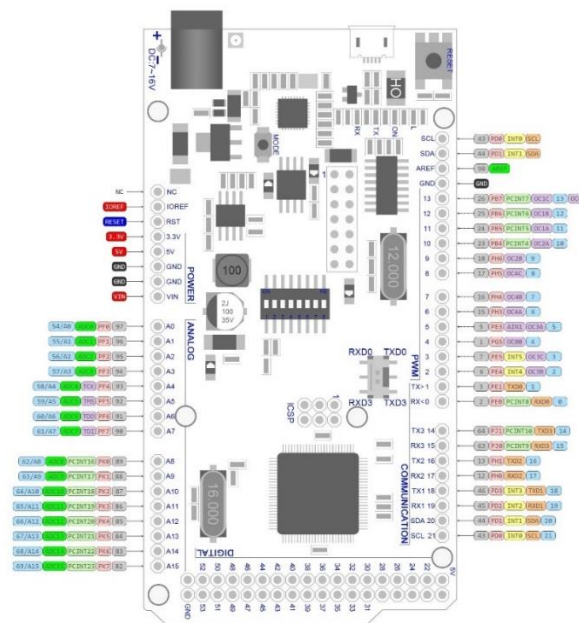
Esta tarjeta puede ser programada usando el IDE de Arduino o mediante el lenguaje LUA (LUA, n.d.). NodeMCU dispone de tres versiones hasta la fecha, mismas que varían en memoria y número de pines. A continuación, las principales especificaciones técnicas de las tarjetas:

- Microcontrolador: Tensilica 32bits.
- SoC: ESP8266.
- Velocidad del procesador: 80MHz a 180 MHz.
- Memoria RAM: 128KB.
- Memoria Flash: 512KB.

- GPIO: 11, cada uno puede ser PWM.
- 1 entrada ADC.
- Soporta estándar 802.11 b/g/n.
- Tensión de alimentación: 3.3V – 10 V DC.
- Antena agregada en la placa.

### 3.2.8.3. Arduino Mega R3 con ESP8266

Esta tarjeta integra el microcontrolador Atmel ATmega2560 y el chip ESP8266 en una sola placa, ver Figura 3.15, donde los componentes están configurados para trabajar juntos o de manera independiente utilizando interruptores DIP (RobotOyn, 2018).



**Figura 3.15.** Esquema pinout de la tarjeta arduino mega R3 con WiFi.  
Fuente: (RobotOyn, 2018)

Características técnicas importantes:

- Microcontrolador: Atmel ATmega2560.
- Chip: ESP8266.
- Conexión micro USB.

- Voltaje de alimentación: 7V – 12V DC.
- Velocidad del reloj: 16MHz para Atmega2560
- Velocidad de reloj: 80/160 MHz para ESP8266
- Entradas/salidas digitales: 54, 15 PWM.
- Entradas analógicas: 16.
- Memoria RAM: 4KB.
- Memoria Flash: 32KB.

En la Tabla 3.1, se realizó un análisis comparativo entre las tres tarjetas de desarrollo antes mencionadas, además se toma en cuenta que el proyecto requiere un total de siete pines digitales para la conexión y control de actuadores. Por lo que se deduce que la tarjeta **nodeMCU** cuenta con las características necesarias para la realización de este proyecto.

**Tabla 3.1**  
*Análisis comparativo entre tarjetas de desarrollo.*

	<b>Arduino Mega+ESP8266</b>	<b>Raspberry Pi 3B+</b>	<b>NodeMCU V2</b>
<b>Microcontrolador</b>	ATmega2560	MK64FX512VMD12	Tensilica L106
<b>Velocidad del procesador</b>	16 MHz	1.4 GHz	80 MHz
<b>Memoria RAM</b>	8 KB	1 GB	128 KB
<b>GPIO</b>	54	40	11
<b>ADC</b>	16	40	1
<b>DAC</b>	-	40	1
<b>PWM</b>	14	40	11
<b>Conector</b>	USB micro-B	USB micro-B	USB micro-B
<b>Voltaje de operación</b>	5.0 V	3.3 V	3.3 V
<b>Voltaje de alimentación</b>	7.0 – 12.0 V	5.0 V	5.0 V
<b>Serial</b>	4	6	2
<b>I2C</b>	1	3	1
<b>Tarjeta SD</b>	-	1	-
<b>Costo</b>	30	75	12

Fuente: (nodeMCU, 2014; Raspberrypi.org, 2019; SparkFun, n.d.)

### 3.2.9. Batería LiPo

La alimentación del sistema será realizada mediante una batería LiPo (litio y polímero) de tamaño y peso reducido. Para seleccionar el amperaje de la batería se realiza un análisis de consumo de corriente de cada dispositivo que compone el sistema, considerando los valores máximos de consumo.

**Tabla 3.2**

*Análisis de consumo de energía para la selección de la batería.*

Cantidad	Hardware	Voltaje de operación	Consumo de corriente individual	Consumo de corriente total
1	Driver TB6612FNG	5 - 15 V	450 mA	450 mA
2	Motorreductores	3 - 6 V	250 mA	500 mA
2	Servomotor MG90	3.3 – 5 V	327 mA	654 mA
1	NodeMCU V2	3.3 – 5 V	80 mA	80 mA

Fuente: Elaborado por el autor

De la **Tabla 3.2** se deduce que el consumo individual de cada robot móvil del sistema es aproximadamente 1684 mA, por lo que se seleccionó una batería que supere el amperaje requerido, ver Figura 3.16.



**Figura 3.16.** *Batería LiPo 7.4V y 2200mAh*  
Fuente: Elaborado por el autor

Utilizando la ecuación (1) se estima el tiempo de autonomía de la batería, con el objetivo de determinar el tiempo de operación del sistema.

$$\text{Tiempo de autonomía} = \frac{\text{Capacidad batería}}{\text{Consumo sistema}} \quad (1)$$

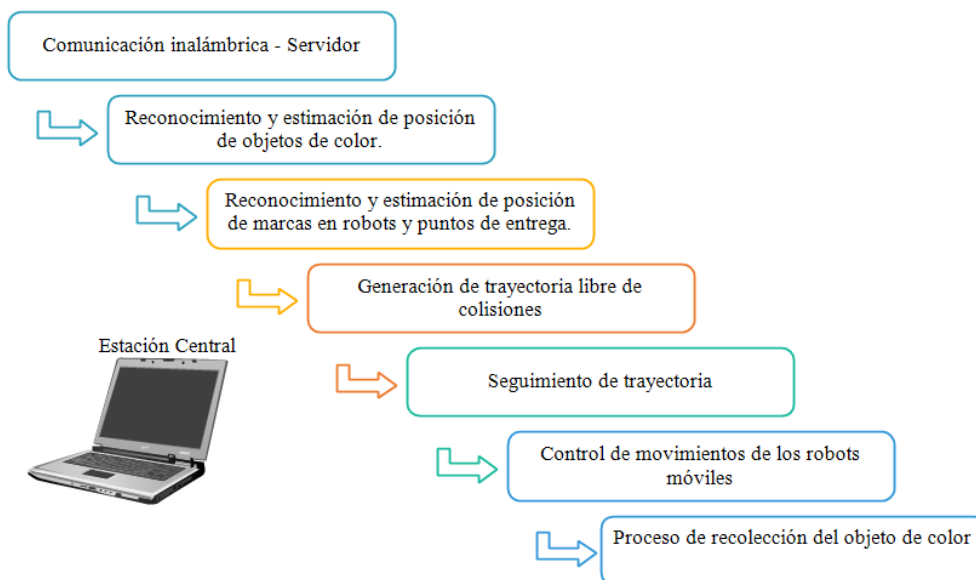
Reemplazando los datos obtenidos anteriormente se tiene:

$$\text{Tiempo de autonomía} = \frac{2200 \text{ mAh}}{1684 \text{ mA}} = 1.31 \text{ h} = 78 \text{ min} \quad (2)$$

Con base al resultado de la ecuación (2) cada robot del sistema puede operar por aproximadamente por 78 minutos, correspondiendo al consumo de la Tabla 3.2.

### 3.3. Software

Para este proyecto se utiliza software libre, tomando en consideración las características técnicas y recursos necesarios para su desarrollo. La estación central de procesamiento del sistema será implementada en un computador portátil, y la programación de los algoritmos de detección, reconocimiento y localización de objetos de color y marcas se realizará a través del intérprete PyCharm y el lenguaje de programación Python 3. A continuación, se observa en la Figura 3.17 los principales procesos que realiza el algoritmo de la estación central de procesamiento.

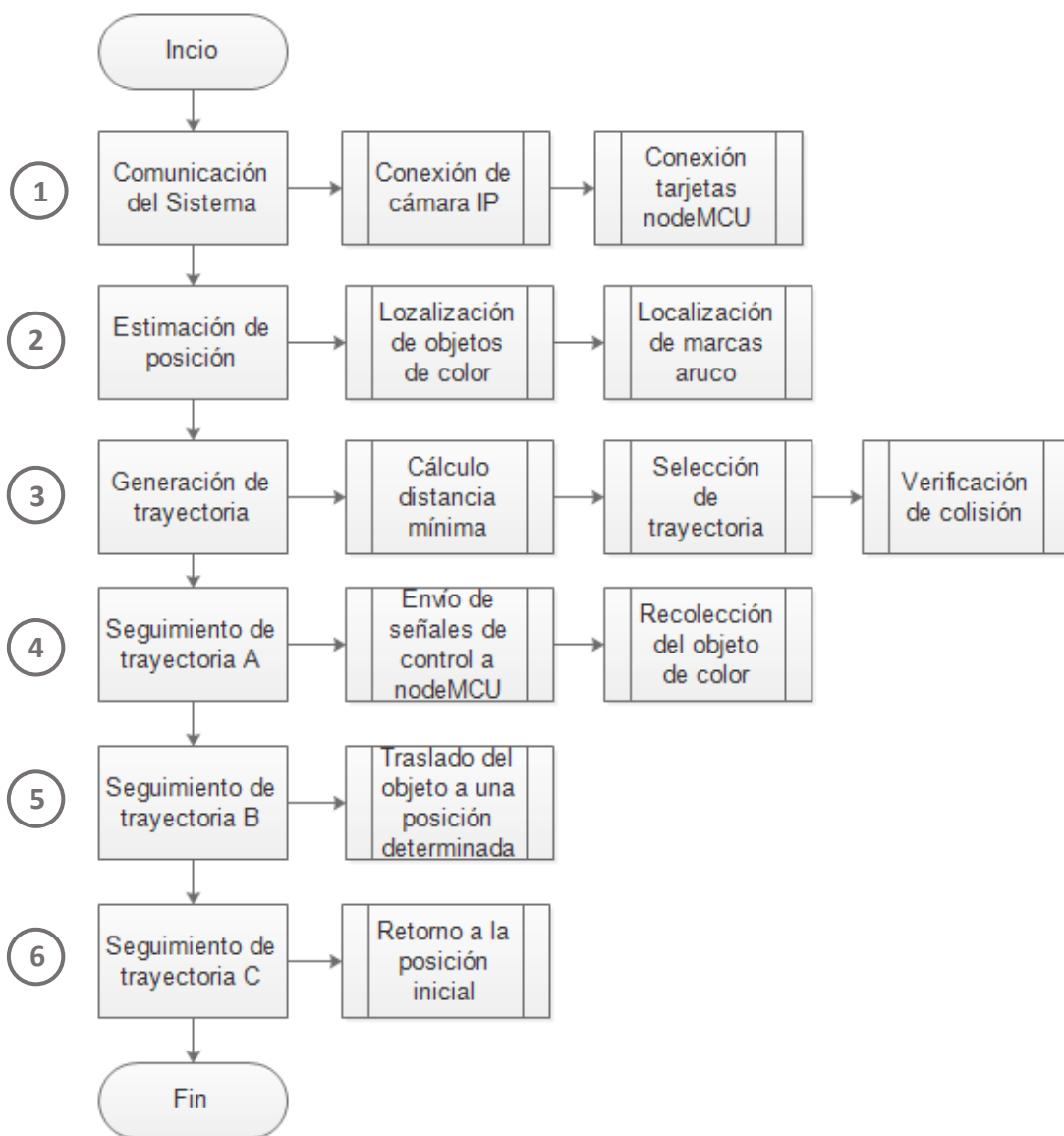


**Figura 3.17.** Diagrama de bloques del software de la estación central.  
Fuente: Elaborado por el autor

El algoritmo de comunicación y control de los robots móviles será implementado en las tarjetas nodeMCU V2 y desarrollado en el IDE de arduino.

### 3.3.1. Etapas del software del sistema

El sistema contará las etapas de software de la Figura 3.18, que comprenden: comunicación inalámbrica del sistema, estimación de posición de los objetos y marcas (robots y puntos de entrega), generación de trayectorias libre de colisiones, seguimiento de trayectorias, navegación de los robots móviles y transporte del objeto de color.

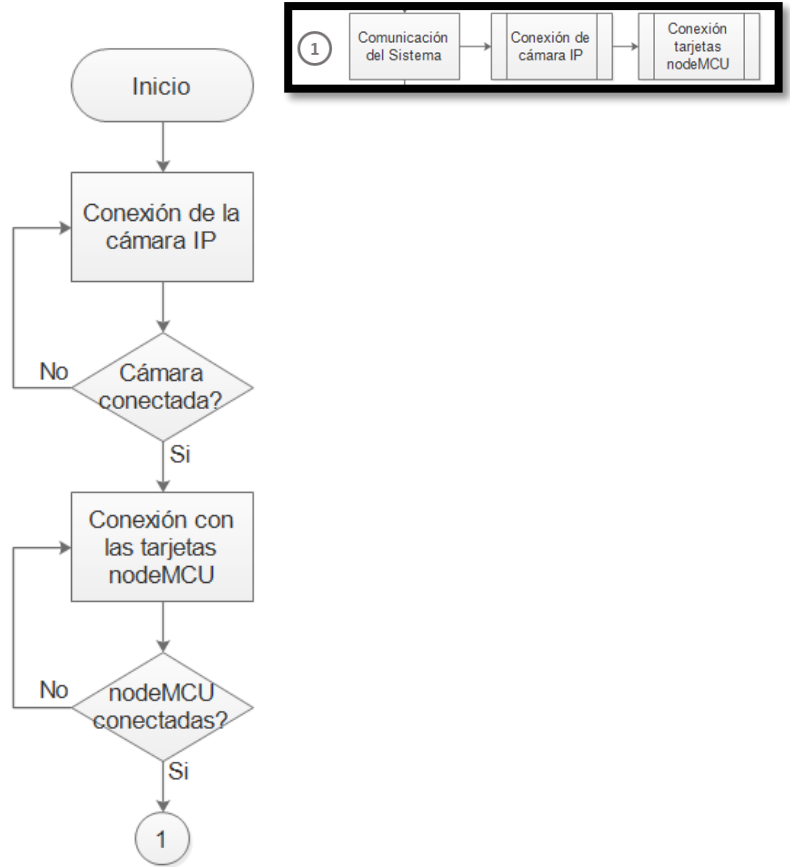


**Figura 3.18.** Diagrama de flujo de las etapas de software del sistema.

Fuente: Elaborado por el autor



En la primera etapa se realizará la comunicación inalámbrica WiFi de la cámara IP central y las tarjetas nodeMCU ensambladas en los robots con la estación central de control. En la Figura 3.19 se visualiza el diagrama de flujo correspondiente a esta etapa.

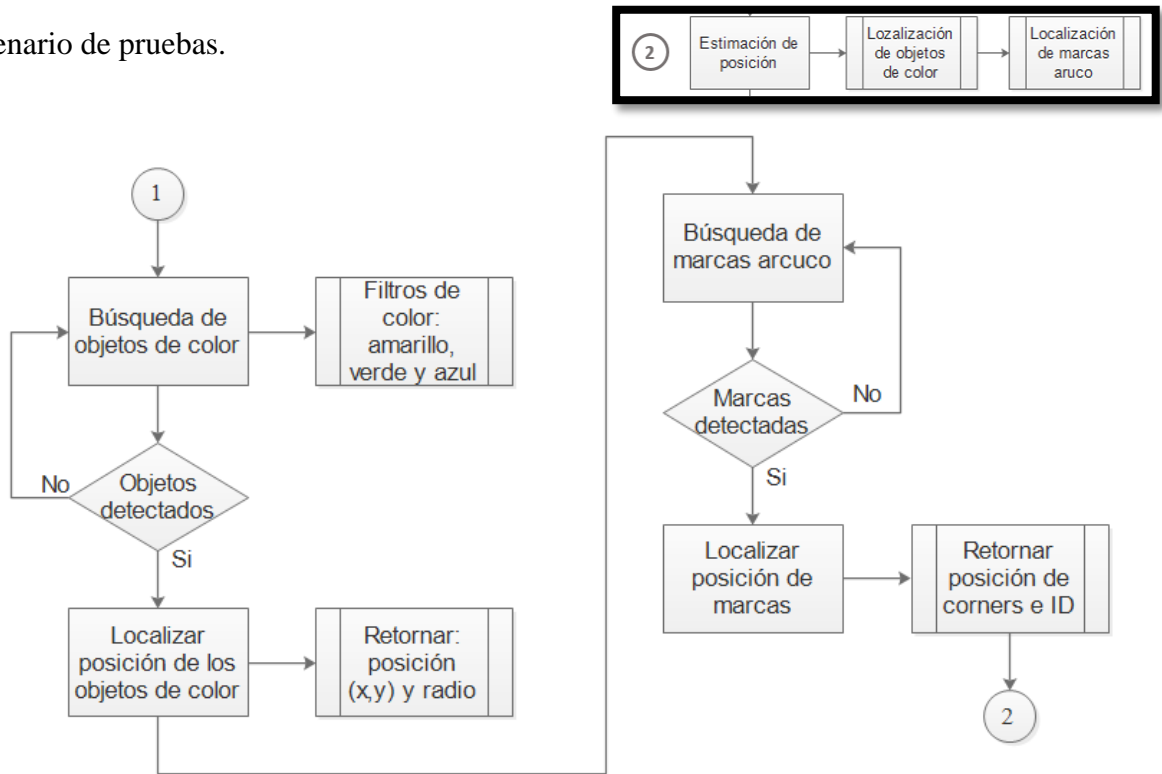


**Figura 3.19.** Diagrama de flujo de la conexión inalámbrica del sistema.  
Fuente: Elaborado por el autor

La segunda etapa abarcará el proceso de detección y localización de los objetos de color, así como las marcas Aruco ubicadas en los robots móviles y las posiciones de entrega utilizando iluminación artificial direccional. Para la detección de los objetos de color se trabajará con filtros de color utilizando el modelo de color HSV y la librería OpenCV en Python, obteniendo la posición central, radio y área de cada objeto.

Las marcas cuadradas utilizadas son creadas mediante la librería Aruco, utilizando esta librería y el modelo de color RGB se obtiene la identificación de las marcas y la posición de las esquinas de las mismas. En la Figura 3.20 se representa el diagrama de

flujo correspondiente a la estimación de posición de objetos de color y marcas en el escenario de pruebas.



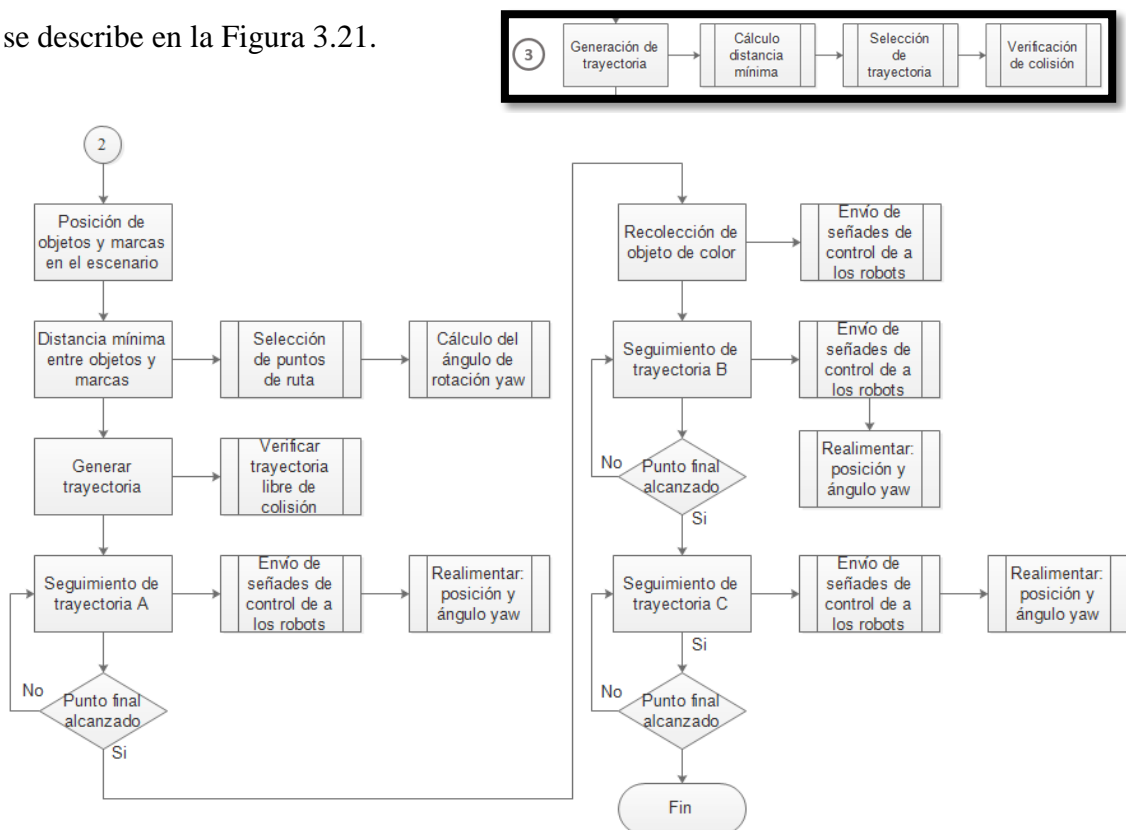
**Figura 3.20.** Diagrama de flujo de la estimación de posición de objetos y marcas.  
Fuente: Elaborado por el autor

En base a los datos de posición en 2D  $(x, y)$  obtenidos de los objetos de color y marcas aruco en la segunda etapa, se generan las trayectorias a seguir para los robots móviles considerando que deben ser libres de colisiones. Las trayectorias a generarse son de dos tipos: una línea recta o una ruta generada por el algoritmo RRT.

