



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA ELECTRÓNICA
EN CONTROL Y REDES INDUSTRIALES

”INTELIGENCIA ARTIFICIAL APLICADA A ROBOT
ASISTENCIAL PARA LA INTERACCIÓN CON NIÑOS EN EDAD
TEMPRANA”

Trabajo de titulación presentado para optar al grado académico de:
INGENIERO EN ELECTRÓNICA, CONTROL Y REDES
INDUSTRIALES

AUTORES: DIANA ISABEL CAYAMBE YAMBAY
JUAN ANDRÉS CASTAÑEDA CHATE
TUTOR: ING. EDWIN VINICIO ALTAMIRANO SANTILLÁN

Riobamba-Ecuador

2016

©2016, Diana Isabel Cayambe Yambay y Juan Andrés Castañeda Chate

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho del Autor.

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES
INDUSTRIALES

El Tribunal de Trabajo de Titulación certifica que: El trabajo de investigación: “INTELIGENCIA ARTIFICIAL APLICADA A ROBOT ASISTENCIAL PARA LA INTERACCIÓN CON NIÑOS EN EDAD TEMPRANA”, de responsabilidad de los señores Diana Isabel Cayambe Yambay y Juan Andrés Castañeda Chate, ha sido minuciosamente revisado por los Miembros del Tribunal de Tesis, quedando autorizada su presentación.

NOMBRE	FIRMA	FECHA
Dr. Miguel Tasambay. PhD. DECANO FIE	-----	-----
Ing. Jorge Paucar. DIRECTOR EIE CONTROL Y REDES INDUSTRIALES	-----	-----
Ing. Edwin Altamirano. DIRECTOR DE TRABAJO DE TITULACIÓN	-----	-----
Ing. Fernando Mejía. MIEMBRO DEL TRIBUNAL DE TRABAJO DE TITULACIÓN	-----	-----
DOCUMENTALISTA SISBIB – ESPOCH	-----	-----
NOTA:	-----	

Nosotros, DIANA ISABEL CAYAMBE YAMBAY y JUAN ANDRÉS CASTAÑEDA CHATE, somos responsables de las ideas, doctrinas y resultados expuestos en este Trabajo de Titulación, y el patrimonio intelectual de la misma pertenece a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO.

Diana Isabel Cayambe Yambay

Juan Andrés Castañeda Chate

DEDICATORIA

Ojala pudiera nombrar a cada una de las personas que estuvieron a mi lado desde que empezó lo que en un principio era un sueño y hoy se convierte en una realidad por ello dedico este logro a toda mi familia de manera especial a mis padres por su amor y apoyo incondicional en todos los momentos de mi vida, a mis hermanas y hermanos por ser un gran ejemplo de valentía y perseverancia, a mis hermosos sobrinos por los momentos de alegría que me regalan a diario y a la familia que escogí, mis amigos, por los momentos compartidos entre risas y tareas; este logro es para ustedes.

DIANA

Dedico este trabajo a toda mi familia por el apoyo incondicional y palabras de aliento, pues lucharon junto a mí hasta llegar al final y conseguir este triunfo. De manera especial a mi hija quien fue mi inspiración en cada paso que daba. A mis compañeros y amigos que estuvieron siempre en el camino apoyándome.

JUAN

AGRADECIMIENTO

El más sincero agradecimiento a la Escuela Superior Politécnica de Chimborazo por abrirnos sus puertas y formarnos como profesionales, a nuestros maestros que a lo largo de la carrera nos transmitieron sus conocimientos y anécdotas profesionales, a nuestro tutor de Trabajo de Titulación por su apoyo constante en el desarrollo del mismo.

DIANA Y JUAN

ÍNDICE DE ABREVIATURAS

A	Amperios
C	Tasa de descarga
CAMSHIFT	Continuously Adaptive Mean Shift
cm	Centímetros
DC	Corriente Directa
GDL	Grados de Libertad.
GPIO	General Purpose Input Output, Puerto de Entrada y Salida de Propósito General
g	Gramo
HD	High Definition (Alta definición)
HDMI	High Definition Multimedia Interface
HSV	Hue, Saturation, Value. (Tono, Saturación, Valor)
IA	Inteligencia Artificial.
IDE	Entorno de desarrollo integrado
Kg	Kilogramo
m	Metros.
m A	Miliamperios
mm	Milímetros
mAh	Mili amperios hora
MP	Megapíxeles.
PIR	Passive Infra Red
RAM	Random Access Memory (Memoria de acceso aleatorio)

RGB	Red, Green, Blue
SO	Sistema operativo
SoC	System on a Chip, Sistema en un Chip
μA	microamperios
V	Voltios.
VA	Visión Artificial.
VGA	Adaptador gráfico de Video
VNC	Virtual network Computing
2D	Dos dimensiones.
3D	Tres dimensiones.

TABLA DE CONTENIDO

PORTADA.....	i
DERECHOS DE AUTOR.....	ii
CERTIFICACIÓN.....	iii
DECLARACIÓN DE RESPONSABILIDAD.....	iv
DEDICATORIA.....	v
AGRADECIMIENTO.....	vi
ÍNDICE DE ABREVIATURAS.....	vii
ÍNDICE GENERAL.....	viii
ÍNDICE DE TABLAS.....	xiii
ÍNDICE DE FIGURAS.....	xiv
ÍNDICE DE ANEXOS.....	xvii
RESUMEN.....	xviii
SUMARY.	xix
INTRODUCCIÓN.....	1
PLANTEAMIENTO DEL PROBLEMA.....	2
JUSTIFICACIÓN.....	4
OBJETIVOS.....	6

CAPITULO I

1. MARCO TEÓRICO REFERENCIAL	7
1.1. Inteligencia Artificial	7
1.2. Visión Artificial	7
1.2.1. Imagen Digital.....	8
1.2.1.1. Modelo RGB.....	9
1.2.1.2. Modelo HSV.....	9
1.2.2. Elementos de un sistema de Visión Artificial.....	10
1.2.2.1. Fuente de Luz.....	11
1.2.2.2. Sensor de Imagen	13
1.2.2.3. Tarjeta de adquisición.....	13
1.2.2.4. Algoritmos de análisis de la imagen	14
1.2.2.5. Módulo de Proceso	14
1.2.2.6. Sistema de respuesta en tiempo real	14
1.2.3. Transformaciones Morfológicas aplicadas al procesamiento de imágenes	15

1.2.3.1.	<i>Dilatación Binaria</i>	15
1.2.3.2.	<i>Erosión Binaria</i>	16
1.2.3.3.	<i>Apertura</i>	16
1.2.3.4.	<i>Cierre</i>	17
1.2.3.5.	<i>Esqueletos</i>	17
1.2.4.	<i>Etapas de un Sistema de Visión Artificial</i>	18
1.3.	Introducción a la Robótica	19
1.4.	Estructura de Robots Móviles	20
1.5.	Dispositivos Sensoriales de los Robots Móviles	20
1.5.1.	<i>Sensor de Movimiento PIR</i>	21
1.5.2.	<i>Sensor de Proximidad</i>	21
1.6.	Locomoción de los Robots Móviles	22
1.6.1.	<i>Locomoción Diferencial</i>	22
1.6.2.	<i>Locomoción triciclo o coche</i>	23
1.7.	Fuente de energía usada en Robots Móviles	24
1.7.1.	<i>Baterías</i>	24
1.8.	Actuadores de Robots Móviles	25
1.8.1.	<i>Motores Reductores</i>	25
1.8.2.	<i>Servomotores</i>	25
1.8.3.	<i>Módulo de relés</i>	26
1.8.4.	<i>Switch Eléctrico</i>	28
1.9.	Algoritmos aplicados al Tracking de Objetos	28
1.9.1.	<i>Algoritmo de Umbralización aplicado al Tracking de Objetos</i>	29
1.9.2.	<i>Algoritmo de segmentación por color aplicado al Tracking de Objetos</i>	30
1.9.3.	<i>Algoritmo CAMShift aplicado al Tracking de Objetos</i>	30
1.10.	Impresión 3D	32
1.11.	La placa Raspberry Pi B+	33
1.11.1.	<i>Conectores disponibles de la placa Raspberry Pi B+</i>	34
1.11.2.	<i>Accesorios de la placa Raspberry Pi B+</i>	36
1.12.	Software utilizado para la programación y diseño del Robot	36
1.12.1.	<i>Sistema Operativo Raspbian Wheezy</i>	36
1.12.2.	<i>Software SolidWorks</i>	37
1.12.3.	<i>Lenguaje C++</i>	37
1.12.4.	<i>OpenCV</i>	38

CAPITULO II

2.	MARCO METODOLÓGICO	39
-----------	---------------------------------	-----------

2.1.	Instalación y Configuración de Software requerido	40
2.1.1.	<i>Instalación y Configuración de Raspbian Wheezy</i>	<i>40</i>
2.1.2.	<i>Instalación y Configuración de OpenCV</i>	<i>41</i>
2.1.3.	<i>Configuración para la conexión remota al Raspberry Pi B+ a través de VNC ...</i>	<i>44</i>
2.2.	Selección de Motores.....	45
2.2.1.	<i>Motores para el desplazamiento del robot</i>	<i>45</i>
2.2.2.	<i>Motores para el movimiento de los brazos del pingüino.....</i>	<i>46</i>
2.2.3.	<i>Motor para el mecanismo de pateo de la pelota</i>	<i>46</i>
2.3.	Selección de Sensores	47
2.3.1.	<i>Sensor de Movimiento.....</i>	<i>47</i>
2.3.2.	<i>Sensor de Proximidad</i>	<i>48</i>
2.4.	Selección de la Fuente de Alimentación	49
2.5.	Selección de la Cámara.....	50
2.6.	Diseño del Robot.....	52
2.6.1.	<i>Diseño y Modelado de la estructura del Robot en SolidWorks.....</i>	<i>52</i>
2.6.2.	<i>Diseño de la tabla de verdad para seleccionar el color a seguir.....</i>	<i>57</i>
2.7.	Configuración de puertos GPIO	57
2.8.	Diagramas de Conexión.....	58
2.8.1.	<i>Diagrama de conexión de Reguladores de voltaje</i>	<i>60</i>
2.8.2.	<i>Diagrama de conexión de Actuadores a la placa Raspberry Pi.....</i>	<i>61</i>
2.8.3.	<i>Diagrama de Conexión de Sensores a la placa Raspberry Pi.....</i>	<i>63</i>
2.8.4.	<i>Diagrama de Conexión del Switch a la placa Raspberry Pi.....</i>	<i>65</i>
2.9.	Conexión y Ensamblaje del Robot.....	65
2.9.1.	<i>Ensamblaje de la estructura externa del Robot</i>	<i>66</i>
2.10.	Códigos de Programación.....	71
2.10.1.	<i>Código de Programación para seleccionar el color de la pelota</i>	<i>71</i>
2.10.2.	<i>Programación del Algoritmo de Umbralización aplicado al tracking</i>	<i>72</i>
	<i>de la pelota.....</i>	<i>72</i>
2.10.3.	<i>Programación para controlar los actuadores.....</i>	<i>75</i>
2.10.4.	<i>Programación para controlar los sensores S1 y S2</i>	<i>77</i>
2.10.5.	<i>Programación para emitir mensajes de audio.....</i>	<i>78</i>
2.11.	Método de Aprendizaje de reconocimiento del objeto.....	79
2.12.	Descripción del funcionamiento del Robot	80
 CAPITULO III		
3.	MARCO DE RESULTADOS, ANÁLISIS Y DISCUSIÓN.....	83
3.1.	Pruebas de Funcionamiento del robot.....	83

3.1.1.	<i>Calibración de Sensores S1 y S2</i>	83
3.1.2.	<i>Calibración de velocidad de Motores</i>	85
3.1.3.	<i>Pruebas de seguimiento de la pelota</i>	86
3.2.	Análisis de Locomoción y Seguimiento de Trayectoria	90
3.3.	Detección y Corrección de errores	92
3.4.	Interacción Niño-Robot	93
3.5.	Resultados	94
	CONCLUSIONES	98
	RECOMENDACIONES	100
	BIBLIOGRAFÍA	
	ANEXOS	

ÍNDICE DE TABLAS

Tabla 2-1.	Lista de paquetes y dependencias utilizadas por OpenCV.	42
Tabla 2-2.	Parámetros de funcionamiento de los motores usados en el robot.....	45
Tabla 2-3.	Posición de los ángulos de los servomotores.	46
Tabla 2-4.	Ventajas y Desventajas del módulo de cámara y la cámara USB.....	51
Tabla 2-5.	Tabla de verdad para seleccionar el color de la pelota	57
Tabla 2-6.	Distribución de Puertos GPIO	58
Tabla 2-7.	Identificadores asignados a actuadores, sensores y módulos de relés.	59
Tabla 2-8.	Comparación entre Algoritmos aplicados al tracking de objetos	73
Tabla 2-9.	Mensajes emitidos por el robot.	79
Tabla 2-10.	Descripción de las etapas de funcionamiento del robot.....	81
Tabla 3-1.	Frecuencia en base al Voltaje aplicado al motor MRD	85
Tabla 3-2.	Coordenadas del objeto localizado a través del módulo de la cámara.....	89
Tabla 3-3.	Parámetros tomados en cuenta para las pruebas de interacción Niño-Robot.....	94

ÍNDICE DE FIGURAS

Figura 1-1.	Imagen Digital.....	8
Figura 1-2.	Valores correspondientes a los colores: rojo,.....	9
Figura 1-3.	Rueda de color del modelo HSV.....	9
Figura 1-4.	Editor de Imagen Digital.....	10
Figura 1-5.	Iluminación Difusa.....	11
Figura 1-6.	Iluminación Frontal.....	12
Figura 1-7.	Iluminación Direccional.....	12
Figura 1-8.	Iluminación Estructurada.....	13
Figura 1-9.	Imagen binarizada y su dilatación.....	15
Figura 1-10.	Imagen binarizada y erosionada.....	16
Figura 1-11.	Imagen con ruido vs imagen aplicada la apertura.....	16
Figura 1-12.	Aplicación de cierre a una imagen binarizada.....	17
Figura 1-13.	Imagen binarizada vs Imagen aplicada operación de esqueletos.....	17
Figura 1-14.	Diagrama de bloques de las etapas de un.....	18
Figura 1-15.	Estructura del Robot diseñada en SolidWorks.....	20
Figura 1-16.	Sensor de Movimiento PIR.....	21
Figura 1-17.	Sensor de Proximidad Ultrasónico.....	22
Figura 1-18.	Sistema de Locomoción Diferencial.....	23
Figura 1-19.	Sistema de Locomoción Triciclo.....	23
Figura 1-20.	Batería LIPO.....	24
Figura 1-21.	Motor reductor Pololu.....	25
Figura 1-22.	Servomotor HS-311.....	26
Figura 1-23.	Módulo de cuatro relés.....	27
Figura 1-24.	Circuito esquemático de un canal del Módulo de relés.....	27
Figura 1-25.	Switch de tres posiciones.....	28
Figura 1-26.	Esquema del Algoritmo de Umbralización.....	29
Figura 1-27.	Esquema del algoritmo de segmentación por color.....	30
Figura 1-28.	Esquema del algoritmo de seguimiento CAMShift.....	31
Figura 1-29.	Vista frontal de una impresora 3D.....	32
Figura 1-30.	Construcción de una pieza en una impresora 3D.....	33
Figura 1-31.	Esquema de la placa Raspberry Pi B+.....	34
Figura 1-32.	Diagrama de Puertos GPIO.....	35
Figura 2-1.	Menú Raspi-Config.....	40
Figura 2-2.	Escritorio de Raspbian Wheezy.....	41

Figura 2-3.	LXTerminal.....	42
Figura 2-4.	Micro motor reductor 50:1	47
Figura 2-5.	Parte posterior del Sensor PIR.....	48
Figura 2-6.	Cargador de Batería LIPO.....	50
Figura 2-7.	Módulo de la cámara de la placa Raspberry Pi	51
Figura 2-8.	Rueda del Robot diseñada en SolidWorks.....	52
Figura 2-9.	Buje de acoplamiento diseñado en SolidWorks.....	53
Figura 2-10.	Motor acoplado a la rueda diseñado en SolidWorks.....	53
Figura 2-11.	Brazo del robot diseñado en SolidWorks.....	53
Figura 2-12.	Mecanismo para patear la pelota diseñado en SolidWorks.....	54
Figura 2-13.	Acople del Motor al mecanismo para patear la pelota	54
Figura 2-14.	Cabeza del Robot diseñada en SolidWorks.....	55
Figura 2-15.	Parte superior del tronco del robot diseñada en SolidWorks	55
Figura 2-16.	Parte inferior del tronco del robot diseñada en SolidWorks	55
Figura 2-17.	Vista Superior de la parte baja del tronco del robot	56
Figura 2-18.	Vista Lateral del robot diseñada en SolidWorks	56
Figura 2-19.	Diagrama de Bloques de conexiones del robot	59
Figura 2-20.	Diagrama de conexión para regular voltaje de 8V DC a 5V DC	60
Figura 2-21.	Diagrama para regular voltaje de 12V DC a 6V DC.....	61
Figura 2-22.	Diagrama de conexión de los motores 131:1 y el micro motor 50:1 a la placa ... Raspberry Pi B+.....	62
Figura 2-23.	Diagrama de Conexión de los Servomotores a la placa Raspberry Pi B+.....	63
Figura 2-24.	Diagrama de Conexión de Sensores a la placa Raspberry Pi B+.....	64
Figura 2-25.	Circuito divisor de voltaje	64
Figura 2-26.	Diagrama de conexión del switch a la placa Raspberry Pi B+.....	65
Figura 2-27.	Ensamblaje de la cabeza del robot.	66
Figura 2-28.	Ensamblaje del tronco del robot.....	66
Figura 2-29.	Ensamblaje y acople del mecanismo de pateo de la pelota.....	67
Figura 2-30.	Ensamblaje y acople de los brazos en	67
Figura 2-31.	Ensamblaje y acople de las ruedas, bujes y motores en el robot.....	68
Figura 2-32.	Ensamblaje y acople de la rueda loca en la parte inferior del robot.....	68
Figura 2-33.	Disposición de los sensores.....	69
Figura 2-34.	Disposición de los altavoces en la parte interna de	69
Figura 2-35.	Ubicación de la placa Raspberry Pi B+, circuitos	70
Figura 2-36.	Disposición de Módulos MR1 y MR2	70
Figura 2-37.	Vista frontal y lateral del robot terminado	71
Figura 2-38.	Código implementado para controlar el switch de selección	72

Figura 2-39.	Código para iniciar la cámara.....	73
Figura 2-40.	Código para invocar a la imagen RGB.....	74
Figura 2-41.	Código para establecer parámetros de rango HSV para el color verde	74
Figura 2-42.	Código para convertir la imagen RGB a HSV.	74
Figura 2-43.	Código para aplicar procesos de umbralización y transformaciones morfológicas.....	74
Figura 2-44.	Código para calcular la posición X,Y.	75
Figura 2-45.	Código para imprimir resultados de posición X, Y	75
Figura 2-46.	Código para controlar un servomotor.....	76
Figura 2-47.	Código para activar los pines del módulo de relés.	76
Figura 2-48.	Código para controlar el sensor ultrasónico	77
Figura 2-49.	Código implementado para controlar el sensor PIR.....	78
Figura 2-50.	Código implementado para emitir archivos audio.....	79
Figura 2-51.	Código para determinar el contorno de un círculo	80
Figura 3-1.	Distancias en cm sensadas por el Sensor Ultrasónico HC-SR04	84
Figura 3-2.	Movimientos detectados por el Sensor PIR HC-SR501.....	85
Figura 3-3.	Tracking de la pelota roja.....	87
Figura 3-4.	Tracking de la pelota azul	87
Figura 3-5.	Tracking de la pelota verde	88
Figura 3-6.	Angulo de visión horizontal de la cámara.	90
Figura 3-7.	Angulo de visión vertical de la cámara	90
Figura 3-8.	Desplazamiento del robot en línea recta.....	91
Figura 3-9.	Desplazamiento del robot con giro hacia la izquierda.....	91
Figura 3-10.	Desplazamiento del robot con giro hacia la derecha	92

ÍNDICE DE ANEXOS

- ANEXO A. DATASHEET RASPBERRY PI B+
- ANEXO B. DATASHEET MODULO DE CÁMARA DE RASPBERRY PI B+
- ANEXO C. DATASHEET MICRO MOTOR REDUCTOR 50:1
- ANEXO D. DATASHEET MOTOR REDUCTOR 131:1
- ANEXO E. DATASHEET SERVOMOTOR HS-311
- ANEXO F. DATASHEET SENSOR ULTRASÓNICO HC-SR04
- ANEXO G. DATASHEET SENSOR PIR HC-SR501
- ANEXO H. CIRCUITO IMPLEMENTADO DEL DIAGRAMA DE CONEXIÓN DEL REGULADOR DE VOLTAJE DE 5V DC.
- ANEXO I. CIRCUITO IMPLEMENTADO DEL DIAGRAMA DE CONEXIÓN DEL REGULADOR DE VOLTAJE DE 5V DC.
- ANEXO J. IMPRESIÓN DE UNA DE LAS PIEZAS DEL ROBOT
- ANEXO K. PARÁMETROS PARA RECONOCER PATRONES DE FIGURAS

RESUMEN

El presente Trabajo de Titulación consiste en la aplicación de la Inteligencia Artificial a un robot asistencial para la interacción con niños en edad temprana en la Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Ingeniería Electrónica en Control y Redes Industriales. Con esta investigación el usuario dispondrá de un robot en forma de pingüino que interactúa con el niño en el típico juego de buscar y patear la pelota. Se utilizó el método Inductivo para obtener ideas generales en la programación a partir de premisas particulares y el método Experimental para el desarrollo de pruebas y corrección de errores en el funcionamiento del robot. El prototipo consta de un minicomputador Raspberry Pi sobre el cual se realizó la programación mediante el uso de bibliotecas del software Open Source que controlan: el tracking de la pelota a través de la cámara OMNI-VISION, los motores reductores que permiten el desplazamiento del robot, los dispositivos sensoriales que detectan la presencia del niño y de obstáculos, los altavoces que emiten mensajes pregrabados aleatoriamente. Se implementó el Algoritmo de Umbralización, mediante el cual el robot busca y sigue la pelota en el transcurso del juego. El robot interactuó con diez niños de 1 a 5 años de edad, el 80% de ellos mostraron mayor interés en el juego durante 25 minutos, incentivando así su desarrollo intelectual y psicomotriz; mientras que 20% de ellos se mostraron tímidos en la interacción con el robot con un tiempo de juego de 5 minutos. El prototipo en forma de pingüino incorpora los elementos básicos de un Sistema de Visión Artificial, logrando así un robot de aspecto amigable con los niños. Se recomienda a los estudiantes de la Escuela de Ingeniería Electrónica, tomar los resultados de esta investigación para profundizar los conocimientos en el amplio campo de la Inteligencia Artificial.

PALABRAS CLAVE

<INTELIGENCIA ARTIFICIAL>, <VISIÓN ARTIFICIAL>, <OPEN SOURCE>, <ALGORITMO>, <MINICOMPUTADOR (RASPBERRY PI)>, <SEGUIMIENTO DE OBJETOS (TRACKING)>, <CÁMARA OMNI VISION>, <ROBOT MÓVIL> <DESARROLLO PSICOMOTRIZ E INTELECTUAL>, <CONTROL Y REDES INDUSTRIALES>.

SUMMARY

This work involves the application Degree of Artificial Intelligence to a medical robot for interacting with children at an early age at the Polytechnic School of Chimborazo, Faculty of Informatics and Electronics, Electronics Engineering in Control and Industrial Networking. Through research, the user will have a penguin shaped robot that interacts with the child in the typical game of seek and kick the ball. The inductive method was used to obtain general ideas in programming from particular premises and the experimental method to test development and correction of errors in the operation of the robot. The prototype consists of a minicomputer Raspberry Pi on which programming is performed by using libraries of Open Source software that control: tracking the ball through the OMNI-VISION camera gear motors that allow the movement of the robot, sensory devices that detect the presence of the child and obstacles, the speakers that emit randomly prerecorded messages. Thresholding Algorithm, by which the robot seeks out and follows the ball during the game was implemented. The robot interacted with ten children 1 to 5 years old, 80% of them showed more interest in the game for 25 minutes, thus encouraging their intellectual and psychomotor development; while 20% of them were shy in the interaction with the robot with a playing time of 5 minutes. The penguin shaped prototype incorporates the basic elements of a machine vision system, thus achieving a friendly looking robot with children. It is recommended to students of the School of Electronic Engineering to take the results of this research to deepen knowledge in the broad field of Artificial Intelligence.

KEYWORDS

<ARTIFICIAL INTELLIGENCE>, <VISION>, <OPEN SOURCE>, <ALGORITHM>, <MINICOMPUTER (RASPBERRY PI)>, <OBJECT TRACKING (TRACKING)>, <OMNI VISION CAMERA>, <MOBILE ROBOT>
<PSYCHOMOTOR AND INTELLECTUAL DEVELOPMENT>, <CONTROL AND INDUSTRIAL NETWORKS>.

INTRODUCCIÓN

Hoy en día la tecnología ha tenido un crecimiento notable, permitiendo al ser humano desarrollar aplicaciones que facilitan tareas que pueden resultar rutinarias y tediosas. Tanto en software como en hardware existe una actualización constante en el mercado.

En lo que se refiere a Hardware tenemos la placa Raspberry Pi que es un minicomputador de bajo costo que cuenta con su propio sistema operativo conocido como Raspbian, salió a la venta en el año 2011, aunque en un principio estuvo dirigido a estudiantes como una herramienta de aprendizaje y plataforma de desarrollo de proyectos pequeños; en la actualidad se ha convertido en una sistema embebido empleado en proyectos de mucha relevancia.

La Robótica en conjunto con la Visión Artificial ha desarrollado aplicaciones dirigidas a la manipulación y reconocimiento de objetos, ensamblaje de piezas, reconocimiento y determinación de la posición de objetos, entre otras. Todas estas aplicaciones han sido dirigidas en su mayoría al área de la industria; por este motivo este trabajo se ha encaminado a colaborar con el desarrollo intelectual y psicomotriz de los niños.

El presente trabajo fusiona diferentes ciencias como: la Robótica, la Visión Artificial, la Electrónica; para crear un prototipo en forma de pingüino que interactuara con el niño en un juego de pelota.

La Visión Artificial permite el tratamiento de las imágenes en tiempo real a través del módulo de cámara. Se escogió el algoritmo más óptimo para realizar la segmentación y seguimiento de la pelota.

El prototipo usa actuadores que le permiten desplazarse en el entorno en el que se encuentra y dispone de sensores que ayudan a conservar la integridad del niño y evitar que el robot no sufra averías por choques. A medida que el juego avance el robot emitirá mensajes pregrabados para así lograr un vínculo con el niño y estimularlo en su desarrollo.

La placa Raspberry Pi sirve de plataforma en este proyecto; puesto que sobre esta placa se instaló el software necesario para controlar los actuadores, sensores y el tracking de la pelota.

Con este trabajo se dio un enfoque diferente a la Visión Artificial y la Robótica; pues está dirigido a los más pequeños del hogar que son los niños.

PLANTEAMIENTO DEL PROBLEMA

Antecedentes

Un desarrollo poco adecuado de las capacidades cognitivas en los niños puede generar problemas en su desarrollo mental, físico y psicológico. Desde hace tiempo atrás la humanidad tenía conocimiento que al dejar jugar a sus hijos de forma independiente ellos potenciaban su capacidad cognitiva, además de que aprendían a relacionarse con otros niños ya que incrementan las habilidades de atención, percepción y memoria.

Los problemas con el desarrollo psicomotriz ha influido negativamente en el ser humano, esta deficiencia impide que la persona explore, investigue, viva sus emociones y conflictos, aprenda a superar situaciones, a enfrentarse a sus límites, a sus miedos y deseos, a relacionarse con los demás y asumir roles; provocando el disminución de habilidades innatas de la humanidad.

“La psicomotricidad es la acción del sistema nervioso central, que crea una conciencia en el ser humano sobre los movimientos que realiza, a través de los patrones motores como la velocidad, el espacio y el tiempo.” Según la web guarderiainfantil.com

Según la web cosasdelainfancia.com “En los primeros años de vida, la Psicomotricidad juega un papel muy importante, porque influye valiosamente en el desarrollo intelectual, afectivo y social del niño favoreciendo la relación con su entorno y tomando en cuenta las diferencias individuales, necesidades e intereses de los niños y las niñas.