La trayectoria en línea recta será utilizada cuando no existan cruces o sobre posiciones entre las rutas generadas para los robots móviles, en caso de existir se utilizará una trayectoria generada por RRT para él o los robots en conflicto, y en el caso que no sea posible la generación de trayectoria el robot o robots con complicaciones, esperarán hasta la culminación de las trayectorias de los demás robots para realizar el proceso de generación de trayectoria nuevamente.

El seguimiento de trayectoria es en tiempo real, para lo cual se deduce el ángulo de rotación perpendicular al eje vertical (yaw) de cada robot en base a los datos de posición del robot y del objeto de color.

El sistema presenta realimentación en los parámetros de posición de robots y marcas, así como del ángulo yaw con el objetivo de lograr alta precisión en el seguimiento de trayectoria. El algoritmo a implementar para la generación y seguimiento de trayectoria se describe en la Figura 3.21.



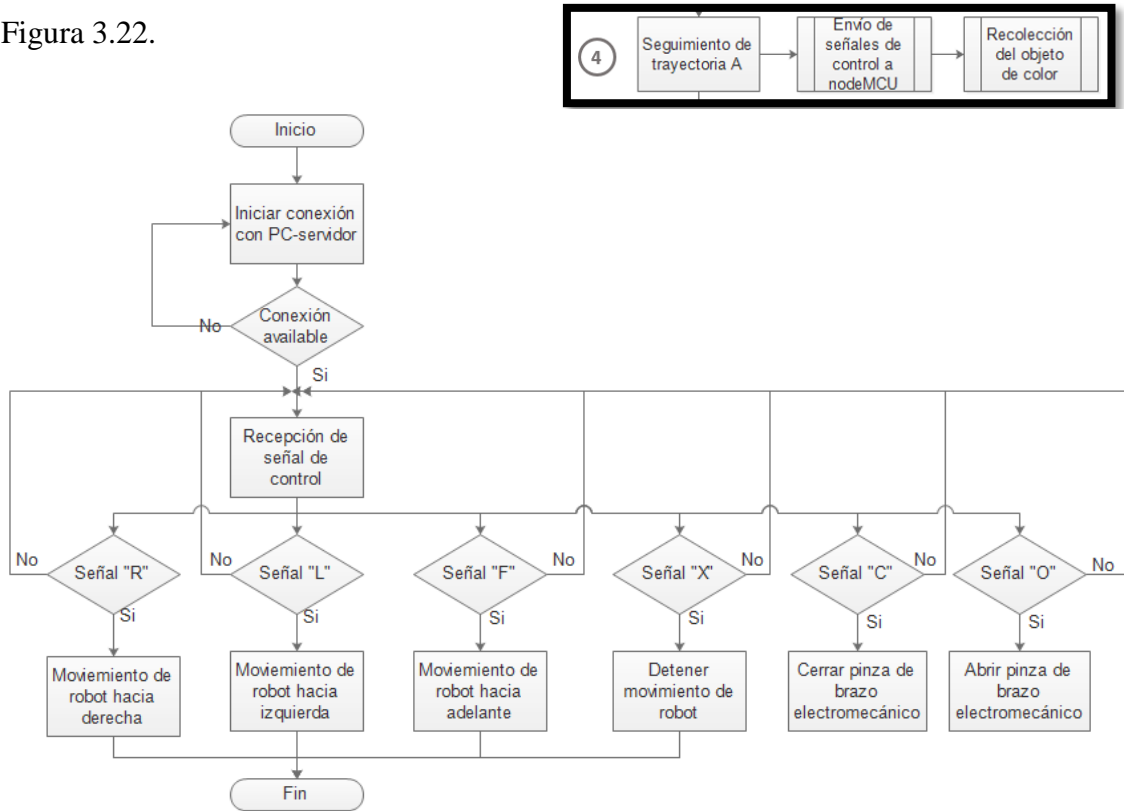
**Figura 3.21.** Diagrama de flujo del algoritmo de generación y seguimiento de trayectoria.

Fuente: Elaborado por el autor

La trayectoria A corresponde al trayecto desde las posiciones iniciales de los robots a los objetos de color correspondientes, la trayectoria B concierne al tramo desde la posición de los A hasta las marcas de entrega de los objetos de color y finalmente la trayectoria C corresponde al trayecto de retorno a las posiciones iniciales de los robots.

La etapa de navegación de los robots móviles estará basada en el valor del ángulo de rotación de cada robot, mismo que se utiliza para el control de los actuadores en los robots móviles a través del envío de mensajes desde la estación central que actúa como servidor a los clientes que son las tarjetas nodeMCU, el algoritmo a implementar se describe en la

Figura 3.22.



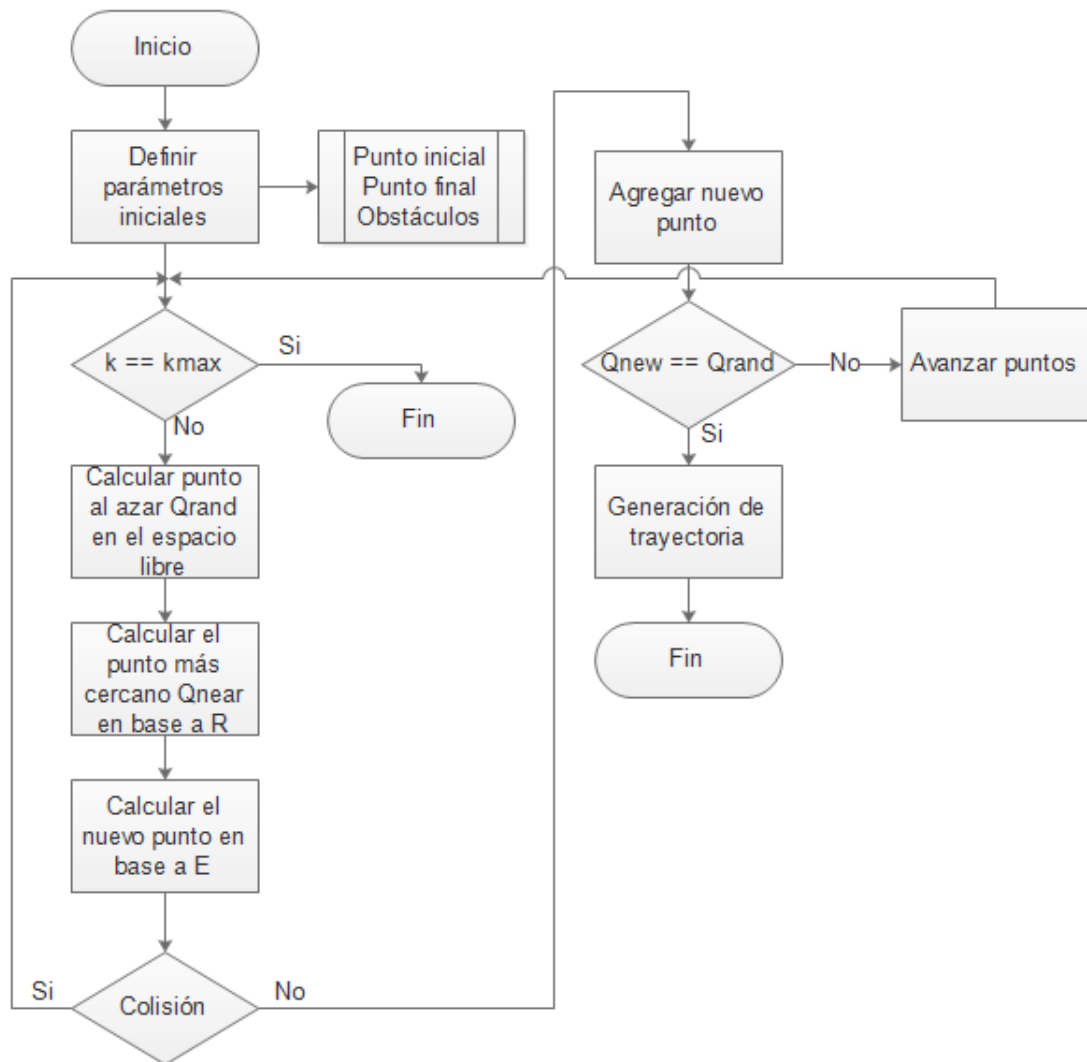
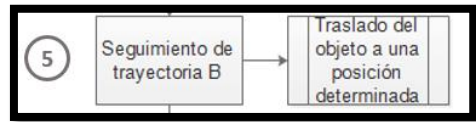
**Figura 3.22.** Diagrama de flujo del algoritmo de control para los robots.  
Fuente: Elaborado por el autor

Para etapa de recolección y transporte del objeto de color se utilizará un brazo electromecánico, accionado desde la estación central cuando el robot móvil se encuentre a una distancia determinada del objeto.

### 3.3.1.1. Algoritmo RRT

El algoritmo RRT requiere de tres parámetros iniciales para operar: punto inicial, punto final y los obstáculos. Su principio de funcionamiento consiste en la construcción de un árbol cuyas ramas crecen cubriendo el espacio libre de colisiones desde el punto

inicial hasta el final, y mediante un postprocesado establece la mejor trayectoria (Diego & Fernando, 2012).



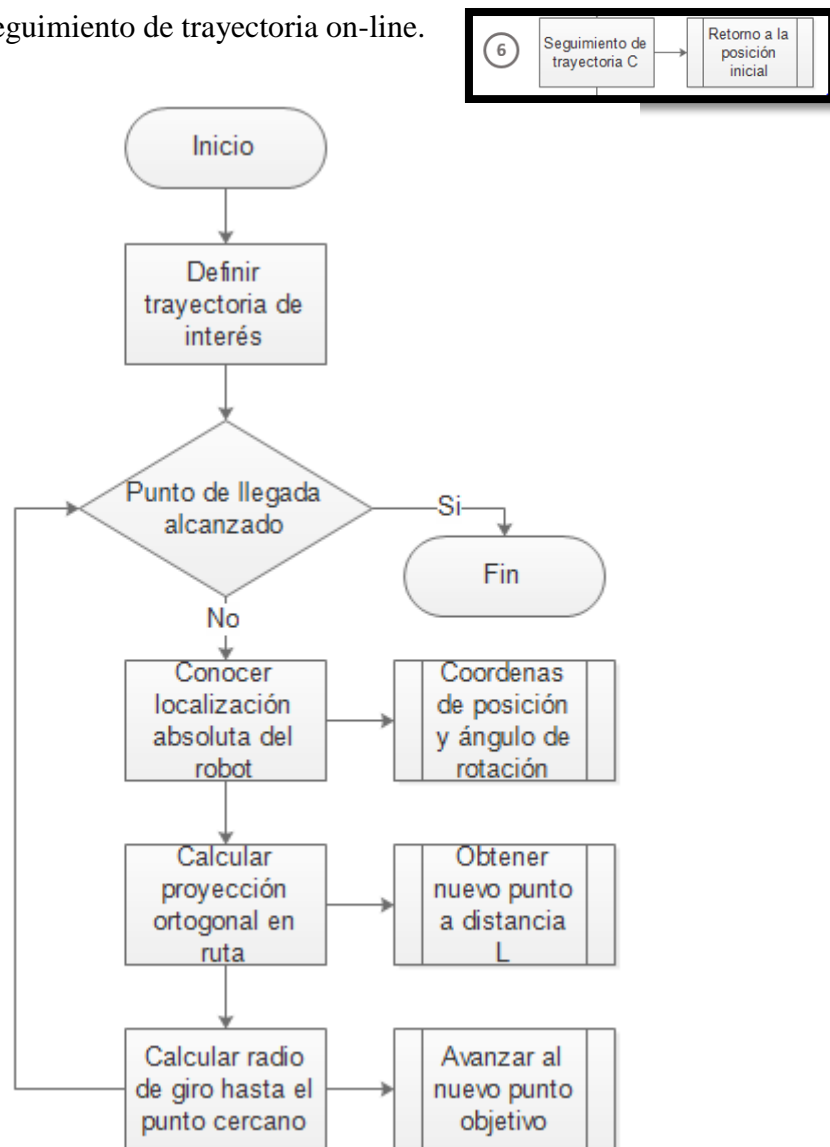
**Figura 3.23.** Diagrama de flujo del procedimiento del algoritmo RRT.  
Fuente: Elaborado por el autor

La Figura 3.23 representa el procedimiento a seguir del algoritmo RRT simple, donde  $K_{max}$  es el número de iteraciones máximas del algoritmo y en caso de no llegar al punto final en el número determinado es interrumpido el proceso.

### 3.3.1.2. Algoritmo Pure Pursuit

Para el seguimiento de trayectoria en tiempo real se utilizará un algoritmo que permita la persecución precisa de la ruta mediante la realimentación de posición y el ángulo de

rotación del robot móvil. Pure Pursuit es un algoritmo cuyo objetivo es el seguimiento de la trayectoria con un bajo error en orientación y posición, para lo cual corrige continuamente este error con respecto a la trayectoria de interés mediante la realimentación de los parámetros de localización y el ángulo de rotación del robot móvil. En la Figura 3.24 se observa el diagrama de flujo de los procesos que realiza el algoritmo pure pursuit para el seguimiento de trayectoria on-line.



**Figura 3.24.** Diagrama de flujo del procedimiento del algoritmo Pure Pursuit.  
Fuente: Elaborado por el autor

### 3.3.2. Python 3

Python es un lenguaje de programación de alto nivel, amigable, flexible y fácil de aprender. Python es software libre de propósito general implementado en todos los sistemas operativos y plataformas habituales, puede usarse y distribuirse libremente incluso para uso comercial (Python.org, 2019). Python 3 es una versión del lenguaje, que incluye un conjunto de optimizaciones haciendo a este lenguaje de programación rápido y, además, admite módulos de terceros permitiendo infinitas posibilidades a la hora de desarrollar algoritmos. Cabe recalcar que Python 3 es incompatible con versiones anteriores.



**Figura 3.25.** *Software Python.*  
Fuente: (Python.org, 2019).

Pycharm es uno de los IDE más completos para Python, ya que permite la ejecución de los scripts a medida que se desarrollan. Además, ofrece: asistencia, análisis y ayuda en la sintaxis.



**Figura 3.26.** *Software IDE PyCharm.*  
Fuente: (Python.org, 2019).

### 3.4. Análisis de costo

El análisis de costo es un recurso imprescindible para la realización del proyecto, ya que implica el recurso monetario necesario para el desarrollo e implementación del proyecto. En la Tabla 3.3 se puede observar el costo de los componentes requeridos para el desarrollo del hardware, el presupuesto para la parte de software es nulo debido a que se trabajará con programas de código abierto. En la Tabla 3.4 se presenta el costo de las herramientas necesarias para la elaboración del proyecto.

**Tabla 3.3**  
*Dispositivos electrónicos.*

Cantidad	Hardware	Valor unitario	Valor total
1	Cámara IP Hikvision	78,00	78,00
3	Driver TB6612FNG	5,40	16,20
3	NodeMCU V2	9,60	28,80
3	Chasis de dos plantas	24,00	72,00
3	Ruedas locas	3,30	9,90
3	Estructura del brazo electromecánico	9,99	29,97
3	Servomotor MG90	7,40	22,20
3	Batería LiPo	42,00	126,00
3	Protoboard	5,00	15,00
3	Módulo regulador de voltaje	5,00	15,00
36	Cables para conexión de 10cm	0,15	5,40
1	Cable de red de 15 metros	15,00	15,00
3	Cilindros de espuma flex	1,00	3,00
1	Escenario de prueba	100,00	100,00
1	Pedestal de soporte para cámara IP	40,00	40,00
Valor total del presupuesto para hardware (USD)			536,47

Fuente: Elaborado por el autor



**Tabla 3.4**  
*Herramientas eléctricas y materiales varios.*

Cantidad	Descripción	Valor unitario	Valor total
1	Pinza ponchadora	15,00	15,00
1	Probador para cables de red	12,00	12,00
1	Multímetro	45,00	45,00
1	Cargador imax B6 para baterías LiPo	40,00	40,00
1	Juego de desarmadores	15,00	15,00
1	Impresión de marcas	3,00	3,00
Valor total del presupuesto para hardware (USD)			130,00

Fuente: Elaborado por el autor

Es necesario añadir el costo de horas hombre requeridas para el desarrollo de este proyecto, el precio se estima a partir del salario básico unificado que en el 2019 se encuentra establecido en \$394, con base a esta cifra y al número de horas laborables en un mes se tiene que el costo por hora hombre es de \$2.46. En la Tabla 3.5 se visualiza el número estimado de horas requeridas para el desarrollo del proyecto de titulación y su costo total.

**Tabla 3.5**  
*Mano de obra.*

Horas	Descripción	Costo hora	Valor total
54	Investigación teórica.	2,46	130,68
48	Desarrollo de algoritmo para reconocimiento de color.	2,46	116,16
48	Desarrollo de algoritmo para reconocimiento de marcas.	2,46	116,16
6	Ensamble de los robots móviles.	2,46	14,52
72	Desarrollo de algoritmo para navegación de los robots.	2,46	174,24
72	Integración de algoritmos	2,46	174,24
30	Pruebas de funcionamiento	2,46	72,60
120	Parte escrita	2,46	290,4
Valor total del presupuesto para mano de obra (USD)			1089,00

Fuente: Elaborado por el autor

Adicional a los valores monetarios calculados en las tablas anteriores (Tabla 3.3, Tabla 3.4 y Tabla 3.5) se añade un 20% por situaciones imprevistas.

En la Tabla 3.6 se realiza el análisis total del presupuesto requerido para el desarrollo del proyecto de titulación.

**Tabla 3.6**  
*Presupuesto para el desarrollo del proyecto.*














Descripción	Costo	20% adicional
<b>Hardware</b>	536,47	643,76
<b>Herramientas</b>	130,00	156,00
<b>Mano de obra</b>	1089,00	1306,80
Presupuesto total para el proyecto		2106,57

Fuente: Elaborado por el autor

La propuesta para desarrollar e implementar este proyecto de acuerdo con la Tabla 3.6 tiene un costo de \$2106,57, presupuesto que será financiado personalmente.

### 3.5. Análisis de tiempo

El presente proyecto de titulación tiene una duración aproximada de cinco meses, el proyecto está dividido en seis etapas, como se puede observar en el diagrama de grant de la Figura 3.27.

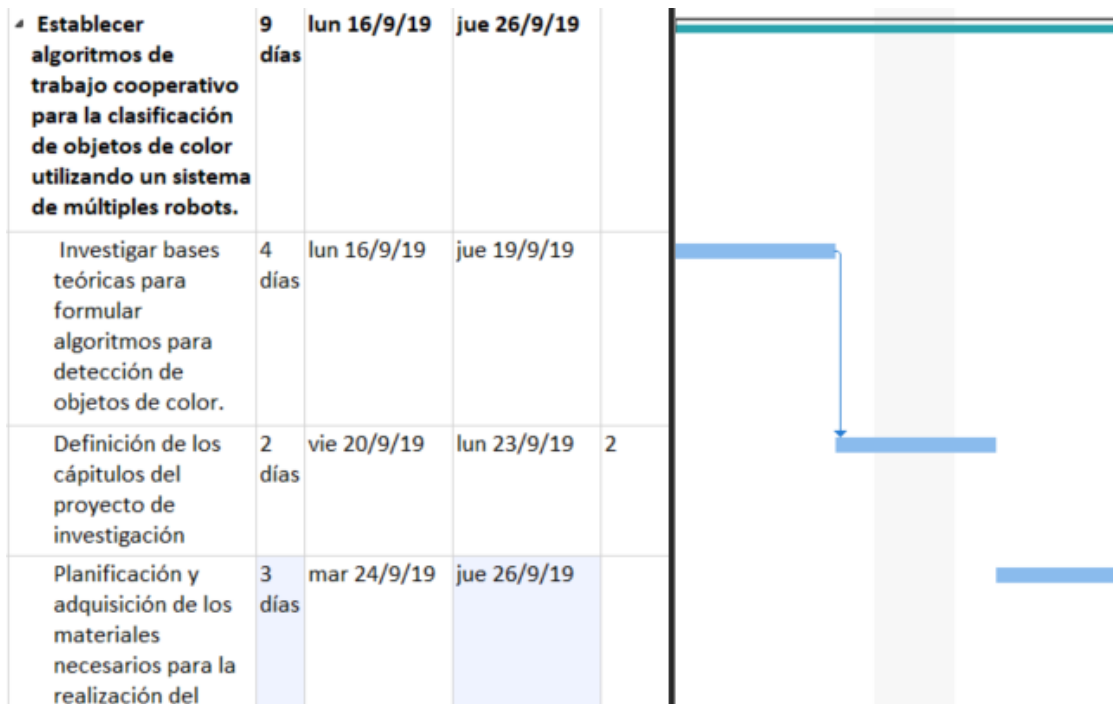
1		Establecer algoritmos de trabajo cooperativo para la clasificación de objetos de color utilizando un sistema de múltiples robots.	9 días	lun 16/9/19	jue 26/9/19
2		Investigar bases teóricas para formular algoritmos para detección de objetos de color.	4 días	lun 16/9/19	jue 19/9/19
3		Definición de los capítulos del proyecto de investigación	2 días	vie 20/9/19	lun 23/9/19
4		Planificación y adquisición de los materiales necesarios para la realización del proyecto	3 días	mar 24/9/19	jue 26/9/19
5		Formular algoritmos de visión artificial para la detección de objetos de color mediante una	8 días	mar 24/9/19	jue 3/10/19
6		Sistema de navegación	31 días	vie 4/10/19	vie 15/11/19
7		Formular algoritmos para navegación autónoma de los robots en el escenario en base a los parámetros obtenidos por	15 días	vie 4/10/19	jue 24/10/19
8		Formular algoritmos para el control de la pinza de los robots	8 días	vie 25/10/19	mar 5/11/19
9		Pruebas de funcionamiento y corrección de algoritmos	8 días	mié 6/11/19	vie 15/11/19
10		Sistema de robots cooperativos	15 días	mié 6/11/19	mar 26/11/19
11		Corrección de los algoritmos de detección y navegación.	15 días	mié 6/11/19	mar 26/11/19
12		Analizar el desempeño del sistema en tiempo real para evaluar la velocidad y precisión del mismo en el proceso de detección y transporte de objetos ante los cambios de iluminación en el	5 días	lun 2/12/19	vie 6/12/19
13		Realizar la parte escrita del proyecto de investigación	91 días	lun 16/9/19	lun 20/1/20

**Figura 3.27.** Planificación de tareas del proyecto.

Fuente: Elaborado por el autor

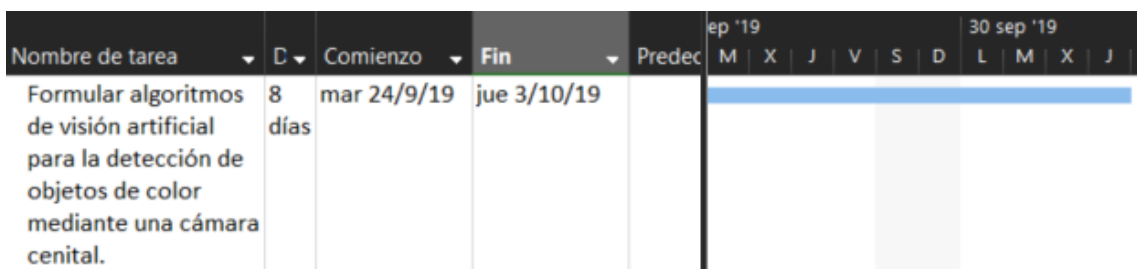
La primera etapa abarca la investigación teórica de algoritmos de visión artificial y navegación autónoma de los robots móviles, así como la adquisición de los materiales y

elementos necesarios para el desarrollo del proyecto. Como se observa en la Figura 3.28, esta etapa tiene una duración de nueve días.



**Figura 3.28.** Diagrama de grant etapa 1.  
Fuente: Elaborado por el autor

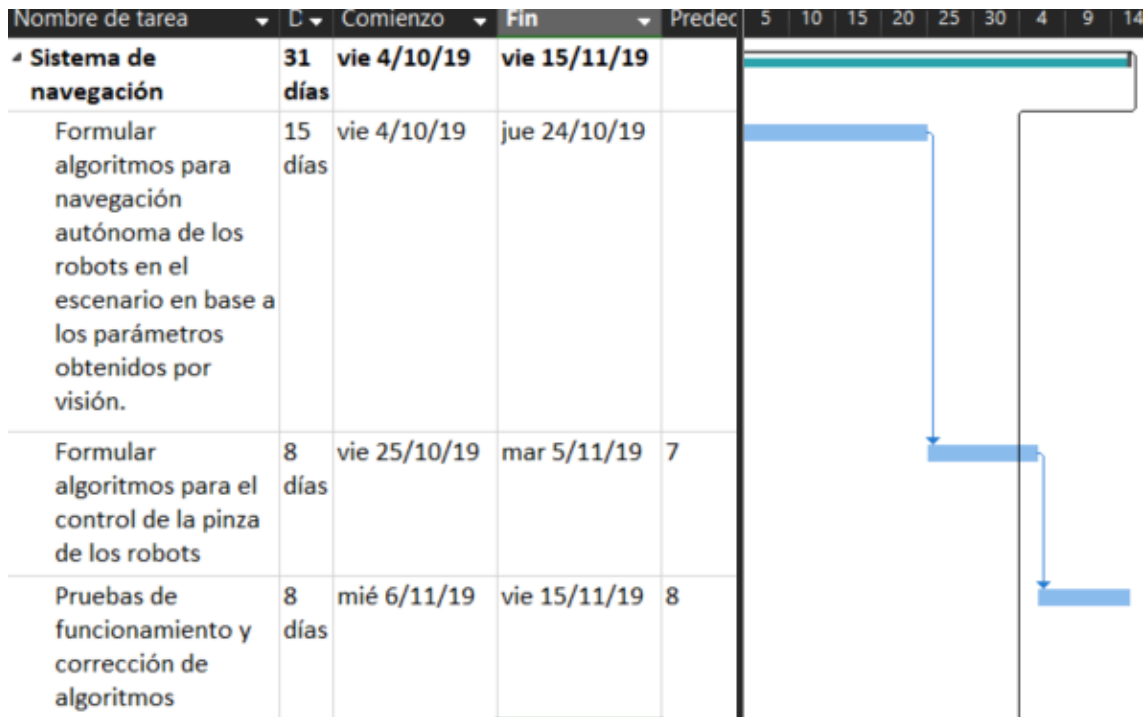
La segunda etapa comprende el desarrollo de algoritmos para detectar objetos de color y marcas, y la familiarización con la cámara IP e instalación de los programas necesarios. Como se observa en la Figura 3.29, esta etapa tiene una duración de ocho días.



**Figura 3.29.** Diagrama de grant etapa 2.  
Fuente: Elaborado por el autor

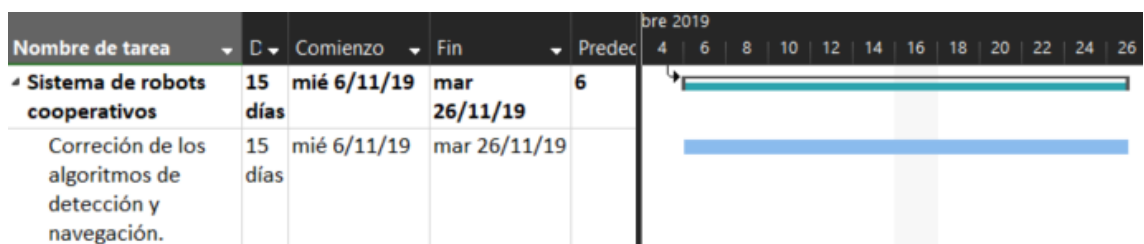
La tercera etapa comprende el desarrollo de algoritmos para la navegación de los robots móviles a través de los datos adquiridos en la etapa anterior, así como la

formulación de algoritmos para el control del brazo electromecánico y las pruebas respectivas de funcionamiento. Como se observa en la Figura 3.30, esta etapa tiene una duración de quince días.



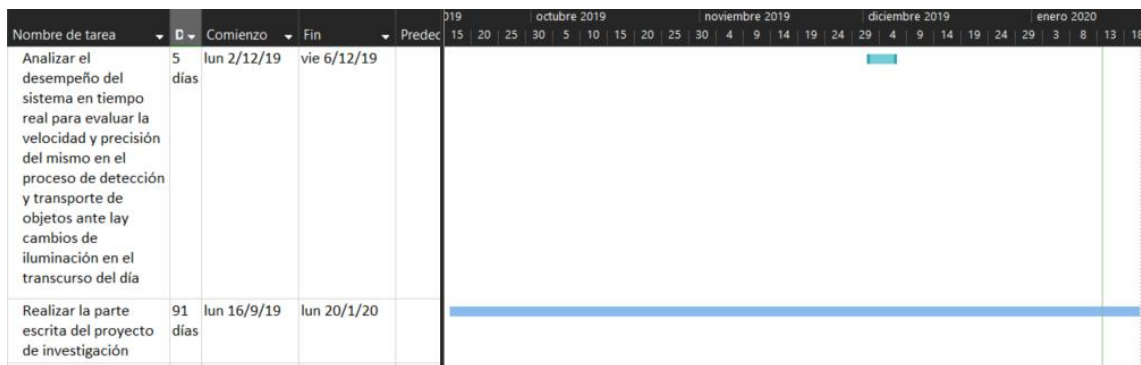
**Figura 3.30.** Diagrama de grant etapa 3.  
Fuente: Elaborado por el autor

La cuarta etapa comprende la corrección de los algoritmos desarrollados en las etapas dos y tres, al finalizar esta etapa el sistema de múltiples robots móviles cooperativos estará completamente funcional. Como se observa en la Figura 3.31, esta etapa tiene una duración de quince días, tiempo requerido para realizar pruebas y correcciones.



**Figura 3.31.** Diagrama de grant etapa 4.  
Fuente: Elaborado por el autor

En la quinta etapa se realiza pruebas de desempeño del sistema de robots cooperativos con el objetivo de evaluar la velocidad y precisión de los robots en las tareas de transporte de objetos, así como la detección y localización de robots y objetos en el escenario de pruebas. Finalmente, la sexta etapa comprende el desarrollo del documento del proyecto. Como se observa en la Figura 3.32, la quinta etapa tiene una duración de 5 días, tiempo requerido para realizar pruebas y observaciones. La duración de la sexta etapa corresponde al tiempo total estimado para el desarrollo del proyecto, el cual es noventa y un días.



**Figura 3.32.** Diagrama de Gantt etapa 5 y 6.  
Fuente: Elaborado por el autor

### 3.6. Ventajas del proyecto

El presente proyecto se enfoca en desarrollar a escala un sistema de robots móviles cooperativos que realicen el proceso de transporte y clasificación de objetos de color con el fin de utilizar el sistema en un proceso real de transporte de sustancias químicas y reemplazar la intervención humana evitando accidentes. A continuación, enumera las principales ventajas del proyecto:

- El hardware requerido para el proyecto es de costo accesible y de fácil acceso en el mercado nacional.

- El software utilizado es open source, y puede ser utilizado para uso comercial sin restricciones.
- La conectividad del sistema es inalámbrica y presenta una estructura de control central, por lo que los robots móviles presentan un diseño flexible.
- La autonomía del sistema es de 78 minutos con baterías LiPo de 2200 mAh, este tiempo puede aumentar trabajando con baterías de mayor capacidad.
- La transmisión de video es en tiempo real, por lo que el sistema no requiere de una tarjeta o disco externo para almacenar datos.

# CAPÍTULO IV

## 4. RESULTADOS

### 4.1. Introducción

En el presente capítulo se describe el proceso de construcción de los robots móviles a nivel de hardware, así como la implementación de algoritmos para detectar, identificar y localizar objetos de color y marcas. Además, algoritmos de control para la navegación de los robots cooperativos dentro del escenario de pruebas mientras realizan el proceso de clasificación y transporte de objetos de color.

Igualmente, en este capítulo se expondrán los resultados obtenidos al implementar el hardware y software del sistema, explicando las diferencias existentes entre el planteamiento teórico y los cambios necesarios en el sistema cooperativo implementado para cumplir con los objetivos planteados.

### 4.2. Implementación de hardware

La construcción de los robots móviles consiste en la integración de todos los componentes de hardware seleccionados anteriormente, ver Tabla 4.1, en cada robot móvil del sistema cooperativo.

**Tabla 4.1**

*Hardware implementado en cada robot móvil del sistema.*

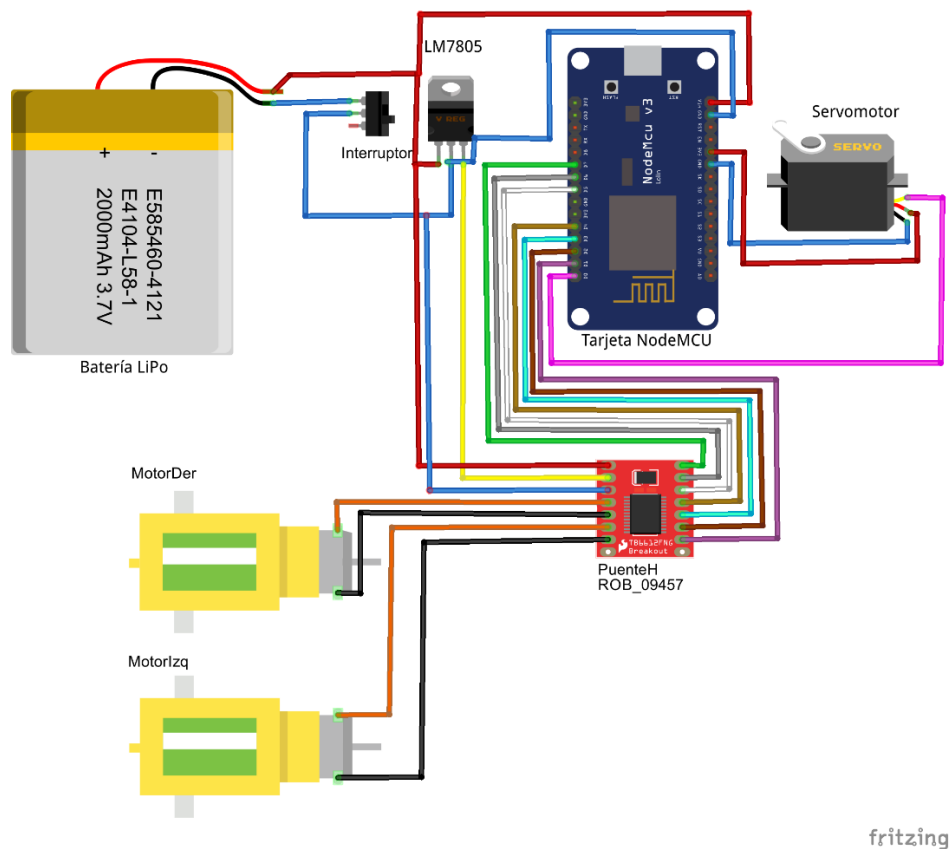
Aplicación	Elementos	Tarea
Adquisición	Cámara IP cenital	Reconocimiento de objetos de color y marcas
Control y procesamiento	NodeMCU V2	Tarjeta de comunicación inalámbrica WiFi. Control de actuadores.
	Driver TB6612FNG	Control de motores DC
Potencia	Motores DC	Desplazamiento del robot
	Servomotores MG90	Control del brazo electromecánico
	Regulador de Voltaje LM7805	Regulación de voltaje a 5(V)
	Baterías LIPO	Fuente de voltaje DC

Fuente: Elaborado por el autor



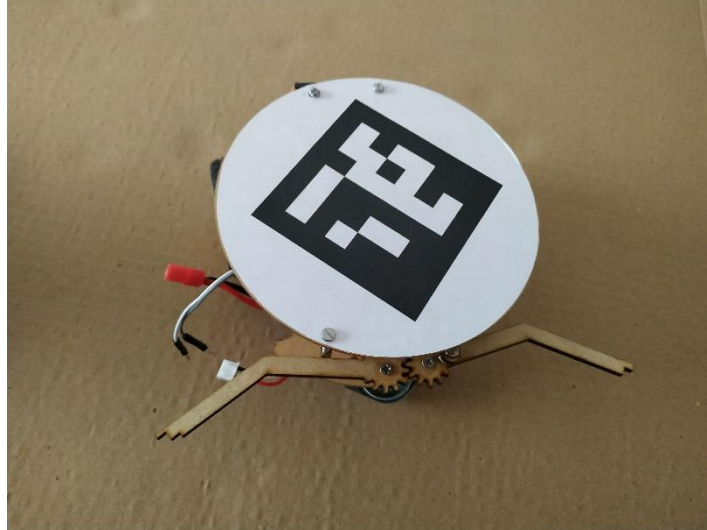
La conexión de los elementos que conforman los robots móviles se realiza por partes como se observa en la Figura 4.1:

- Primero se conecta la etapa de regulación de voltaje, es decir se conecta el encapsulado LM7805 con la batería LiPo y el interruptor respetando polaridad.
- Segundo se conecta la alimentación para la tarjeta de desarrollo NodeMCU y para el controlador de los motores TB6612FNG.
- Tercero se conecta los pines de control y señal para cada motor entre la tarjeta nodeMCU y el driver TB6612FNG.
- Cuarto se conectan los motores al puente TB6612FNG, y se revisan las conexiones.
- Finalmente se conecta el servomotor para control del brazo electromecánico.



**Figura 4.1.** Diagrama de conexión de los robots móviles.  
Fuente: Elaborado por el autor

En las figuras, Figura 4.2 y Figura 4.3 se observa la distribución de los elementos acoplados en el chasis de los robots móviles, obteniendo una estructura que realiza la búsqueda y clasificación de objetos de color en un entorno de pruebas sin riesgo de desconexión o fallos en su circuitería.

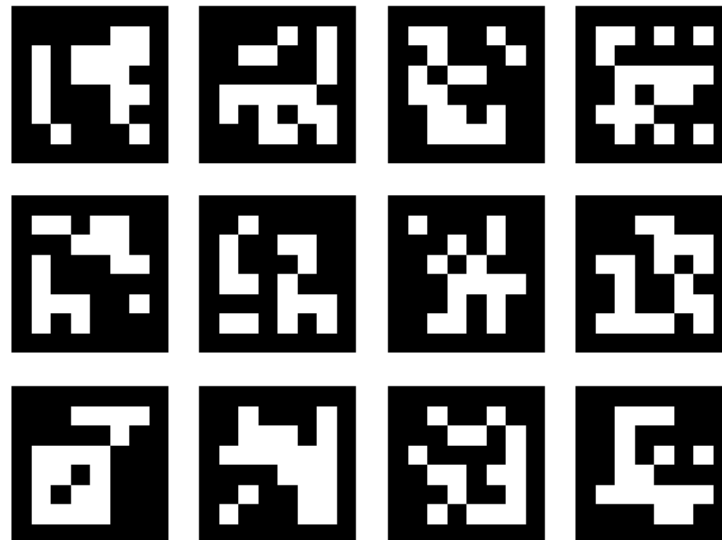


**Figura 4.2.** *Robots Móviles vista superior.*  
Fuente: Elaborado por el autor



**Figura 4.3.** *Robots Móviles vista lateral.*  
Fuente: Elaborado por el autor

Sobre el chasis de los robots se colocan marcas de identificación, ver Figura 4.2, con las cuales se obtendrá la posición y orientación de los robots dentro del entorno de pruebas, las marcas a utilizar se observan en la Figura 4.4.



**Figura 4.4.** *Marcas aruco de identificación.*

Fuente: Elaborado por el autor

En la Tabla 4.2 se aprecia la distribución de las marcas para los robots, las posiciones de entrega y las posiciones de origen.

**Tabla 4.2**

*Distribución de las marcas aruco de identificación.*

Marca	Identificación
Primera	Robot uno
Segunda	Robot dos
Tercera	Robot tres
Cuarta	Posición de entrega color verde
Quinta	Posición de entrega color azul
Sexta	Posición de entrega color amarillo
Séptima	Posición de origen orden aleatorio
Octava	Posición de origen orden aleatorio
Novena	Posición de origen orden aleatorio

Fuente: Elaborado por el autor

La cámara cenital se coloca en medio del entorno de pruebas a una altura determinada por la capacidad de detección de los objetos de color y las marcas. Para lo cual se realiza pruebas de detección a tres diferentes alturas (1.60 m, 1.50m y 1.40 m), utilizando la estructura de la Figura 4.5.

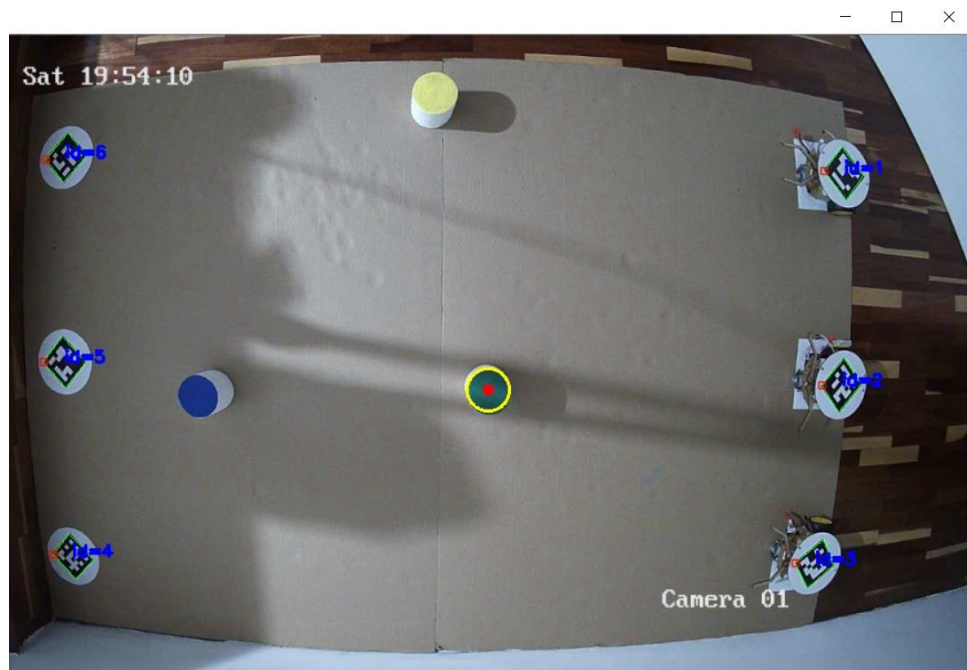


**Figura 4.5.** *Estructura para la cámara cenital.*  
Fuente: Elaborado por el autor

A 1.60 metros la cámara no detecta la marca tres y los objetos de color como se observa en la Figura 4.6, en cambio a la altura de 1.50 metros la cámara presenta problemas de detección en los objetos de color como se muestra en la Figura 4.7 y finalmente a 1.40 metros de altura, ver en la Figura 4.8., la cámara no presenta inconvenientes en la detección de marcas y colores siendo selecciona esta altura para desarrollar el proyecto de titulación.



**Figura 4.6.** Colocación de la cámara IP a una altura de 1.60 metros.  
Fuente: Elaborado por el autor



**Figura 4.7.** Colocación de la cámara IP a una altura de 1.5 metros.  
Fuente: Elaborado por el autor



**Figura 4.8.** Colocación de la cámara IP a una altura de 1.40 metros.  
Fuente: Elaborado por el autor

El tamaño del entorno de pruebas está relacionado con ubicación de la cámara, siendo su tamaño máximo 2.10 x 1.5 metros, ver Figura 4.9.