- A nivel motor, le permitirá al niño dominar su movimiento corporal.
- A nivel cognitivo, permite la mejora de la memoria, la atención y concentración y la creatividad del niño.
- A nivel social y afectivo, permitirá a los niños conocer y afrontar sus miedos y relacionarse con los demás.”

Para ello se busca soluciones al normal desarrollo de sus habilidades cognitivas, atacando el problema en los primeros años de vida del ser humano, ya sea en el proceso de vida adecuado o enfermedades que no les permite perfeccionar sus habilidades al máximo.

Hoy en día se puede utilizar la tecnología para ayudar a los niños desde una edad muy temprana y personas con capacidades especiales, creando módulos y maquinas psicomotrices, estas han sido de gran ayuda y siempre se está tratando de mejorar dichas maquinas aprovechando el avance de la tecnología.

Ortiz Rojas Jaime, 2004 realizó una tesis de grado titulada “DISEÑO Y REALIZACIÓN DE MAXCOTA, UN ROBOT INTELIGENTE PARA NIÑOS”, se basa en la interacción del niño con el robot, juegos tales como: “buscar al usuario”, “huir del usuario” y “navegar en su entorno” y esto nos servirá para determinar que atractivo crea el vínculo con el niño.

Sobrado Malpartida Eddie Ángel en 2003 realizó una tesis de maestría titulada “SISTEMA DE VISIÓN ARTIFICIAL PARA EL RECONOCIMIENTO Y MANIPULACIÓN DE OBJETOS UTILIZANDO UN BRAZO ROBOT “, su estudio se basó en la visión artificial uso tracking de un objeto para su localización, procesamiento de imagen en movimiento.

Torres Vidal Ernesto en 2009 realizó un trabajo final de tesis de grado titulada “SISTEMA DE INTELIGENCIA ARTIFICIAL PARA EL CONTROL DE ROBOTS AUTÓNOMOS”, su trabajo se basó en la inteligencia artificial, en el cual encuentra un objeto y usa un sistema de control RF (joystick) para su localización.

A partir de los estudios mencionados se propone diseñar un prototipo que localice un objeto (pelota), a través de una cámara y de forma autónoma se dirija hacia él y empujándolo hacia el niño para realizar en mismo proceso y así interactuarán mutuamente para establecer un vínculo y así lograr un desarrollo óptimo de las capacidades psicomotrices del niño.

Formulación del problema

En la actualidad existe una gran diversidad de trabajos basados en la Robótica, Inteligencia Artificial y Visión Artificial, la mayoría de ellos enfocados a mejorar la vida de las personas. Existen diversos tipos de robots, como son los industriales, los móviles, los de servicio y los denominados robots inteligentes; cada uno tiene una aplicación distinta.

Es de conocimiento que en la etapa de la infancia es cuando el ser humano adquiere la mayoría de habilidades cognitivas y psicomotoras que le servirán para desenvolverse con mayor

facilidad en la sociedad. Al no contar con las herramientas necesarias y la debida atención para el desarrollo del niño, este crecerá con limitaciones intelectuales y motrices.

Es por esta razón que se realizará el Proyecto de Titulación, con el que se pretende ayudar a los más pequeños del hogar a desarrollar sus capacidades al máximo con la asistencia de un Robot que interactuará con el niño en un juego de pelota. Al jugar el niño con el robot ejercitará sus órganos sensoriales, sus movimientos y su lenguaje

Sistematización del problema

1. ¿Qué tecnología a utilizar de los diferentes sistemas embebidos existentes será eficiente en este proyecto para lograr los objetivos propuestos?
2. ¿Qué tipo de locomoción será la más óptima para el tipo de movimiento que realizará el prototipo?
3. ¿Ayudará la Visión Artificial a que el prototipo localice una pelota en el espacio; dando así una ubicación más exacta?
4. ¿Qué herramientas de OPEN-SOURCE serán las más indicadas para resolver los algoritmos propuestos?
5. ¿Cómo influirá el prototipo en el desarrollo intelectual y psicomotriz del niño?

JUSTIFICACIÓN

Justificación teórica

Con la inteligencia artificial los mecanismos no harán acciones repetitivas, al contrario realizarán acciones donde ellos por si solos tomarán las decisiones según sea lo más óptimo tomando en cuenta el entorno y las limitaciones que este tenga.

La robótica tiene como finalidad ayudar al ser humano a tener vida más sencilla, para esto existen mecanismos autónomos que realizan diferentes procesos con una finalidad específica y con el avance de la tecnología surge la necesidad de que el mecanismo tome decisiones por sí solo.

Los juguetes psicomotrices han logrado tener efecto positivo, la mayoría son comercializados para niños de 6 años en adelante, y esto genera la idea de diseñar un prototipo que interactúe con los niños en edad temprana.

Con la tecnología adecuada se busca que los niños se sientan a gusto con el prototipo y a su vez les sirva como ayuda en su desarrollo en una etapa de su vida, ayudándolo a descubrir habilidades que él tiene, pero tratando de perfeccionarlas y estimulando a su memoria, todo esto con la ayuda de un modelo robótico.

Justificación aplicativa

Para lograr los objetivos planteados se tendrá que utilizar recursos de un sistema embebido que será nuestra placa de control y mando de nuestro prototipo, el cual permitirá controlar los diferentes recursos a usar como son motores dc, cámara de video (módulo raspberry OMNI-VISION), sensores y diferentes componentes pasivos.

El sistema embebido a usar será la placa Raspberry Pi B, que es un minicomputador compacto y servirá de plataforma de programación y control de periféricos de salida, elementos eléctricos y electrónicos.

Un recurso fundamental de este proyecto será la cámara de video que cuenta con resolución de 5 MP, capaz de capturar vídeo a 1080p (1920x1080 pixeles) ya que con la visión artificial y el software adecuado permitirá la localización del objeto en el espacio lo que se conoce como tracking.

Los actuadores del prototipo serán los motores, con los cuales se moverá en el espacio en diferentes direcciones. Los servomotores serán usados para controlar el movimiento de los brazos del robot, dichos movimientos serán constantes a medida que el robot se desplace.

Los sensores a usar permitirán al robot detectar la presencia del niño, así como también obstáculos como por ejemplo paredes.

El robot interactuará con el niño en el típico juego de encontrar y patear la pelota, para esto el robot tendrá prioridad de no chocar y lastimar al niño, el prototipo al buscar y patear la pelota

esperara un tiempo determinado para que el niño realice el mismo procedimiento, además reproducirá mensajes pregrabados a medida que avance el juego; para así lograr un vínculo afectivo con el niño y evitar que el juego se convierta en aburrido y monótono.

El propósito general de este proyecto es ayudar a la niñez con un robot de entretenimiento para los más pequeños, que a la vez colabora con su desarrollo psicomotriz y capacidades cognitivas, el niño al interactuar desarrollará habilidades comunes del ser humano como caminar, correr, “Cultivará la capacidad perceptiva a través del conocimiento, de los movimientos y de la respuesta corporal y además ayuda a crear una conciencia y un respeto a la presencia y al espacio de los demás.” Según la web guarderiainfantil.com.

OBJETIVOS

Objetivos generales

- Aplicar la inteligencia artificial a robot asistencial para la interacción con niños en edad temprana.

Objetivos específicos

- Utilizar el mejor algoritmo enfocado el tracking de objetos que permita al prototipo localizar y mover el objeto para realizar un nuevo seguimiento.
- Realizar la programación del prototipo mediante el software OPEN SOURCE.
- Realizar el diseño del prototipo en forma de pingüino de aspecto amigable para los niños, usando las herramientas de SOLIDWORKS.
- Localizar la pelota a través de una cámara aplicando visión artificial, para así lograr la interacción con el niño.
- Reproducir mensajes pregrabados en el prototipo, aleatoriamente a medida que avanza el juego.
- Realizar la interacción del niño con el robot usando una pelota pequeña.

CAPITULO I

1. MARCO TEÓRICO REFERENCIAL

1.1. Inteligencia Artificial

La Inteligencia Artificial (IA) es la rama de la ciencia que se encarga del estudio de la inteligencia en elementos artificiales y, desde el punto de vista de la ingeniería, propone la creación de elementos que posean un comportamiento inteligente como son: el aprendizaje, la capacidad de adaptación a entornos cambiantes, la creatividad, etc. (Pazos, Pedreira, Rabuñal y Pereira, 2007, p 30).

La IA abarca muchos campos como son: la Neurociencia, la Psicología, las Matemáticas, Biología, etc. La IA apoyada en otras ciencias consolida sistemas que actúan de forma racional, adaptándose a un ambiente específico.

1.2. Visión Artificial

Se puede decir que la Visión Artificial (VA) describe la adquisición de una o varias imágenes bidimensionales del entorno; para obtener de ellas su estructura y propiedades dentro de un mundo tridimensional que puede ser dinámico. Las imágenes pueden ser monocromáticas o colore (Eddie Sobrado, 2003, p 1-1).

La VA intenta recoger el mayor número de características de una escena u objeto que se obtendrían mediante la visión humana; sin embargo está muy lejos de igualarla. A pesar de las limitaciones con las que se enfrenta la VA, también presenta ventajas para facilitar mecanismos y son las siguientes:

- Permite medir de una forma más exacta magnitudes físicas.
- Mejor realización de tareas repetitivas.
- Efectuar trabajos de bajo nivel de proceso.

- Ayudar al ser humano en tareas rutinarias.
- Colaborar en el desarrollo integral y psicomotriz de niños.

El campo de Aplicación de la Visión Artificial es amplio; se usa en áreas de producción como: Control de Calidad, Biomédica, Robótica, entre otros. El presente documento hace referencia a la aplicación en el campo de la Robótica; para lo cual se vinculan varias ciencias; para así lograr un Sistema de Visión Artificial óptimo a los objetivos planteados.

1.2.1. Imagen Digital

Una imagen digital es una representación bidimensional de una imagen, mediante el uso de bits ya sean 0 o 1; dichas imágenes son capturadas a través de una cámara fotográfica. Una imagen digital presenta características como: dimensión y espacio de color.

Una imagen digital está formada por una retícula rectangular, esta a su vez está formada por celdas, que son la menor unidad en color de una imagen digital. A la mencionada celda se le conoce con el nombre de pixel. Cada pixel está formado por tres subpixeles: rojo, verde y azul; cada uno con diferente intensidad de brillo (Enrique Benimeli, 2011, p.1).

La resolución de una imagen digital es la cantidad de pixeles por pulgada. A una mayor resolución se obtendrá una información digital más detallada d la imagen.

Existen diversos códigos diseñados para representar espacios o modelos de colores de una imagen digital. Para el proyecto se utilizará dos tipos de modelos para el procesamiento de imágenes: el Modelo RGB y el modelo HSV.

En la **Figura 1-1**. Se ilustra una imagen digital en la que se muestra los pixeles de los que está formado el ojo.

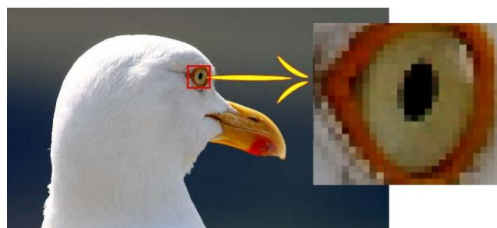


Figura 1-1. Imagen Digital
Fuente: Los colores en Informática, 2011.

1.2.1.1. Modelo RGB

El modelo RGB de un color hace referencia a la composición del color según la intensidad de brillo de los colores primarios que son: rojo, verde y azul. Mediante la mezcla de los colores citados a distintos niveles de intensidad es posible generar una gama de colores. Un color está formado por una codificación de 8 bits. La intensidad de cada color se mide en un rango que va desde valores de 0 a 255 (Enrique Benimeli, 2011, p 3-4).

En la **Figura 1-2.** se ilustra los valores correspondientes a los colores: rojo verde, azul, negro, blanco y gris.

R	G	B
255	0	0
Rojo		

R	G	B
0	255	0
Verde		

R	G	B
0	0	255
Azul		

R	G	B
0	0	0
Negro		

R	G	B
255	255	255
Blanco		

R	G	B
128	128	128
Gris		

Figura 1-2. Valores correspondientes a los colores: rojo, verde, azul, negro blanco y gris.

Fuente: Los colores en Informática, 2011.

1.2.1.2. Modelo HSV

El modelo HSV (Hue, Saturation, Value/ Tono Saturación, Valor) define un color en parámetros de sus componentes; como son matiz, saturación y valor.

El modelo HSV está basado en una rueda de color ilustrado en la **Figura 1-3.** En la que se indica el tono mediante un ángulo mientras que la saturación y el brillo se denotan mediante un porcentaje.

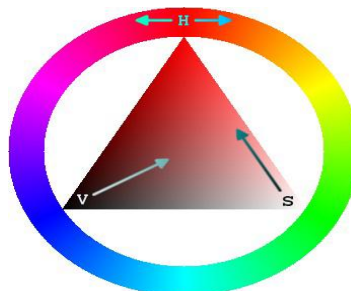


Figura 1-3. Rueda de color del modelo HSV

Fuente: Los colores en Informática, 2011.

- El matiz (H) se expresa en un rango de grados que vienen dados en valores de 0 a 360°. A los colores rojos, amarillo y verde les corresponden valores de 0°, 60° y 120° respectivamente.
- Los valores de Saturación (S) vienen dados en rangos de porcentaje que van desde a al 100%. Mientras más baja sea la saturación el color tenderá a decolorarse.
- El valor (V) se mide en porcentajes que van desde 0 a 100%. Dichos valores dependen de la saturación.

El modelo HSV es el más utilizado en algoritmos de procesamiento de imágenes, ya que este modelo permite procesar el color de una forma similar en la que el ser humano percibe el mismo color (Enrique Benimeli, 2011, p.5-6).

En la **Figura 1-4.** se muestra el editor de una imagen, mediante el cual es posible definir el rango de un color específico.

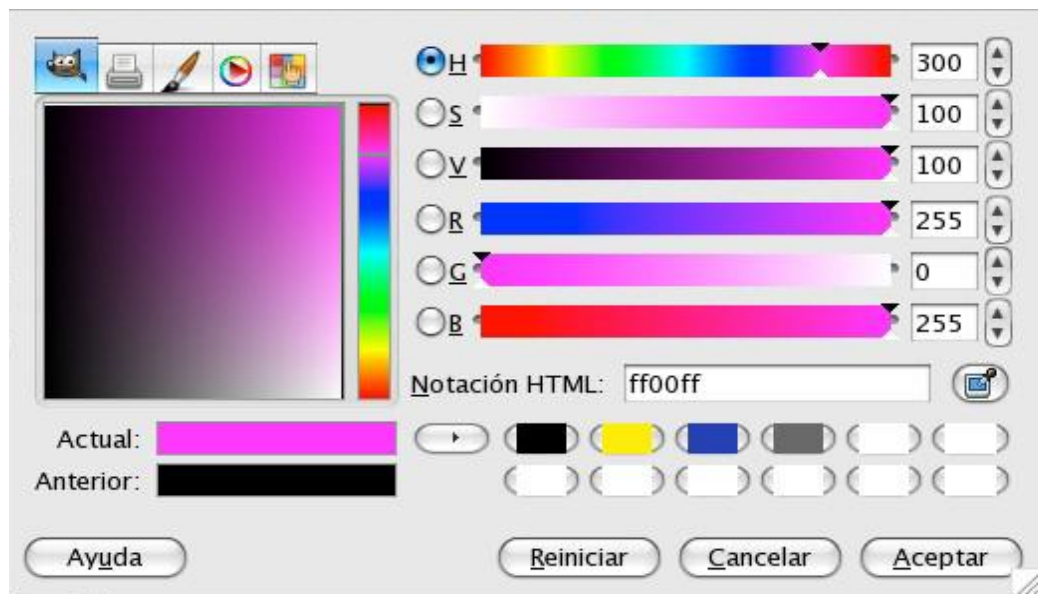


Figura 1-4. Editor de Imagen Digital.
Fuente: Los colores en Informática, 2011.

1.2.2. Elementos de un sistema de Visión Artificial

En la Visión Artificial intervienen dos subsistemas principales; el primero se encarga de adquisición de la imagen, mientras que el segundo se encarga de procesar dichas imágenes. Dichos subsistemas están conformados a la vez por los siguientes elementos:

1.2.2.1. Fuente de Luz

La fuente de Luz es la encargada de iluminar la escena a visualizar, este es un apartado de vital importancia; ya que tiene gran influencia en el óptimo desempeño del sistema de VA; pues dependiendo de la cantidad de luz emitida, el fondo y la posición de la cámara en la escena; los colores pueden variar. Dichos parámetros deben ajustarse al objeto que se desea iluminar (Eddie Sobrado, 2003, p 2-3,2-6).

Existen diversos tipos de iluminación que se aplican a sistemas de Visión Artificial dependiendo de las necesidades y condiciones del entorno, además de rescatar los rasgos de una determinada escena u objeto. Los tipos de iluminación son los siguientes:

- **Iluminación de Retroalimentación Difusa:** En este tipo de iluminación se coloca la fuente de luz contra la cámara y el objeto a iluminar se encuentra en medio. Por lo general esta iluminación se usa para obtener las características de un objeto, sin embargo se pierde los detalles del resto de la escena. Además el objeto o pieza a iluminar debe encontrarse fija.

En la **Figura 1-5.** se muestra la Iluminación Difusa aplicada a una escena u objeto.

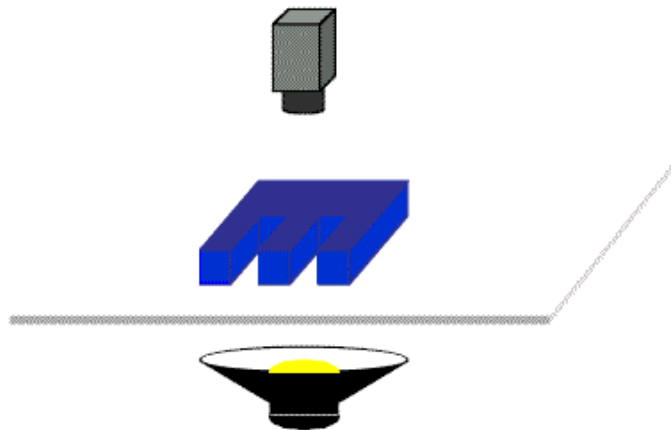


Figura 1-5. Iluminación Difusa

Fuente: Sistema de VA para el reconocimiento y manipulación de objetos utilizando un brazo robot, 2003.

- **Iluminación Frontal:** Es la más usada ya que se enfoca al objeto o escena frontalmente. Sin embargo se puede tener problemas con objetos que sean demasiado reflectores, provocando brillos.

En la **Figura 1-6.** se muestra la Iluminación Frontal aplicada a una escena u objeto.

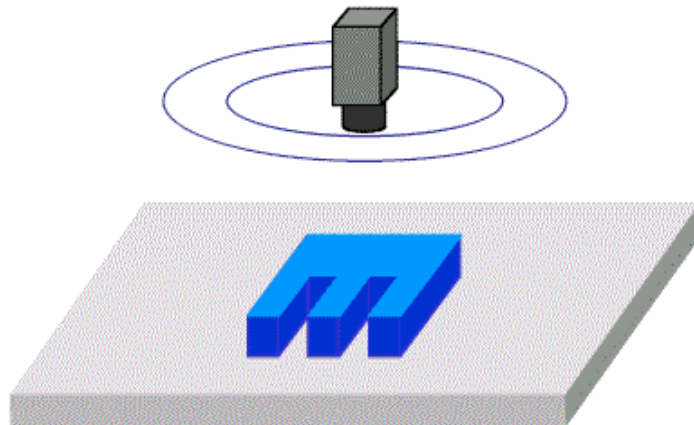


Figura 1-6. Iluminación Frontal

Fuente: Sistema de VA para el reconocimiento y manipulación de objetos utilizando un brazo robot, 2003.

- **Iluminación Direccional:** Consiste en iluminar desde algún sentido la escena, puede ser desde alguna esquina o desde el centro de la habitación; para así obtener características específicas del objeto.

En la **Figura 1-7.** se muestra la Iluminación Direccional aplicada a una escena u objeto.

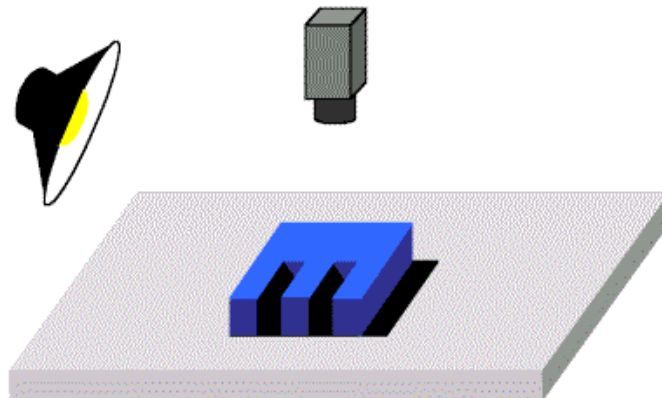


Figura 1-7. Iluminación Direccional

Fuente: Sistema de VA para el reconocimiento y manipulación de objetos utilizando un brazo robot, 2003.

- **Iluminación Estructurada:** En esta tipo de iluminación se usa por lo general una fuente de luz láser; la misma que atraviesa unas rejillas para posteriormente enfocar el objeto o escena. Generalmente se usa para conocer la forma de objetos.

En la **Figura 1-8.** se muestra la Iluminación Estructurada aplicada a una escena u objeto.

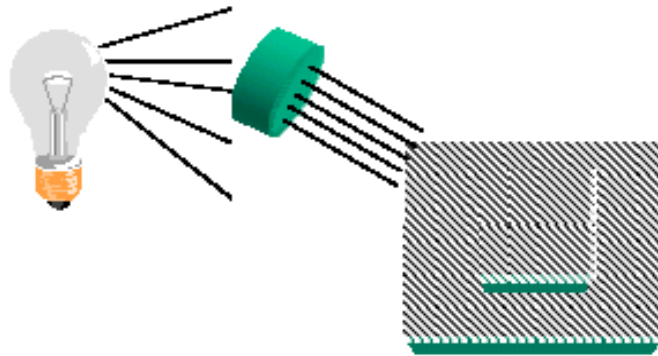


Figura 1-8. Iluminación Estructurada

Fuente: Sistema de VA para el reconocimiento y manipulación de objetos utilizando un brazo robot, 2003.

1.2.2.2. Sensor de Imagen

Este subsistema está destinado a la adquisición y captura de las características del objeto o escena a analizar. Para dicho sistema se utilizan diversos tipos de cámara dependiendo de la necesidad de cada proyecto (Universidad Nacional de Quilmes, Cátedra: Visión Artificial, 2005, p.3).

En el presente trabajo se tiene dos opciones en cuanto al sensor de imagen; la primera es el módulo de la cámara de la placa Raspberry Pi y otra alternativa es el uso de una cámara USB.

1.2.2.3. Tarjeta de adquisición

La tarjeta de adquisición es la encargada de transportar la imagen capturada por la cámara hacia el módulo de proceso o computador.

En este componente radican aspectos importantes para el procesamiento óptimo de la imagen como son: la velocidad de transmisión, la profundidad del pixel, la velocidad de transferencia de datos. Todos estos parámetros aseguran que la imagen recibida por el computador conserve en su mayoría las características reales del objeto o escena (Aspectos de un Sistema de Visión Artificial, 2005, p.4).

1.2.2.4. *Algoritmos de análisis de la imagen*

Existen diversos algoritmos para analizar una imagen; todo depende de la aplicación que se le quiera dar. En base a lo descrito en el documento Aspectos de un Sistema de Visión Artificial (2005, p.5-6) existen diversas etapas para un proceso de reconocimiento de formas en una escena; a continuación se describen:

- **Filtrado o Pre-procesamiento:** En esta etapa se aplican transformaciones a la imagen capturada para así corregir las variaciones en la escena ocasionadas por ruido, deficiencia en la fuente luz, etc.
- **Aislamiento de los Objetos de interés:** Permite separar por regiones la escena filtrada mediante diversas técnicas adaptadas a la realidad de cada situación.
- **Cálculo, clasificación y selección de características:** Una vez definidos los objetos a analizar es necesario conocer sus características; las mismas que deben tener valores numéricos diferentes para objetos distintos y en el caso de pertenecer a la misma clase de objetos deben tener valores semejantes.

1.2.2.5. *Módulo de Proceso*

Es el sistema encargado de recibir y analizar las imágenes. La selección del módulo de proceso depende de las necesidades del sistema de Visión Artificial, hay que tener en cuenta la velocidad al procesar la imagen en tiempo real (Aspectos de un Sistema de Visión Artificial, 2005, p.7-8). En el presente trabajo se utilizó como Módulo de Proceso a la placa Raspberry Pi.

1.2.2.6. *Sistema de respuesta en tiempo real*

Este sistema usa la información recibida desde la escena para tomar decisiones con respecto al entorno, como pueden ser de movimiento o desplazamiento; mediante actuadores, sensores, baterías, entre otros (Aspectos de un Sistema de Visión Artificial, 2005, p.9-10).

1.2.3. Transformaciones Morfológicas aplicadas al procesamiento de imágenes

Las Transformaciones Morfológicas se encargan de modificar la geometría y forma de un objeto, mediante el uso de herramientas matemáticas que permiten obtener valores en cuanto a las características de forma y estructura de la imagen tratada.

Entre las principales utilidades que brindan las Transformaciones Morfológicas tenemos: separación de objetos unos de otros, obtención de contornos, descomposición de formas de objetos complejas en formas de objetos simples, supresión de ruidos y simplificación de formas, descripción de objetos basado en área, perímetros y la reconstrucción de elementos distorsionados en una imagen (Técnicas y Algoritmos Básicos de Visión Artificial, 2006, p 55-60).

A continuación se detallan las principales herramientas matemáticas que ofrecen las Transformaciones Morfológicas.

1.2.3.1. Dilatación Binaria.

La dilatación Binaria produce un efecto de engrosamiento de los bordes de los objetos, con esto se logra aumentar el contorno de objetos y unir líneas discontinuas.

Dada una imagen A y un elemento estructural B; ambas son imágenes binarias con fondo blanco; se procede a desplazar el elemento estructural B por toda la imagen A; en caso de coincidir un elemento de la matriz de la imagen A con un pixel del elemento estructural B el pixel en común se pone a uno (Edmans, 2006, p 55).

En la **Figura 1-9.** se ilustra una imagen binarizada y su dilatación

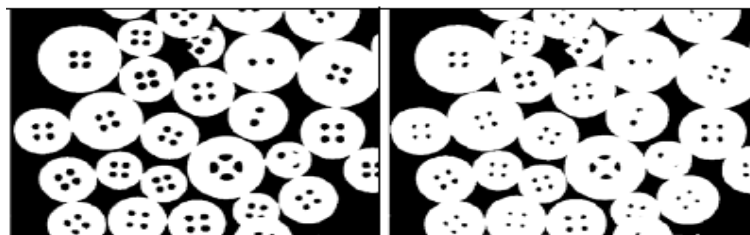


Figura 1-9. Imagen binarizada y su dilatación
Fuente: Técnica y Algoritmos Básicos de Visión Artificial

1.2.3.2. Erosión Binaria

La erosión binaria es una función dual que permite combinar los resultados obtenidos de la dilatación y erosión; mientras que la dilatación engrosa y expande los bordes de los objetos reduce los contornos de los objetos para así separar los objetos que están unidos en pequeñas secciones de sus contornos (Edmans, 2006, p 57).

En la **Figura 1-10.** se ilustra una imagen binarizada y su erosión.

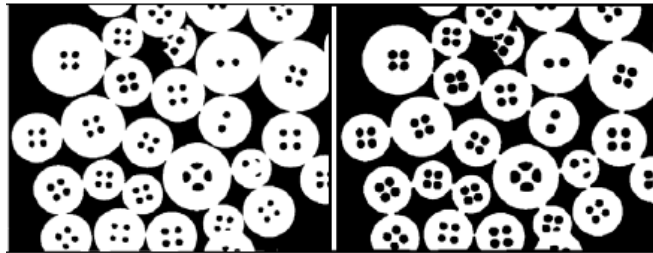


Figura 1-10. Imagen binarizada y erosionada
Fuente: Técnica y Algoritmos Básicos de Visión Artificial

1.2.3.3. Apertura

La operación de apertura de una imagen consiste en realizar primero la erosión y luego la dilatación de la misma. La apertura puede estar formada por una combinación de números de erosiones y dilataciones diferentes y es utilizada para separar objetos de diferentes formas, descomponer un objeto en partes más sencillas, aumentar agujeros que están dentro de objetos, extracción de formas (Edmans, 2006, p 58).

En la **Figura 1-11.** se ilustra una imagen binarizada y su apertura.