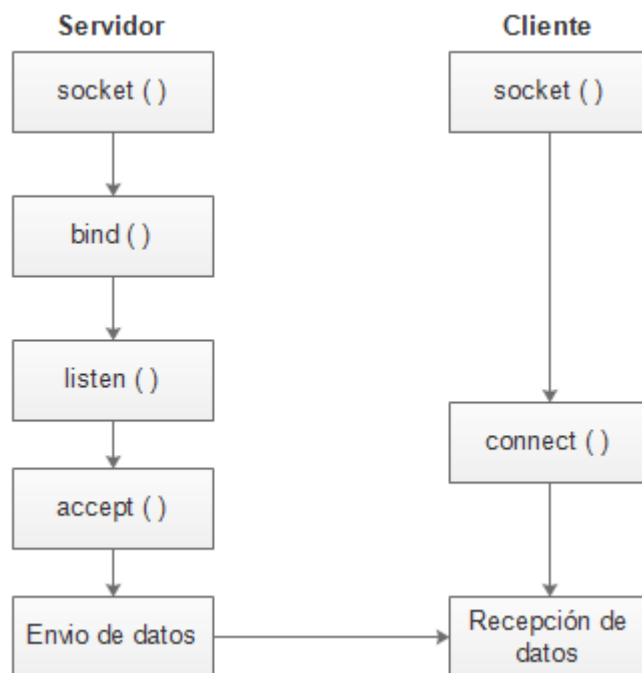
**Figura 4.9.** Entorno de pruebas.  
Fuente: Elaborado por el autor.

### 4.3. Desarrollo del sistema de adquisición de datos

La adquisición de datos se realizó empleando un computador portátil que cuenta con el intérprete Pycharm y el lenguaje de programación Python. Los datos de la cámara son registrados y procesados a través de comunicación WiFi. A continuación, se presenta las configuraciones realizadas para establecer la comunicación, la adquisición de datos y el control de actuadores.

#### 4.3.1. Configuración de la estación central de control

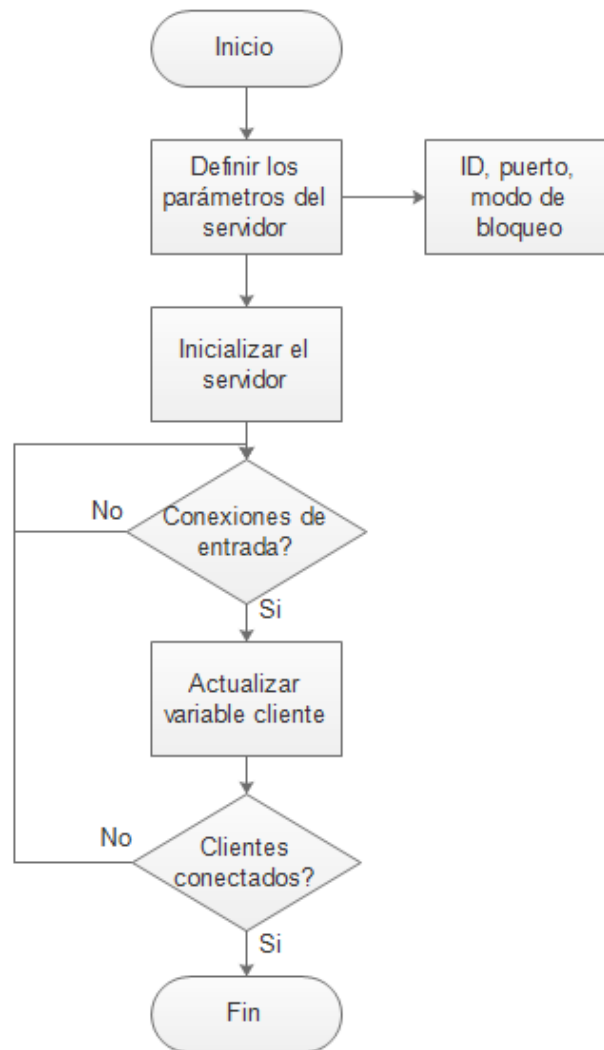
La estación central de control del sistema cooperativo es programada en el IDE Pycharm con el lenguaje de programación Python versión 3.7. Se estableció una comunicación TCP cliente/servidor entre la estación central de control y la tarjeta nodeMCU a través del módulo socket de Python, donde la estación de control actúa como servidor escuchando los mensajes entrantes de los clientes, las tarjetas nodeMCU, en la Figura 4.10 se observa el diagrama cliente/servidor utilizado.



**Figura 4.10.** Diagrama Cliente/Servidor del sistema.  
Fuente: Elaborado por el autor



En la Figura 4.11 se muestra el diagrama de flujo que representa la conexión del servidor con los clientes. Inicialmente se definen los parámetros del servidor para luego ser inicializado, el siguiente paso es la conexión de todos los clientes con él servidor, y una vez conectados todos los clientes se procede a la etapa de configuración de la cámara IP.



**Figura 4.11.** Diagrama de flujo de la comunicación Server del sistema.  
Fuente: Elaborado por el autor

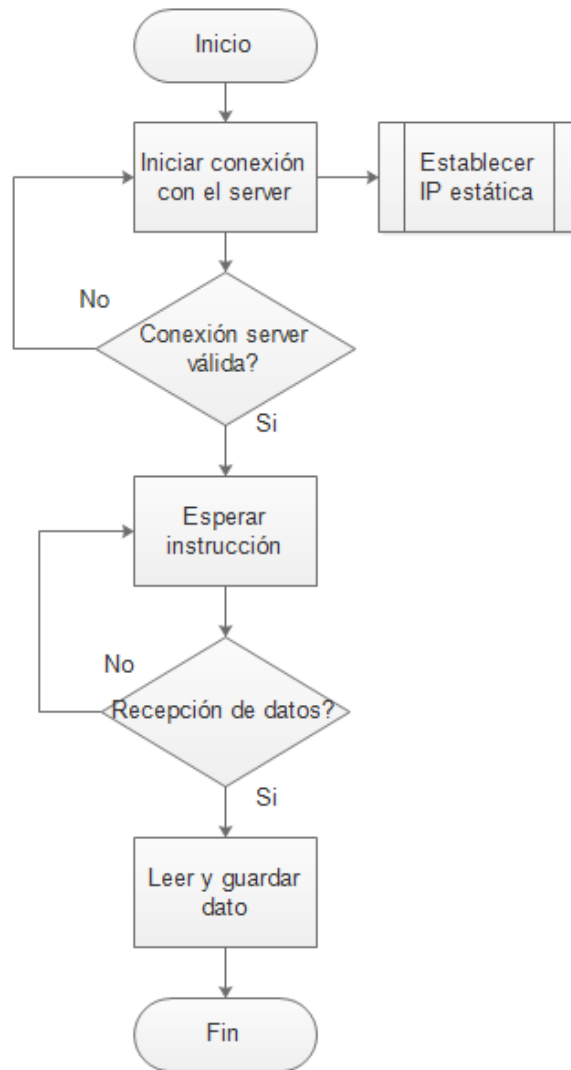
Para la identificación de objetos de color y marcas se utiliza la librería OpenCV versión 3.0 para Python, la misma que es utilizada en aplicaciones de visión artificial en los campos investigativo, empresarial y educativo debido a su compatibilidad con diversos



sistemas operativos (Linux, Windows, Mac OS, IOS, entre otros) y lenguajes de programación (C #, Java, C y Python).

#### 4.3.2. Configuración de la tarjeta nodeMCU

Esta tarjeta de desarrollo es programada en el IDE de Arduino, y está conectada inalámbricamente a la estación central de procesamiento por comunicación WiFi. La tarjeta nodeMCU trabaja a una velocidad de procesamiento de 160 MHz. A continuación, en la figura se describe la recepción de datos realizada por esta tarjeta con respecto a la estación de control.



**Figura 4.12.** Diagrama de flujo del proceso de comunicación de los clientes.

Fuente: Elaborado por el autor

Adicional la tarjeta nodeMCU se encarga de establecer las IP estáticas correspondientes a los robots móviles, en la Tabla 4.3 se observa las IP seleccionadas para cada robot.

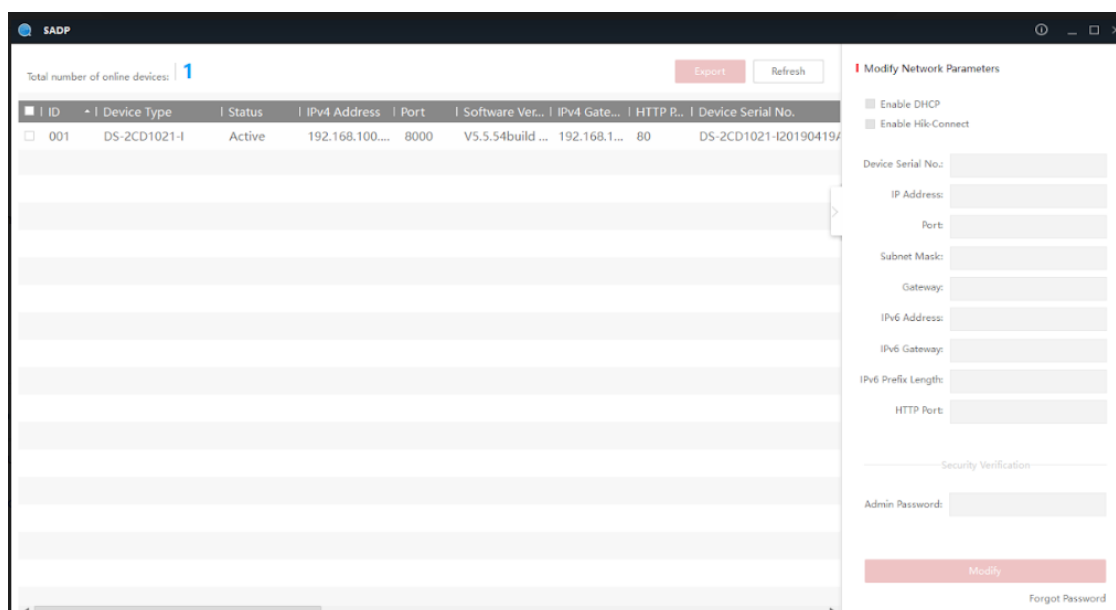
**Tabla 4.3**  
*IP estáticas para los robots móviles.*

Robot	IP
<b>Robot uno</b>	192.168.100.111
<b>Robot dos</b>	192.168.100.112
<b>Robot tres</b>	192.168.100.111

Fuente: Elaborado por el autor

### 4.3.3. Configuración de la cámara IP

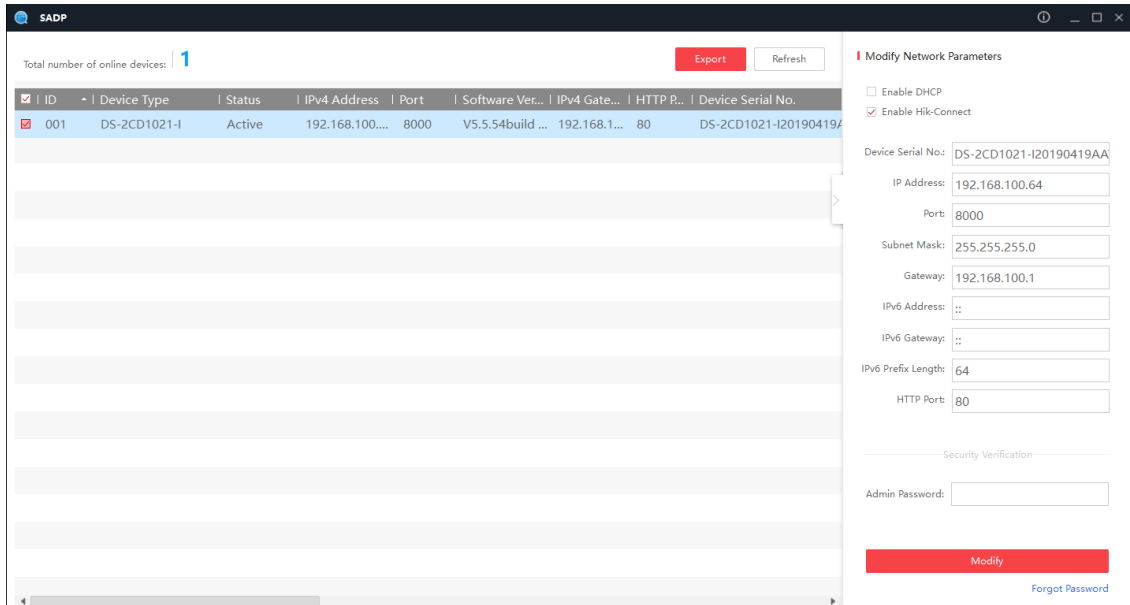
La cámara IP Hikvision por defecto está configurada con la IP 192.0.0.64, siendo habitualmente inaccesible desde la mayoría de redes. Para poder acceder a su configuración se debe utilizar el software SADP (Search Active Device Protocol) de Hikvision, mediante el cual se puede visualizar la cámara en la red y modificar sus configuraciones, ver la Figura 4.13.



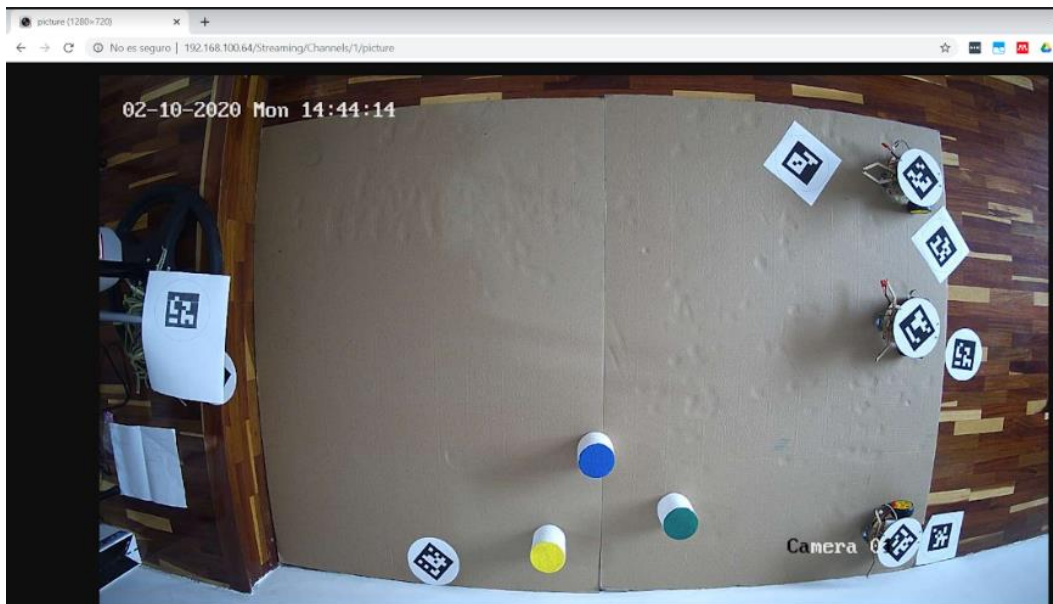
**Figura 4.13.** Visualización de la cámara IP en el software SADP.

Fuente: Elaborado por el autor

En SADP realizamos el cambio de la IP por defecto a la IP estática 192.168.100.64, ver la Figura 4.14, una vez realizado el cambio se puede acceder desde cualquier navegador a la cámara IP, ver Figura 4.15.



**Figura 4.14.** Configuración de la IP de la cámara en el software SADP.  
Fuente: Elaborado por el autor



**Figura 4.15.** Visualización de la cámara IP en el navegador.  
Fuente: Elaborado por el autor

#### **4.4. Desarrollo del sistema de procesamiento**

La estación central obtiene los parámetros de posición de los objetos de color y marcas ubicados en el entorno de prueba a través del procesamiento de la información captada por la cámara IP. Al iniciar la ejecución de los algoritmos del sistema, se estableció un tiempo de espera de cinco segundos para que la cámara IP se estabilice. Con los valores obtenidos de las posiciones de los objetos de color, marcas y robots en el entorno de prueba se implementa algoritmos para la planificación y generación de trayectorias A, B, y C libre de colisiones.

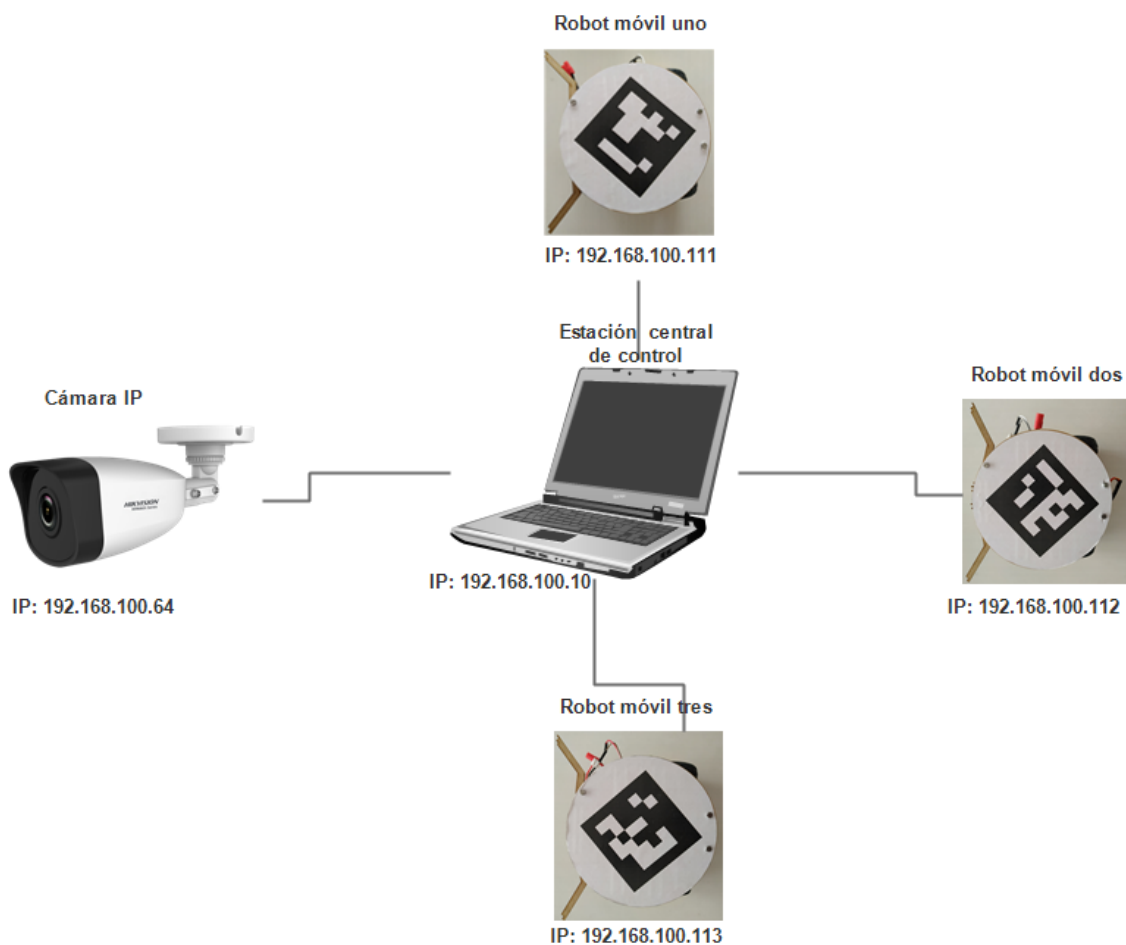
Donde la trayectoria A comprende el tramo que debe recorrer un robot hacia un objeto de color, la trayectoria B representa el trayecto recorrido por el robot con el objeto de color hacia la posición de entrega del mismo y finalmente la trayectoria C comprende el recorrido que debe realizar el robot hacia su posición de origen.

Una vez generadas las trayectorias, se procede a la etapa de seguimiento donde la estación central se encarga del control de los actuadores en los robots móviles mediante el envío de datos correspondientes con las posiciones en tiempo real de los robots, marcas de entrega y marcas de origen a las tarjetas nodeMCU acopladas en los robots móviles.

Las tarjetas nodeMCU reciben caracteres desde la estación central para el control de los motores, siendo “F” el carácter dispuesto para que el robot avance, “L” el carácter que indica el giro hacia la izquierda del robot, “R” que señala el giro hacia la derecha del robot y “X” para el paro total de los motores. También recibe caracteres para el control del brazo electromecánico donde, el carácter “C” dispone el cierre de la pinza y “O” estipula la apertura de la pinza del brazo electromecánico, ver Figura 3.22.

#### 4.4.1. Control centralizado

El sistema cooperativo presenta un control centralizado bajo los conceptos de cooperación, comunicación y coordinación con una arquitectura de red en estrella, donde existe una estación central que se conecta inalámbricamente a los robots móviles y a la cámara IP, ver la Figura 4.16. La estación central se encarga además de la implementación, ejecución y procesamiento de los algoritmos de localización, planificación, seguimiento y captura.



**Figura 4.16.** *Arquitectura del control del sistema cooperativo.*  
Fuente: Elaborado por el autor

#### **4.4.2. Tareas de cooperación y coordinación entre los robots móviles**

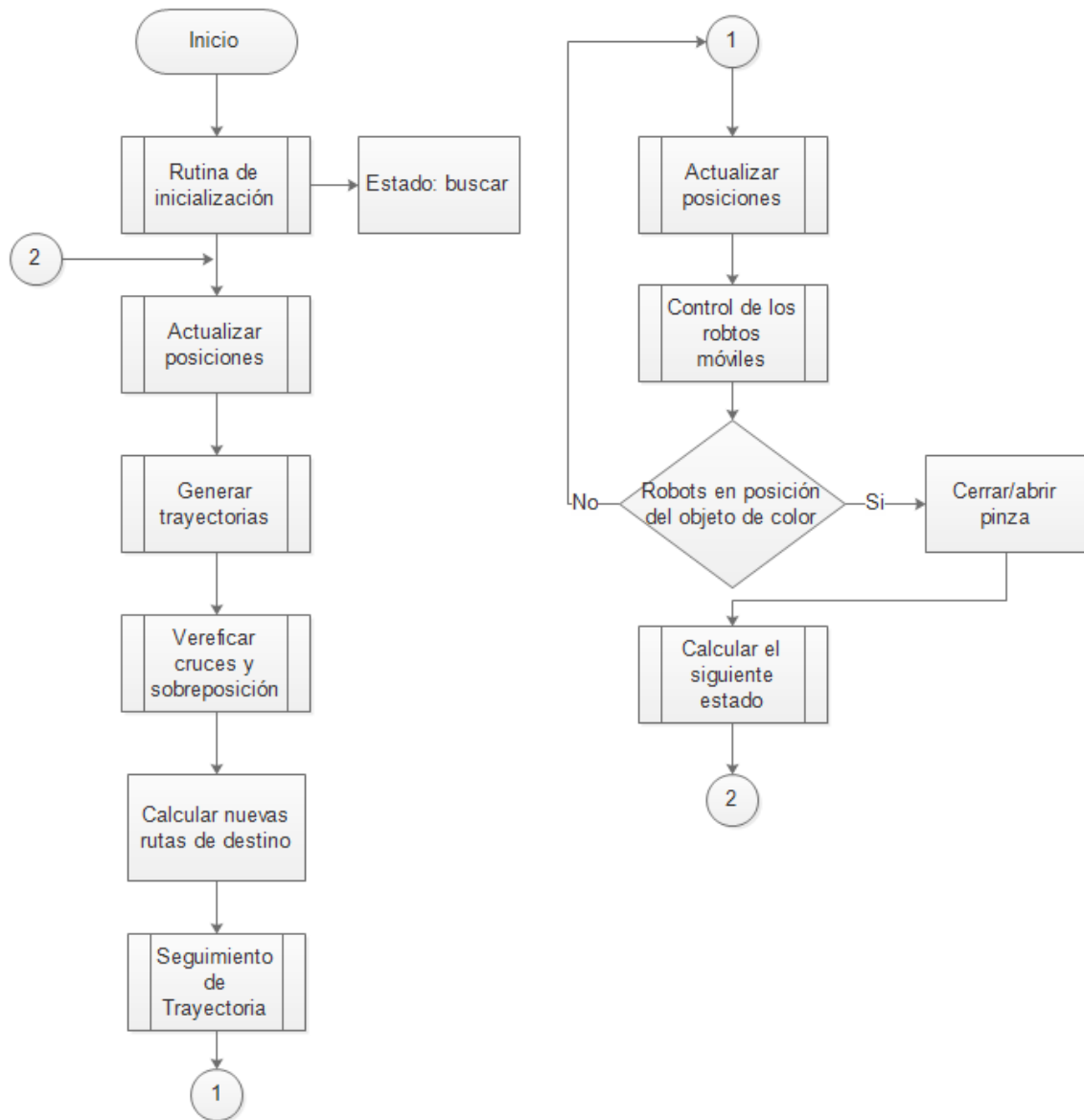
Los algoritmos del sistema permiten realizar la búsqueda de todos los objetos de color dentro del escenario de pruebas y obtener sus posiciones. Finalizado el proceso de distribución de los primeros objetos de color, el sistema tiene la capacidad de volver a realizar una nueva rutina de detección y distribución de objetos.

La utilización de tres robots móviles bajo el control centralizado permite reducir el tiempo de distribución y clasificación de los objetos de color, debido a que si se utilizara un solo robot móvil se clasificaría un solo objeto de color en un tiempo determinado, frente al caso de utilizar tres robots para clasificar tres objetos de color en el mismo intervalo de tiempo.

Las tareas de coordinación implementadas para los robots móviles permiten que por medio del sistema de comunicación establecido los robots inicien su movimiento al mismo instante, dirigiéndose hacia los objetos de color determinados por la estación de control central. Además, con la información del entorno capturada por la cámara cenital el sistema crea trayectorias de navegación para los robots, analizando posibles casos de colisión y realizando validaciones en base a la mínima distancia recorrida para seleccionar el robot que debe avanzar y el que debe detenerse evitando así colisiones en las trayectorias.

#### **4.5. Desarrollo de algoritmos del sistema**

El sistema cooperativo para la clasificación y distribución de objetos de color aplicando técnicas de visión artificial, está compuesto por los procesos y rutinas de la Figura 4.17 desarrollados para la estación central de control del sistema.

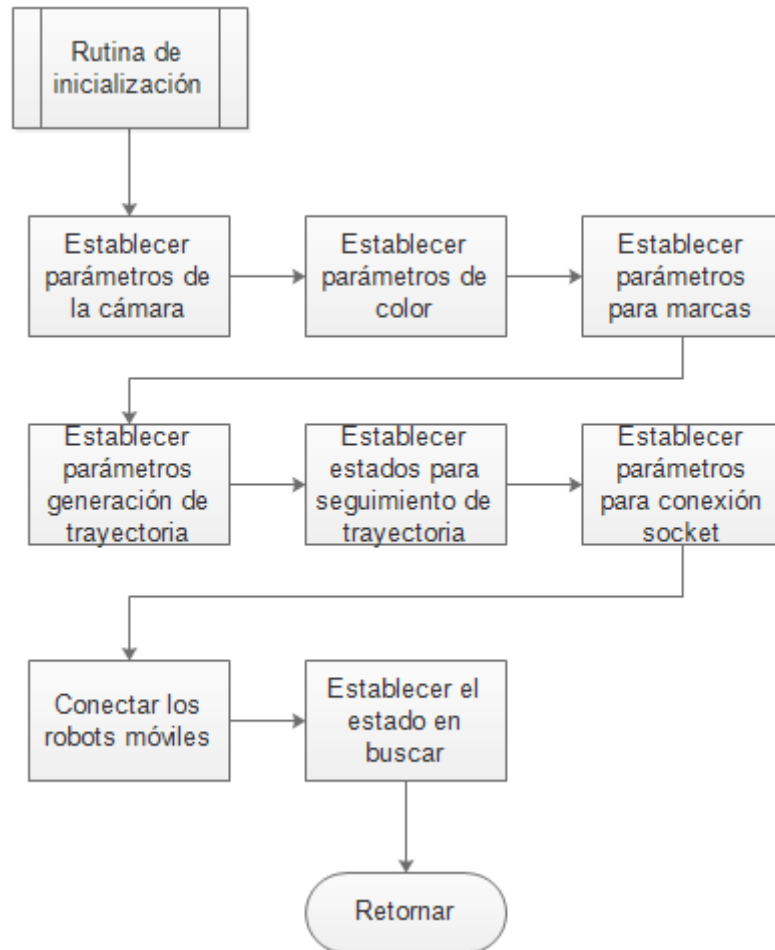


**Figura 4.17.** Diagrama de flujo del software del sistema de desarrollar.  
Fuente: Elaborado por el autor

#### 4.5.1. Algoritmo de inicialización y conexión del sistema

La primera rutina del sistema establece los parámetros iniciales para: el funcionamiento de la cámara IP, la detección de los objetos de color, la localización de las marcas aruco, la conexión de los robots mediante socket, la generación y seguimiento de trayectorias. Además, se encarga de la conexión de los robots móviles y la cámara con la estación central de control para iniciar con la localización de objetos de color y marcas.

En la Figura 4.18 se aprecia los procesos que realiza la rutina de inicialización del sistema cooperativo.



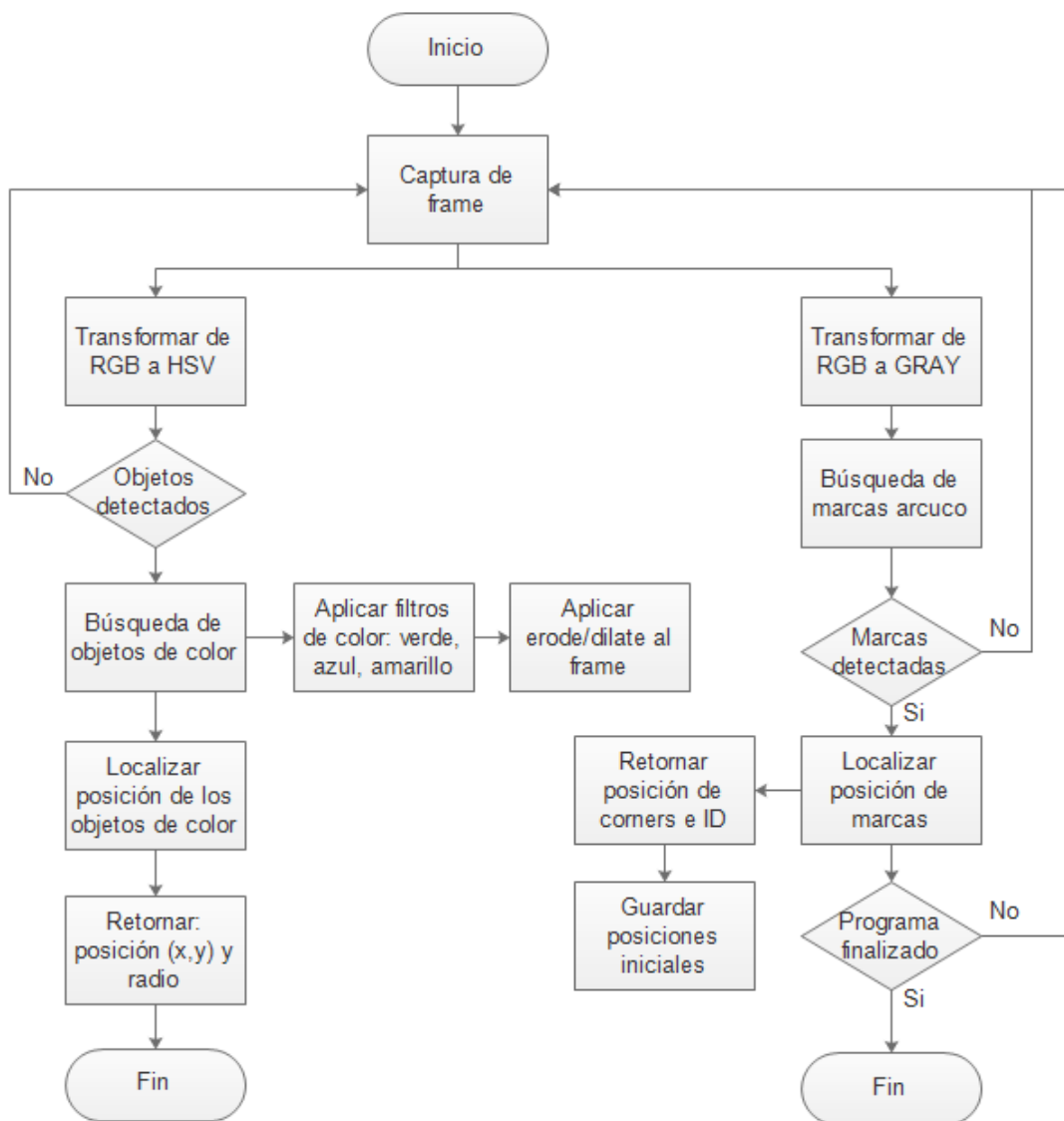
**Figura 4.18.** Diagrama de flujo de la rutina de inicialización del sistema.  
Fuente: Elaborado por el autor

#### 4.5.2. Algoritmo de localización de objetos de color y marcas

Los procesos realizados en la rutina de actualización de posiciones se observan en la Figura 4.19, donde se determina las posiciones en el escenario de pruebas para los objetos de color y las marcas. El proceso de detección y localización de los objetos de color se realiza una sola vez, debido a que solamente se utilizan sus posiciones para la generación de trayectoria A, la misma que es off-line.



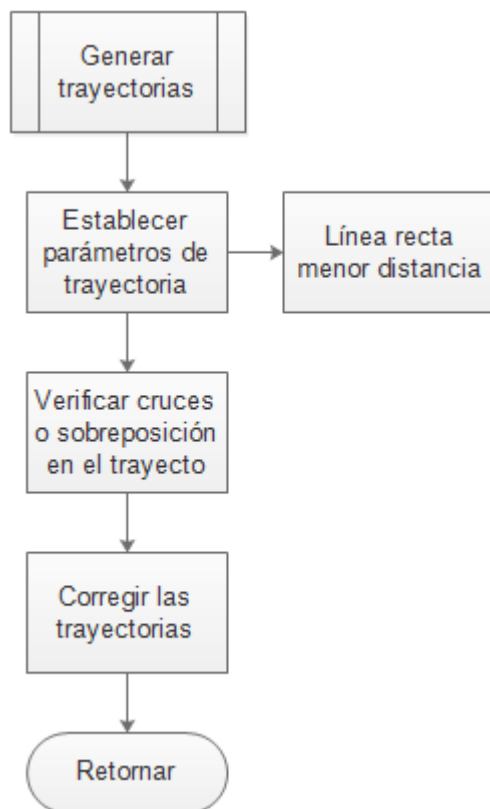
El proceso de localización de las marcas es repetitivo en el programa, ya que sus valores son necesarios para el proceso de seguimiento de trayectoria. En el proyecto se trabaja con nueve marcas aruco repartidas en: las posiciones de partida, de los robots y en las posiciones de entrega de los objetos de color. Para su detección se transforma el frame capturado en el modelo de color RGB a GRAY, y se aplica la función de filtrado de la librería Aruco, obteniendo la identificación (ID) de las marcas y sus cuatro corners. Del proceso se retorna únicamente los corners de la diagonal principal, es decir de la que presenta un recuadro en rojo.



**Figura 4.19.** Diagrama de flujo de la rutina de actualización de posiciones.  
Fuente: Elaborado por el autor

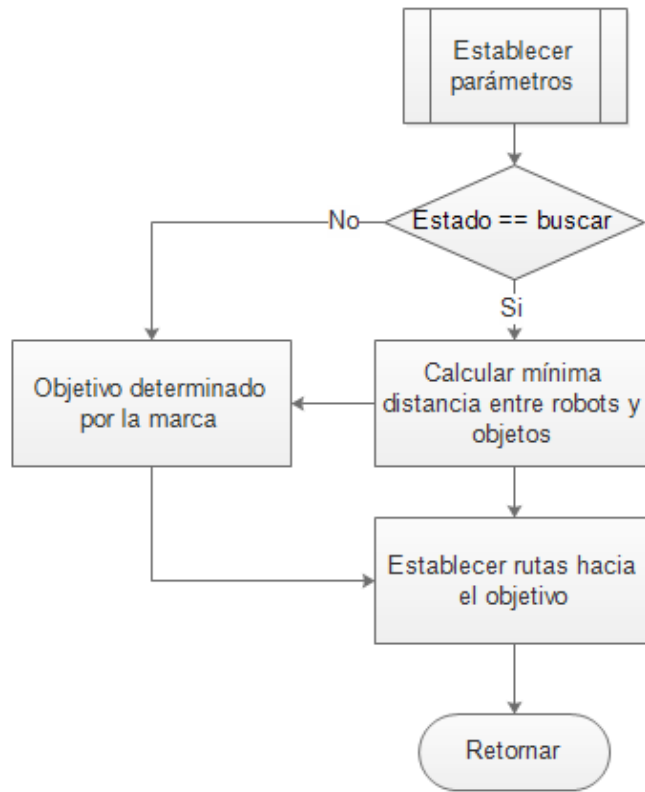
### 4.5.3. Algoritmo de generación de trayectorias

El algoritmo de generación de trayectoria comprende la formación de las rutas a seguir por los robots en base a los datos de posición obtenidos de los objetos de color y las marcas, ver Figura 4.20. Primero se generan trayectorias en línea recta considerando la menor distancia existente entre los robots y su objetivo, ver Figura 4.21. Se verifica si existen cruces o sobreposición en las trayectorias generadas, de no existir se retornan y continúa el programa con el siguiente proceso; en caso de existir conflictos en la trayectoria se corrigen mediante el algoritmo RRT, ver Figura 4.22.

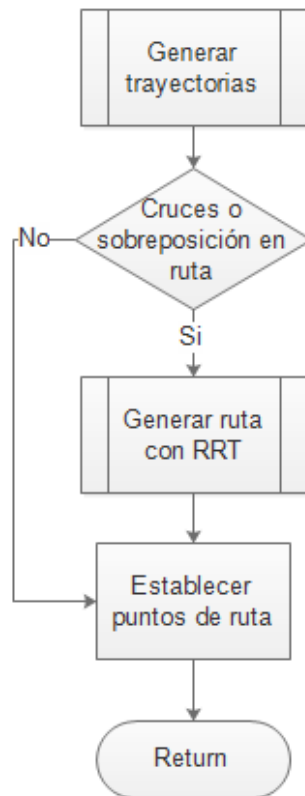


**Figura 4.20.** Diagrama de flujo de la rutina de generación de trayectoria.

Fuente: Elaborado por el autor



**Figura 4.21.** Diagrama de flujo de la evaluación de distancia entre robots y objetos.  
Fuente: Elaborado por el autor

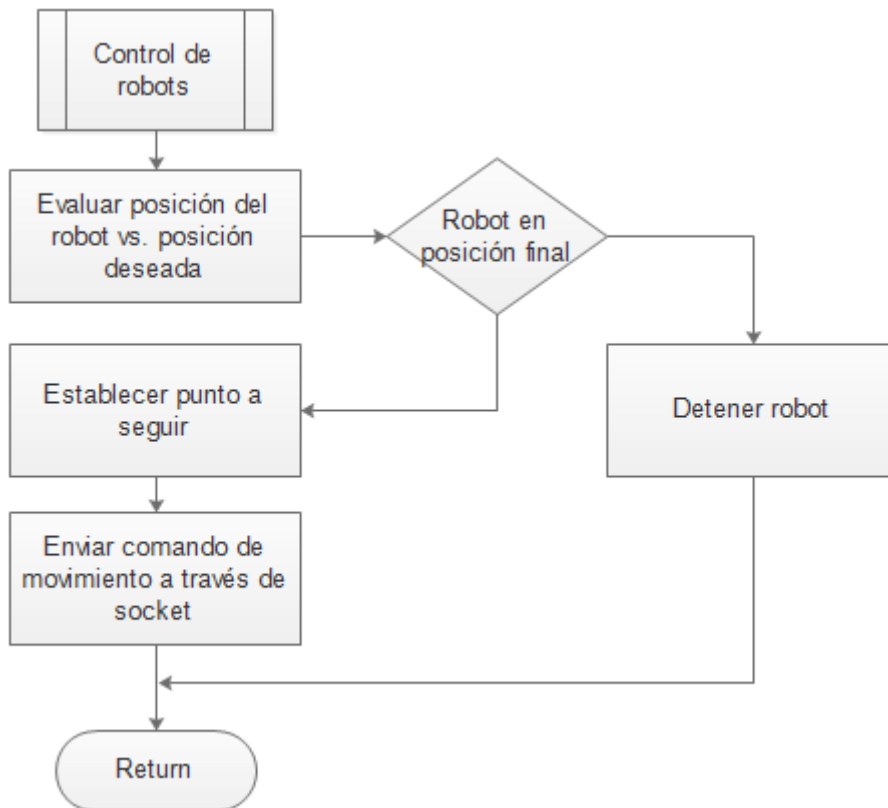


**Figura 4.22.** Diagrama de flujo de la generación de trayectoria con RRT.  
Fuente: Elaborado por el autor

#### 4.5.4. Algoritmo de seguimiento de trayectoria

Para el seguimiento de trayectoria es necesario evaluar la posición actual del robot móvil con respecto a la posición de llegada, para detener el algoritmo en el caso de alcanzar el final de la trayectoria o agregar un nuevo punto a seguir, siendo necesario actualizar el valor de las posiciones de las marcas, ver Figura 4.23.

La navegación de los robots en el escenario de prueba es controlada desde la estación central mediante el envío de señales de movimiento (derecha, izquierda, adelante y atrás) en base a la evaluación realizada en el seguimiento de trayectoria.

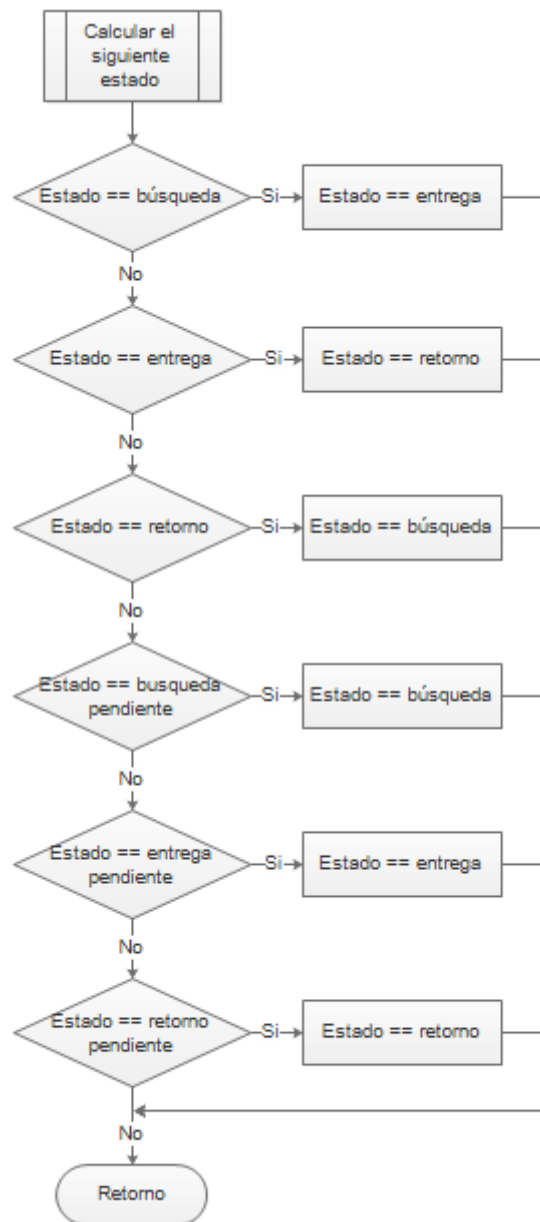


**Figura 4.23.** Diagrama de flujo del algoritmo de seguimiento de trayectoria

Fuente: Elaborado por el autor

En la Figura 4.24 se analiza el cambio de estado, que controla las tareas de iniciación del sistema cooperativo, la generación y seguimiento de trayectoria para los tres trayectos

correspondientes a la recolección de los objetos de color, su traslado a la posición de entrega y el retorno a la posición de la que partió el robot.