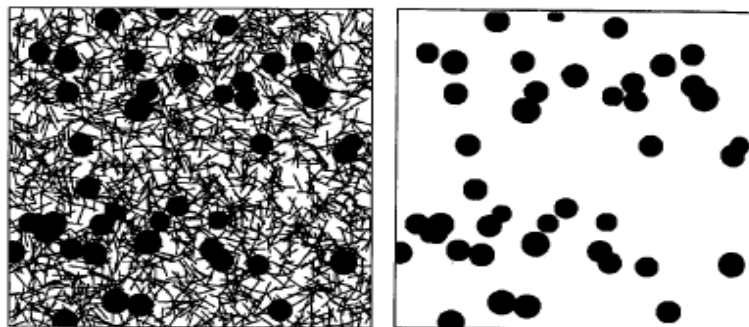


Figura 1-11. Imagen con ruido vs imagen aplicada la apertura.
Fuente: Técnica y Algoritmos Básicos de Visión Artificial

1.2.3.4. Cierre

El cierre es la operación opuesta a la apertura, realizando primero la dilatación y luego la erosión de la imagen. La combinación de la dilatación y erosión en segundo lugar ayuda en el filtrado, segmentación de la imagen (Edmans, 2006, p 58).

En la **Figura 1-12**, se ilustra la aplicación de cierre a una imagen binarizada.

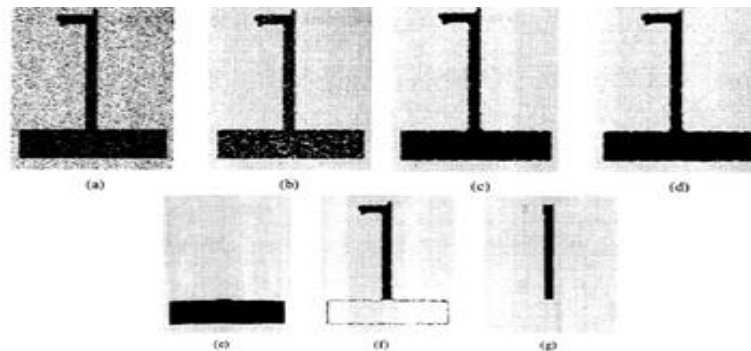


Figura 1-12. Aplicación de cierre a una imagen binarizada
Fuente: Técnica y Algoritmos Básicos de Visión Artificial

1.2.3.5. Esqueletos

En esta operación se reducen los objetos de una imagen aun esqueleto del grosor de un pixel. En este tipo de operación el tiempo juega un papel importante ya que solo se puede implementar si se cuenta con el hardware específico para acelerar los cálculos de dicha operación (Edmans, 2006, p 60).

En la **Figura 1-13**, se ilustra el resultado de aplicar la operación de esqueletos a la imagen de la izquierda.

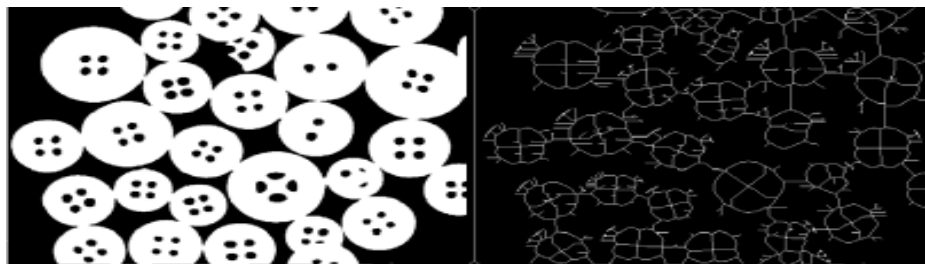


Figura 1-13. Imagen binarizada vs Imagen aplicada operación de esqueletos.
Fuente: Técnica y Algoritmos Básicos de Visión Artificial

1.2.4. Etapas de un Sistema de Visión Artificial

Un sistema de Visión Artificial necesita de un conjunto de procesos ordenados sistemáticamente; los mismos que están destinados a procesar las características de una escena u objeto, para que posteriormente en conjunto con sensores, actuadores; se pueda cumplir con el objetivo específico del sistema (Técnicas y Algoritmos Básicos de Visión Artificial, 2006, p 15).

Las etapas de un sistema de Visión Artificial son:

- Etapa Sensorial.
- Etapa de pre-procesado.
- Etapa de segmentación.
- Etapa de parametrización.
- Etapa de clasificación.
- Etapa de actuación según resultados.

En la **Figura 1-14**. se muestra el diagrama de un Sistema Visión Artificial.

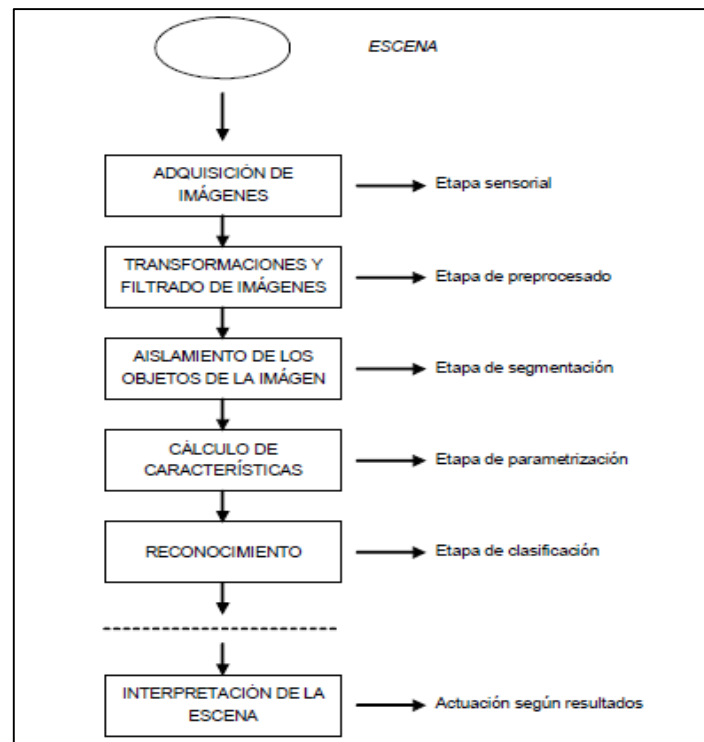


Figura 1-14. Diagrama de bloques de las etapas de un Sistema de Visión Artificial.

Fuente: Técnica y Algoritmos Básicos de Visión Artificial

Un sistema de Visión Artificial debe seguir un conjunto de procesos consecutivos. En primer lugar se adquiere la imagen o escena a través de la cámara. Luego la imagen capturada es procesada; aquí se obtiene una información más precisa al eliminar ruido en la misma. En estas dos primeras etapas se realiza un tratamiento y análisis de la imagen más simple y global.

Posteriormente se realiza la segmentación de la imagen; aquí se separa en regiones u objetos de ser el caso. Luego de tener la escena separada por regiones se analiza los objetos características como el color, la textura, la forma, el diámetro, el área; etc. En estas etapas se le da un tratamiento a la imagen más personalizado de las regiones que la conforman.

Al tener la imagen separada en regiones es posible seleccionar la región u objeto a la que se desea interpretar en un entorno estructurado y en tiempo real.

Una vez cumplido el proceso descrito actuará un sistema de respuesta, que en conjunto con el Sistema de Visión Artificial cumplirán el objetivo final del procedimiento.

1.3. Introducción a la Robótica

La robótica es la ciencia que aplica la informática al diseño y construcción de aparatos que realizan operaciones o trabajos orientados comúnmente a aplicaciones industriales y de navegación (Santos y Duro, 2005).

Con el fin de facilitar al ser humano en diversas áreas de su trabajo y vida diaria, se empezó a incursionar en el área de la robótica, creando desde maquinas sencillas hasta prototipos que adopten características propias del ser humano.

La Robótica Móvil permite a un prototipo o mecanismo adoptar características de movimiento para desplazarse en un ambiente y la vez interactuar con él. Un robot móvil contará con sensores que le permitirán percibir cambios en su entorno y actuadores para su desplazamiento. Además de una programación que controlara y enviara instrucciones a cada parte del prototipo.

Para lograr la autonomía de un robot se están desarrollando diversas técnicas que están relacionadas al tratamiento de imágenes mediante la Visión Artificial.

1.4. Estructura de Robots Móviles

La estructura es la forma física que se le va a dar al robot, sirve de protección para las partes eléctricas y electrónicas. Con la estructura se puede visualizar las funciones que cumplen tanto actuadores como sensores, ya que se convierten en extensiones de los mismos. Hay que tener en cuenta que la estructura debe ser flexible y a la vez resistente; ya que debe adaptar sus movimientos a los cambios del entorno (Vigueras y Fernández, p.25).

Existen diversos materiales para la construcción de la estructura de un robot, como son: plástico, metales, fibras de carbono, etc. Dependiendo del medio en el que se desenvuelva y las funciones que cumplirá el robot se seleccionará el material indicado.

La **Figura 1-15.** muestra la estructura del Robot, tanto la parte frontal como la parte lateral; diseñada en SolidWorks.

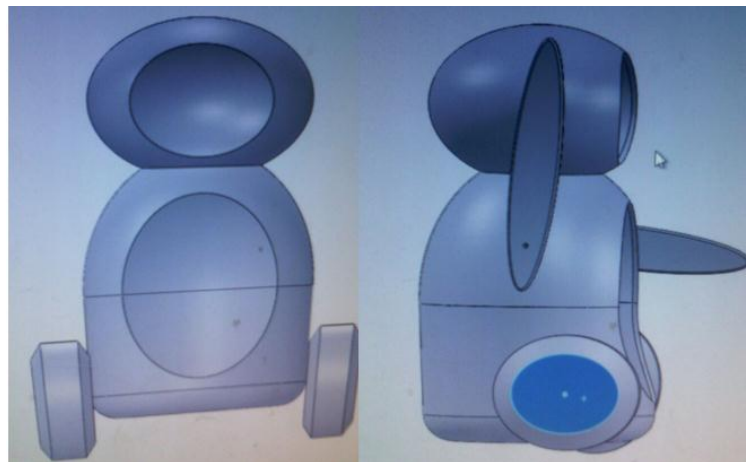


Figura 1-15. Estructura del Robot diseñada en SolidWorks.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

1.5. Dispositivos Sensoriales de los Robots Móviles

Los seres humanos contamos con cinco sentidos; los mismos que nos permiten apreciar sabores, olores, tacto, sonidos, lo lejos o cerca que se encuentra un objeto; para así descifrar las condiciones que nos rodean y estar preparados a cualquier cambio.

Al igual que los seres humanos los robots móviles para poder desenvolverse dentro un ambiente específico necesitan recibir alertas o señales que les permiten diferenciar algún cambio en su entorno; para ello se les incorpora dispositivos sensoriales. Entre los más importantes se tiene: sensores de movimiento y proximidad.

1.5.1. Sensor de Movimiento PIR

El sensor PIR o Pasivo Infrarrojo pertenecen a la familia de los sensores piroeléctricos, puesto que emiten una señal eléctrica al estar en contacto una señal infrarroja. Son les denomina pasivos porque a diferencia de los activos; estos no emiten una señal infrarroja más bien la reciben dicha señal (Villegas, 2011).

La señal que recibe el sensor PIR es provocada por el movimiento del ser humano o animales en su entorno. Además cuenta con un filtro que impide las falsas señales de radiaciones que pueden ser provocadas por rayos del sol, radiofrecuencias.

En la 0se muestra el Sensor PIR HC-SR501 utilizado en el prototipo.



Figura 1-16. Sensor de
Movimiento PIR
Realizada por: Cayambe, Diana; Castañeda, Juan, 2016.

1.5.2. Sensor de Proximidad

Este tipo de sensores detectan los objetos cercanos a él; al ser interrumpida una señal que emite el transmisor del mismo; entonces el receptor obtiene una señal eléctrica que es transformada en valores de distancia medida desde el sensor hacia el objeto intruso.

La **Figura 1-17.** muestra un sensor de proximidad ultrasónico.



Figura 1-17. Sensor de Proximidad Ultrasónico
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

1.6. Locomoción de los Robots Móviles

La locomoción hace referencia a la manera en que el robot se desplazara en una superficie solida; dependiendo de las fuerzas que existen entre dicha superficie y la estructura del robot. Entre los elementos más usados para generar el movimiento del robot tenemos a las ruedas, cintas y patas. Las ruedas permiten el soporte y equilibrio de mayor peso, por lo que son los más usados para implementar robots móviles.

1.6.1. Locomoción Diferencial

La locomoción diferencial hace referencia a la estructura que le permitirá desplazarse al robot realizando trayectorias rectilíneas y curvas. La estructura está basada en dos ruedas dispuestas a cada lado del robot. Para realizar el cambio de dirección se debe modificar la velocidad de una de las ruedas. Este tipo de estructura es muy usada en robots pequeños (Sistemas de Locomoción de Robot Móviles).

Esta estructura es de bajo costo y un diseño sencillo, sin embargo presenta problemas de equilibrio y control en el robot.

En la **Figura 1-18.** Se ilustra la forma de desplazamiento de un sistema de Locomoción Diferencial.

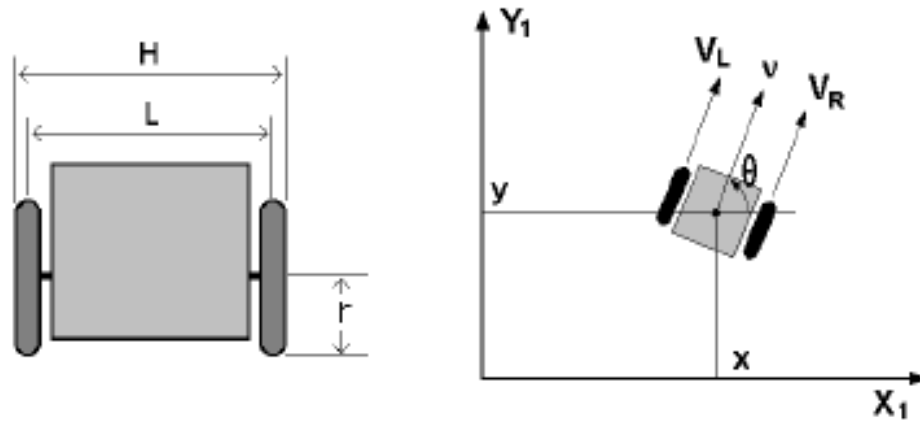


Figura 1-18. Sistema de Locomoción Diferencial
Fuente: Sistema de Locomoción de robots móviles.

1.6.2. Locomoción triciclo o coche

Al implementar este tipo de locomoción el robot podrá moverse en líneas rectas o curvas y girar. Para conseguir dichos movimientos son necesarias dos ruedas fijas y una rueda loca o rueda de castor. La rueda loca es una rueda no controlada que le otorga equilibrio al robot; al formar lo que se conoce como un triciclo en conjunto con las ruedas fijas controladas (Sistemas de Locomoción de Robot Móviles).

En la **Figura 1-19.** se aprecia la forma de desplazamiento de un Sistema de Locomoción Triciclo.

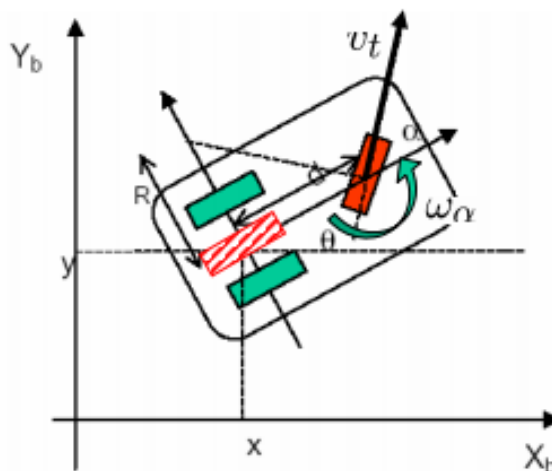


Figura 1-19. Sistema de Locomoción Triciclo
Fuente: Sistema de Locomoción de robots móviles

1.7. Fuente de energía usada en Robots Móviles

La autonomía en los robots móviles es una parte esencial, por esta razón la energía de la que dependerá el robot debe encontrarse dentro de su estructura y no depender de una fuente externa.

Existen diversas formas de energizar a un robot, como por ejemplo las placas solares, bombonas de aire, células de hidrogeno, células nucleares, baterías, entre otras; todos ellas permiten almacenar y proveer energía al prototipo.

1.7.1. Baterías

Las baterías son elementos eléctricos que transforman energía química en energía eléctrica. Los tipos de baterías comúnmente utilizadas y que podemos encontrar en el mercado son: pilas alcalinas, Baterías Níquel Cadmio (Ni-Cd), Baterías Níquel e Hidruro Metálico (Ni-MH), Baterías Ion-Litio, Baterías de polímeros de litio (Li-Po), (Viguera y Fernández, p. 30).

Existen diferentes parámetros a tomar en cuenta en el momento de escoger una batería que más se adapte a la aplicación; como son: tensión nominal, capacidad de la batería, tasa de descarga y peso.

En la **Figura 1-20**. se muestra una batería LIPO de la marca Turnigy, usada para alimentar los actuadores del robot y la placa Raspberry Pi.



Figura 1-20. Batería LIPO
Fuente: www.hobbyking.com

1.8. Actuadores de Robots Móviles

Para que los robots puedan desplazarse en un ambiente es necesario el acople de actuadores los mismos que pueden ser del tipo hidráulico, neumático y eléctricos. Los más usados en robots móviles son los actuadores eléctricos, ya que en este grupo encontramos los motores reductores, servomotores, switch, entre otros. Existen diversos tipos de motores los mismos que se adaptan a las necesidades de la aplicación (Vigueras y Fernández, p. 29).

1.8.1. Motores Reductores

Los motores reductores están creados a base de una cadena de engranajes que permiten controlar su velocidad de una forma eficiente y segura. Con esto se logra tener valores proporcionales entre la velocidad y la potencia transmitida del motor.

En la **Figura 1-21**. Se puede apreciar un Motor reductor 131:1 de la marca Pololu utilizados para accionar las ruedas del robot.



Figura 1-21. Motor reductor Pololu.
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

1.8.2. Servomotores

Este tipo de motores se caracterizan porque se posicionan mediante un ángulo de 0 a 180 grados, lo que permite tener un control más exacto de la articulación del robot.

El ancho de pulsos o más conocido como PWM indica el ángulo de giro del motor. Los valores de dichos pulsos vienen dados de 1ms a 2 ms; por ende 1ms corresponde a 0° y 2ms a 180°. Para mantener el servomotor en una posición es necesario enviar pulsos continuos de anchura de la posición deseada.

En la **Figura 1-22.** se puede apreciar un Servomotor HS-311 de la marca Hitec utilizados para el accionamiento de los brazos del robot.



Figura 1-22. Servomotor HS-311
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

1.8.3. Módulo de relés

El Módulo de relés es un dispositivo formado por cuatro canales que funcionan a 5V, son capaces de manejar cargas de hasta 10A y 250V. Generalmente placas como Arduino o la placa Raspberry Pi B+ no puedan manejar cargas que se alimentan de voltajes altos y que consumen corrientes mayores a 1A, para ello se usa el Módulo de relés que sirve de canal de comunicación entre la placa Raspberry Pi y la carga a controlar (Módulo de 4 relés para Arduino, 2015).

En la **Figura 1-23.** se muestra un Módulo de cuatro relés utilizado para controlar los motores reductores 131:1.



Figura 1-23. Módulo de cuatro relés
 Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En el mercado se encuentran Módulos de relés que vienen desde 1 canal hasta los 16 canales; cada canal posee el mismo diagrama de conexión interna. El relé tiene una bobina de 5V y contactos que manejan cargas de hasta 250V y 10A, además posee una bornera de tres contactos, un contacto normalmente abierto (NO), un contacto normalmente cerrado (NC) y un contacto común.

El módulo de relés usado en el proyecto cuenta con un jumper selector que permite alimentar los relés desde una fuente externa y no desde la placa Raspberry Pi.

En la **Figura 1-24.** se muestra el Circuito esquemático de un canal del Módulo de relés

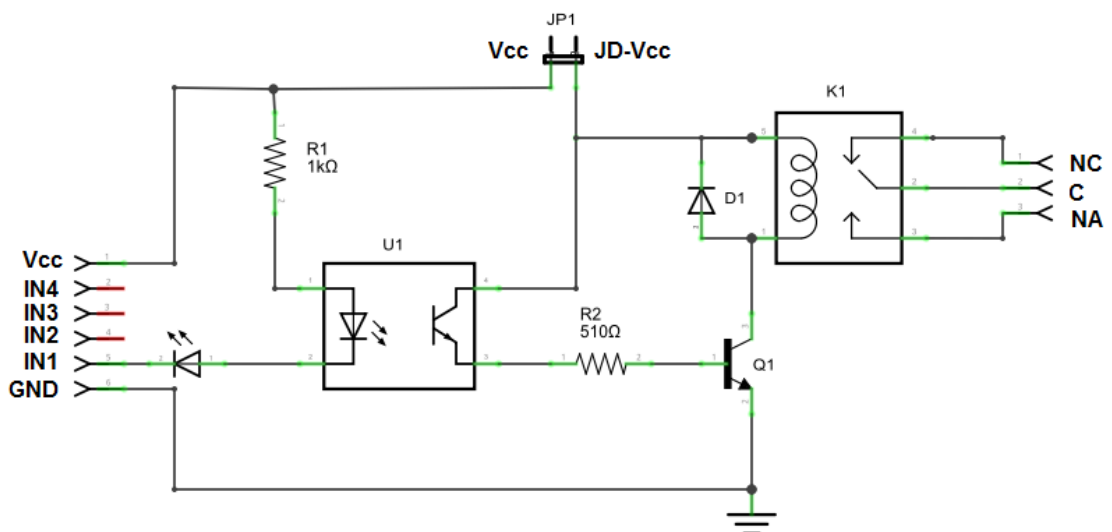


Figura 1-24. Circuito esquemático de un canal del Módulo de relés
 Fuente: www.profetolocka.com.

1.8.4. *Switch Eléctrico*

El switch eléctrico es un dispositivo que permite desviar la corriente que circula en un circuito. Las características a tomar en cuenta al momento de seleccionar un interruptor son las siguientes:

- El número de polos que son circuitos individuales controlados por el switch;
- El número de vías es la cantidad de posiciones que posee el switch.

En la **Figura 1-25**. se muestra un Switch de tres posiciones utilizado para seleccionar el color de la pelota.

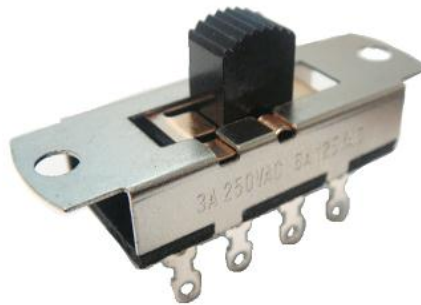


Figura 1-25. Switch de tres posiciones
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

1.9. Algoritmos aplicados al Tracking de Objetos

El tracking de objetos es una de las aplicaciones más comunes de la VA, pues tiene diversas aplicaciones entre ellas tenemos el control de seguridad, la manipulación de objetos, robots que se desenvuelven en un entorno en tiempo real (Ilbay, 2015, p 62-70).

En este trabajo se aplica a un robot móvil, por ende se debe tener en cuenta factores que influirán en el seguimiento del objeto tanto en el entorno físico como la programación enfocada al seguimiento.

Existen diversos métodos o técnicas aplicables al tracking de objetos; los mismos que se basen en enfocar diferentes características del objeto o escena a tratar.

1.9.1. Algoritmo de Umbralización aplicado al Tracking de Objetos

El algoritmo de Umbralización se utiliza cuando existe una diferencia notoria entre el fondo de la escena y el objeto a seguir. Consiste en agrupar los píxeles según la intensidad luminosa; hasta encontrar el umbral necesario que realizará una binarización de la escena y separará el objeto poniéndolo en blanco, mientras que el fondo será de color negro (Iibay, 2015, p 65-67).

Sin embargo en este algoritmo hay que tener cuidado con la iluminación, ya que si varía de forma constante complicará el proceso de segmentación y seguimiento del objeto.

El proceso de ejecución del Algoritmo de Umbralización se ilustra en la **Figura 1-26**.

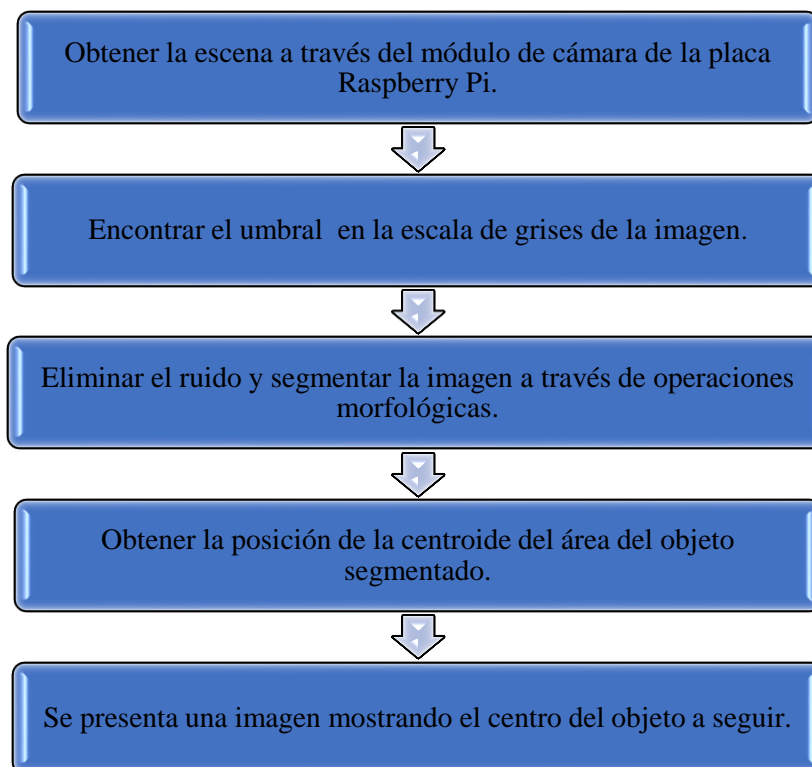


Figura 1-26. Esquema del Algoritmo de Umbralización aplicado al Tracking de Objetos

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En el Sistema de Visión Artificial del presente proyecto se implementó el Algoritmo de Umbralización para el tracking de la pelota.

1.9.2. Algoritmo de segmentación por color aplicado al Tracking de Objetos

Los algoritmos que emplean la segmentación por color, utilizan un color previamente definido y los compara con los colores de cada pixel de la imagen analizada. Cuando existe coincidencia de colores se procederá a segmentar la imagen (Mafla y Ortiz, 2014, p 59).

En la **Figura 1-27.** se presenta los pasos a seguir del algoritmo de segmentación por color.

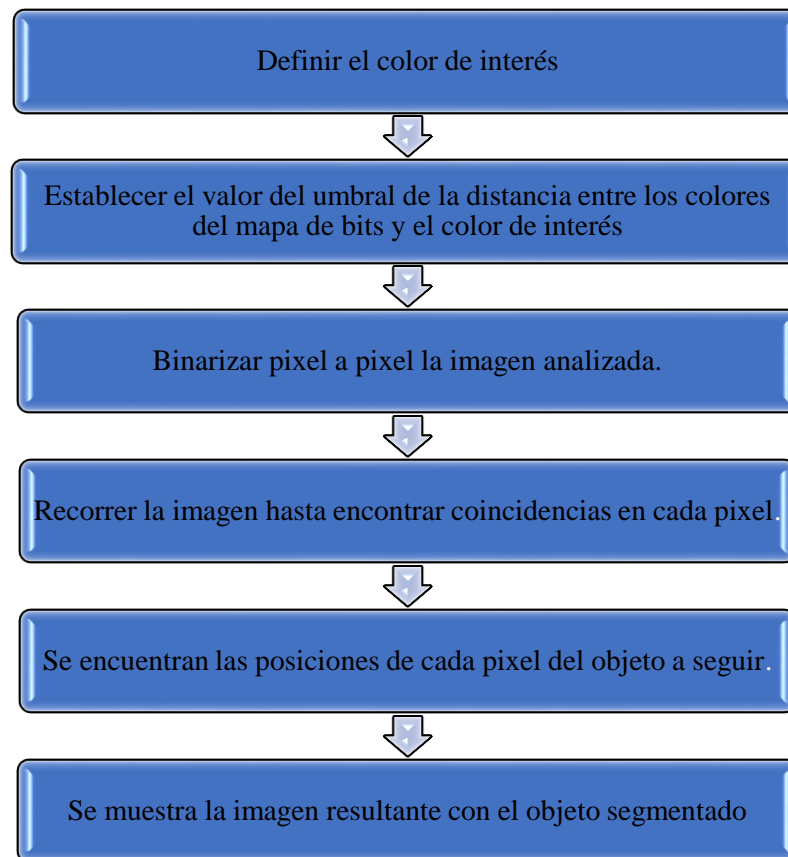


Figura 1-27. Esquema del algoritmo de segmentación por color aplicado al Tracking de Objetos.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

1.9.3. Algoritmo CAMShift aplicado al Tracking de Objetos

Este algoritmo trata a las imágenes en base a cambios en su tamaño y posición. Las imágenes ingresadas serán del tipo RGB y se trasladarán al modelo HSV para posteriormente ser tratadas por el algoritmo CAMShift. El Proceso a seguir para la ejecución del algoritmo es el siguiente:

- Creación de una ventana en la que se seleccionará el objeto a seguir en la imagen.
- Definir el histograma de la componente matiz en la ventana creada.
- Generar una imagen a escala de grises, aquí cada pixel tendrá diferente intensidad y así se podrá reconocer si el pixel pertenece al objeto a seguir
- Se aplica el filtro Canny, el mismo que permitirá detectar bordes.
- Se realiza la operación AND entre la imagen ingresada e imagen procesada.
- A la imagen resultante de la operación se le traslada a escala de grises.