**Figura 4.24.** Diagrama de flujo de la actualización de los estados  
Fuente: Elaborado por el autor

#### 4.6. Implementación de algoritmos

Los algoritmos que fueron implementados en la estación central de control del sistema cooperativo son los siguientes:

- Conexión inalámbrica entre los elementos del sistema

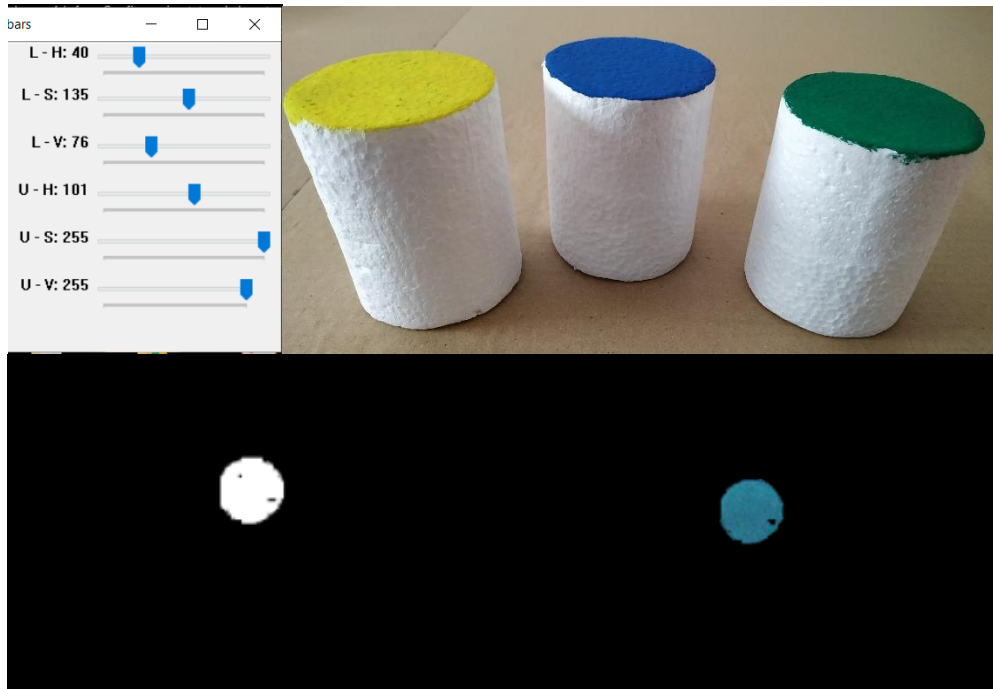
- Detección y localización de los objetos de color y marcas en el escenario de pruebas.
- Generación de trayectoria off-line, es decir la trayectoria se envía a los robots antes de que empiecen a desplazarse para los tramos A, B y C descritos anteriormente.
- Seguimiento de trayectoria on-line, donde la cámara proporciona continuamente información del entorno, es decir en tiempo real, para los tramos A, B y C.
- Control de navegación de los robots móviles y del brazo electromecánico.

En las tarjetas nodeMCU acopladas en los robots móviles se implementó algoritmos para la comunicación inalámbrica y algoritmos para el control de los actuadores (motores y servomotor) mediante las señales enviadas desde la estación central de procesamiento.

#### **4.6.1. Algoritmo de detección y localización de objetos de color y marcas**

Para localizar la posición de las marcas y los objetos de color en el escenario de pruebas se utilizó visión artificial, utilizando filtros de color e identificación de córner o esquinas en las marcas aruco. Para la detección de objetos de color es necesario adquirir en el espacio de color HSV el rango de valores para cada objeto, por lo que se debe transformar la imagen captada por la cámara al modelo de color HSV.

Se utiliza un algoritmo de filtrado de color basado en la función **inRange** ( ) de OpenCV, la cual trabaja con tres parámetros: imagen, umbral inferior y umbral superior. Los valores para los umbrales son determinados experimentalmente como se observa en la Figura 4.25, y en la Tabla 4.4 se muestran los valores correspondientes a los tres colores de los objetos.



**Figura 4.25.** *Obtención del rango de valores HSV para el color verde.*  
Fuente: Elaborado por el autor

**Tabla 4.4**

*Rango de valores HSV para los objetos de color.*

Color	Rango inferior	Rango superior
Azul	36, 160, 160	185, 255, 255
Verde	39, 135, 75	101, 255, 255
Amarillo	17, 130, 160	44, 250, 255

Fuente: Elaborado por el autor

Una vez adquiridos los valores HSV para todos los objetos de color, se procede a calcular su centro y radio en píxeles y se almacenan los valores iniciales de posición. Para localizar la posición de las macas en el escenario se utiliza la librería Aruco construida con OpenCV, esta librería nos permite detectar el ID de las marcas y conocer la posición de sus corners de manera fácil, rápida y robusta como se observa en la Figura 4.26.



**Figura 4.26.** *Detección de marcas Aruco vista aérea.*  
Fuente: Elaborado por el autor

Al trabajar con la librería Aruco es necesario transformar la imagen del espacio de color RGB a GRAY para no tener inconvenientes con la detección, debido a que las marcas son de color blanco y negro. Una vez detectadas las marcas se procede a obtener las posiciones en píxeles de los corners de la diagonal principal referenciada por el cuadrado rojo en un corner, ver Figura 4.26, y se guardan los valores iniciales de las marcas 1, 2 y 3 correspondientes a los robots móviles. Estos valores son utilizados para generar la trayectoria “A” que comprende la recolección de los objetos de color.

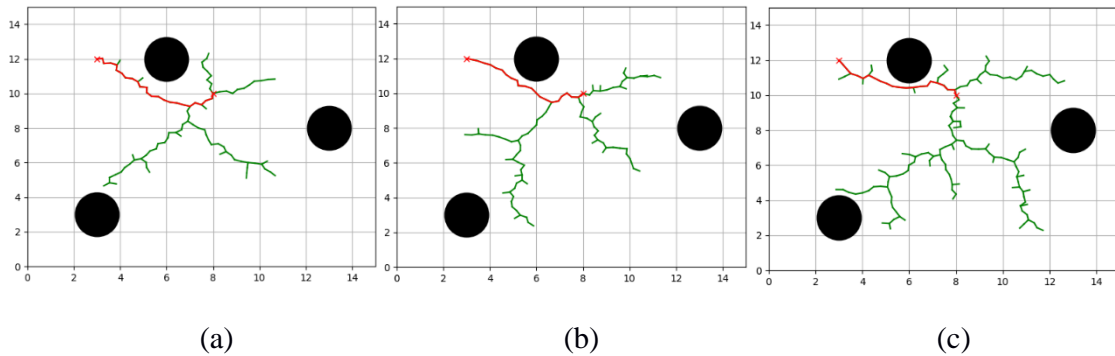
Además, el ID y el valor de los corners son actualizados continuamente para el proceso de seguimiento de trayectoria en tiempo real. Con el valor de los corners de la diagonal principal y el centro de los objetos de color se calcula el ángulo de rotación para el robot móvil.



#### 4.6.2. Algoritmo de generación de trayectorias

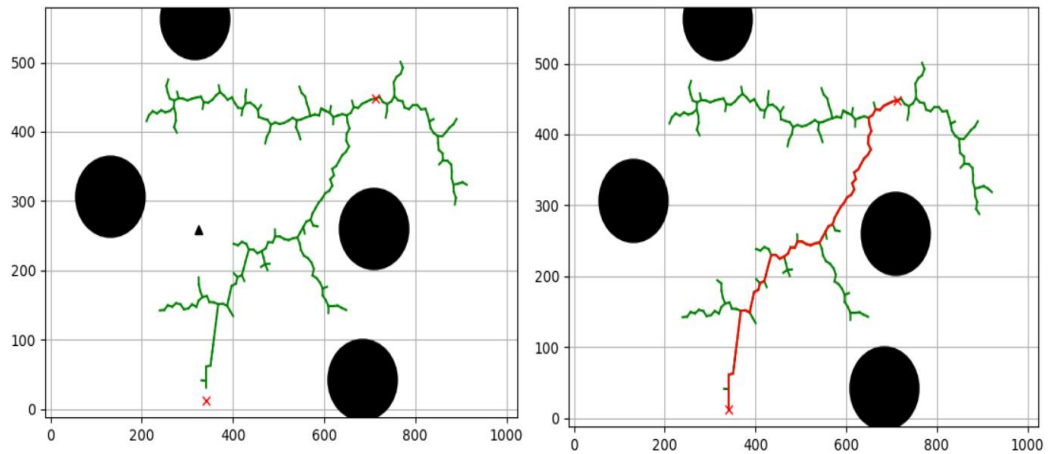
Se trabaja con dos algoritmos de generación de trayectorias: línea recta y RRT. El primero genera trayectorias rectilíneas aplicando la ecuación de la recta entre dos puntos, tomando como punto inicial la posición del robot y como punto final la posición del objeto de color. Este algoritmo se utiliza únicamente cuando no existan conflictos de cruce de trayectos o sobreposición de robots.

El algoritmo RRT genera trayectorias fortuitas libre de colisiones con cada ejecución, en la Figura 4.27 se muestra la generación de tres trayectorias diferentes para dos mismos puntos de inicio y fin.



**Figura 4.27.** *Generación de trayectoria con el algoritmo RRT para puntos fijos.*  
Fuente: Elaborado por el autor

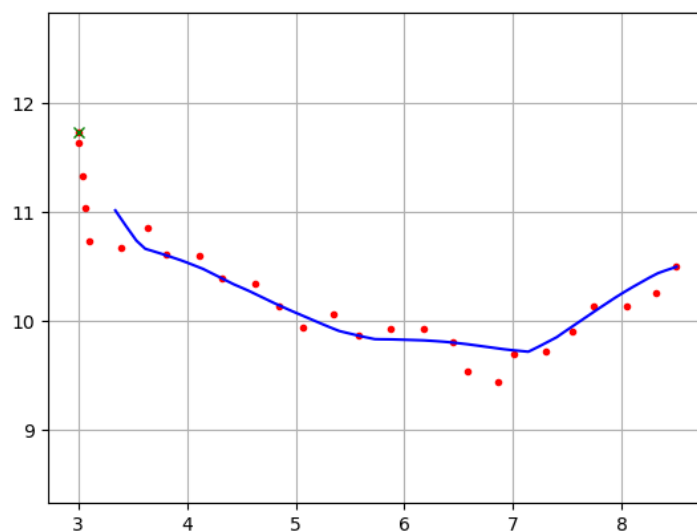
Los parámetros requeridos por el algoritmo son: punto de partida, punto de llegada y los obstáculos. Las dimensiones del escenario de pruebas en pixeles son de 1024 x 576, siendo este espacio el utilizado por el algoritmo RRT. En la Figura 4.28 se observa la generación del trayecto B para el robot tres con respecto al objeto de color amarillo, donde el tamaño de ramificación es de 20 pixeles



**Figura 4.28.** *Generación de trayectoria mediante el algoritmo RRT en el escenario.*  
Fuente: Elaborado por el autor

### 4.6.3. Algoritmo de seguimiento de trayectoria

Se implementó el algoritmo pure pursuit debido al bajo error de posición presentado y a su factibilidad de configuración por contar con un solo parámetro de control. Como se puede observar en el ejemplo de la Figura 4.29 el algoritmo permite al robot seguir la trayectoria sin la necesidad de pasar por todos los puntos de la misma, siendo adecuado su utilización para las trayectorias generadas mediante el algoritmo RRT, resultando en un seguimiento suave y con buenos resultados rutas creadas con ramificaciones.



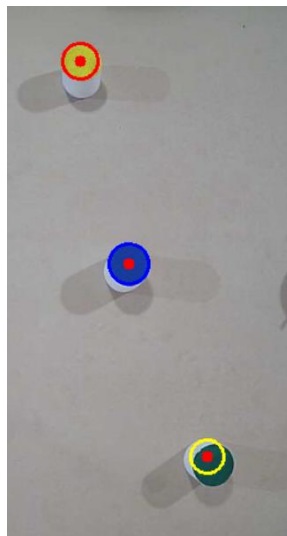
**Figura 4.29.** *Seguimiento de trayectoria aplicando el algoritmo pure pursuit.*  
Fuente: Elaborado por el autor

## 4.7. Pruebas de funcionamiento y análisis de resultados

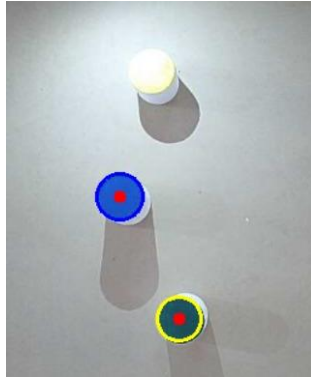
En esta sección se analiza los resultados obtenidos en las pruebas realizadas para comprobar el desempeño del sistema.

### 4.7.1. Reconocimiento de objetos de color

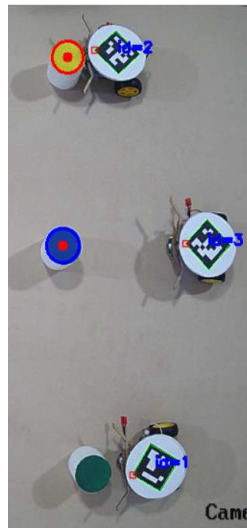
Para evaluar si el sistema reconoce los objetos de color en distintos lugares del escenario en diferentes condiciones de iluminación se procede a realizar pruebas de detección y localización de los objetos, en la Figura 4.30 se observa la detección de objetos de color en un ambiente de iluminación artificial adecuada, y en la Figura 4.31 se visualiza cuando la iluminación no es adecuada. Del mismo modo se realiza pruebas de detección utilizando iluminación natural adecuada, ver Figura 4.32, e inadecuada, ver Figura 4.33.



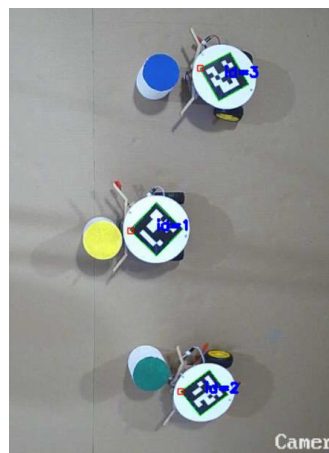
**Figura 4.30.** *Detección de objetos con iluminación artificial adecuada.*  
Fuente: Elaborado por el autor



**Figura 4.31.** *Detección de objetos con iluminación artificial inadecuada.*  
Fuente: Elaborado por el autor



**Figura 4.32.** *Detección de objetos con iluminación natural adecuada.*  
Fuente: Elaborado por el autor



**Figura 4.33.** *Detección de objetos con iluminación natural inadecuada.*  
Fuente: Elaborado por el autor

En la Tabla 4.5 se observan los resultados obtenidos de la experimentación con iluminación adecuada y en la Tabla 4.6 se observa los resultados con iluminación inadecuada.

**Tabla 4.5**

*Resultados de las pruebas con iluminación adecuada.*

<b>Color</b>	<b>Iluminación</b>	<b>Configuración</b>	<b>Tipo</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
Azul	Adecuada	Primera	Natural	Si	Si	No	Si	Si
Azul	Adecuada	Primera	Artificial	Si	Si	Si	Si	Si
Verde	Adecuada	Primera	Natural	Si	Si	Si	No	No
Verde	Adecuada	Primera	Artificial	No	Si	Si	Si	Si
Amarillo	Adecuada	Primera	Natural	Si	Si	Si	Si	Si
Amarillo	Adecuada	Primera	Artificial	No	Si	Si	Si	Si
Azul	Adecuada	Segunda	Natural	Si	Si	Si	Si	No
Azul	Adecuada	Segunda	Artificial	Si	Si	Si	Si	Si
Verde	Adecuada	Segunda	Natural	Si	No	Si	Si	No
Verde	Adecuada	Segunda	Artificial	Si	Si	Si	Si	Si
Amarillo	Adecuada	Segunda	Natural	No	Si	No	Si	Si
Amarillo	Adecuada	Segunda	Artificial	Si	Si	Si	Si	Si
Azul	Adecuada	Tercera	Natural	Si	Si	Si	Si	Si
Azul	Adecuada	Tercera	Artificial	Si	Si	Si	Si	Si
Verde	Adecuada	Tercera	Natural	No	Si	Si	Si	No
Verde	Adecuada	Tercera	Artificial	Si	Si	Si	Si	Si
Amarillo	Adecuada	Tercera	Natural	No	Si	Si	No	Si
Amarillo	Adecuada	Tercera	Artificial	Si	Si	No	Si	Si

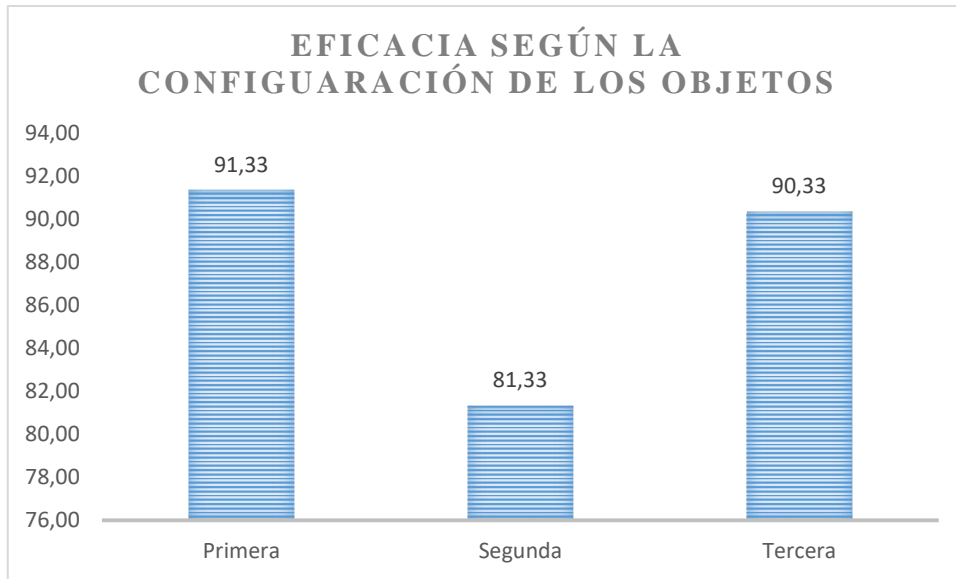
Fuente: Elaborado por el autor

**Tabla 4.6***Resultados de las pruebas con iluminación adecuada.*

<b>Color</b>	<b>Iluminación</b>	<b>Configuración</b>	<b>Tipo</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
Azul	Inadecuada	Primera	Natural	No	Si	No	Si	Si
Azul	Inadecuada	Primera	Artificial	Si	Si	No	Si	Si
Verde	Inadecuada	Primera	Natural	Si	No	Si	No	No
Verde	Inadecuada	Primera	Artificial	No	Si	Si	No	Si
Amarillo	Inadecuada	Primera	Natural	Si	No	Si	Si	Si
Amarillo	Inadecuada	Primera	Artificial	No	Si	No	Si	Si
Azul	Inadecuada	Segunda	Natural	Si	No	Si	Si	No
Azul	Inadecuada	Segunda	Artificial	Si	Si	No	Si	Si
Verde	Inadecuada	Segunda	Natural	Si	No	Si	Si	No
Verde	Inadecuada	Segunda	Artificial	Si	No	Si	No	Si
Amarillo	Inadecuada	Segunda	Natural	No	Si	No	Si	Si
Amarillo	Inadecuada	Segunda	Artificial	Si	No	No	Si	Si
Azul	Inadecuada	Tercera	Natural	No	Si	Si	No	Si
Azul	Inadecuada	Tercera	Artificial	Si	No	Si	Si	Si
Verde	Inadecuada	Tercera	Natural	No	Si	No	Si	No
Verde	Inadecuada	Tercera	Artificial	No	Si	No	Si	Si
Amarillo	Inadecuada	Tercera	Natural	No	Si	Si	No	Si
Amarillo	Inadecuada	Tercera	Artificial	Si	Si	Si	No	Si

Fuente: Elaborado por el autor

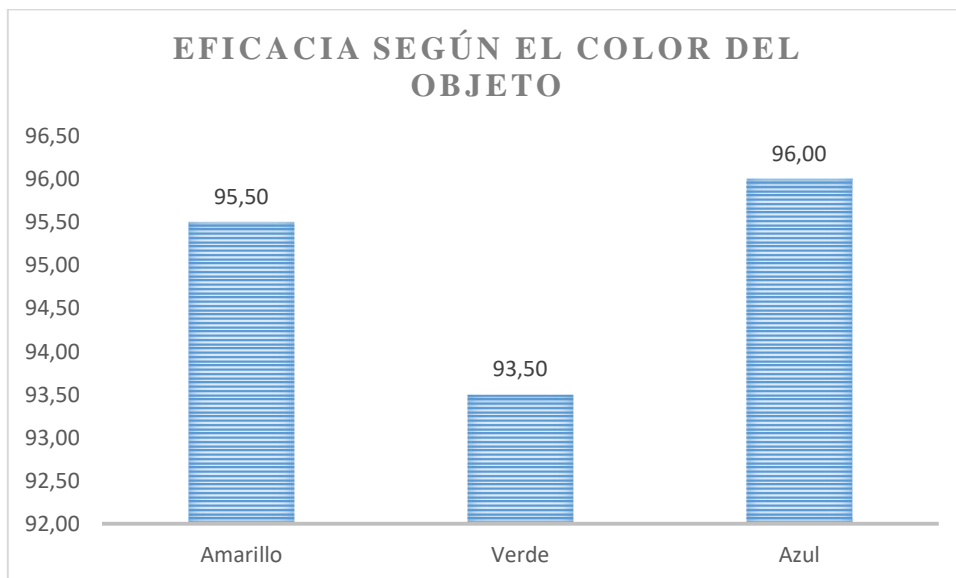
Centrándose en la configuración de distribución de los objetos en el escenario de pruebas se han obtenido los resultados mostrados en la Figura 4.34, donde la eficiencia obtenida en referencia a la detección es de 91,33% para la primera configuración, 81,23% para la segunda y 90,33% para la tercera.