En la **Figura 1-28**. Se puede apreciar el esquema del algoritmo de seguimiento CAMShift.

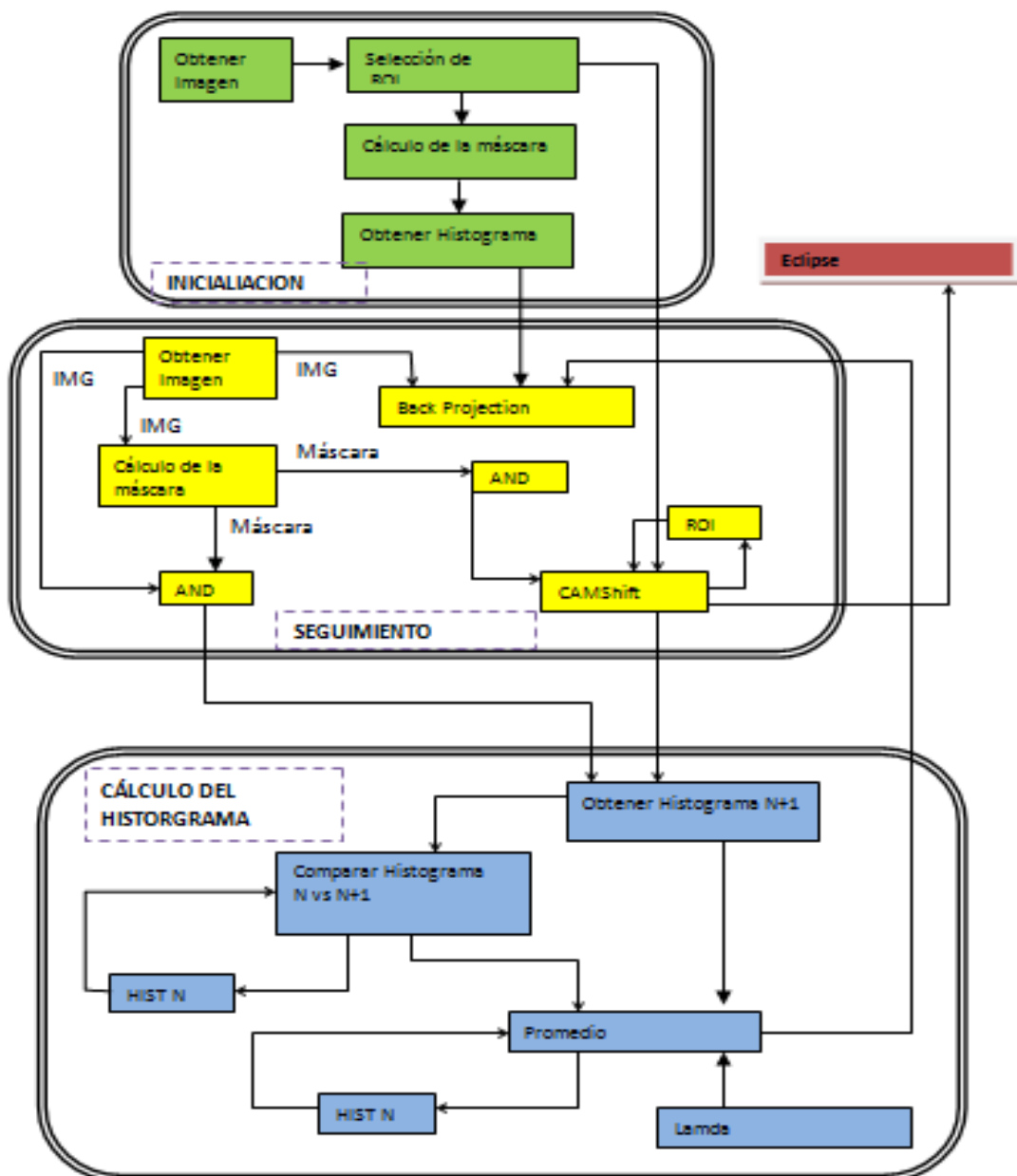


Figura 1-28. Esquema del algoritmo de seguimiento CAMShift

Fuente: GIAR Algoritmo de Seguimiento de objetos basado en visión asistida por computador en tiempo real utilizando CAMShift e histogramas ponderados.

1.10. Impresión 3D

Una impresora 3D es una máquina que permite crear objetos o prototipos. Realiza una fabricación aditiva; es decir, crea objetos en una sucesión de capas que van desde la parte inferior hasta la superior (Impresión 3D, p 1).

Hoy en día la impresión 3D es una gran herramienta en el ámbito industrial ya que se puede crear a partir de piezas pequeñas prototipos o figuras realmente llamativas. Para el proceso se usa un hilo fino de plástico colocado en un cabezal de extrusión que será controlada por un ordenador.

En la **Figura 1-29**. se observa Vista frontal de una impresora3D de la marca SaisCartesian PRUSA I3X.

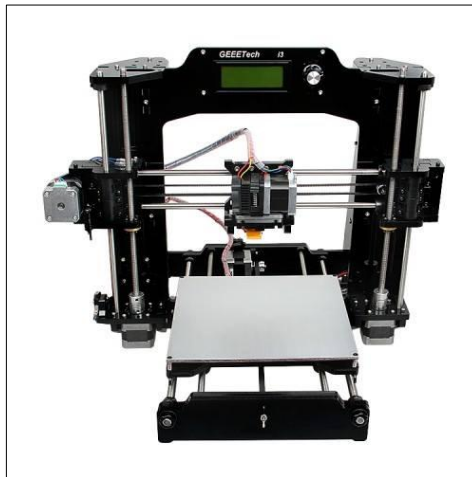


Figura 1-29. Vista frontal de una impresora3D.
Fuente: SAIS 3D - Impresión 3D - 3D Printing

El proceso general de impresión 3D basado en el documento Impresión 3D de Stratasys (p 2-5) consiste en tres etapas: el pre-procesado, la construcción y el post-procesado; a continuación se describen las etapas:

- En el pre-procesado se abre el archivo de diseño CAD; posteriormente se importa el archivo para elegir opciones de color, material y crear segmentos o capas de la pieza a imprimir. Con los segmentos creados el software genera las instrucciones de construcción que seguirá el cabezal de extrusión.

- En la etapa de construcción damos la orden a la impresora pulsando el botón Imprimir, para que así empiece a crear la pieza. El hilo de plástico es introducido al cabezal de extrusión al mismo que se le aplica calor para ablandarlo. Por último el sistema va colocando capas de aproximadamente 0,13 mm, hasta formar la pieza.
- Finalmente en el post-procesado solo queda retirar los soportes que sostienen a la pieza y limpiar los restos de material.

En la **Figura 1-30**, se muestra cómo se va formando una pieza al posicionarse desde la parte inferior de la pieza hasta el término de la misma.



Figura 1-30. Construcción de una pieza en una impresora 3D

Fuente: La impresión 3D ¿Cómo Funciona?, 2011

1.11. La placa Raspberry Pi B+

La placa Raspberry Pi es un microcomputador desarrollado en Reino Unido, se basa en arquitectura ARM que contiene funciones de señal: analógica, digital, mixta, memoria RAM. Dicho microcomputador contiene todos los elementos necesarios para su correcto arranque, integrados en una sola placa (Salcedo, 2015, p 28-36). En el ANEXO A se muestra la hoja de especificaciones de la placa Raspberry Pi B+.

Raspberry Pi cuenta con diversas versiones para su distribución en el mercado, como son: Modelo A, Modelo Brev1, Modelo Brev2 y Modelo B+. En el presente proyecto se usará el Modelo B+ V1.2, el mismo que presenta las siguientes especificaciones:

- Procesador Broadcom BCM2835 SoC
- Posee una arquitectura Core ARM11.
- La CPU cuenta con una velocidad de 700MHz
- La GPU Dual Core VideoCore
- Memoria SDRAM 512 MBytes.
- El sistema operativo inicia desde una tarjeta Micro SD

- Requiere una energía de 600mA hasta 1.8A a 5 VDC.
- La temperatura máxima de funcionamiento es de 85°C.

1.11.1. Conectores disponibles de la placa Raspberry Pi B+

La placa Raspberry Pi cuenta con diversos tipos de conectores como por ejemplo conectores USB, conector de cable Ethernet, conector de audio, entre otros. Posteriormente se dará a conocer las funciones que realizan cada uno de los conectores que conforman la placa Raspberry Pi (Raspberry Pi Robotics Projects, 2014, p 10-33).

En la **Figura 1-31**. se visualiza como están dispuestos en la placa Raspberry Pi B+.

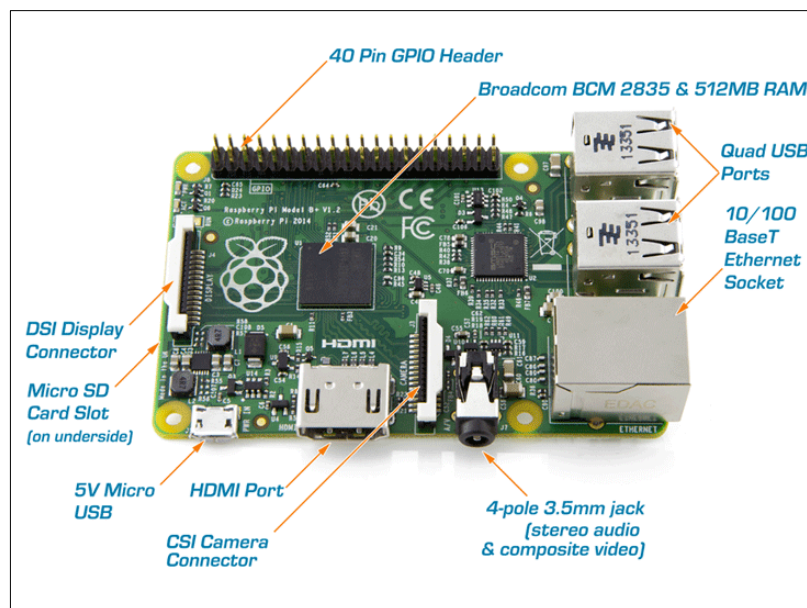


Figura 1-31. Esquema de la placa Raspberry Pi B+

Fuente: <https://www.pcdiga.com>

A continuación se describe la función de cada conector en la placa Raspberry Pi B+.

- **DSI Display:** Permitirá la conexión de pantallas LCD que soporten DSI.
- **Memory Card Slot:** Aquí se insertará la Micro SD, previamente instalada en ella el sistema operativo a usar.
- **Conector Micro USB:** Este conector permite alimentar a la placa Raspberry Pi B+ con una señal de 600mA a 1.8 A a 5VDC.

- **HDMI Port:** Permite la conexión de un televisor analógico y también a monitores del tipo RCA.
- **CSI Camera Connector:** Nos permite la conexión del Módulo de Cámara que nos proporciona la Fundación de Raspberry Pi
- **Puerto de Audio:** Permite la salida de audio en forma stereo.
- **Puerto Ethernet:** Mediante un conector RJ45 permite la conexión física de la placa Raspberry Pi B+ a una red LAN Ethernet
- **Puertos USB:** Cuenta con cuatro puertos USB 2.0, provee de alimentación eléctrica a periféricos y dispositivos electrónicos.
- **Puertos GPIO:** Este puerto cuenta con 40 pines como se muestran en la **Figura 1-32**, que pueden ser configurados de acuerdo a las necesidades del trabajo, dichos pines operan a 3.3 VDC, no cuentan con una protección contra sobretensiones; por lo que se debe evitar conectar cualquier tipo de dispositivo con lógica TTL/CMOS de 5VDC.

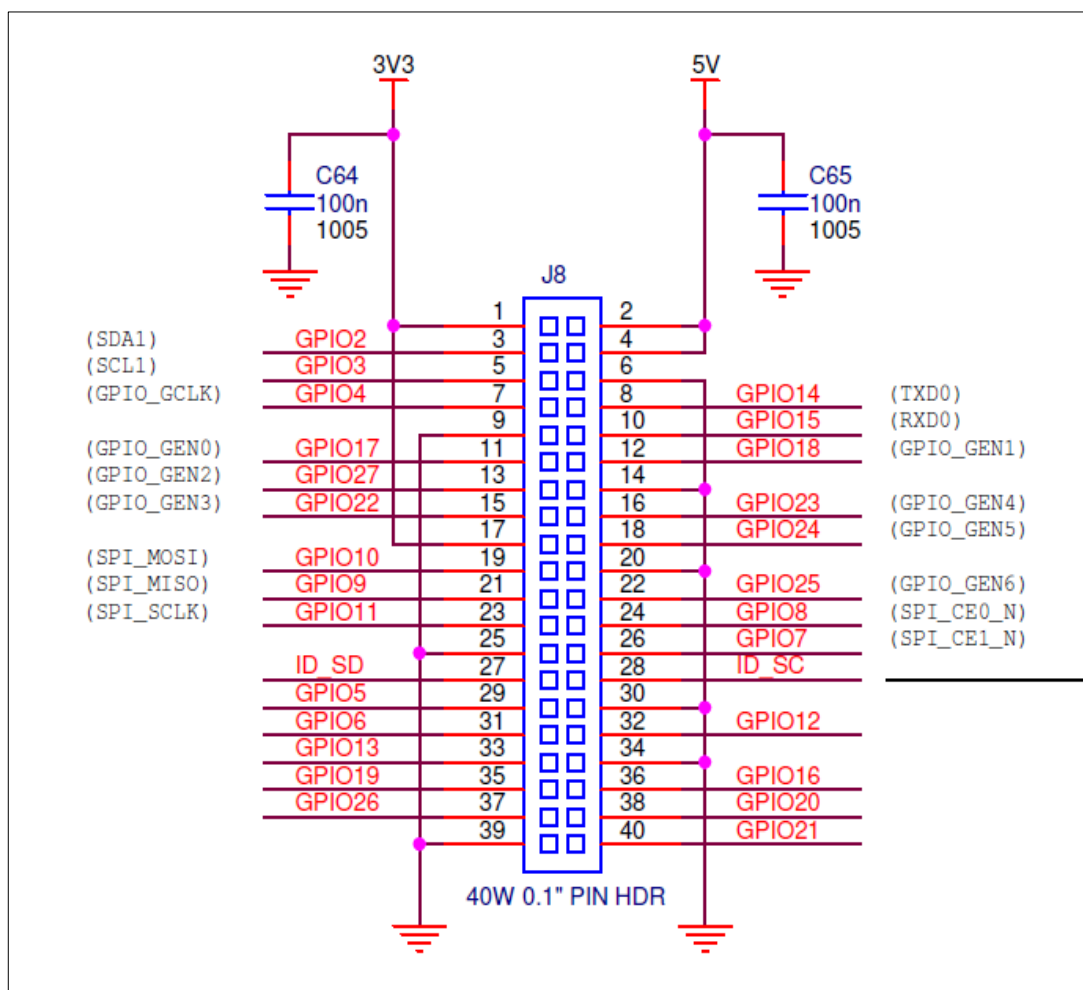


Figura 1-32. Diagrama de Puertos GPIO

Fuente: <https://www.raspberrypi.org>

1.11.2. Accesorios de la placa Raspberry Pi B+

La placa Raspberry Pi B+ cuenta con accesorios que nos otorgan una mejor experiencia al usar como computador personal o en cualquier proyecto educativo (Datasheet Raspberry Pi, 2014). A continuación se enlistan.

- **Módulo de Cámara:** Presenta un sensor de imagen OmniVision. Captura imágenes de 5 megapíxeles y videos de 1080p/(30fps); posee una velocidad de fotogramas de 720p/(60fps).
- **Suministrador de energía:** Es un cargador que suministra 5 voltios DC
- **Tarjeta de Memoria:** En ella se instala el sistema operativo que sirve de plataforma en la placa Raspberry Pi.
- **Cable HDMI-VGA:** Mediante este cable se transformara la señal de la placa Raspberry Pi y la transformara a VGA para así obtener un entorno grafico del sistema operativo instalado.
- **Adaptador WIFI USB:** Mediante este adaptador se podrá acceder a internet de vía WIFI sin necesidad de un cable de red.

1.12. Software utilizado para la programación y diseño del Robot

Para la realización del prototipo se hace necesario diversos software, desde el sistema operativo Raspbian Wheezy que sirve de plataforma al software de programación, software de compilación. El sistema operativo Windows 8.1 servirá de plataforma para el software de diseño del robot.

1.12.1. Sistema Operativo Raspbian Wheezy

La plataforma de Raspberry Pi ofrece diversos tipos de sistemas operativos como Raspbian, Noobs. Estos sistemas operativos son libres y están basados en Debian acoplándose a las necesidades de Raspberry Pi. En cuanto a Raspbian tenemos las versiones de Jessie, Jessie Lite y Wheezy, las mismas que son actualizadas a sus versiones superiores (Raspberry Pi for Dummies 2013, p 19-31).

Raspbian Wheezy usa LXDE como escritorio y Midori como navegador web. Cuenta con diversos Programas orientados a la programación como Python. Actualmente Raspbian cuenta con una tienda de aplicaciones llamada Pi Store.

1.12.2. Software SolidWorks

SolidWorks es un software de diseño en 3D asistido por computadora, permite modelar piezas de una forma versátil y precisa; ya que se puede describir minuciosamente las piezas a modelar. Además se puede obtener una animación en tiempo real del diseño creado (Introducción a SolidWorks, p 17-22).

SolidWorks contiene tres módulos que son: Pieza, Ensamblaje y Dibujo; los mismos que se encuentran asociados entre sí, lo que quiere decir que si se realiza algún cambio en el módulo pieza automáticamente se actualizarán los demás módulos.

- El módulo de pieza permite crear modelos tridimensionales a partir de diagramas en 2D; para posteriormente volverlos sólidos.
- El módulo de ensamblaje permite crear conjuntos a partir de los diseños creados en el módulo de piezas. A través de este módulo podemos verificar que no existan choques o interferencias entre las piezas que conforman el ensamblaje
- El módulo de dibujo permite obtener las vistas estándar, las acotaciones, los materiales, secciones y cortes, entre otras.

1.12.3. Lenguaje C++

Es un lenguaje de programación orientada a objetos y estructurada. Está formado por un conjunto de funciones; las mismas que ejecutan un proceso con entradas y salidas previamente definidas.

Las funciones son una gran herramienta dentro de la programación ya que permiten encerrar cierta parte del código y convertirla en un subproceso; para así llamarlo cada vez sea necesario (Diseño de Algoritmos y su Programación, 2013, p 8-10).

1.12.4. OpenCV

OpenCV es una biblioteca open SOURCE orientado al procesamiento de imágenes y visión por computadora en tiempo real. Entre sus aplicaciones más importantes se tiene:

- Interacción Hombre-máquina.
- Segmentación y reconocimiento de objetos.
- Seguimiento de Movimiento.
- Robots Móviles.

Además OpenCV cuenta con una gama de módulos los mismos que están destinados a aplicaciones en tiempo real. Con el conjunto de módulos integrados se puede resolver la mayoría de procesos encaminados a visión por computador (Brahmbhatt, p 1-23).

A continuación se describen los módulos asociados a OpenCV:

- **Core:** contiene estructuras de datos y funciones aritméticas.
- **Improg:** contiene funciones dedicadas a la manejo de imágenes.
- **Highgui:** abarca las funciones de lectura y escritura tanto de fotografías como de video.
- **Video:** permite realizar un análisis de movimiento y seguimiento de objetos en tiempo real.
- **Calib3d:** mediante este módulo se puede calibrar la cámara y reconstruir en 3D una imagen.
- **Features2d:** permite extraer características de la escena
- **Objdetect:** abarca algoritmos de detección de objetos como rostros.
- **GPU:** contiene algoritmos seleccionados para una rápida ejecución de hardware.

CAPITULO II

2. MARCO METODOLÓGICO

El proceso de diseño e implementación del proyecto denominado Inteligencia Artificial aplicada a robot asistencial para la interacción con niños en edad temprana fue desarrollado aplicando tres métodos.

Los métodos Inductivo, Heurístico y Experimental se distribuyen en tres etapas del trabajo. El primer método es usado en la etapa de recolección de información, el segundo método se aplica para la programación y diseño del robot y el tercer método se utilizó en la etapa de prueba y corrección de fallas y errores en el robot.

Se aplicó el método inductivo para la etapa de recolección de información, ya que este método obtiene ideas generales a partir de premisas particulares. Pues la fusión de ideas particulares tomadas de ciencias como la Electrónica, la Robótica, la Inteligencia Artificial y la Visión Artificial permite llegar a una idea general de como diseñar un Sistema de VA aplicado a un robot móvil.

Se recurrió al método heurístico para realizar la programación en OpenCV, el diseño de la estructura en SolidWorks, disposición de los sensores y actuadores; el método se basa en el uso de reglas y conocimientos empíricos para explorar procedimientos que permitan hallar la solución. Con la ayuda de esto método y de la aplicación de las técnicas de programación y diseño aprendidas a lo largo de la carrera se logró diseñar la estructura del robot; así como también la programación que controla los elementos del robot.

Para realizar las pruebas funcionamiento del prototipo se empleó el método experimental; haciendo uso de la lógica empírica y la experimentación. Se realizó las pruebas de los sensores, motores, el tracking de la pelota; para así determinar las fallas y errores que presentaba el robot, posteriormente se realizó la debida calibración de los elementos para minimizar los errores y obtener un desempeño optimo del Sistema de VA, desempeño de los actuadores y sensores del prototipo.

2.1. Instalación y Configuración de Software requerido

Para evitar errores en el momento de la ejecución de los programas que realiza el tracking de la pelota y controla los actuadores y sensores del robot, es necesario una instalación y configuración adecuada, empezando por el sistema operativo Raspbian Wheezy hasta el software de programación y compilación.

2.1.1. Instalación y Configuración de Raspbian Wheezy

De los sistemas operativos disponibles para Raspberry Pi se ha elegido Raspbian Wheezy, pues es una plataforma que cuenta con las aplicaciones necesarias para el proyecto.

Se ha considerado necesario mostrar el proceso de instalación y configuración del sistema operativo, ya que es un software poco conocido en el entorno.

Una vez insertada la tarjeta SD en la placa Raspberry Pi B+, previamente cargada la imagen iso del sistema operativo Raspbian Wheezy; automáticamente se mostrará en la pantalla el proceso de carga del sistema y posteriormente el menú de Raspi-Config ilustrado en la **Figura 2-1**.

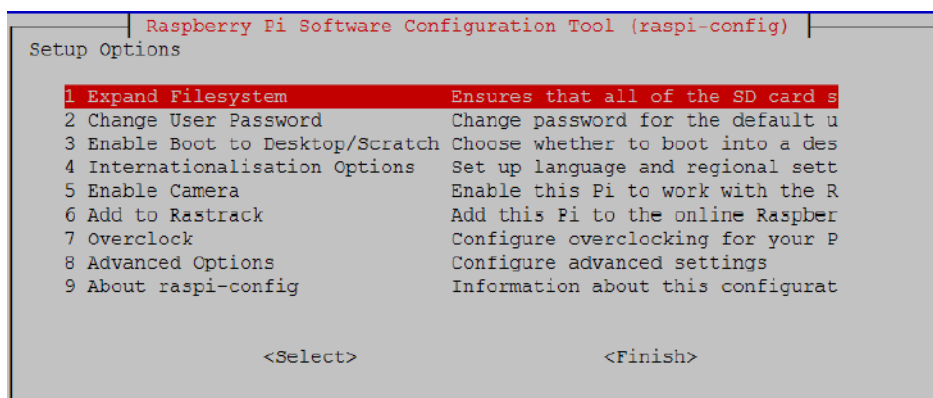


Figura 2-1. Menú Raspi-Config

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

A continuación se describe los cambios realizados sobre el menú Raspi-Config.

- Se seleccionará la opción 1 para expandir la memoria y así contar con más espacio de almacenamiento.

- La opción 2 no es necesario modificar. En el proyecto se mantendrá el nombre del usuario y la contraseña por defecto que son “pi” y “raspberry” respectivamente.
- La opción 3 permitirá que la Raspberry Pi inicie en un entorno grafico al seleccionar la opción Graphical Desktop.
- En la opción 4 se configurará el idioma, la zona horaria, la distribución del teclado. En este caso se elegirá el idioma es_ES.UTF-8 UTF-8, la zona Horaria de Bogotá, y el teclado PC genérico 105 teclas (intl) en español latinoamericano.
- La opción 5 permitirá habilitar el módulo de la cámara.
- En la opción 6 se registra a la placa de Raspberry Pi en un sitio de web, que permite localizar a las placas en todo el mundo. No se consideró necesario registrar la Raspberry Pi.
- En la opción 7 presenta opciones para aumentar la velocidad del procesador. En el proyecto se mantuvo la velocidad por defecto.
- En la opción 8 no se realizó ninguna modificación.

Una vez realizados los cambios en el menú de Raspi-Config se mostrará el escritorio de Raspbian Wheezy como se ilustra en **Figura 2-2**.

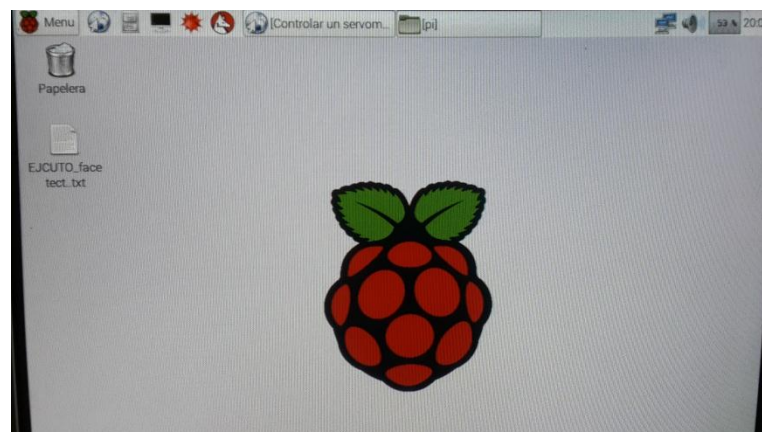


Figura 2-2. Escritorio de Raspbian Wheezy
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.1.2. *Instalación y Configuración de OpenCV*

Una vez configurado el sistema operativo Raspbian Wheezy, se procede a la instalación y configuración de OpenCV. Primero se debe abrir el LXTerminal en el cual se introducirán los comandos necesarios para instalar los paquetes que usa OpenCV. La pantalla del LXTerminal se muestra en la **Figura 2-3**.

Primero se debe realizar una actualización y limpieza del sistema; con la ayuda de los siguientes comandos.

- *sudo apt-get update*
- *sudo apt-get upgrade.*

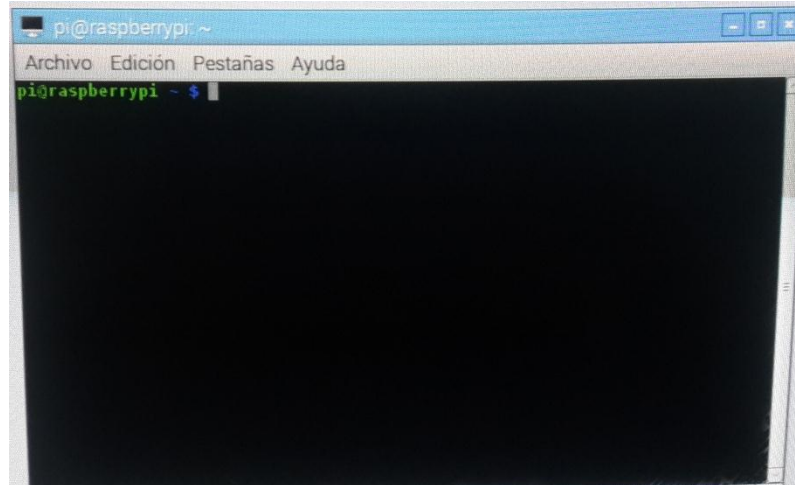


Figura 2-3. LXTerminal
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Una vez ejecutados los comandos mencionados se procede a la instalación de los paquetes y dependencias necesarios para trabajar en el software OpenCV. Para ello ejecutamos el comando *sudo apt-get install <nombre del paquete>*. En la **Tabla 2-1.** se enlistan los paquetes y dependencias usadas por OpenCV.

Tabla 2-1. Lista de paquetes y dependencias utilizadas por OpenCV.

PAQUETES Y DEPENDENCIAS UTILIZADAS POR OPENCV		
libopencv-dev	libhighgui2.3	python-matplotlib
libpng3	libtiffxx0c2	libgstreamer0.10-0-dbg
libdc1394-22-dev	libgtk2.0-dev	libjasper-dev
build-essential cmake pkg-config	python-opencv	python-pandas
libpnglite-dev	libtiff-tools	libgstreamer0.10-0
libdc1394-22	libpngwriter0-dev	libopenexr-dev
libavformat-dev	opencv-doc	python-nose
zlib1g-dbg	libjpeg8	libgstreamer0.10-dev
libdc1394-utils	libpngwriter0c2	libgdal-dev
x264	libcv-dev	v4l-utils
zlib1g	libjpeg8-dev	libxine1-ffmpeg

libv4l-0	libswscale-dev	python-tk
v4l-utils	libcvaux-dev	libgtkglext1-dev
zlib1g-dev	libjpeg8-dbg	libxine-dev
libv4l-dev	libjpeg-dev	python3-dev
Ffmpeg	libhighgui-dev	libpng12-0
Pngtools	libavcodec-dev	libxine1-bin
libpython2.6	libwebp-dev	python3-tk
libcv2.3	python-numpy	libpng12-dev
libtiff4-dev	libavcodec53	libunicap2
python-dev	libpng-dev	python3-numpy
libcvaux2.3	python-scipy	libpng++-dev
libtiff4	libavformat53	libunicap2-dev
python2.6-dev	libtiff5-dev	libeigen3-dev
qtcreator	synaptic	gucvview
python-rpi.gpio		

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

A continuación descargamos e instalamos el Software OpenCV con los siguientes comandos:

- *wget http://sourceforge.net/projects/opencvlibrary/files/opencv-unix/2.4.10/opencv-4.10.zip*
- *unzip openCV-2.4.10.zip*
- *cd opencv-2.4.10*
- *mkdir build*
- *cd build*
- *cmake -D CMAKE_BUILD_TYPE=RELEASE -D INSTALL_C_EXAMPLES=ON -D INSTALL_PYTHON_EXAMPLES=ON -D BUILD_EXAMPLES=ON -D WITH_QT=ON -D CMAKE_INSTALL_PREFIX=/usr/local -D WITH_OPENGL=ON -D WITH_V4L=ON -D BUILD_NEW_PYTHON_SUPPORT=ON -D WITH_TBB=ON ..*
- *make*
- *sudo make install*

Es necesario realizar varias modificaciones en dos archivos para evitar errores en la compilación de códigos en OpenCV. A continuación se describe el proceso de modificación de dichos archivos.

El primer archivo a modificar se denomina `opencv.conf` y para abrirlo ejecutamos el comando *sudo nano /etc/ld.so.conf.d/opencv.conf*, luego añadir la línea */usr/local/lib* en el archivo y

guardar los cambios. A continuación abrir el archivo `bash.bashrc` con el comando *sudo nano /etc/bash.bashrc*, añadir las siguientes líneas al archivo y guardar los cambios.

```
PKG_CONFIG_PATH=$PKG_CONFIG_PATH:/usr/local/lib/pkgconfig  
export PKG_CONFIG_PATH
```

Realizadas las configuraciones mencionadas finalmente las herramientas y bibliotecas que ofrece el software OpenCV están listas para usarse dentro del código de programación.

2.1.3. Configuración para la conexión remota al Raspberry Pi B+ a través de VNC

Para el presente proyecto es necesario trabajar en la placa Raspberry Pi B+ desde otro equipo, para así controlar el robot sin la necesidad de que la placa esté conectada a un monitor, mouse y teclado a través de un cable.

Para lograr el manejo remoto de la placa Raspberry Pi B+ se utilizó el sistema Virtual Network Computing o más conocido como VNC. El sistema VNC es una aplicación cliente –servidor que envía los eventos de teclado y mouse desde el controlador del computador remoto a la placa Raspberry Pi; el computador remoto recibe actualizaciones que permiten visualizar el escritorio de la Placa Raspberry Pi B+.

Para realizar la conexión entre servidor y cliente VNC es necesario un adaptador WI-FI conectado en el servidor VNC, en este caso en la placa raspberry Pi B+.

A continuación se describen los pasos para realizar la configuración del servidor VNC en la placa Raspberry Pi B+.

- **Instalar el paquete TightVNC:** *sudo apt-get install tightvncserver*
- **Ejecutar el comando:** *tightvncserver*
- **Digitar la contraseña para acceder al escritorio remoto:** *raspberry*
- **Iniciar un servidor VNC desde el terminal:** *vncserver :1 -geometry 1920x1080 -depth 24*
- **Detener el displaymanager usando el comando:** *service lightdm stop*

A continuación se describen los pasos para realizar la configuración del cliente VNC en el computador remoto con SO Ubuntu.

- En el menú Aplicaciones/Internet seleccionar Remote Desktop Viewer.
- En la ventana de Escritorio Remoto seleccionar en Conexión.
- En la ventana Conexión establecer el Protocolo VNC
- En Host digitar la dirección IP de la Raspberry Pi B+, seguido del número 1 de la siguiente manera: 192.168.0.6:1
- Hacer clic en conectar e ingresar la contraseña establecida en el servidor VNC y finalmente nos encontraremos en escritorio de Raspberry Pi B+.

2.2. Selección de Motores

Para lograr el desplazamiento del robot, el movimiento de los brazos y el empuje de la pelota, se utilizaron motores reductores y servomotores. Se les dio una función específica a cada motor aprovechando las características como frecuencia, torque, peso, corriente que consumen, voltaje de operación; con las que cuentan cada uno.

En la **Tabla 2-2.** se describen los parámetros de funcionamiento de: micro motor reductor 50:1, motor reductor 131:1 y del servomotor HS-311; en base a la hoja de especificaciones mostradas en el 0, ANEXO D y ANEXO E respectivamente.

Tabla 2-2. Parámetros de funcionamiento de los motores usados en el robot.

Parámetros Motores	Frecuencia (rpm)	Torque (kg-cm)	Peso (g)	Corriente que consume (mA)	Voltaje de operación (V DC)
Motor Reductor	40 a 6 V DC 80 a 12 V DC	9 a 6V DC 18 a 12V DC	205	2500 a 6V DC 5000 a 12V DC	6 - 12
Micro Motor Reductor	630 a 6 V DC	0.5 a 6V DC	9.5	1600 a 6V DC	6
Servomotor HS-311	----	3.02 a 4.8V DC 3.53 a 6V DC	43	180 a 6V DC	4.8 -6

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

2.2.1. Motores para el desplazamiento del robot

Para elegir los motores que le permitirán al robot desplazarse, se tuvo en cuenta variables como: el peso del robot, el piso en el que se moverá, la velocidad necesaria. Por esta razón se

seleccionó el motor reductor con una relación de transmisión de 131:1 de la marca Pololu ilustrado en la **Figura 1-21.** del Capítulo 1, pues cuenta con las características necesarias para cumplir con un desplazamiento óptimo del robot.

Las ruedas accionados por los motores 131:1 deberán tener una velocidad que le permita al robot desacelerar sin cambios bruscos; para así evitar que el robot se detenga de golpe y sufra daños.

2.2.2. *Motores para el movimiento de los brazos del pingüino*

El movimiento que realizarán los brazos del robot será constante y repetitivo, siempre y cuando el robot se encuentre en movimiento. Los motores adecuados para este tipo de movimiento son los denominados servomotores ilustrados en la **Figura 1-22.** del Capítulo I.

La **Tabla 2-3.** muestra las posiciones de los ángulos de los servomotores para realizar el movimiento de las aletas del robot.

Tabla 2-3. Posición de los ángulos de los servomotores.

Valores Movimiento del Brazo del robot	ANCHO DE PULSO (ms)	POSICIÓN DEL ÁNGULO
Posición inicial del brazo	1.5 ms	90°
Movimiento hacia adelante	1.6 ms	108°
Movimiento hacia atrás	1.4 ms	72°

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

2.2.3. *Motor para el mecanismo de pateo de la pelota*

Para que el robot pueda mover la pelota se necesita una fuerza de empuje aplicada a la pieza que realizará el pateo de la pelota. La fuerza de empuje debe ser lo suficientemente fuerte como para mover la pelota a una distancia de 3 m, pero a la vez debe ser sutil para que la pelota no se aleje

demasiado o impacte en el niño, causándole daño. Por este motivo se escogió el micro motor reductor con una relación de transmisión de 50:1.

En la **Figura 2-4.** se muestra un micro motor reductor 50:1 de la marca Pololu.



Figura 2-4. Micro motor reductor 50:1
Fuente: www.pololu.com

2.3. Selección de Sensores

Al momento de seleccionar los sensores se tuvo en cuenta el entorno en el que se desenvolverá y la aplicación a la que está enfocado. El robot contará con dos sensores dispuestos en la estructura externa del robot.

2.3.1. *Sensor de Movimiento*

Este sensor será de gran importancia en el robot, pues su función es la de mantener una distancia considerable con el niño, para así evitar que el robot le cause daño.

El Sensor PIR HC- SR502 ilustrado en la **Figura 1-16.** del capítulo 1, tiene las características necesarias para cumplir con los requerimientos en el prototipo. El sensor recibirá señales de radiación infrarroja provocadas por el movimiento del niño, dichas señales permitirán mantener una distancia considerable entre el niño y el robot.

En base a la hoja de especificaciones del sensor de movimiento PIR HC-SR501 mostrada en el ANEXO G. se describen los principales parámetros de operación.

- **Voltaje de alimentación:** 3 a 5 V DC
- **Consumo de Corriente:** menor a 100 uA
- **Rango de medida:** 3m a 7m
- **Ángulo de medición:** 100°

Además el sensor PIR cuenta con dos resistencias variables que sirven para la calibración. La primera resistencia se usa para ajustar la sensibilidad del sensor en un rango de 3 a 7 metros; la segunda resistencia es utilizada para calibrar el tiempo de espera en un rango de 5 a 300 segundos.

El sensor PIR posee un puente que generalmente es usado para realizar pruebas. Si el puente está colocado en la posición L del sensor PIR la salida se mantiene baja tres segundos después de que el movimiento fue detectado. Si el puente está colocado en la posición H el sensor PIR detecta el movimiento pero se desactiva automáticamente si el movimiento no es detectado.

En la **Figura 2-5.** se ilustra la parte posterior del Sensor PIR, donde se encuentran las resistencias de calibración y el puente de pruebas.

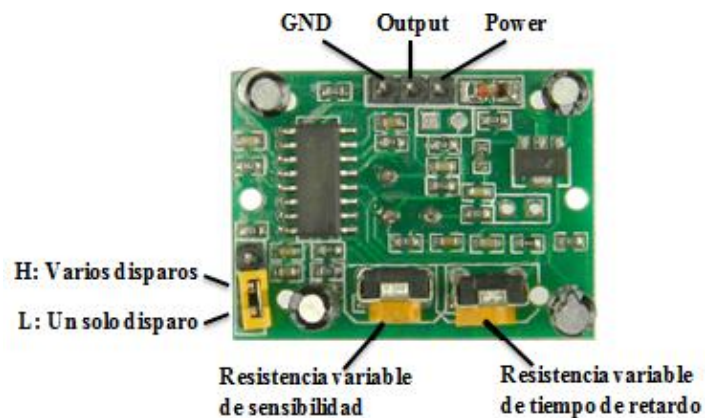


Figura 2-5. Parte posterior del Sensor PIR
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.3.2. *Sensor de Proximidad*

El sensor de Proximidad seleccionado es el sensor Ultrasónico HC- SR04 ilustrado en la **Figura 1-17.** del Capítulo I, y le permitirá al prototipo mantener una distancia considerable frente a obstáculos para así evitar daños en la estructura del robot.

En base a la hoja de especificaciones del sensor de distancia ultrasónico HC-SR04 mostrada en el ANEXO F. se describen los principales parámetros de operación.

- **Voltaje de alimentación:** 5 V DC
- **Consumo de corriente:** 15 mA
- **Frecuencia:** 40 Hz
- **Rango de medida:** 2cm a 4m
- **Angulo de medición:** 15°

2.4. Selección de la Fuente de Alimentación

El robot deberá desplazarse de forma autónoma en un entorno definido por lo que es necesario el uso de una fuente de alimentación que pueda acoplarse a la estructura interna del robot, sin la necesidad de conexiones externas que limiten su libre movimiento.

Para suministrar energía al robot se empleó una Batería formada por celdas de 3.7V DC cada una de la marca Turnigy, ya que permite alimentar a la placa Raspberry Pi B+ y a los actuadores por separado. Hay que tener en cuenta que cada puerto GPIO ofrece una corriente de 16 mA, para así saber que actuadores pueden alimentarse directamente desde la placa Raspberry Pi B+.

A la placa Raspberry Pi B+ se encuentran conectadas las señales que controlarán los sensores, lo motores reductores, servomotores, el módulo de la cámara y los altavoces; cabe recalcar que mientras más elementos se conecten a la placa será necesario un mayor nivel de corriente.

Tanto los sensores como el switch tienen un consumo de corriente que viene dado en valores menores a los 16 mA, por lo que es factible alimentarlos desde la placa Raspberry Pi B+.

Los servomotores tienen un consumo de energía mayor a los 16 mA, por lo que es necesario alimentarnos por separado de la placa Raspberry Pi B+.

Los motores reductores acoplados a las ruedas y al mecanismo de pateo de la pelota, necesitan una batería que les proporcione 5 A y 6 a 12 V DC; por esta razón no se pueden proveer de energía que proporcionan los puertos GPIO, únicamente se conectará a la placa el cable a través del cual el programa enviará señales de actuación a los motores.

En base a la hoja de especificaciones de la Batería se describen los principales parámetros de operación.

- **Capacidad:** 2200 mAh, es decir la batería suministra 2200 mA en una hora.
- **Constante de Descarga:** 20 C, está definida en función de la capacidad de la batería.
- **Configuración:** Posee tres celdas cada una de 3.7 V, en total se tendría 11.1 V
- **Peso:** 188 g.

Otro aspecto a tener en cuenta en las baterías es la carga; pues aquí es necesario realizar un balance de las celdas; es decir, saber que celda se encuentra más descargada; así evitaremos daños en la batería e incluso que haya una explosión.

En la **Figura 2-6.** se muestra el Cargador de Baterías LIPO.



Figura 2-6. Cargador de Batería LIPO
Fuente: www.mercadolibre.com.ec

2.5. Selección de la Cámara

La placa Raspberry Pi B+ soporta la conexión de diversos tipos de cámara USB y a la vez cuenta con un módulo de cámara propio de la placa. Para seleccionar la cámara que más beneficios aporte al sistema de VA se necesita conocer la forma de trabajo tanto de una cámara USB como del módulo de cámara en situaciones de tiempo real; así como también las características físicas de cada una.

En la **Tabla 2-4.** se muestra las ventajas y desventajas de usar el módulo de cámara de la placa Raspberry Pi B+ vs una cámara USB en un Sistema de VA.

Tabla 2-4. Ventajas y Desventajas del módulo de cámara y la cámara USB

MÓDULO DE CÁMARA	CÁMARA USB
VENTAJAS:	VENTAJAS:
<ul style="list-style-type: none"> • Se conecta directamente a la GPU • No afecta el rendimiento de la CPU • Mejor resolución • Mas imágenes por segundo • Son pequeñas 	<ul style="list-style-type: none"> • Posee un cable de conexión largo. • Puede enfocar con facilidad.
DESVENTAJAS:	DESVENTAJAS:
<ul style="list-style-type: none"> • La cinta de conexión es pequeña. • Dificultad al enfocar 	<ul style="list-style-type: none"> • Menos imágenes por segundos. • Afecta al rendimiento de la CPU. • Consume más recursos.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

En base a la información que se observa en **Tabla 4-2.** se ha escogido el módulo de la cámara propio de la placa de Raspberry Pi; ya que brinda mejores características en cuanto a procesamiento de imágenes, tamaño; entre otras, para el Sistema de VA del robot.

A continuación se describen las características del módulo de la cámara basado en la Hoja de especificaciones mostrada en el ANEXO B.

- Posee un sensor de 5 MP.
- Soporta videos de 1080p, 720p y 640x480p60.
- Cuenta con una resolución de 2592 x 1944 pixeles.

En la **Figura 2-7.** se ilustra el Módulo de la cámara de la placa de Raspberry Pi.

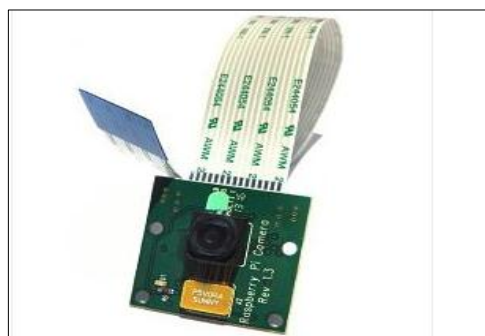


Figura 2-7. Módulo de la cámara de la placa de Raspberry Pi

Fuente: www.raspberrypi.org

2.6. Diseño del Robot

El diseño del robot engloba aspectos como la estructura que será el cuerpo del robot diseñado en el software SolidWorks para imprimir las piezas en la Impresora 3D, también será necesario el diseño de una Tabla de verdad que permitirá escoger el color de la pelota con la que se iniciará el juego.

2.6.1. Diseño y Modelado de la estructura del Robot en SolidWorks

Para el diseño y modelado de la estructura del robot se usó la plataforma de SolidWorks; debido a la versatilidad y facilidad que brindan sus módulos en la creación y ensamblaje de piezas. En el ANEXO J se aprecia la impresión de una de las piezas del robot.

- **Para el desplazamiento:** se diseñaron ruedas con una ligera forma ovalada para imitar el caminado de un pingüino; las mismas que están dispuestas en los lados izquierdo y derecho del robot en la parte inferior. Los motores reductores 131:1 están acoplados a las ruedas.

En la **Figura 2-8.** se muestra el diseño de las ruedas en SolidWorks con las que el robot se desplazará.



Figura 2-8. Rueda del Robot
diseñada en SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-9.** se muestra el diseño del buje de acoplamiento en SolidWorks, el mismo que permitirá el acople entre el motor y la rueda

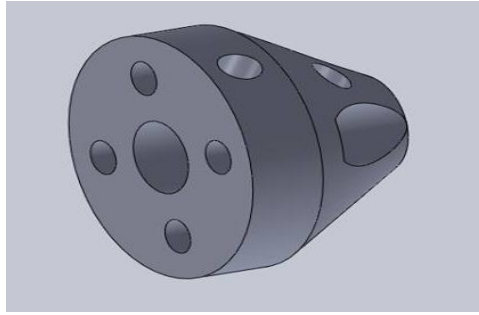


Figura 2-9. Buje de acoplamiento
diseñado en SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-10**, se muestra el acoplamiento del motor con las ruedas en SolidWorks con las que el robot se desplazará.

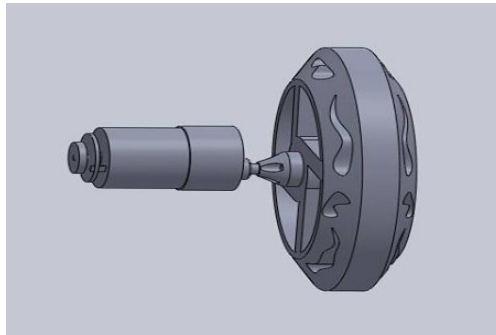


Figura 2-10. Motor acoplado a la rueda
diseñado en SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- **Para el movimiento de los brazos:** se diseñaron dos extremidades en forma de aletas de pingüino, a las que se les acoplará los servomotores HS-311 para darles movimiento. Los brazos están colocados en la parte superior del tronco del robot tanto a los saldos izquierdo y derecho del mismo.

En la **Figura 2-11**, se ilustra el diseño de los brazos del robot diseñado en SolidWorks.

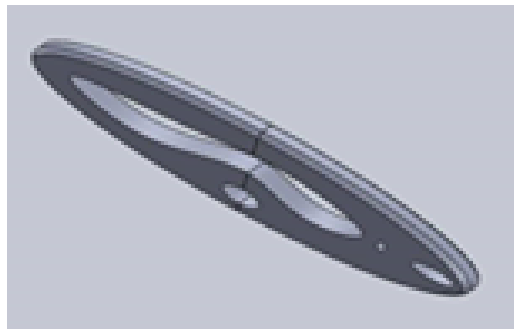


Figura 2-11. Brazo del robot diseñado en
SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- **Para el mecanismo de pateo de la pelota:** se realizó el diseño de una palanca que será accionada por el micro motor reductor 50:1. El mecanismo está ubicado en la parte media del estómago del robot.

En la **Figura 2-12.** se ilustra el mecanismo que pateará la pelota.

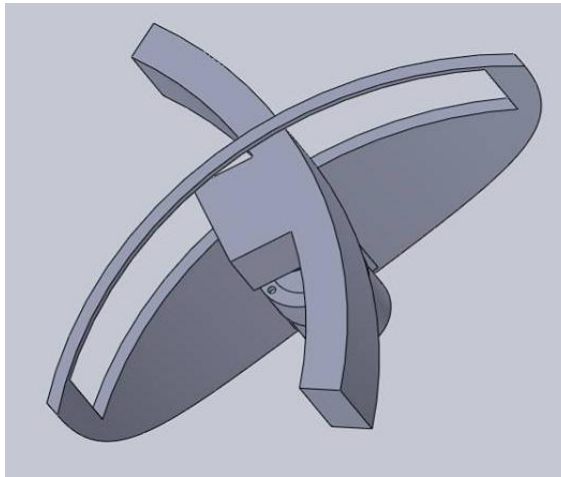


Figura 2-12. Mecanismo para patear la pelota diseñado en SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-13.** se ilustra el acople del micro motor reductor al mecanismo que pateará la pelota.

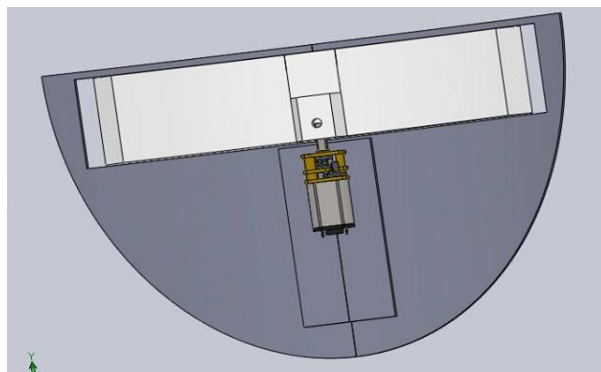


Figura 2-13. Acople del Motor al mecanismo para patear diseñado en SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- El diseño del cuerpo del robot se realizó en tres partes principales. En primer lugar se diseñó la cabeza, luego la parte superior del tronco y finalmente la parte inferior del tronco del robot.

En la **Figura 2-14.** se ilustra el diseño de la cabeza del robot en SolidWorks.

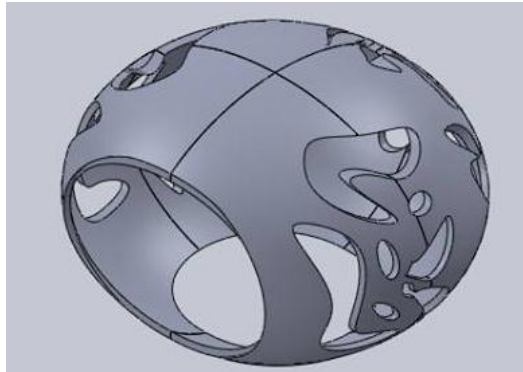


Figura 2-14. Cabeza del Robot diseñada en SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-15.** se ilustra el diseño de la parte superior del tronco del robot en SolidWorks.

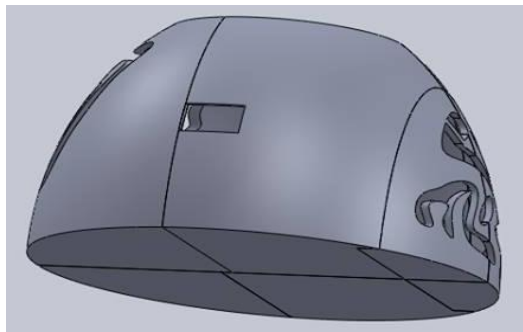


Figura 2-15. Parte superior del tronco del robot diseñado en SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-16.** se ilustra el diseño de la parte inferior del tronco del robot en SolidWorks.

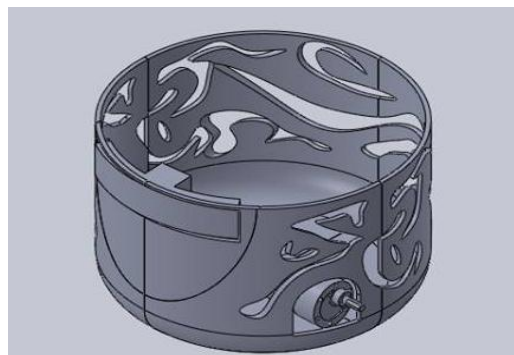


Figura 2-16. Parte inferior del tronco del robot diseñado en SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-17.** se puede apreciar que en la parte inferior del tronco del robot se encuentra adaptado el mecanismo para patear la pelota, así como también se observa el motor que se acoplará a la rueda izquierda del robot en SolidWorks.

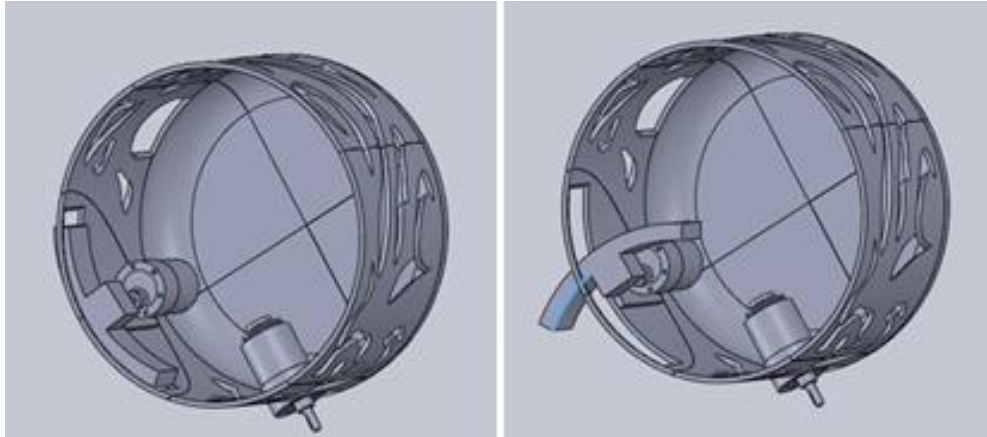


Figura 2-17. Vista Superior de la parte baja del tronco del robot diseñada en SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-18.** se muestra la vista lateral del diseño final del robot en SolidWorks.