**Figura 4.34.** *Eficacia de las distintas configuraciones de los objetos en el escenario.*

Fuente: Elaborado por el autor

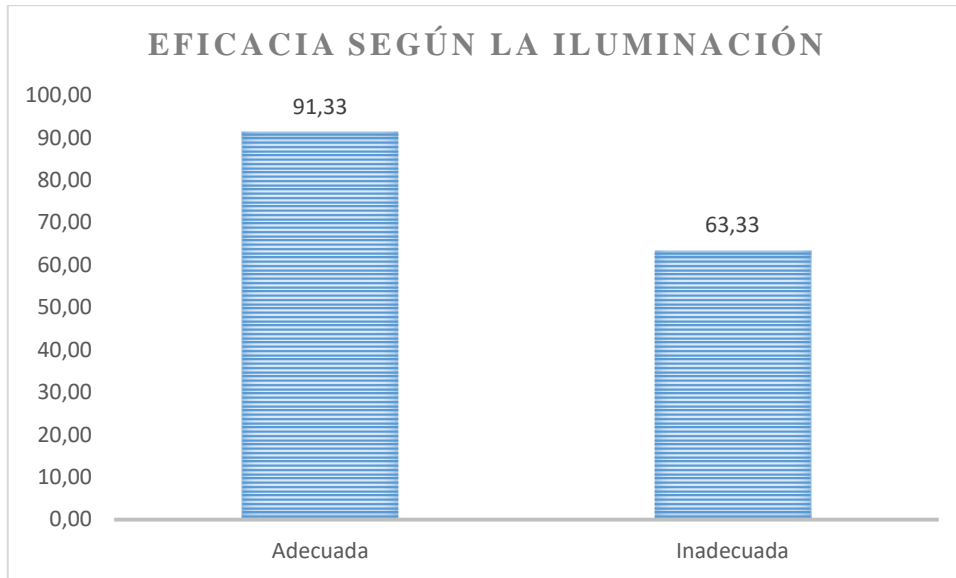
Por otro lado, en la Figura 4.35 se presentan los resultados dependiendo el color de los objetos. En este caso se puede ver que el color amarillo (95,5%) y el azul (96%) se consiguen mayores tasas de acierto que con el color verde (93,5%).



**Figura 4.35.** *Eficacia según el color del objeto.*

Fuente: Elaborado por el autor

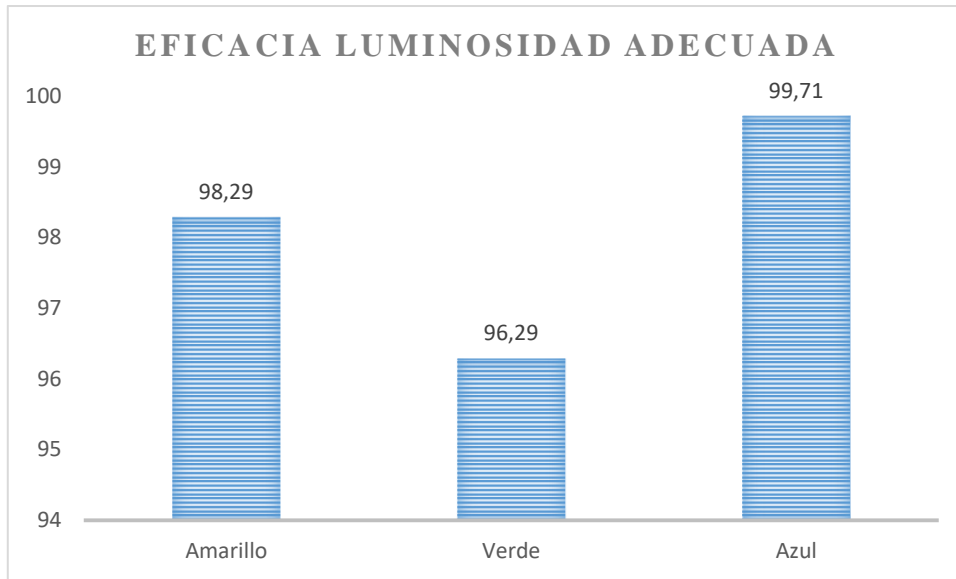
En la Figura 4.36 se observa que la efectividad de la detección baja considerablemente cuando la iluminación en la que se realiza las pruebas es inadecuada. Siendo su porcentaje de detección del 63,33%.



**Figura 4.36.** Eficacia según la iluminación.  
Fuente: Elaborado por el autor

Si en cambio se tiene en cuenta solo las pruebas realizadas con iluminación adecuada, que es realmente cuando se utilizará el algoritmo de detección, se consiguen los resultados mostrados en Figura 4.37. Donde los objetos de color amarillo (98,29%), azul (99,77%) y verde (98,26) se detectan casi a perfección, razón por la cual la eficiencia del sistema es alta y la detección de los objetos robusta.





**Figura 4.37.** *Eficacia para iluminación adecuada.*

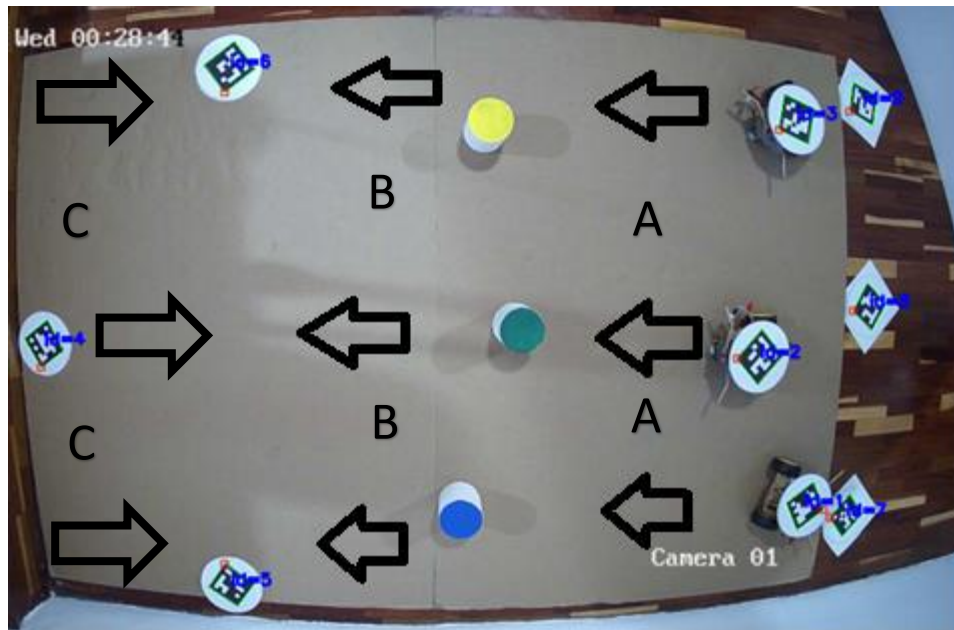
Fuente: Elaborado por el autor

#### **4.7.2. Generación y seguimiento de trayectorias**

Para evaluar las características de los algoritmos implementados se realizaron pruebas para tres configuraciones de los objetos de color en el entorno de pruebas. Se estableció la realización de un total de cinco pruebas para cada configuración.

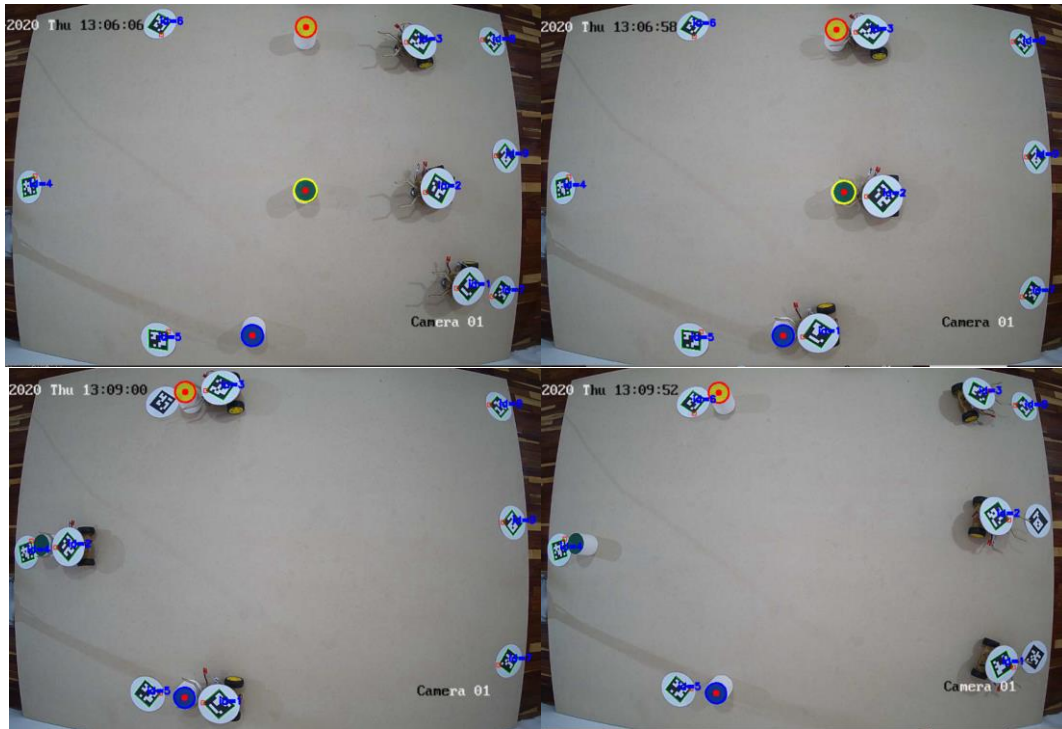
##### **4.7.2.1. Primera configuración de distribución**

La primera distribución de objetos en el escenario de pruebas presenta generación de trayectorias en línea recta para los trayectos A, B y C; ya que en esta configuración los robots avanzan directamente hacia los objetos de color y luego hacia las estaciones de entrega de los objetos sin cruces o superposiciones en sus rutas, ver Figura 4.38.



**Figura 4.38.** *Primera distribución de objetos en el escenario.*  
Fuente: Elaborado por el autor

En la Figura 4.39 se observa en secuencia de imágenes el seguimiento de las trayectorias: A que corresponde al tramo de búsqueda y recolección del objeto de color, B comprende el trayecto hacia la estación de entrega del robot y C es la ruta de retorno a la posición de origen de los robots. Los resultados obtenidos de los experimentos son presentados en la Tabla 4.7.



**Figura 4.39.** *Secuencia de imágenes del seguimiento de trayectoria de la primera configuración.*

Fuente: Elaborado por el autor

**Tabla 4.7**

*Resultados obtenidos para 5 pruebas realizadas en la primera configuración.*

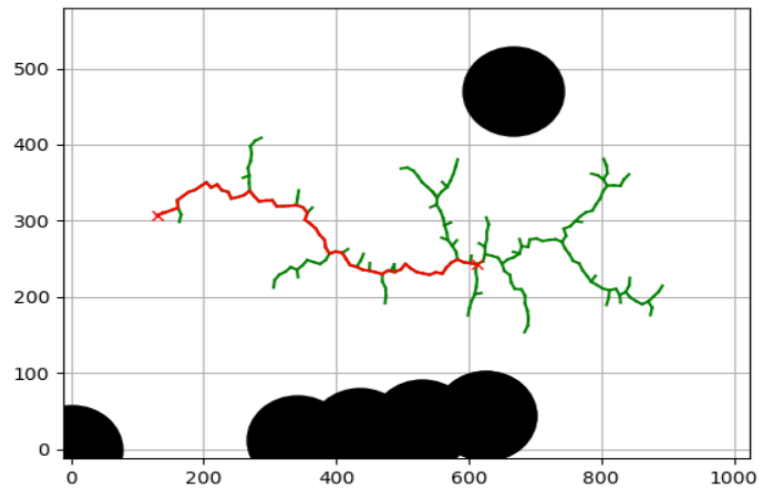
Parámetro	Sistema	Mínimo	Máximo
Tiempo de ejecución (min)	Todos	3,36	5,68
Número de objetos detectados	Cámara	1	3
Número de objetos recogidos	Robot 1	1	1
	Robot 2	1	1
	Robot 3	1	1
Número de objetos entregados	Robot 1	1	1
	Robot 2	1	1
	Robot 3	1	1

Fuente: Elaborado por el autor

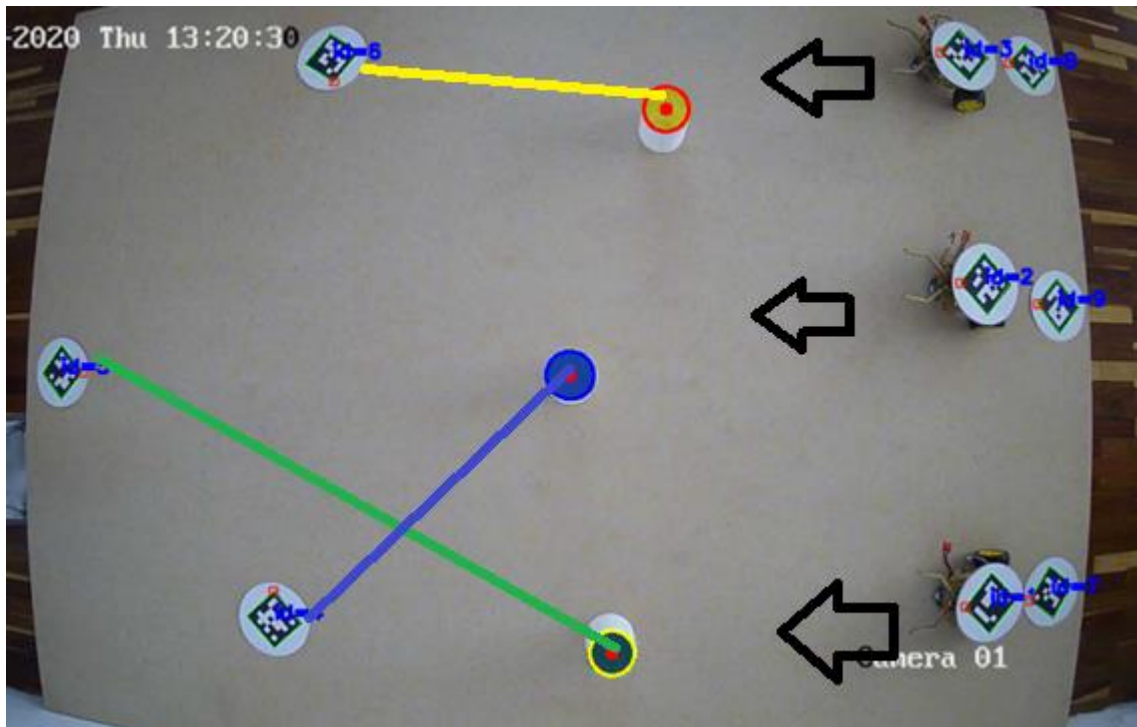
#### 4.7.2.2. Segunda configuración de distribución

La segunda configuración de distribución de objetos en el escenario de pruebas presenta generación de trayectorias en línea recta para el trayecto A, para el trayecto B se

tiene generada la ruta con RRT para el robot tres correspondiente al color verde, ver Figura 4.40, y para el trayecto C se generan rutas en línea hacia la posición de origen, ver Figura 4.41.

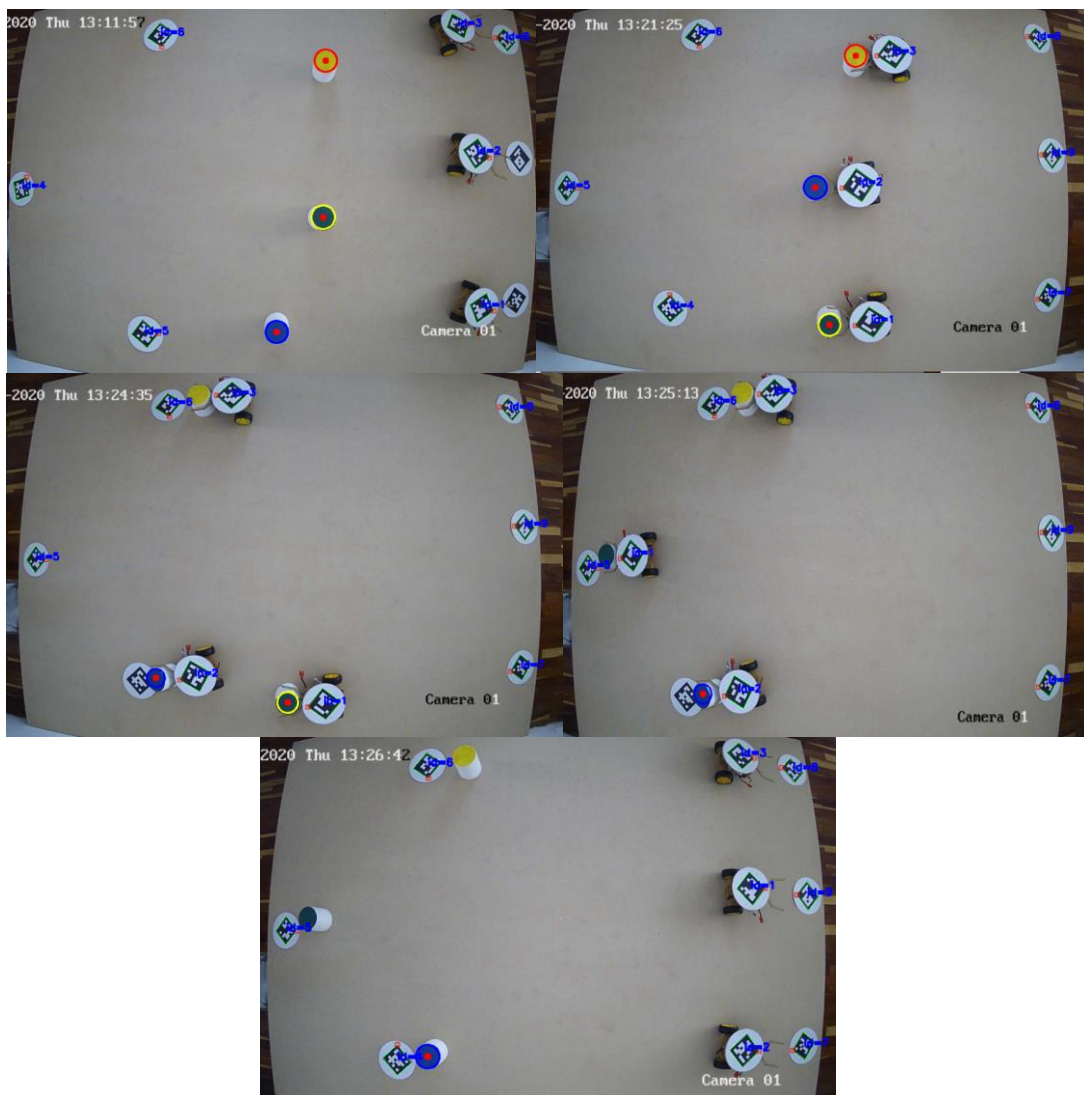


**Figura 4.40.** *Generación de trayectoria para el trayecto B.*  
Fuente: Elaborado por el autor



**Figura 4.41.** *Segunda distribución de objetos en el escenario.*  
Fuente: Elaborado por el autor

En la prueba se visualiza en la secuencia de imágenes el seguimiento de las trayectorias: A que corresponde al tramo de búsqueda en línea recta, B que comprende el trayecto hacia la estación de entrega del robot presenta dos trayectorias en línea recta y una generada con RRT y C presenta trayectorias en línea recta hacia los puntos de partida de los robots. Los resultados obtenidos de los experimentos son presentados en la Tabla 4.8.