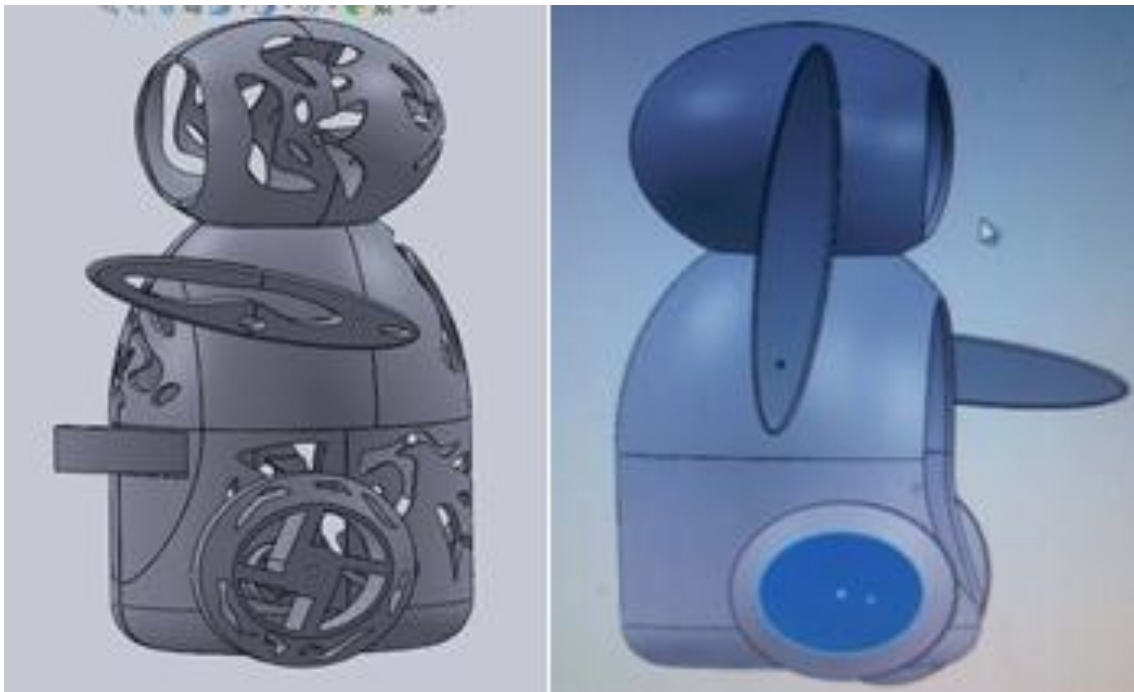


Figura 2-18. Vista Lateral del robot diseñada en SolidWorks

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.6.2. *Diseño de la tabla de verdad para seleccionar el color a seguir.*

El usuario del robot tendrá la posibilidad de elegir el color de la pelota con la que se iniciará el juego. Para ello se creó una tabla de verdad que facilitará la programación y permitirá escoger entre tres colores.

En la **Tabla 2-5.** se muestra combinación de los tres bits, cada bit corresponde al estado de un pin del switch previo su accionamiento.

- En el caso de que el pin correspondiente al color rojo tenga un valor de 1 los dos pines restantes del switch tendrán un valor de 0.
- En el caso de que el pin correspondiente al color verde tenga un valor de 1 los dos pines restantes del switch tendrán un valor de 0.
- En el caso de que el pin correspondiente al color azul tenga un valor de 1 los dos pines restantes del switch tendrán un valor de 0.

Tabla 2-5. Tabla de verdad para seleccionar el color de la pelota

TABLA DE VERDAD			
Color Rojo	0	0	1
Color Verde	0	1	0
Color Azul	1	0	0

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

2.7. Configuración de puertos GPIO

A los puertos GPIO serán conectados los diversos elementos que interactuarán con el entorno, como son: los actuadores y sensores. Previo a la programación que permitirá controlar dichos elementos es necesario importar la biblioteca RPi.GPIO para poder utilizar los puertos GPIO en el código de programación.

En la **Tabla 2-6.** se muestra la distribución de los puertos GPIO que controlarán los elementos del robot.

Tabla 2-6. Distribución de Puertos GPIO

NÚMERO DE PIN	PUERTOS GPIO	SEÑAL INGRESADA AL PUERTO GPIO	ELEMENTO A CONTROLAR
37	GPIO 26	IN1	Motor Reductor 131:1 de la rueda derecha
35	GPIO 19	IN2	
33	GPIO 13	IN3	Motor Reductor 131:1 de la rueda izquierda
31	GPIO 06	IN4	
38	GPIO 20	IN1	Micro Motor Reductor 50:1 para el mecanismo de pateo de la pelota.
40	GPIO 21	IN2	
32	GPIO 12	Pulso	Servomotor del brazo derecho.
36	GPIO 16	Pulso	Servomotor del brazo izquierdo.
12	GPIO 18	Trigger	Sensor Ultrasónico.
16	GPIO 23	Echo	
18	GPIO 24	Señal	Sensor PIR.
11	GPIO 17	Señal switch	Color Rojo
13	GPIO 27	Señal switch	Color Verde
15	GPIO 22	Señal switch	Color Azul
2, 4		5V DC	
6,14,20,30, 34, 9, 25, 39		GND	
1,3		3.3V DC	

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

2.8. Diagramas de Conexión

Se desglosará la conexión de cada uno de los elementos que conforman el robot. Hay que tener en cuenta que los motores reductores, servomotores y los sensores poseen conexiones independientes uno de otro, en lo que se refiere a alimentación. Por esta razón se describe la conexión de cada elemento de forma autónoma.

Para mayor comodidad al momento de describir los procesos del Robot se les asignarán identificadores a los actuadores y sensores utilizados.

En la **Tabla 2-7.** se describe los identificadores respectivos para cada elemento utilizado en el robot.

Tabla 2-7. Identificadores asignados a actuadores, sensores y módulos de relés.

ELEMENTO	UBICACIÓN	IDENTIFICADOR
Sensor Ultrasónico HC-SR04	Ubicado en la parte frontal de robot para detectar obstáculos	S1
Sensor PIR HC-SR501	Ubicado en la parte frontal de robot para detectar movimiento	S2
Micro motor reductor 50:1	Ubicado en el mecanismo de pateo	MP
Motor reductor 131:1	Ubicado en la rueda derecha del robot	MRD
Motor reductor 131:1	Ubicado en la rueda izquierda del robot	MRI
Servomotor HS-311	Ubicado en el brazo derecho del robot	MBD
Servomotor HS-311	Ubicado en el brazo izquierdo del robot	MBI
Módulo de dos relés	Ubicado en la parte interna del robot	MR1
Módulo de cuatro relés	Ubicado en la parte interna del robot	MR2

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-19.** se ilustra el Diagrama de Bloques de conexiones de cada uno de los elementos y reguladores de voltaje utilizados para alimentar el Sistema de Visión Artificial del robot.

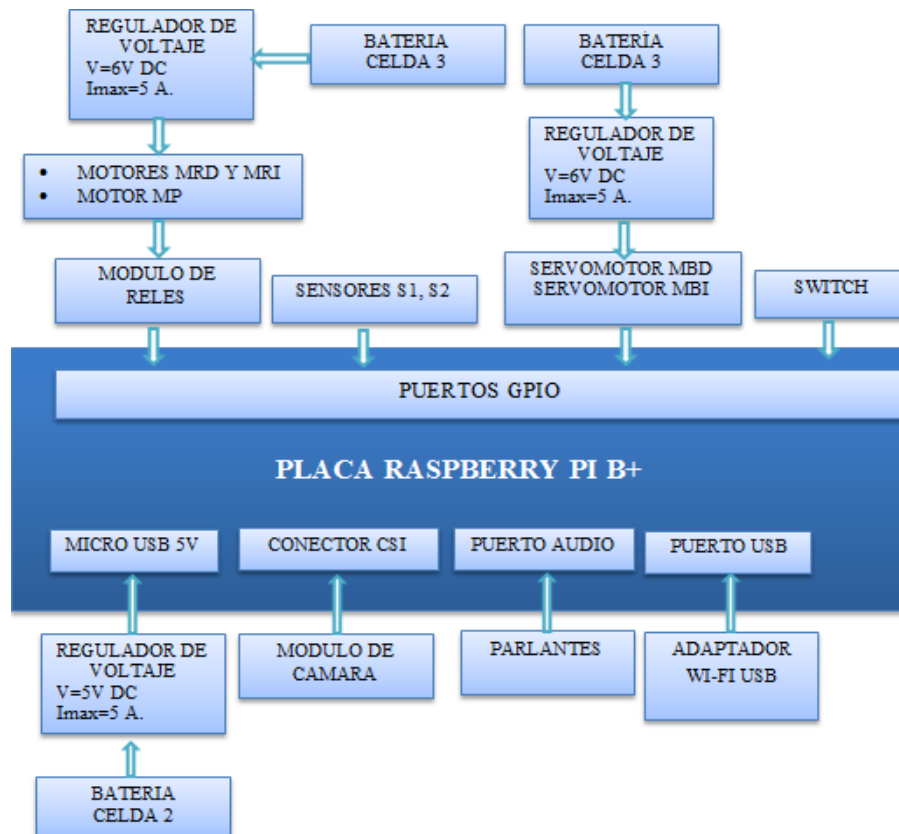


Figura 2-19. Diagrama de Bloques de conexiones del robot
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.8.1. Diagrama de conexión de Reguladores de voltaje

Es necesario regular el nivel de voltaje proveniente de la batería, de acuerdo a las necesidades de la placa Raspberry Pi B+ y los actuadores del robot; para así evitar daños en los elementos y obtener un desempeño óptimos de los mismos.

- Se necesita un regulador de voltaje de 5V DC para alimentar la placa Raspberry Pi B+. La celda dos de la batería proporciona 8V DC a dicha se le conecta el circuito regulador de voltaje que alimenta a la placa Raspberry Pi B+.

En la **Figura 2-20.** se ilustra el diagrama de conexión para regular un voltaje de entrada de 8V DC.

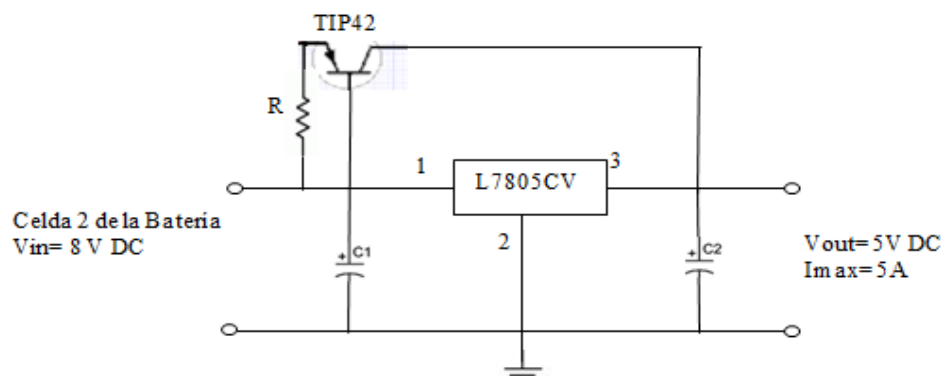


Figura 2-20. Diagrama de conexión para regular voltaje de 8V DC a 5V DC
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016..

En el ANEXO H se muestra la imagen de la baquelita con el circuito regulador de voltaje de 5V DC, lista para ser instalada en la estructura del robot.

- Para los motores reductores MRD, MRI y MP se necesita un nivel de voltaje de 6V DC, para así mantener una velocidad constante en los motores MRD y MRI que le permita al robot desplazarse sin riesgo de una desaceleración brusca provocada por el exceso de velocidad y lograr un movimiento fuerte y rápido del mecanismo que le permitirá empujar la pelota al robot con la acción del motor MP.

En la **Figura 2-21.** se ilustra el diagrama de conexión para regular un voltaje de entrada de 12V DC proveniente de la tercera celda de la batería, a 6V DC con una corriente máxima de 5 A que alimenta a los motores MRI, MRD y MP.

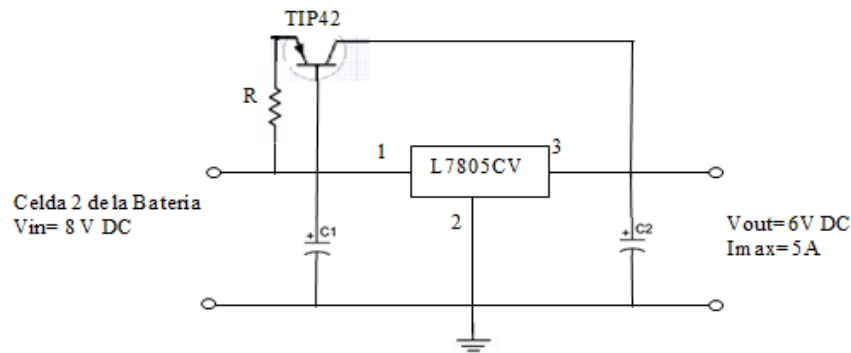


Figura 2-21. Diagrama para regular voltaje de 12V DC a 6V DC
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En el ANEXO I se muestra la imagen de la baquelita con el circuito regulador de voltaje de 6V DC, lista para ser instalada en la estructura del robot.

2.8.2. Diagrama de conexión de Actuadores a la placa Raspberry Pi

Para la conexión de los motores MRD, MRI y MP a la placa Raspberry Pi B+ es necesario el uso de Modulo de Relés; para el proyecto se ha utilizado un módulo de cuatro relés y un módulo de dos relés.

A continuación se describen las conexiones realizadas entre los actuadores, el módulo de relés y la placa Raspberry Pi.

- Al módulo MR1 va conectado el motor MP.
- Al módulo MR2 van conectados los motores MRD Y MRI.
- Las salidas IN1y IN2 del módulo MR1 van conectadas a dos puertos GPIO de la placa Raspberry Pi B+.
- Las salidas IN1, IN2, IN3 y IN del módulo MR2 van conectadas a cuatro puertos GPIO de la placa Raspberry Pi B+.

Para alimentar los Módulos de relés se utilizaron dos fuente: la primera de 5V DC proveniente la placa raspberry Pi que alimentará las señales IN1, IN2, IN3, IN4 del módulo de relés MR1 y las señales IN1, IN2 del módulo de relés MR2; la segunda fuente proveniente del regulador de voltaje de 6V tomado de la tercera celda de la batería.

Además es necesario remover el jumper que une los puntos JD-VCC y VCC.; esta conexión es segura ya que mantiene aislada completamente la carga de la placa Raspberry Pi B+ y el consumo de energía de los relés es tomado de la batería.

En la **Figura 2-22.** se ilustra el Diagrama de conexión de los motores 131:1 y el micro motor 50:1 a través de los módulos de relés MR1 y MR2 la placa Raspberry Pi B+.

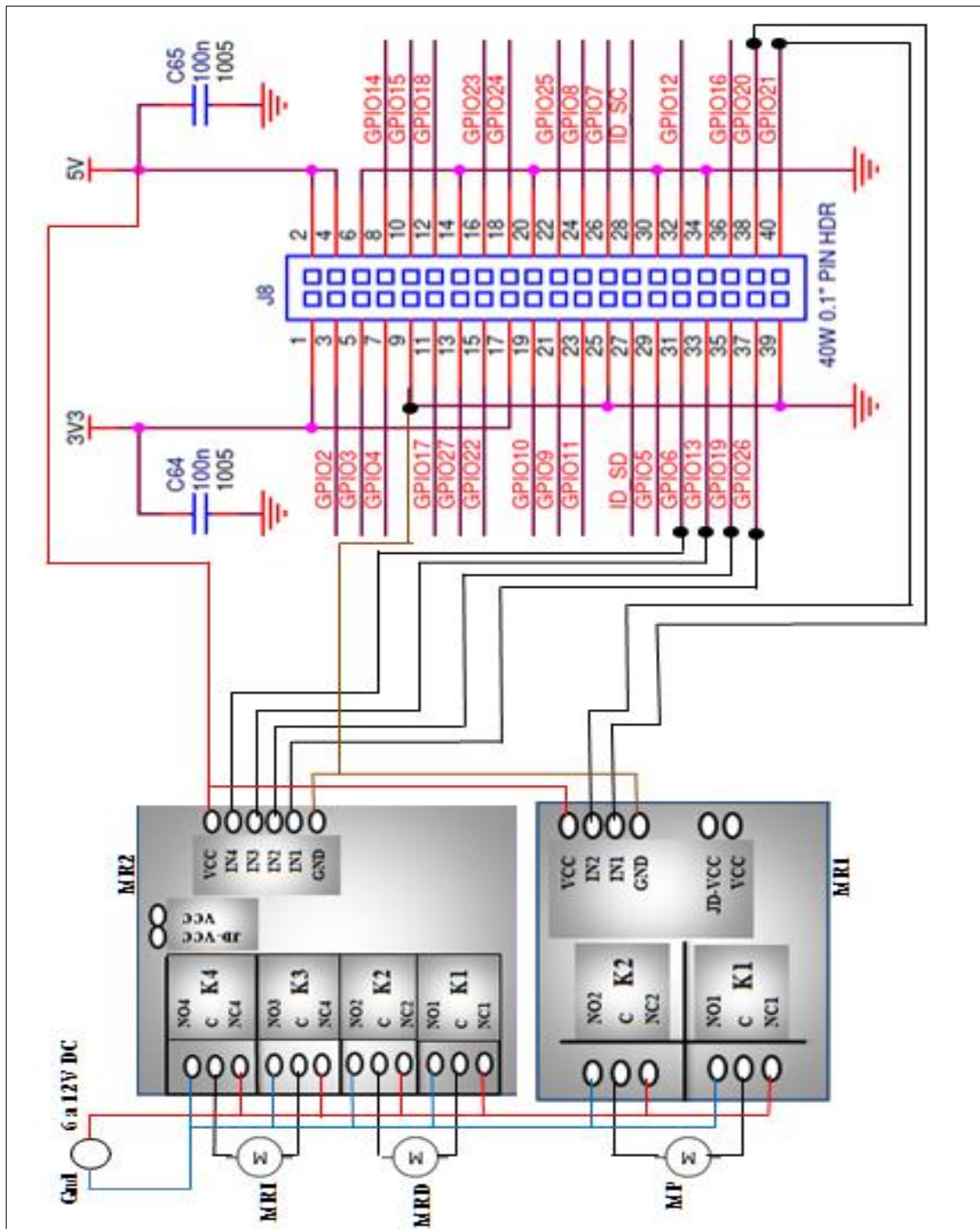


Figura 2-22. Diagrama de conexión de los motores 131:1 y el micro motor 50:1 a la placa Raspberry Pi B+.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Para la conexión la alimentar a los servomotores se usó una fuente externa, proveniente del regulador de voltaje 5VDC; a los puertos GPIO únicamente se conectaron las señales de Pulso de cada servomotor.

En la **Figura 2-23**. se ilustra el diagrama de conexión de los Servomotores HS-311a la placa Raspberry Pi.

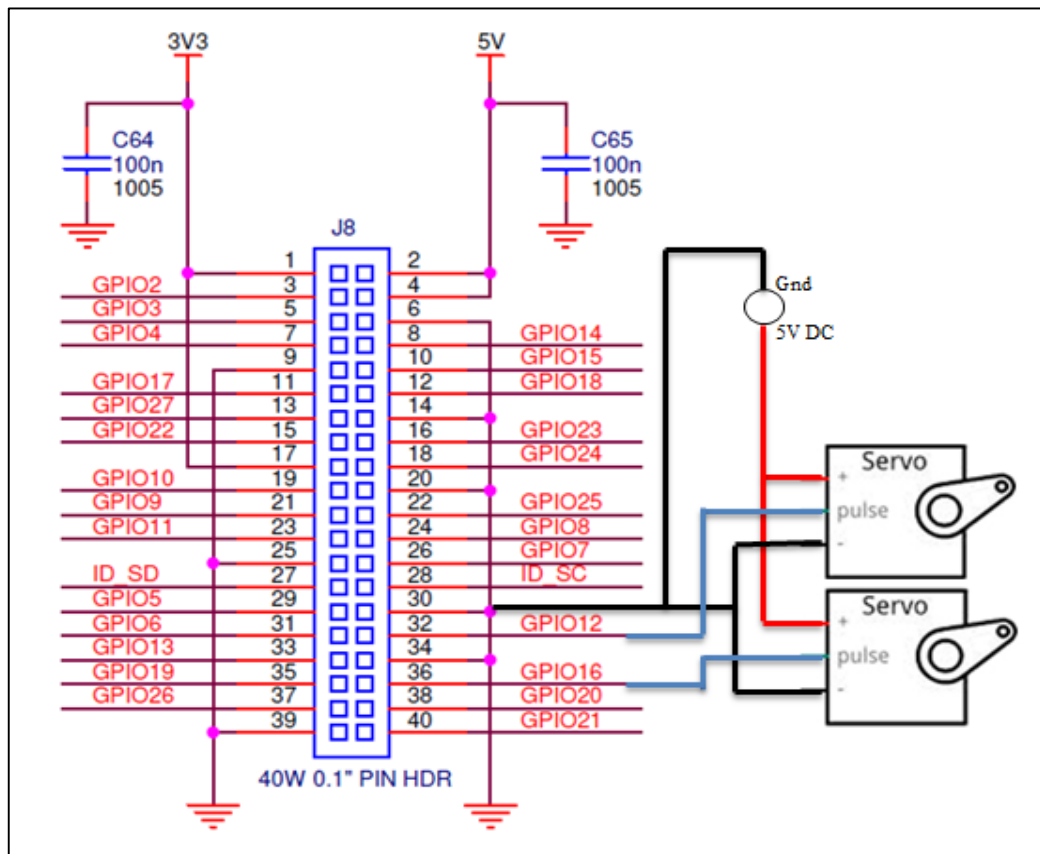


Figura 2-23. Diagrama de Conexión de los Servomotores a la placa Raspberry Pi B+.
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.8.3. Diagrama de Conexión de Sensores a la placa Raspberry Pi

El sensor PIR usará tres pines para su operación; GND, la salida digital y 3 a 5 V DC, al detectar un movimiento en la habitación la señal digital emitirá un pulso de 3.3 V DC.

Para la conexión del sensor Ultrasónico es necesario opera a 5 V DC, la señal de disparo o trigger se activa con 3.3 V DC proveniente del puerto GPIO; la señal de echo se activará y enviará 5V DC al puerto GPIO que realizará la lectura de dicha señal, por lo que es necesario el diseño de un divisor de voltaje que convierta los 5V DC a 3.3V DC.

En la **Figura 2-24.** se ilustra el diagrama de conexión del Sensor PIR y el Sensor Ultrasónico a la placa Raspberry Pi.

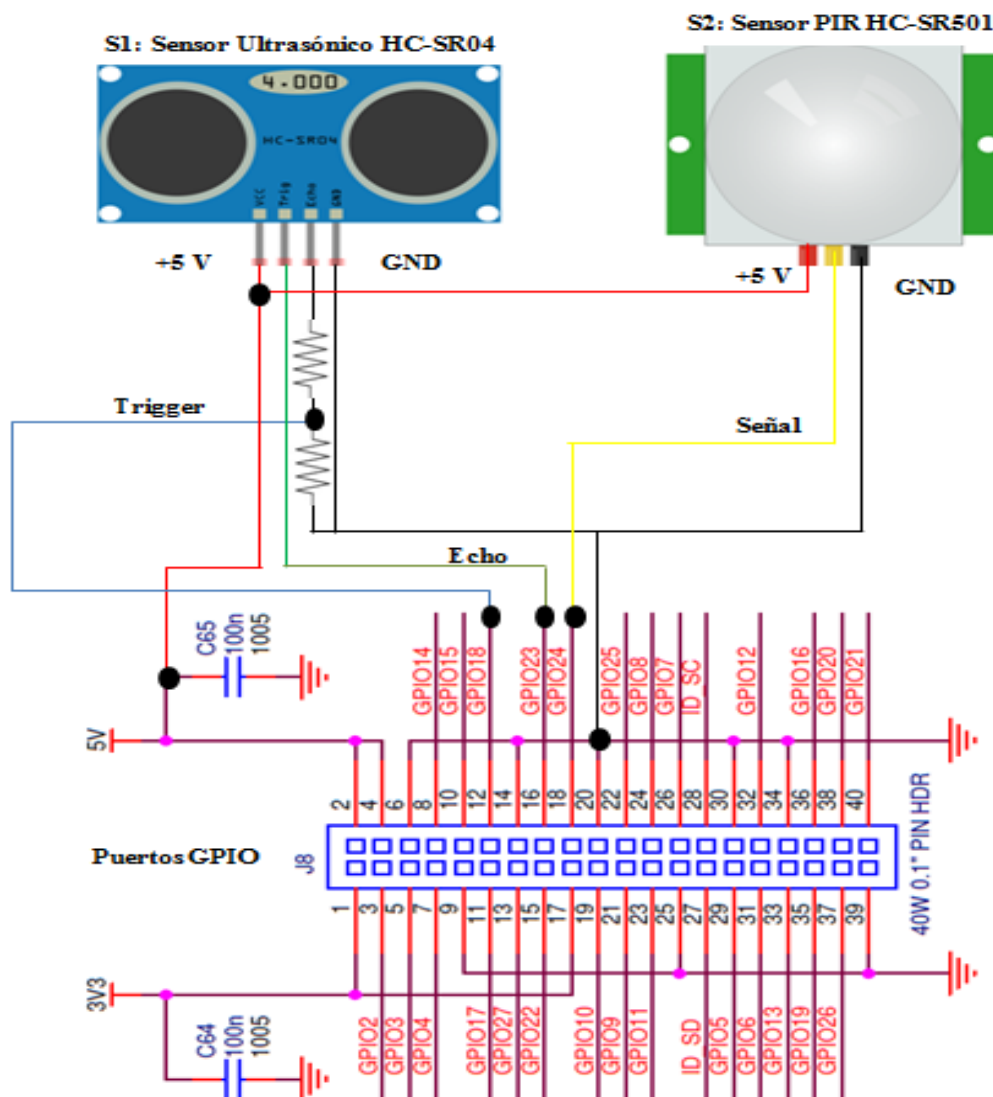


Figura 2-24. Diagrama de Conexión de Sensores a la placa Raspberry Pi B+.
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-25.** se muestra el circuito divisor voltaje utilizado por el sensor ultrasónico.

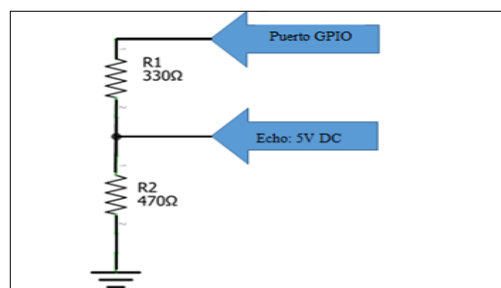


Figura 2-25. Circuito divisor de voltaje
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.8.4. Diagrama de Conexión del Switch a la placa Raspberry Pi

Para que el robot empiece el tracking de la pelota será necesario escoger el color con anterioridad; para lo cual se usó un switch de tres estados. Cada estado pertenece un color de la pelota, que puede ser: rojo, verde o amarillo.

En la **Figura 2-26.** se ilustra el Diagrama de conexión del Switch a la placa Raspberry Pi.

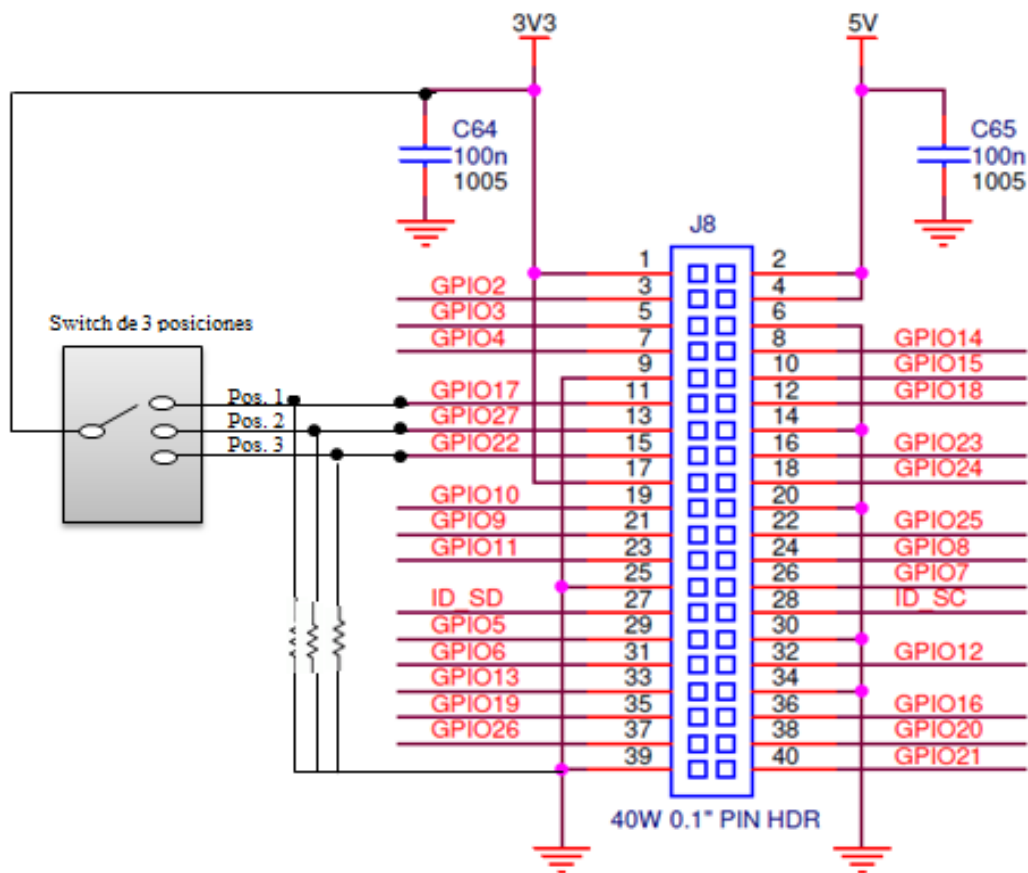


Figura 2-26. Diagrama de conexión del switch a la placa Raspberry Pi B+.
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.9. Conexión y Ensamblaje del Robot.

En este apartado se muestra el proceso de ensamblaje de las piezas del robot impresas, así como también las implementaciones físicas de los diagramas de conexión de los elementos como son actuadores, módulo de la cámara, altavoces, entre otros.

2.9.1. *Ensamblaje de la estructura externa del Robot*

El robot fue diseñado en piezas para posteriormente ser armado; para una mejor explicación se subdividirán en el ensamblaje de: la cabeza, el tronco, las ruedas con los motores MRD y MRI, los brazos en forma de aletas con los motores MBD y MBI, circuitos de conexión a la placa Raspberry Pi B+, la disposición de los sensores S1, S2, el módulo de la cámara y los altavoces.

Para lograr que las piezas permanezcan unidas se realizó la mezcla del pegamento Super Glue con bicarbonato de sodio, así se logró solidez en la unión de piezas de la estructura del robot.

- La cabeza del robot se formó de cinco piezas. Las cuatro piezas forman la parte trasera y frontal de la cabeza y una quinta pieza ubicada en la cara del robot.

En la **Figura 2-27.** se ilustra el ensamblaje de la cabeza del robot.



Figura 2-27. Ensamblaje de la cabeza del robot.
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- El tronco del robot se divide en tres partes: el tronco superior, el tronco inferior y la parte frontal del tronco en donde se ubica el estómago del pingüino.

En la **Figura 2-28.** se ilustra el ensamblaje del tronco del robot.



Figura 2-28. Ensamblaje del tronco del robot.
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- El mecanismo para patear la pelota se ubica en la parte inferior del estómago del pingüino y está formado por tres piezas. Mediante el accionamiento del motor MP acoplado a la pieza se empujará la pelota.

En la **Figura 2-29.** se ilustra el ensamblaje y acople del mecanismo de pateo de la pelota en el robot.



Figura 2-29. Ensamblaje y acople del mecanismo de pateo de la pelota
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Los servomotores están ubicados en las partes laterales superiores del robot, una vez colocados los servomotores se les acoplará los brazos en forma de aletas con la ayuda de tornillos.

En la **Figura 2-30.** se ilustra el ensamblaje y acople de los servomotores MBD, MBI y los brazos en forma de aletas en el robot.



Figura 2-30. Ensamblaje y acople de los brazos en forma de aletas y los servomotores.
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Los bujes serán acoplados al eje de cada motor para posteriormente colocar las ruedas. El conjunto de las ruedas, bujes y motores se ubicarán en las partes laterales inferiores del robot.

En la **Figura 2-31.** se ilustra el ensamblaje y acople de las ruedas, bujes y motores MRD y MRI en el robot.



Figura 2-31. Ensamblaje y acople de las ruedas, bujes y motores en el robot
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Para que el robot mantenga el equilibrio se colocó una rueda loca en la parte inferior del robot. Formando en conjunto con las ruedas lo que se conoce como Locomoción de Triciclo.

En la **Figura 2-32.** se ilustra el ensamblaje y acople de la rueda loca en la parte inferior del robot.



Figura 2-32. Ensamblaje y acople de la rueda loca en la parte inferior del robot
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- El sensor S1y el módulo de la cámara están ubicados en la parte superior del estómago del robot, mientras que el sensor S2 está localizado en la unión de la cabeza con el tronco del robot. El switch está ubicado en la parte trasera del robot.

En la **Figura 2-33**, se ilustra la disposición de los sensores S1 y S2, el módulo de la cámara y el switch en el exterior de la estructura del robot.



Figura 2-33. Disposición de los sensores S1 y S2 en el robot

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Los altavoces fueron colocados en la parte interna de la cabeza del robot.

En la **Figura 2-34**, se ilustra la disposición de los altavoces en la parte interna del robot.

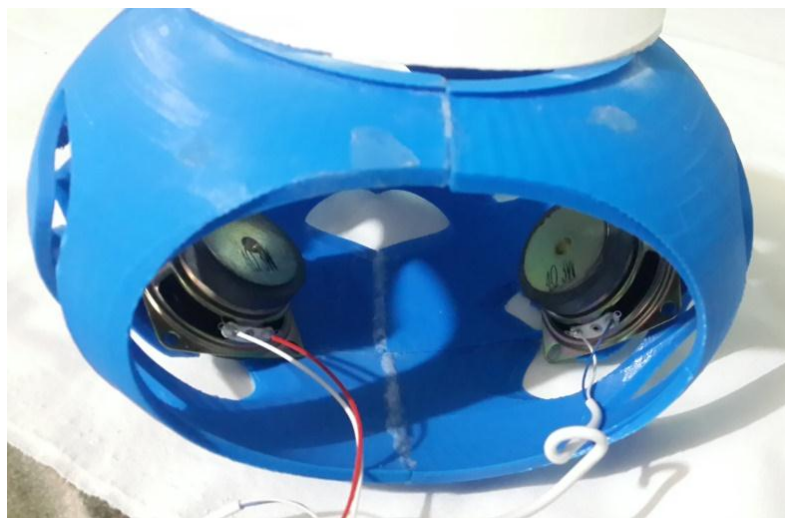


Figura 2-34. Disposición de los altavoces en la parte interna de la cabeza del robot.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- La placa Raspberry Pi B+, los circuitos de regulación de voltaje, el divisor de voltaje para la lectura de la señal del sensor ultrasónico, la batería LIPO; deben estar completamente aislados del medio externo para así evitar la manipulación inapropiada de los elementos mencionados.

En la **Figura 2-35.** se ilustra la disposición de la placa Raspberry Pi B+, los circuitos reguladores de voltaje, el divisor de voltaje y la batería LIPO en la parte interna del robot.



Figura 2-35. Ubicación de la placa Raspberry Pi B+, circuitos y batería en la parte interna del robot.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-36.** se ilustra la disposición de los módulos MR1 y MR2 en la parte interna del robot.

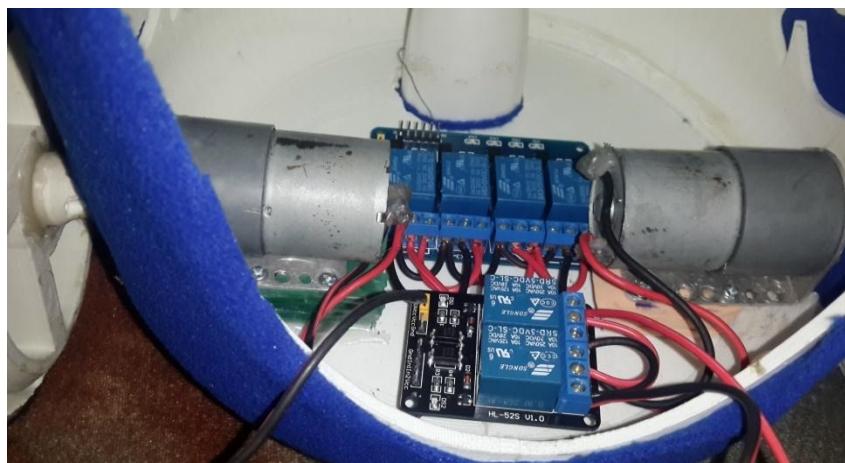


Figura 2-36. Disposición de Módulos MR1 y MR2

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Una vez colocados los elementos del robot en su estructura tanto interna como externa, se procedió a recubrir la estructura del robot para que tenga un aspecto amigable.

En la **Figura 2-37**. se ilustra la vista frontal y lateral del robot terminado.



Figura 2-37. Vista frontal y lateral del robot terminado
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.10. Códigos de Programación

La programación general le permite al robot controlar cada una de sus partes y tomar decisiones en tiempo real adaptándose al entorno y situación en la que se encuentra; está conformada por cuatro subprogramas combinados entre sí.

El primer subprograma permite la selección del color de la pelota, el segundo subprograma está enfocado al Tracking de la pelota, el tercer subprograma controla los motores y el último subprograma controla los sensores de presencia y proximidad.

2.10.1. Código de Programación para seleccionar el color de la pelota

Para realizar la selección del color de la pelota se utilizó un switch que enviará un cero o un uno lógico al puerto GPIO dicha señal será procesada por el programa y a partir de la lectura de dicha señal se iniciará el tracking de la pelota.

En la **Figura 2-38**. se muestra el código implementado para la selección del color de la pelota.

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setup(17, GPIO.OUT) ## GPIO 17 como salida
GPIO.setup(27, GPIO.OUT) ## GPIO 27 como salida
GPIO.setup(22, GPIO.OUT) ## GPIO 27 como salida

if GPIO.output(17, True) ## Enciendo el 17
{
    GPIO.output(27, False) ## Apago el 27
    GPIO.output(22, False) ## Apago el 27
    time.sleep(1) ## Esperamos 1 segundo
    else
    {
        if GPIO.output(27, True) ## Enciendo el 17
        {
            GPIO.output(22, False) ## Apago el 27
            time.sleep(1) ## Esperamos 1 segundo
        }
        else
        GPIO.output(22, True) ## Enciendo el 27
        time.sleep(1) ## Esperamos 1 segundo
    }
}

print "Ejecucion finalizada"
GPIO.cleanup() ## Hago una limpieza de los GPIO
```

Figura 2-38. Código implementado para controlar el switch de selección de color

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.10.2. Programación del Algoritmo de Umbralización aplicado al tracking de la pelota

Una vez conocidos los algoritmos asociados al tracking de objetos se ha escogido el Algoritmo de Umbralización, pues es el que más beneficios aporta al Sistema de VA del robot.

Para implementar un algoritmo óptimo es necesario conocer las características que ofrecen: el Algoritmo de Umbralización, el Algoritmo de Segmentación por color y el Algoritmo CamShift; al Sistema de Visión Artificial del robot.

En la Tabla 2-8. se ilustra las características que ofrece cada algoritmo al Sistema de Visión Artificial mediante la comparación de los mismos.

Tabla 2-8. Comparación entre Algoritmos aplicados al tracking de objetos

Algoritmo Umbralización	Algoritmo de Segmentación por Color	Algoritmo CAMShift
<ul style="list-style-type: none"> • Trabaja con imágenes tipo HSV. • Costo computacional medio. • Resultados muy confiables en entornos controlados de luz. • Capacidad relativamente media para adaptarse a cambios de luz. 	<ul style="list-style-type: none"> • Trabaja en imágenes de tipo RGB • Costo Computacional medio. • Resultados confiables en entornos controlados de luz. • Se puede perder en la búsqueda del objeto al aplicarse cambios bruscos de luz. 	<ul style="list-style-type: none"> • Trabaja con imágenes RGB y con su respectivo histograma. • Costo computacional medio-alto, dependiendo de la precisión que requiera el Sistema de Visión Artificial. • Respuestas confiables en entornos controlados de luz. • Presenta dificultad para dar seguimiento al objeto en cambios externos de luz.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

En base a lo expuesto en la **Tabla 2-8.** se comprueba que el algoritmo de Umbralización es el más óptimo para el Sistema de VA del robot, a continuación se describen las razones:

- Al procesar la imagen en el modelo HSV es posible establecer un color en un rango de valores máximos y mínimos; por ejemplo ir de verdes claros a oscuros.
- Al tener un costo computacional bajo se puede usar los recursos del CPU de la placa Raspberry Pi en el procesamiento de audio, conexión remota, entre otras
- Este algoritmo puede adaptarse a los cambios de luz en la escena, cabe recalcar que los cambios de luminosidad no deben ser extremos.

A continuación se describe el código de programación del proceso del Algoritmo de Umbralización aplicado al Sistema de Visión Artificial del robot.

- Apertura de la cámara: inicia el proceso de activar la cámara mediante la librería picamera.

En la **Figura 2-39.** se muestra el código para iniciar la cámara.

```
from picamera.array import PiRGBArray
from picamera import PiCamera
```

Figura 2-39. Código para iniciar la cámara

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Invocación a la imagen RGB. En la **Figura 2-40**, se muestra el código

```
#graba una imagen de la camara
for frame in camera.capture_continuous(rawCapture, format="bgr", use_video_port=True):
    image = frame.array
```

Figura 2-40. Código para invocar a la imagen RGB

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Establecer parámetros del rango HSV del objeto que se va a buscar. En la **Figura 2-41**, se muestra el código para establecer parámetros de rango HSV para el color verde.

```
#Establecemos el rango de colores que vamos a detectar
#En este caso de verde oscuro a verde-azulado claro
verde_bajos = np.array([70,100,30], dtype=np.uint8)
verde_altos = np.array([109,255,255], dtype=np.uint8)
#Crear una mascara con solo los pixeles dentro del rango de verdes
mask=cv2.inRange(hsv,verde_bajos,verde_altos)
```

Figura 2-41. Código para establecer parámetros de rango HSV para el color verde

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Convertir la imagen RGB a imagen HSV. En la **Figura 2-42**, se muestra el código para convertir la imagen RGB a HSV.

```
hsv=cv2.cvtColor(image,cv2.COLOR_BGR2HSV)
```

Figura 2-42. Código para convertir la imagen RGB a HSV.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Aplicar procesos de umbralización y transformaciones morfológicas. En la **Figura 2-43**, se muestra el código para aplicar procesos de umbralización y transformaciones morfológicas.

```
#Crear una mascara con solo los pixeles dentro del rango de verdes
mask=cv2.inRange(hsv,verde_bajos,verde_altos)
#Encontrar el area de los objetos que detectan la cámara
moments =cv2.moments(mask)
area=moments['m00']
```

Figura 2-43. Código para aplicar procesos de umbralización y transformaciones morfológicas.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Calcular posición X y Y. En la **Figura 2-44.** se muestra el código para calcular la posición X,Y.

```
#Buscamos el centro x,y del objeto
x=int (moments ['m10']/moments ['m00'])
y=int (moments ['m01']/moments ['m00'])
```

Figura 2-44. Código para calcular la posición X,Y.
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Se realiza la impresión de resultados. En la **Figura 2-45.** se muestra el código para imprimir resultados de posición X, Y.

```
#mostramos las coordenadas en pantalla
print "x= ",x
print "y= ",y

#Dibujamos una marca en el centro del objeto
cv2.rectangle (image, (x, y) , (x+2, y+2) , (0, 0, 255) , 2)
```

Figura 2-45. Código para imprimir resultados de posición X, Y
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.10.3. Programación para controlar los actuadores

Para controlar los actuadores se implementó un código de programación usando herramientas que permiten configurar los puertos GPIO a los que se enviarán las órdenes para accionar los motores.

- Para el control de los servomotores MBD y MBI, se implementó el código que permite mover los brazos del robot en un ángulo de 90°, dicho movimiento se efectuará siempre y cuando los motores MRD y MRI estén accionados.

En la **Figura 2-46.** se muestra del código implementado para controlar un servomotor utilizado en los brazos del robot.


```

import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)# COLOCAMOS EL BOARD
GPIO.setup(11, GPIO.OUT) # ESCOGEMOS EL GPIO 11 EN MODO SALIDA
frequencyHertz = 50 #FRECUENCIA ELEGIDA 50 HZ
pwm = GPIO.PWM(11, frequencyHertz) # EXPRESAMOS LA SALIDA DE PWM
leftPosition = 0.75 # VALOR INICIAL IZQ
rightPosition = 2.5 # VALOR INICIAL DER
middlePosition = (rightPosition - leftPosition)/2 + leftPosition #POSICION MEDIA DEL MOTOR
positionList = [leftPosition, middlePosition, rightPosition, middlePosition] # PONER UNA CADENA
#DE VALORES DE LAS POSICIONES
msPerCycle = 1000 / frequencyHertz # FRECUENCIA DE PROCESOS
for i in range(3):
    for position in positionList: #FOR PARA QUE SE UBIQUE EN LOS PUESTOS POSICIONADOS
        dutyCyclePercentage = position * 100 / msPerCycle #POSICION
        print "Posicion : " + str (position)
        print "Duty Cycle." + str (dutyCyclePercentage) + "%\n"
        print ""
        pwm.start(dutyCyclePercentage) #SE ENCIENDE EL PWM SEGUN LA RUTA DE POSICION
        time.sleep(.5)#ESPERA DE 5 SEGUNDOS
pwm.stop()
GPIO.cleanup()

```

Figura 2-46. Código para controlar un servomotor

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Para el control de los motores MRD y MRI, se implementó el código que permite accionar en el momento que centre el área de la pelota. Las señales tratadas en la programación serán IN1, IN2, IN3, IN4 proveniente del módulo de relés.

En la **Figura 2-47.** se muestra del código implementado para activar los pines del módulo de relés y así controlar los motores de las ruedas del robot y el mecanismo para empujar la pelota.

```

#PROGRAMA PARA ACTIVAR LOS RELES
import RPi.GPIO as GPIO
from time import sleep

GPIO.setmode(GPIO.BCM) #SE ACTIVA EL GPIO EN MODO BCM
GPIO.setup(18, GPIO.OUT) #SE USA EL GPIO 18 COMO SALIDA

while (True):
    GPIO.output(18, GPIO.HIGH) #SE ACTIVA LA SALIDA 18
    sleep(5) #TIEMPO DE 5 SEGUNDOS
    GPIO.output (18, GPIO.LOW) #SE DESACTIVA LA SALIDA 18
    sleep(5) #TIEMPO 5 SEGUNDOS

```

Figura 2-47. Código para activar los pines del módulo de relés.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.10.4. Programación para controlar los sensores S1 y S2

La programación que controla las señales recibidas por los sensores S1 y S2 le permitirán al robot tomar decisiones con respecto a distancias frente a obstáculos y detectar movimiento.

- El sensor ultrasónico cuenta con dos señales: trigger y echo; dichas señales serán procesadas en el código implementado.

En la **Figura 2-48**, se muestra el código implementado para controlar el sensor ultrasónico.

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)

TRIG = 23
ECHO = 24

print "Proyección de distancia en trabajo"

GPIO.setup(TRIG,GPIO.OUT) #SE PONE EL GPIO TRIG COMO SALIDA
GPIO.setup(ECHO,GPIO.IN) #SE PONE EL GPIO ECHO COMO ENTRADA

GPIO.output(TRIG,False)# INICIALIZAMOS EL TRIG EN BAJO
print "Espere a que el sensor informe"
time.sleep(2) #ESPERA DE DOS SEGUNDOS

GPIO.output(TRIG, True) #ACTIVAMOS EL GPIO EN ALTO
time.sleep(0.00001)# TIEMPO PEQUEÑO PARA EL ENVIO DE SEÑAL
GPIO.output(TRIG,False) #APAGAMOS EL TRIG

while GPIO.input(ECHO)==0:
    pulse_star=time.time() #CONDICION POR SI NO HALLO OBJETO QUE BUSCAR

while GPIO.input(ECHO)==1:
    pulse_end=time.time()# CONDICION SI HALLO OBJETO
    pulse_duration = pulse_end - pulse_star #RECORTAMOS LA DURACION TOTAL DEL IMPULSO

    distance = pulse_duration * 17150 #REALIZAMOS LA CONVERSION A DISTANCIA
    distance = round (distance,2) #CAMBIAMOS FORMATO DEL VALOR

    print "Distance:", distance, "cm" #IMPRIMIMOS LA DISTANCIA
GPIO.cleanup()
```

Figura 2-48. Código para controlar el sensor ultrasónico
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- El sensor PIR cuenta con una señal, la misma que se activará en caso de detectar movimiento. La señal recibida por el puerto GPIO será tratada en el código implementado.

En la **Figura 2-49**, se muestra el código implementado para controlar el sensor PIR.

```

import RPi.GPIO as GPIO
import time
from time import gmtime, strftime
GPIO.setmode(GPIO.BCM)
PIR_PIN = 24          #AL PIN 24 SE LE DA UN ALIAS PIR_PIN
GPIO.setup(PIR_PIN, GPIO.IN) #SE PONE EL GPIO TRIG COMO ENTRADA

try:
    while True:
        if GPIO.input(PIR_PIN): # PREGUNTA SI LA ENTRADA DEL PIN ES POSITIVA
            GPIO.output(24,True)
            time.sleep(1)
            timex = strftime("%d-%m-%Y %H:%M:%S", gmtime())
            print timex + "MOVIMIENTO DETECTADO" #IMPRIME SI DETECTA UN MOVIMIENTO
            time.sleep(1)
except KeyboardInterrupt:
    print "salida"
    GPIO.cleanup()

```

Figura 2-49. Código implementado para controlar el sensor PIR.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.10.5. Programación para emitir mensajes de audio

Como se mencionó en apartados anteriores el robot deberá emitir mensajes de audio pregrabados a medida que el juego avance para evitar que el niño se aburra fácilmente y crear un vínculo amistoso entre el prototipo y el niño.

Se han grabado con la ayuda de un celular cinco mensajes en formato mp3 que serán reproducidos en distintos momentos del juego. El primero y último mensaje se reproducirá una sola vez en la secuencia de programación.

En la **Tabla 2-9**, se muestra los mensajes que emitirá el robot a través de los parlantes instalados en la estructura interna de la cabeza del prototipo.

Tabla 2-9. Mensajes emitidos por el robot.

NÚMERO DE MENSAJE	MENSAJE A REPRODUCIR	MOMENTO EN EL SE REPRODUCIRÁ EL MENSAJE
1	“Hola mi nombre es Skiper y estoy feliz de jugar contigo”	Al iniciar el juego
2	“Empuja la pelota y yo la buscaré”	Después del saludo inicial
3	“Bien hecho, ahora es mi turno”	Luego de que exista un movimiento de la pelota.
4	“Oh no encuentro la pelota, la seguiré buscando”	Luego del segundo intento de localizar la pelota con la cámara.
5	“Me divertí jugando contigo, hasta pronto”	Al finalizar el juego.

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 2-50.** se muestra del código implementado que permitirá al robot emitir mensajes.

```
import pygame #Importa la biblioteca de audio
pygame.mixer.init()
pygame.mixer.music.load(SaludoInicial.mp3) #Reproduce el audio .mp3
pygame.mixer.music.play()
while pygame.mixer.music.get_busy() == True:
```

Figura 2-50. Código implementado para emitir archivos audio

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

2.11. Método de Aprendizaje de reconocimiento del objeto

Para el aprendizaje de reconocimiento del objeto se utilizaron dos métodos explicados a continuación:

- En primera instancia se propuso un rango de color en espacio HSV lo más cercano posible a los colores reales de las pelotas escogidas para las pruebas, con ello se elimina distintos objetos que pudieran aparecer en la escena con características similares pero no idénticas al objeto.
- Para el siguiente método se utilizó la función de OpenCV Canny que sirvió para procesar los bordes y las esquinas de una imagen. Sabiendo que esta función se ejecuta por medio de

rangos y que toma en cuenta los gradientes de la escena; entonces el programa debe realizar la lectura de una imagen modelo y determinar el rango adecuado del objeto al cual se le va a ser el seguimiento para posteriormente guardar la información en una variable propuesta.

Los pasos a seguir para detectar figuras con OpenCV son:

PASO 1: Crear la máscara de un color.

PASO 2: Eliminar ruido de la máscara.

PASO 3: Aplicar un hough circles a la máscara lo cual ayudará a detectar mejor los bordes.

PASO 4: Detectar los diferentes contornos: guardaremos los puntos de cada contorno.

PASO 5: Aproximar cada contorno para verificar si es la figura deseada.

En la **Figura 2-51**, se muestra el código para determinar el contorno de un círculo, que es el se necesita para detectar la figura de la pelota.

```
// Determinar si es un círculo
double area = cv::contourArea(contours[i]);
cv::Rect r = cv::boundingRect(contours[i]);
int radius = r.width / 2;

if (std::abs(1 - ((double)r.width / r.height)) <= 0.2 &&
    std::abs(1 - (area / (CV_PI * std::pow(radius, 2)))) <= 0.2)
    setLabel(dst, "CIR", contours[i]);
```

Figura 2-51. Código para determinar el contorno de un círculo

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Finalmente en el muestreo de la imagen se debe comparar los distintos objetos para comprobar si se encuentran dentro del rango establecido tanto en valores del Modelo HSV como en los rangos establecidos para detectar los bordes. En el **ANEXO K** se ilustran los parámetros para reconocer patrones de diferentes figuras.

2.12. Descripción del funcionamiento del Robot

Para una mejor explicación del funcionamiento del robot se subdividió en las etapas primordiales que efectúa el Sistema de Visión Artificial para cumplir con los requerimientos deseados.

El robot toma decisiones de acuerdo al entorno en el que se encuentre, el entorno está basado en parámetros como: el color de la pelota, la ubicación de la pelota, la presencia cercana del niño al robot, la presencia de obstáculos, reproducción de mensajes en diferentes etapas del juego.

En la **Tabla 2-10.** se describe las etapas de funcionamiento del robot en un entorno dado.

Tabla 2-10. Descripción de las etapas de funcionamiento del robot

ETAPA	DESCRIPCIÓN
<p style="text-align: center;">INICIO</p>	<ul style="list-style-type: none"> • Se selecciona el color de la pelota mediante la manipulación del switch de tres estados. • Se procede a encender el robot mediante la pulsación de un interruptor. • El robot reproduce una sola vez el saludo inicial “Hola mi nombre es Skipper y estoy feliz de jugar contigo” • El robot reproduce una sola vez el mensaje “Empuja la pelota y yo la buscaré”
<p style="text-align: center;">RECEPCIÓN DE SEÑALES DE LOS SENSORES S1 Y S2</p>	<ul style="list-style-type: none"> • Los sensores S1 y S2 inmediatamente empezarán a enviar señales a través de los puertos GPIO a los que están conectados; para mantener una distancia programada entre el robot y el niño, entre el robot y obstáculos. Este bucle se ejecutara desde que inicia el juego hasta que se termina el mismo.
<p style="text-align: center;">TRACKING DE LA PELOTA</p>	<ul style="list-style-type: none"> • El módulo de la cámara captura la escena para aplicarle el Algoritmo de Umbralización y así separar la pelota del resto de la escena capturada. • Una vez detectada la pelota se procede a verificar que la pelota se encuentre en el centro del ángulo de visión del módulo dela cámara.
<p style="text-align: center;">ACCIONAMIENTO DE LOS MOTORES MRD, MRI, MBD Y MBI PARA BUSCAR LA PELOTA</p>	<ul style="list-style-type: none"> • Los servomotores acoplados a los brazos del robot se accionarán siempre y cuando este accionados los motores MRD y MRI; realizarán un movimiento constante de 0 a 45°.

	<ul style="list-style-type: none"> • Se accionarán los motores MRD y MRI del robot al mismo tiempo si la pelota está centrada en el ángulo de visión del módulo de la cámara. • Se accionará únicamente el motor MRD del robot si la pelota está ubicada al lado izquierdo del centro del ángulo de visión del módulo de la cámara. • Se accionará únicamente el motor MRI del robot si la pelota está ubicada al lado derecho del centro del ángulo de visión del módulo de la cámara. • En caso de no encontrar la pelota el robot girará sobre su propio eje hacia la derecha hasta encontrarla. • Si al segundo intento de búsqueda el robot no encuentra la pelota reproducirá el mensaje “Oh no encuentro la pelota, la seguiré buscando”.
<p style="text-align: center;">ACCIONAMIENTO DEL MOTOR MP</p>	<ul style="list-style-type: none"> • Una vez que el robot se encuentra en frente de la pelota a una distancia menor a 5cm se accionara el motor MP que empujará la pelota • El robot espera 7 segundos luego de haber empujado la pelota y emite el mensaje “Bien hecho, ahora es mi turno” • El robot regresa a la etapa del tracking de la pelota para continuar con el juego
<p style="text-align: center;">FIN</p>	<ul style="list-style-type: none"> • Una vez finalizado el juego el robot reproducirá el siguiente mensaje “Me divertí jugando contigo, hasta pronto”

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.
Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

CAPITULO III

3. MARCO DE RESULTADOS, ANÁLISIS Y DISCUSIÓN

Este capítulo es de gran importancia, ya que se efectúa la medición de los resultados obtenidos mediante las pruebas realizadas a cada elemento del Sistema de VA; y así garantizar el funcionamiento óptimo del robot en un entorno dado. A medida que se realizan las pruebas se van corrigiendo los errores detectados en los elementos del Sistema de VA del robot.