**Figura 4.42.** *Secuencia de imágenes del seguimiento de la trayectoria de la segunda configuración.*

Fuente: Elaborado por el autor

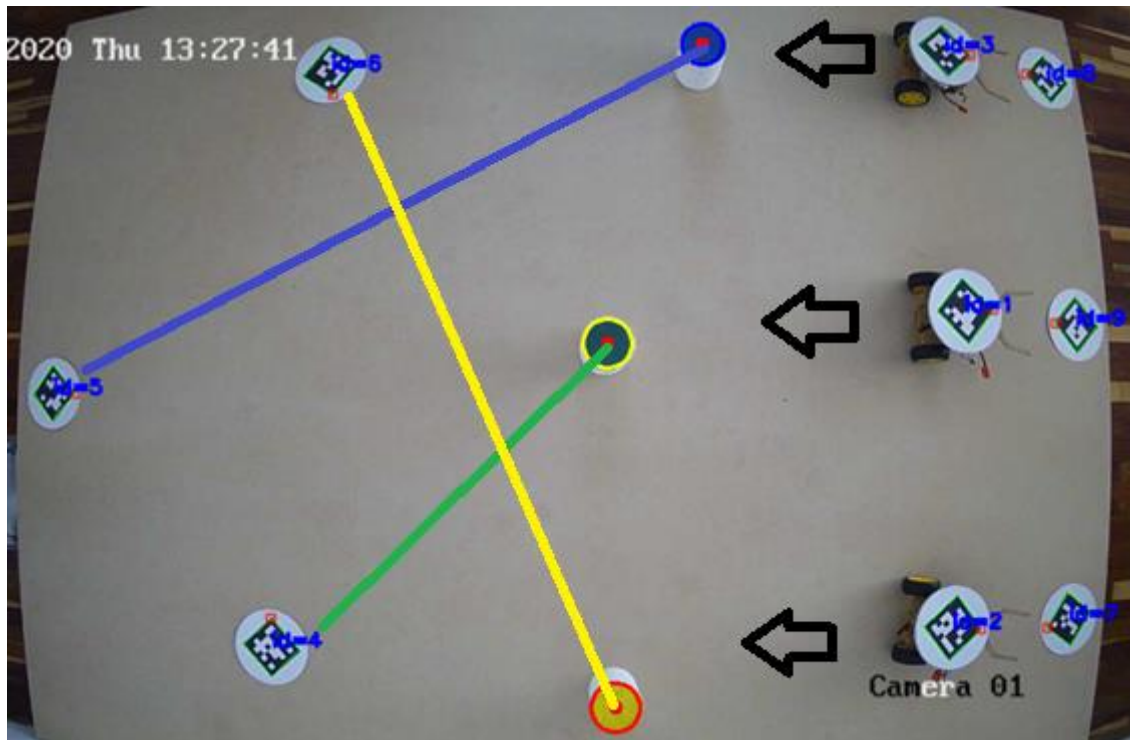
**Tabla 4.8***Resultados obtenidos para 5 pruebas realizadas en la segunda configuración.*

Parámetro	Sistema	Mínimo	Máximo
Tiempo de ejecución (min)	Todos	5,96	7,68
Número de objetos detectados	Cámara	1	3
Número de objetos recogidos	Robot 1	1	1
	Robot 2	1	1
	Robot 3	1	1
Número de objetos entregados	Robot 1	1	1
	Robot 2	1	1
	Robot 3	1	1

Fuente: Elaborado por el autor

**4.7.2.3. Tercera configuración de distribución**

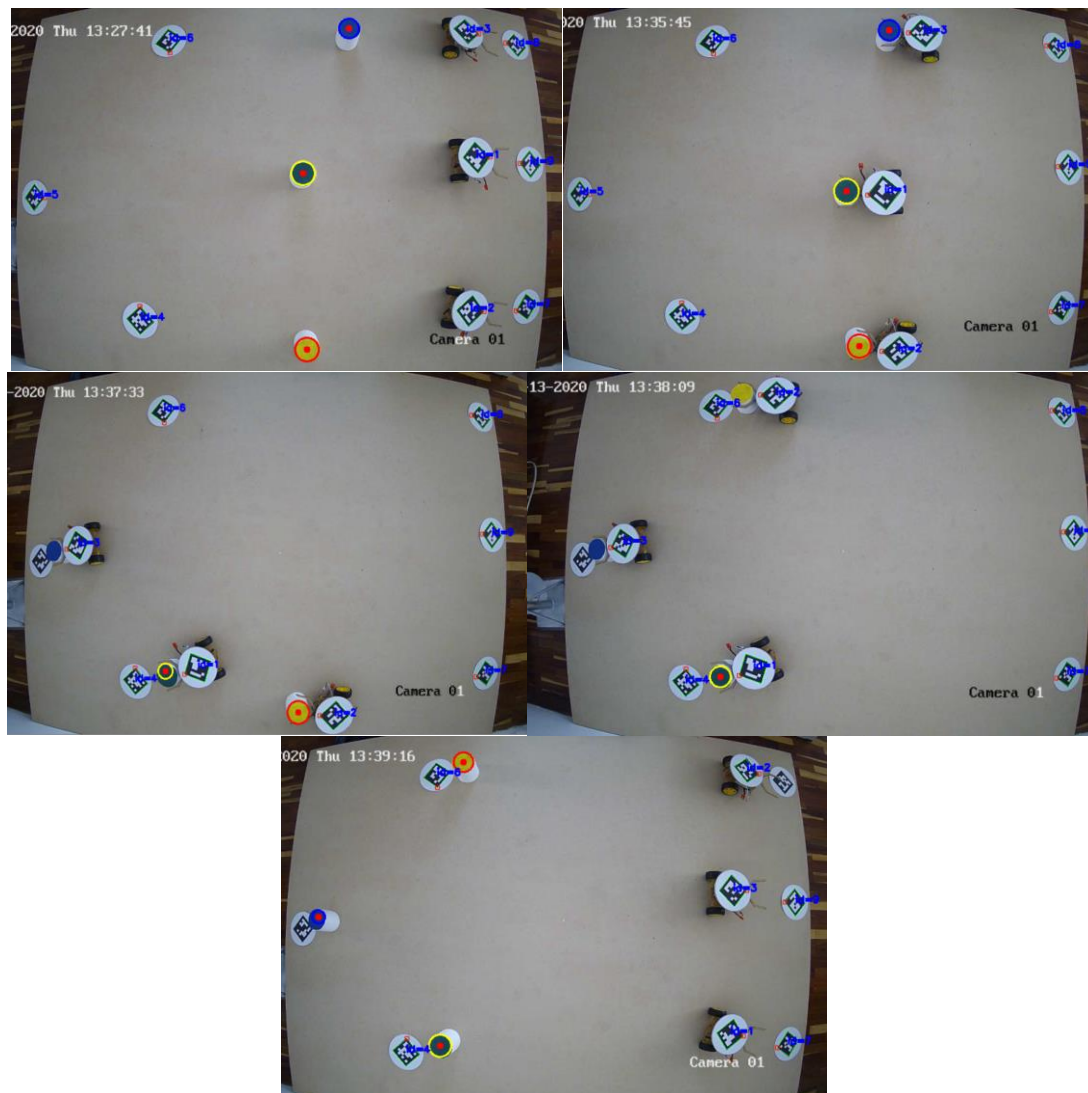
La tercera configuración de distribución de objetos en el escenario de pruebas presenta generación de trayectorias en línea recta para el trayecto A, para el trayecto B se tiene dos rutas en línea recta y una generada mediante RRT para el robot dos correspondiente al color amarillo, y para el trayecto C se generan rutas en línea hacia la posición de origen, ver Figura 4.43.



**Figura 4.43.** Tercera configuración de distribución de objetos en el escenario.  
Fuente: Elaborado por el autor

En la prueba se visualiza en la secuencia de imágenes el seguimiento de las trayectorias: A que corresponde al tramo de búsqueda en línea recta, B que comprende el trayecto hacia la estación de entrega del robot presenta dos trayectorias en línea recta y una generada con RRT y C presenta trayectorias en línea recta hacia los puntos de partida de los robots. Los resultados obtenidos de los experimentos son presentados en la Tabla 4.8.





**Figura 4.44.** Secuencia de imágenes del seguimiento de la trayectoria de la tercera configuración.

Fuente: Elaborado por el autor

**Tabla 4.9**

Resultados obtenidos para 5 pruebas realizadas en la tercera configuración.

Parámetro	Sistema	Mínimo	Máximo
Tiempo de ejecución (min)	Todos	4,56	8,03
Número de objetos detectados	Cámara	1	3
Número de objetos recogidos	Robot 1	1	1
	Robot 2	1	1
	Robot 3	1	1
Número de objetos entregados	Robot 1	1	1
	Robot 2	1	1
	Robot 3	1	1

Fuente: Elaborado por el autor



## 5. CONCLUSIONES

- El algoritmo de reconocimiento de objetos de color utiliza el modelo de color HSV y filtros para los diferentes colores. El algoritmo fue probado en condiciones de iluminación adecuada e inadecuada a diferentes horas del día y en los espacios de color RGB y HSV, observando que el modelo HSV presenta una eficiencia en la detección del 84,4% con respecto al 66,7% presentado por el modelo RGB, concluyendo un reconocimiento de color robusto y eficaz en tiempo real para el modelo HSV, disminuyendo así el factor ambiental en el sistema cooperativo.
- Se estableció la altura ideal de detección de la cámara IP para objetos de color y marcas a partir de varias experimentaciones en las cuales se varió la altura del pedestal en el rango de 1.60 metros a 1.40 metros en intervalos de 10cm, verificando la detección correcta de los objetos y marcas. De la experimentación se concluye que la colocación de la cámara a alturas superiores de 1.40 metros presenta errores en el reconocimiento de los objetos de color y las marcas.
- La implementación de la estación central de control que coordine las tareas de comunicación, localización y navegación es indispensable para el transporte y clasificación de objetos con robots de diseño simple. De las pruebas realizadas se concluye que al implementar un control central en el sistema cooperativo se redujo la necesidad de una alta capacidad de procesamiento en los robots abaratando su costo, además el sistema puede ser escalado con facilidad.
- La aplicación del algoritmo RRT constituyó una gran alternativa en la planificación de trayectorias que presentaban conflictos de cruce o sobreposición. El algoritmo ha demostrado ser sencillo y de fácil

implementación, aunque existieron situaciones que no era capaz de resolver. Dichas situaciones se deben en gran medida a las dimensiones seleccionadas para el escenario de pruebas.

- La construcción de los robots móviles permitió realizar el proceso de distribución de los objetos de color posicionados aleatoriamente, en un tiempo promedio de 3,36 minutos y se pudo comprobar que la certeza de llegar al objetivo propuesto en cada trayecto es de 100% para todas las pruebas realizadas.

## **6. RECOMENDACIONES**

- Se recomienda incluir en el sistema un algoritmo que permita implementar el concepto de inteligencia cooperativa, es decir que en caso de defecto de uno o más robots otro tome su lugar con el objetivo para desarrollar robusta ante fallos.
- Se recomienda aplicar el uso de sistemas cooperativos en líneas de producción, ya que el sistema podría realizar distintos procesos al mismo tiempo optimizando y mejorando los tiempos de producción.
- Al utilizar baterías LiPo como fuente de alimentación de los robots del sistema se recomienda usar un módulo de protección y además controlar que el nivel de descarga no exceda el indicado por el fabricante con la finalidad de prolongar el tiempo de vida de la batería.

## 7. REFERENCIAS BIBLIOGRÁFICAS

- Álvarez Romero, L. M., & Figueroa Montenegro, J. E. (2010). *IMPLEMENTACIÓN DE ALGORITMOS DE NAVEGACIÓN UTILIZANDO LA PLATAFORMA DEL IROBOT CREATE Y MÓDULOS DE COMUNICACIÓN INALÁMBRICA XBEE*. Retrieved from <http://repositorio.espe.edu.ec/handle/21000/2643>
- Apolo Peñaloza, A. K., & Velasco Barrera, A. A. (2013). *Desarrollo de un sistema de robotica cooperativa entre dos elementos robots tipo Robonova*. Universidad de las Fuerzas Armadas - ESPE.
- Avantor Performance Materials. (2014). *Catálogo de sustancias químicas*. Retrieved from <https://www.bdcint.com.do/wp-content/uploads/2014/10/Catálogo-Avantor-2013-2014-Español.pdf>
- Carrascal Quintero, J. A. (2010). *Almacenamiento de Reactivos y Sustancias Químicas* (pp. 14–18). pp. 14–18. Retrieved from [http://www.usbcartagena.edu.co/phocadownload/facultades/salud/GUIA\\_ALMACENAMIENTO\\_REACTIVOS.pdf](http://www.usbcartagena.edu.co/phocadownload/facultades/salud/GUIA_ALMACENAMIENTO_REACTIVOS.pdf)
- CENAPRED, Dirección de análisis y gestión de riesgos, & Subdirección de riesgos sanitarios y toxicología. (2016). *Manual de seguridad en el laboratorio de muestras ambientales del CENAPRED*. Retrieved from [https://www.gob.mx/cms/uploads/attachment/file/204535/15.-MANUAL\\_SEGURIDAD\\_LABORATORIO\\_MUESTRAS\\_AMBIENTALES\\_CENAPRED.pdf](https://www.gob.mx/cms/uploads/attachment/file/204535/15.-MANUAL_SEGURIDAD_LABORATORIO_MUESTRAS_AMBIENTALES_CENAPRED.pdf)
- Chen, X., Kundu, K., Zhang, Z., Ma, H., Fidler, S., & Urtasun, R. (2016). Monocular 3D Object Detection for Autonomous Driving. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2147–2156. Retrieved from [123](https://www.cv-</a></p></div><div data-bbox=)

foundation.org/openaccess/content\_cvpr\_2016/papers/Chen\_Monocular\_3D\_Object\_CVPR\_2016\_paper.pdf

- Chicala, J., Jacho, R., Atiencia, L., & Vintimilla, B. (2006). Reconocimiento y seguimiento de objetos móviles en un sistema de fútbol robótico. *Revista Tecnológica ESPOL*, *xx*. Retrieved from [https://www.researchgate.net/profile/Boris\\_Vintimilla/publication/28791943\\_Reconocimiento\\_Y\\_Seguimiento\\_De\\_Objeto\\_Moviles\\_En\\_Un\\_Sistema\\_De\\_Futbol\\_Robotico/links/00b7d532d34fdda76e000000/Reconocimiento-Y-Seguimiento-De-Objetos-Moviles-En-Un-Sistema-De-Fut](https://www.researchgate.net/profile/Boris_Vintimilla/publication/28791943_Reconocimiento_Y_Seguimiento_De_Objeto_Moviles_En_Un_Sistema_De_Futbol_Robotico/links/00b7d532d34fdda76e000000/Reconocimiento-Y-Seguimiento-De-Objetos-Moviles-En-Un-Sistema-De-Fut)
- Colmenares, M. C. (2014). BASIS FOR MANAGEMENT OF HAZARDOUS CHEMICALS IN TEACHING LABORATORIES OF THE UNIVERSITY OF CARABOBO'S CHEMICAL ENGINEERING SCHOOL. *Ingeniería y Sociedad*, *9*, 30–42. Retrieved from <http://servicio.bc.uc.edu.ve/ingenieria/revista/IngenieriaYSociedad/a9n1/art03.pdf>
- Conlago Guatemal, C. R., & Yunda Sangoluisa, J. A. (2016). *Sistema Automático de detección y reconocimiento de señales de tránsito en intersecciones viales para aplicaciones en vehículos inteligentes*. (Universidad de las Fuerzas Armadas ESPE). Retrieved from <https://repositorio.espe.edu.ec/bitstream/21000/12242/1/T-ESPE-053557.pdf>
- Constante Prócel, P. N., & Gordón Garcés, A. M. (2015). *Diseño e Implementación de un sistema de visión artificial para clasificación de al menos tres tipos de frutas*. (Escuela Politécnica Nacional). Retrieved from <https://bibdigital.epn.edu.ec/bitstream/15000/11368/1/CD-6457.pdf>
- Corporation, C. (2019). Iluminación de visión artificial. Retrieved from Cognex website: <https://www.cognex.com/es-ar/what-is/machine->

vision/components/lighting

De La Cruz, C., Carelli, R., & Gava, C. C. (2006). Control centralizado de formacion usando una cámara omnidireccional. *IV Jornadas Argentinas de Robótica*.

Diego, L. G. D., & Fernando, G. B. D. (2012). *Tesis Doctoral Nuevas aportaciones en algoritmos de planificación para la ejecución de maniobras en robots autónomos no holónomos*. Retrieved from

[http://rabida.uhu.es/dspace/bitstream/handle/10272/5501/Nuevas\\_aportaciones\\_en\\_algoritmos\\_de\\_planificacion.pdf?sequence=2](http://rabida.uhu.es/dspace/bitstream/handle/10272/5501/Nuevas_aportaciones_en_algoritmos_de_planificacion.pdf?sequence=2)

Diestra, P., & Guazhambo Ortiz, M. A. (2012). *ESTUDIO SOBRE LAS*

*TECNOLOGIAS PARA CONTROL DE VIGILANCIA DE CÁMARAS IP POR MEDIO DE INTERNET DESDE DISPOSITIVOS MÓVILES* (Universidad ISrael).

Retrieved from <http://157.100.241.244/bitstream/47000/582/1/UISRAEL-EC- SIS-378.242- 265.pdf>

Efrain Ernesto, A. V., Arturo, Z. L., & Juan, V. C. (2015). Implementación de reconocimiento de objetos por color y forma en un robot móvil (Vol. 91).

ESPE. (2014). *Instructivo para la clasificación y almacenamiento de reactivos*.

Retrieved from <https://studylib.es/doc/6138179/instructivo-para-la-clasificacion-y-almacenamiento-de-rea...>

Etxeberria, J. A. (2010). *Algoritmo de reconocimiento de forma y color para una plataforma robótica* (Universidad del País Vasco / Euskal Herriko Unibertsitatea).

Retrieved from <http://www.ehu.eus/documents/1545039/1570316/10jaristondo.pdf>

Giacomett, J. (2017). *INSTRUCTIVO PARA LA CLASIFICACIÓN Y*

*ALMACENAMIENTO DE REACTIVOS DE LABORATORIO*. Retrieved from

<http://decv.espe.edu.ec/wp-content/uploads/2017/06/II-CLASIFICA-Y-ALMACENAMIENTO-DE-REACTIVOS-LAB.pdf>

- Gonzales, R. C., & Woods, R. E. (2007). Color Image Processing. In Prentice Hall (Ed.), *Digital Image Processing* (3rd ed., pp. 282–344). New Jersey.
- Gonzalo Rafael Rojo Aguirre. (2014). *Seguridad y medio ambiente en planta química. QUIE0108*. Retrieved from [https://books.google.es/books?id=RaenBAAAQBAJ&dq=ambientes+de+manipulacion+de+sustancias+tóxicas&lr=&hl=es&source=gbs\\_navlinks\\_s](https://books.google.es/books?id=RaenBAAAQBAJ&dq=ambientes+de+manipulacion+de+sustancias+tóxicas&lr=&hl=es&source=gbs_navlinks_s)
- Hikvision. (2019). Network Camera. Retrieved from hikvision website: <https://www.hikvision.com/en/Products/Network-Camera>
- Illanes Tibanlombo, J. C. (2018). *Adaptación del sistema robótico móvil autónomo cooperativo SMELLROBSENSE para mapeo 2D de obstáculos, temperatura y concentración de gases en una ambiente controlado*. (Universidad de las Fuerzas Armadas ESPE). Retrieved from <https://repositorio.espe.edu.ec/bitstream/21000/15528/1/T-ESPE-040601.pdf>
- Inglett, J. E., & Rodríguez-Seda. (2017). Object transportation by cooperative robots. *SoutheastCon*, 1–6.
- Javier, C., Arias, V., Cesar, J., & Sotelo, T. (n.d.). *Diseño de un Sistema de Visión Artificial para la Clasificación de Chirimoyas basado en medidas*.
- Jiménez Camacho, E. (2009). *Medición de distancias por medio de procesamiento de imágenes y triangulación, haciendo uso de cámaras de video*. Escuela de Ingeniería, Universidad de las Américas Puebla.
- Kravtchenko, V. (1992). Tracking Color Objects in Real Time (The University of British Columbia). Retrieved from <https://www.cs.ubc.ca/grads/resources/thesis/Nov99/Vladimir-Kravtchenko.pdf>
- La Serna Palomino, N., & Román Concha, U. (2009). Técnicas de Segmentación en Procesamiento Digital de Imágenes. *Revista de Ingeniería de Sistemas e*

*Informática*, 6(2), 9–16. [https://doi.org/10.1016/S0186-1042\(14\)70160-3](https://doi.org/10.1016/S0186-1042(14)70160-3)

Lara, A., Urrea, C., & Barbosa, K. A. (2015). Controller design of industrial cooperative robots. *2015 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, 73–78.  
<https://doi.org/10.1109/Chilecon.2015.7400355>

Lin, L., Wang, X., Yang, W., & Lai, J.-H. (2015). Discriminatively Trained And-Or Graph Models for Object Shape Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(5), 952–972.  
<https://doi.org/10.1109/TPAMI.2014.2359888>

LUA. (n.d.). The Programming Language Lua. Retrieved October 20, 2018, from <https://www.lua.org/>

Mejía, J. L. M. (2015). *Diseño de un sistema de video-vigilancia inalámbrico para la ciudad de Cayambe*. Escuela Politécnica Nacional.

Murray, R. M. (n.d.). *Recent Research in Cooperative Control of Multivehicle Systems*.  
<https://doi.org/10.1115/1.2766721>

nodeMCU. (2014). NodeMcu - An open-source firmware based on ESP8266 wifi-soc. Retrieved October 20, 2018, from [http://nodemcu.com/index\\_en.html#fr\\_54747661d775ef1a3600009e](http://nodemcu.com/index_en.html#fr_54747661d775ef1a3600009e)

Osorio Giraldo, R. D. (2009). *Manual de técnicas de laboratorio químico*. Universidad de Antioquía.

Pérez Álvarez, P. J. (2017). *Desarrollo de un algoritmo de visión artificial mediante el procesamiento digital de imágenes con el sensor Kinect orientado a Linux* (Universodad de las Fuerzas Armadas). Retrieved from <https://repositorio.espe.edu.ec/bitstream/21000/13477/1/T-ESPE-057383.pdf>

Pérez Checa, A. R. (2016). *Aplicación de algoritmos de visión artificial para el control*



*de robots cooperativos móviles dentro de espacios semi-estructurados*. Escuela Politécnica Nacional.

Possegger, H., & Mauthner, T. (2015). In Defense of Color-based Model-free Tracking. *N Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2113–2120. Retrieved from [https://www.cv-foundation.org/openaccess/content\\_cvpr\\_2015/papers/Possegger\\_In\\_Defense\\_of\\_2015\\_CVPR\\_paper.pdf](https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Possegger_In_Defense_of_2015_CVPR_paper.pdf)

Python.org. (2019). Python.

Raspberrypi.org. (2019). Raspberry Pi. Retrieved from <https://www.raspberrypi.org/documentation/faqs/#introduction>

Revista Negocios de Seguridad. (2007). Cámara IP. *Negocios de Seguridad*, 31, 84–104. Retrieved from [http://www.rnds.com.ar/articulos/031/RNDS\\_084W.pdf](http://www.rnds.com.ar/articulos/031/RNDS_084W.pdf)

RobotOyn. (2018). MEGA+WiFi R3.

Rojas, J. J. B., & Alonso Pérez, M. A. (2008). *Uso del sistema HSI para asignar falso color a objetos en imágenes digitales* (Vol. 54). Retrieved from <http://www.scielo.org.mx/pdf/rmfe/v54n2/v54n2a11.pdf>

Solís Ortega, R. (2016). *Algoritmo cooperativo con enjambres de robots para la exploración de una zona y comunicación de una ruta hacia un objetivo*. Instituto Tecnológico de Costa Rica.

Soria, C., Carelli, R., Kelly, R., & Zannatha, J. M. I. (2004). Control de Robots Cooperativos por Medio de Vision Artificial. *XVI Congreso de La Asociación Chilena de Control Automático*.

SparkFun. (n.d.). Advanced Arduino Comparison Guide - SparkFun Electronics. Retrieved October 20, 2018, from 2014 website: [https://www.sparkfun.com/advanced\\_arduino\\_comparison\\_guide](https://www.sparkfun.com/advanced_arduino_comparison_guide)

STMicroelectronics. (2000). *L298 DUAL FULL-BRIDGE DRIVER*. Retrieved from  
[https://www.sparkfun.com/datasheets/Robotics/L298\\_H\\_Bridge.pdf](https://www.sparkfun.com/datasheets/Robotics/L298_H_Bridge.pdf)

Tait, K. (2015). *Industria Farmaceutica*. 20. Retrieved from  
<http://www.insht.es/InshtWeb/Contenidos/Documentacion/TextosOnline/EnciclopediaOIT/tomo3/79.pdf>

Vezhnevets, V., Sazonov, V., & Andreeva, A. (2003). A Survey on Pixel-Based Skin Color Detection Techniques. *GraphiCon Conference*.