En los apartados siguientes se describen las pruebas realizadas a: los motores MRD y MRI para el desplazamiento del robot, al motor MP, a los servomotores MBD y MBI, a los sensores S1 y S2 y las pruebas realizadas al tracking de la pelota en base a un color seleccionado.

3.1. Pruebas de Funcionamiento del robot

Las pruebas de funcionamiento del robot se subdividieron en tres partes:

- **Pruebas para el funcionamiento de los sensores S1 y S2:** se realiza la calibración en cuanto a sensibilidad y precisión.
- **Pruebas para el funcionamiento de los motores MRD y MRI:** se realiza la calibración basado en la velocidad.
- **Pruebas para el tracking de la pelota:** se realiza la toma de coordenadas de la pelota en tiempo real.

3.1.1. Calibración de Sensores S1 y S2

Para la calibración de los sensores se implementaron las conexiones de diagramas de la **Figura 23-2**. para posteriormente capturar los resultados leídos por los puertos GPIO asignados a lectura de las señales de los sensores S1 y S2.

- Para comprobar el funcionamiento del sensor Ultrasónico en la pantalla del LXTerminal de la placa Raspberry Pi se imprimirá la distancia en cm, entre la parte frontal del robot y el obstáculo detectado.

En la **Figura 3-1.** se muestran distancias sensadas en cm por el Sensor Ultrasónico HC-SR04.

```

root@raspberrypi:/home/pi# sudo python sensor_prof.py
18. 6084866524
12. 9821896553
13. 0312561989
12. 8104567528
13. 1334781647
13. 20707798
13. 0149006844
13. 6691212654
14. 2374753952
14. 3519639969
14. 3887639046
15. 2474284172
15. 349650383
15. 4559612274
15. 4355168343
15. 3864502907
15. 194272995
15. 4518723488
15. 3169393539
15. 505027771
15. 0225400925
13. 8572096825
13. 5832548141
11. 9722366333
11. 7473483086
11. 4733934402
11. 4938378334
12. 0376586914
11. 2525939941
11. 1626386642
11. 2157940865
10. 7169508934
10. 9745502472
10. 7005953789
10. 4757070541
10. 6842398643
10. 5820178986
9. 87873077393
10. 1363301277
9. 91144180298
10. 1690411568
9. 55570936203
9. 34717655182

```

Figura 3-1. Distancias en cm sensadas por el Sensor Ultrasónico HC-SR04
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Para probar el funcionamiento del sensor PIR nos dirigiremos al LXTerminal de la placa Raspberry Pi; en él se imprimirá el mensaje “Movimiento Detectado” en caso de que el sensor perciba el movimiento del niño.

En la **Figura 3-2.** se muestran los movimientos detectados por el Sensor PIR HC-SR501, en la pantalla del LXTerminal.

```

root@raspberrypi:/home/pi# sudo python sensorpir.py
24-05-2016 04:20:05MOVIMIENTO DETECTADO
24-05-2016 04:20:07MOVIMIENTO DETECTADO
24-05-2016 04:20:11MOVIMIENTO DETECTADO
24-05-2016 04:20:13MOVIMIENTO DETECTADO
24-05-2016 04:20:15MOVIMIENTO DETECTADO
24-05-2016 04:20:17MOVIMIENTO DETECTADO
24-05-2016 04:20:19MOVIMIENTO DETECTADO
24-05-2016 04:20:21MOVIMIENTO DETECTADO
24-05-2016 04:20:23MOVIMIENTO DETECTADO
24-05-2016 04:20:25MOVIMIENTO DETECTADO
24-05-2016 04:20:27MOVIMIENTO DETECTADO
24-05-2016 04:20:29MOVIMIENTO DETECTADO
24-05-2016 04:20:33MOVIMIENTO DETECTADO
24-05-2016 04:20:35MOVIMIENTO DETECTADO
24-05-2016 04:20:37MOVIMIENTO DETECTADO
24-05-2016 04:20:39MOVIMIENTO DETECTADO
24-05-2016 04:20:41MOVIMIENTO DETECTADO
24-05-2016 04:20:43MOVIMIENTO DETECTADO
24-05-2016 04:20:45MOVIMIENTO DETECTADO
24-05-2016 04:20:47MOVIMIENTO DETECTADO
24-05-2016 04:20:49MOVIMIENTO DETECTADO
24-05-2016 04:20:51MOVIMIENTO DETECTADO
24-05-2016 04:21:01MOVIMIENTO DETECTADO
24-05-2016 04:21:03MOVIMIENTO DETECTADO
24-05-2016 04:21:05MOVIMIENTO DETECTADO
24-05-2016 04:21:07MOVIMIENTO DETECTADO
24-05-2016 04:21:09MOVIMIENTO DETECTADO
24-05-2016 04:21:11MOVIMIENTO DETECTADO

```

Figura 3-2. Movimientos detectados por el Sensor PIR HC-SR501
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

3.1.2. Calibración de velocidad de Motores

Para encontrar la velocidad adecuada a la cual el robot se desplaza mediante el accionamiento de los motores MRD y MRI se les aplico diferentes niveles de voltaje, que van desde los 4V DC hasta los 12V DC.

En la **Tabla 3-1.** se ilustra las revoluciones por minuto de los motores MRD y MRI alimentados a diferentes niveles de voltaje.

Tabla 3-1. Frecuencia en base al Voltaje aplicado al motor MRD

VOLTAJE (V DC)	FRECUENCIA (RPM)	VOLTAJE (V DC)	FRECUENCIA (RPM)
4	26	8.5	56
4.5	30	9	60
5	33	9.5	63
5.5	36	10	66
6	40	10.5	70
6.5	43	11	73
7	46	11.5	76
7.5	50	12	80

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

En base a las pruebas realizadas a los motores aplicándoles diferentes valores de voltaje; se ha obtenido los siguientes resultados:

- Cuando se les aplicaba a los motores MRD y MRI voltajes menores a 5V DC la velocidad reducía considerablemente, y provocaba que el robot tome más tiempo en llegar a la ubicación de la pelota.
- Cuando se les aplicaba a los motores MRD y MRI voltajes de 6 a 7 V DC la velocidad de los motores provocaba una desaceleración sutil.
- Cuando se les aplicaba a los motores MRD y MRI voltajes mayores a 8V DC la velocidad incrementaba y provocando una desaceleración brusca de los motores.

Considerado los resultados obtenidos en cuanto a velocidad y desaceleración de los motores MRD y MRI se ha decidido aplicar un voltaje de 6V DC a dichos motores, pues se observó que el robot mantenía una velocidad promedio para desplazarse y desaceleraba de una manera sutil.

A los servomotores MBD y MBI se les aplicó voltajes de 4V DC provenientes de la primera celda de la batería con los que se logró movimientos constantes de 0 a 45°.

Al motor MP se le aplicó un voltaje de 6V DC provenientes de la segunda celda de la batería regulada previamente a 6V DC, con lo que se logró un movimiento rápido y fuerte para empujar la pelota.

3.1.3. Pruebas de seguimiento de la pelota

Se realizaron las pruebas del tracking o seguimiento para cada uno de los tres colores entre los que se puede elegir para iniciar el juego. A cada color se les asignó un rango de valores en el Modelo HSV.

- En base a los parámetros establecidos en el rango del modelo HSV para rojos bajos (172, 100, 30) y rojos altos (255, 255, 255) mediante un editor de imágenes, se realizaron las pruebas para seguimiento de la pelota roja.

En la **Figura 3-3.** se ilustra el tracking de la pelota roja.

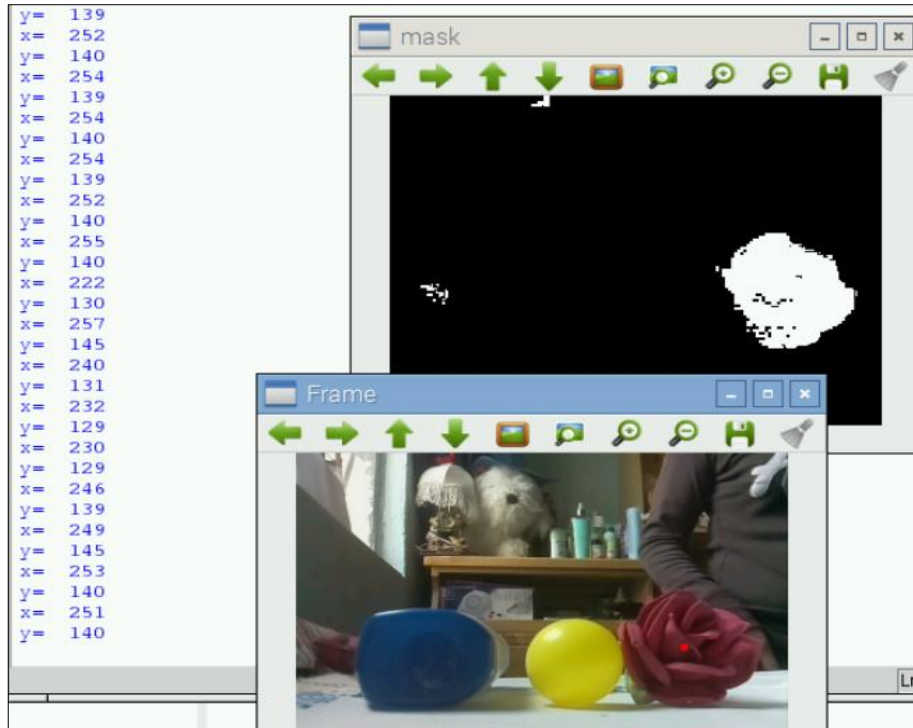


Figura 3-3. Tracking de la pelota roja.
 Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- En base a los parámetros establecidos en el rango del modelo HSV para azules bajos (70, 100, 30) y azules altos (109, 255, 255) mediante un editor de imágenes, se realizaron las pruebas para seguimiento de la pelota azul.

En la **Figura 3-4.** se ilustra el tracking de la pelota azul.

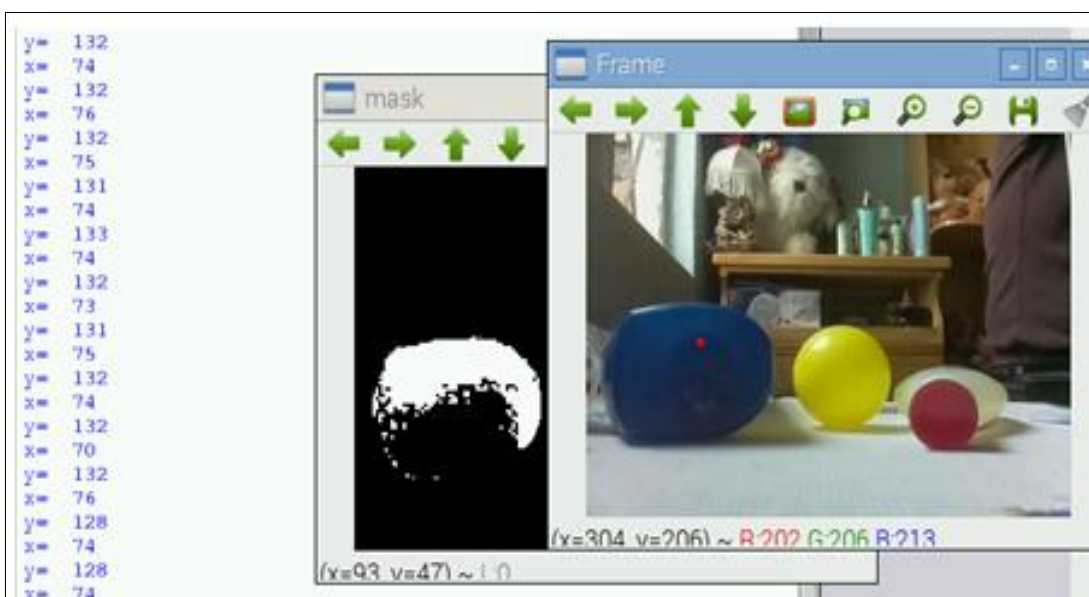


Figura 3-4. Tracking de la pelota azul
 Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Finalmente En base a los parámetros establecidos en el rango del modelo HSV para verdes bajos (31, 100, 30) y verdes altos (69, 255, 255) mediante un editor de imágenes, se realizaron las pruebas para el seguimiento de la pelota verde.

En la **Figura 3-5.** se ilustra el tracking de la pelota verde.

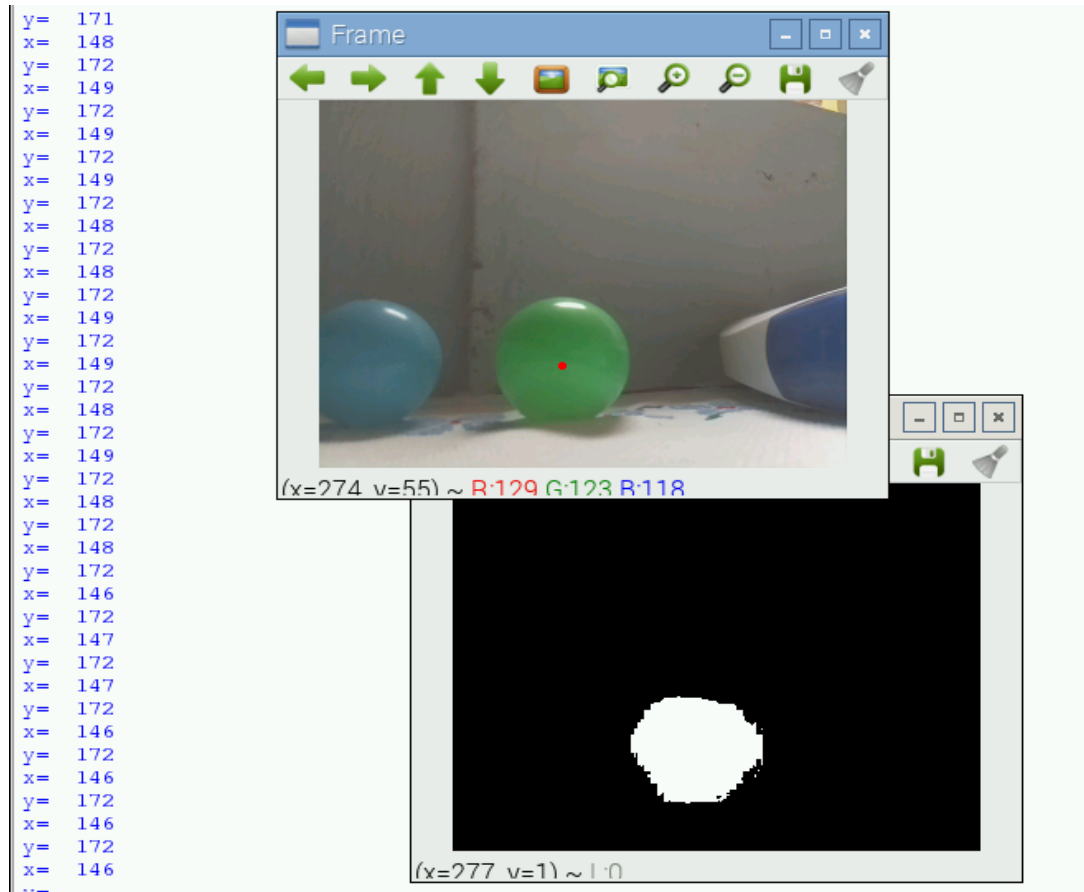


Figura 3-5. Tracking de la pelota verde
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

El código permite conocer las coordenadas X, Y del objeto localizado; en base a las coordenadas obtenidas el robot accionará los motores MRD y MRI para desplazarse hacia la ubicación de la pelota.

Además en la pantalla del LXTerminal se podrá visualizar las coordenadas y verificar que así que las coordenadas de la ubicación de la pelota son certeras.

En la **Tabla 3-2.** se muestran los datos de las coordenadas tomados en las pruebas con los tres tipos de colores.

Tabla 3-2. Coordenadas del objeto localizado a través del módulo de la cámara.

PELOTA ROJA		PELOTA AZUL		PELOTA VERDE	
Eje X	Eje Y	Eje X	Eje Y	Eje X	Eje Y
251	137	63	118	200	147
253	140	64	150	202	156
245	138	92	149	199	150
247	142	92	149	196	149
246	149	92	150	198	150
244	148	93	149	197	158
240	147	92	144	195	160
245	146	92	145	190	164
249	144	93	144	193	150
248	149	92	146	196	158
246	145	96	144	198	145
240	145	97	145	200	138
247	147	95	144	200	134
244	146	96	145	148	140
250	148	97	119	151	139
246	145	97	149	166	131
248	144	97	145	170	140
247	143	95	147	253	149
244	146	97	145	239	137
244	143	96	144	153	134
245	143	97	143	157	145
249	149	97	144	180	150
247	147	95	146	185	153
245	147	98	145	149	139
246	145	94	148	184	137
246	144	97	144	190	149
249	146	96	150	190	156
245	149	98	144	186	149
247	145	97	145	172	137
250	140	100	150	180	135
245	149	97	145	195	149
249	145	97	144	190	156
247	145	95	146	193	149
245	147	98	144	196	137
250	146	90	145	198	135

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

3.2. Análisis de Locomoción y Seguimiento de Trayectoria

Una vez encontrada la pelota a través del módulo de la cámara el robot se desplazara de acuerdo a la ubicación de la pelota en el ángulo de visión del módulo de la cámara con un ángulo de visión de $54^{\circ} \times 41^{\circ}$.

En la **Figura 3-6**. se ilustra el ángulo de visión horizontal del módulo de cámara

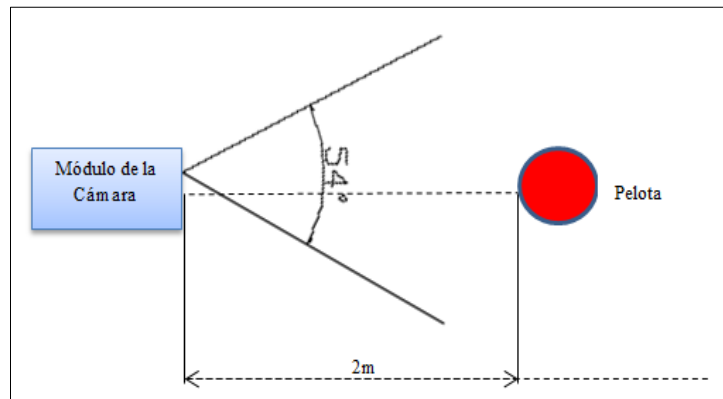


Figura 3-6. Ángulo de visión horizontal de la cámara.
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 3-7**. se ilustra el ángulo de visión vertical del módulo de cámara

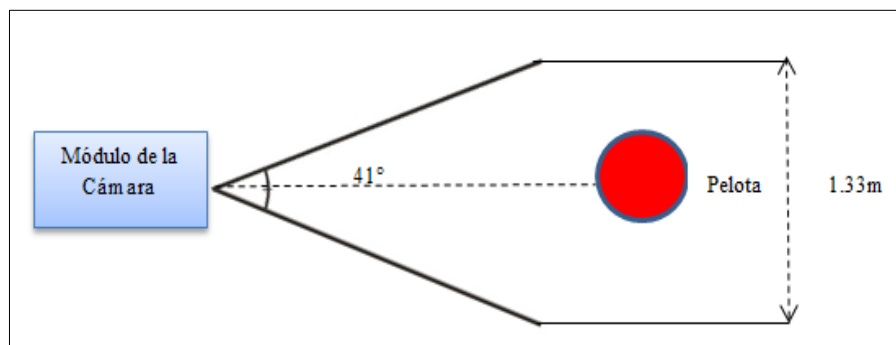


Figura 3-7. Ángulo de visión vertical de la cámara
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

A continuación se describen los desplazamientos posibles en base a la ubicación de la pelota en el ángulo de visión:

- Se desplazará en línea recta si la pelota encontrada se ubica en el centro del ángulo de visión del módulo de la cámara.

En la **Figura 3-8.** se muestra el desplazamiento en línea recta del robot.

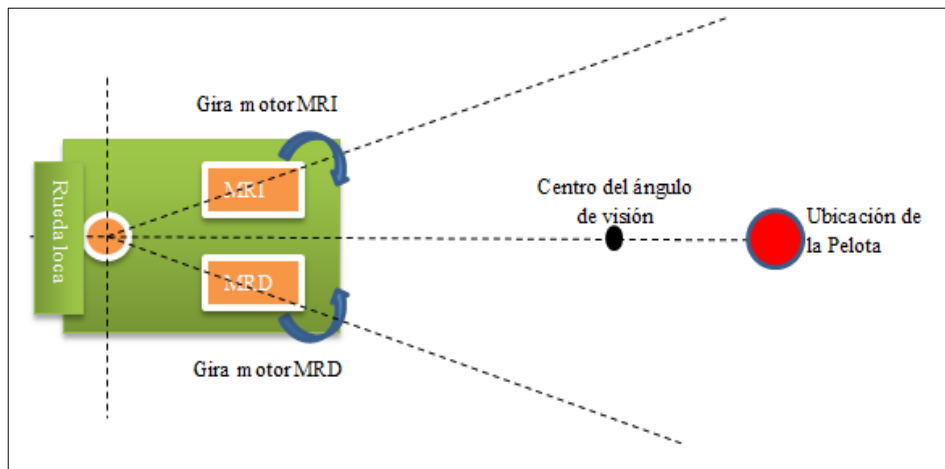


Figura 3-8. Desplazamiento del robot en línea recta
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

- Si la pelota encontrada no se halla en el centro del ángulo de visión del módulo de la cámara, el robot deberá accionar el motor MRD si la pelota se encuentra a la izquierda del centro del ángulo de visión; mientras que si la pelota se encuentra a la derecha del centro del ángulo de visión el robot deberá accionar el motor MRI; dichos desplazamientos los realiza hasta centrar la pelota en el ángulo de visión.

En la **Figura 3-9.** se muestra el desplazamiento con giro hacia la izquierda del robot.

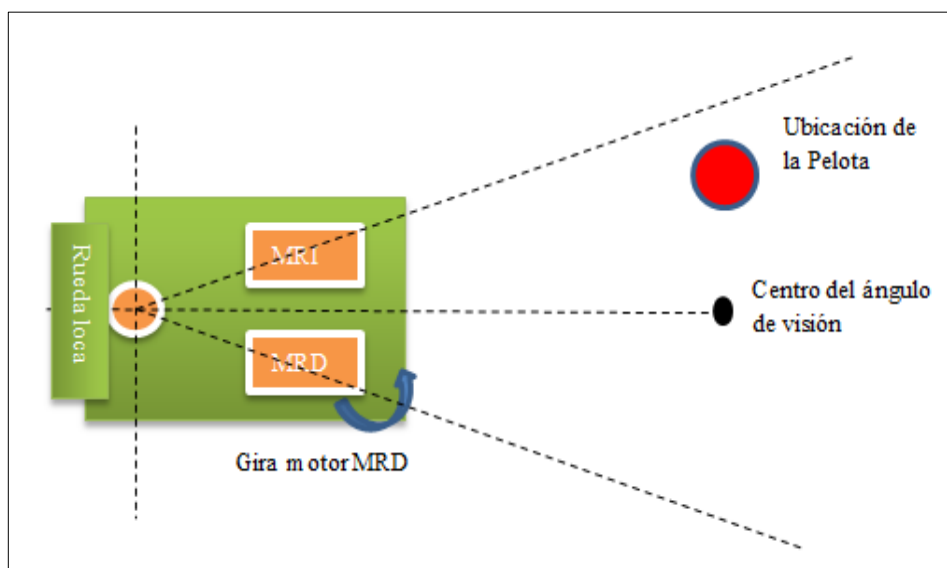


Figura 3-9. Desplazamiento del robot con giro hacia la izquierda
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

En la **Figura 3-10**, se muestra el desplazamiento con giro hacia la derecha del robot.

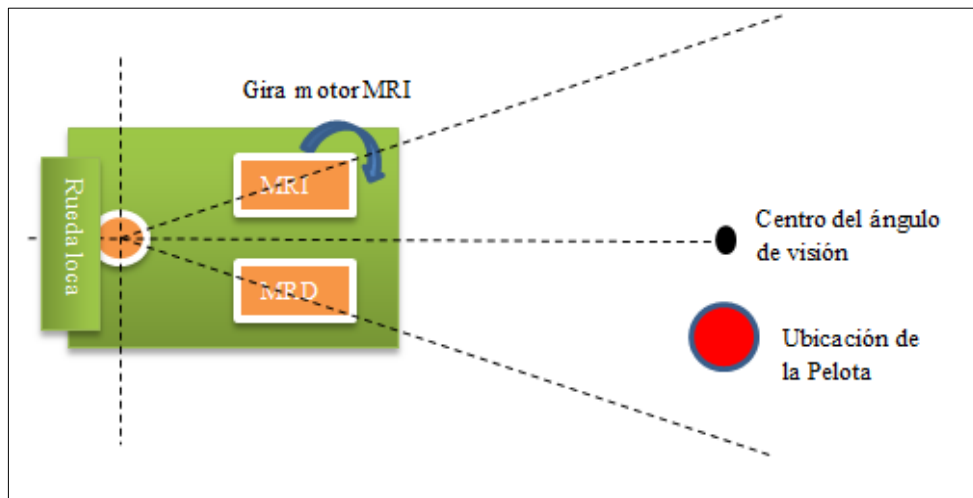


Figura 3-10. Desplazamiento del robot con giro hacia la derecha
Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

3.3. Detección y Corrección de errores

Los errores detectados y corregidos tanto en la parte de programación como en la parte de hardware se describen a continuación:

- En un principio se colocó en el código de programación una distancia máxima entre el robot y un obstáculo de 20 cm, la misma que es transmitida a la placa Raspberry Pi B+ por el sensor ultrasónico HC-SR04; sin embargo se observó que esta distancia no era suficiente para evitar que el robot impactara con el obstáculo, por lo que en base a pruebas realizadas la distancia óptima es de 50 cm.
- En cuanto a la velocidad de los motores MRD y MRI en un comienzo se les aplicaron niveles de voltajes mayores a los 7V DC y se observó que el robot en el momento de detenerse lo realizaba de una manera rápida y brusca tendiendo a caerse hacia adelante. En base a pruebas realizadas se tomó la media de los valores medidos siendo el voltaje de 6V DC el indicado para este proceso.
- La cámara se colocó en un principio en la nariz del robot; sin embargo se notó dificultades al momento de centrar la pelota en el ángulo de visión vertical del módulo de la cámara; por ello se decidió colocar la cámara en la parte superior del tronco del robot; permitiéndole así encontrar la pelota con mayor facilidad.

3.4. Interacción Niño-Robot

Previo al juego se procedió a explicar al niño sobre el funcionamiento del robot Skiper, para así ayudar a la interacción entre el niño y el robot; además de evitar confusiones en el niño sobre el juego.

- Se procedió a presentarle el robot Skiper al niño; indicando lo que el robot hace.
- Luego se le indico el proceso del juego al niño.

Debido a la orientación educativa infantil del proyecto se tuvo la necesidad de realizar pruebas en distintos niños sobre el funcionamiento del robot en los cuales hemos propuestos dos ítems de evaluación como: nivel de interés del niño y tiempo de interacción voluntaria del niño. Los resultados de las pruebas de interacción realizadas son descritos tanto cualitativa como cuantitativamente.

Las pruebas de interacción se realizaron a cuatro niños de diferentes edades, a continuación se describe los resultados cualitativos con respecto a la reacción de cada niño frente al robot, además de su actitud en el juego:

- En la edad de 1 a 2 años los niños empiezan a caminar y decir sus primeras palabras. Al interactuar el robot con un niño de esta etapa se observó que en un principio el niño le llamo la atención; comenzando el juego, sin embargo necesito de la ayuda de una persona adulta para seguir la dinámica del mismo. Se notó el interés del niño en permanecer junto al robot e ir tras la pelota, estimulando así sus capacidades psicomotrices y sensoriales.
- A partir de los 2 a 4 años de edad los niños empiezan a explorar el mundo que los rodea, desarrolla habilidades psicomotrices; adema empiezan a relacionarse con otros niños. Al interactuar el robot con un niño de esta etapa se observó mucha más emoción e interés en el juego, el niño trato de comunicar a sus familiares la experiencia del juego, además despertó su curiosidad sobre el robot, trataba de imitar los movimientos y mensajes emitidos por robot.
- A partir de los 4 a 6 años de edad los niños se tornan más curiosos al preguntar por todo lo desconocido, empiezan a mostrar independencia al escoger su ropa y ponérsela. Al interactuar el robot con el niño, se observa la curiosidad del niño, puesto que realiza muchas preguntas con respecto al robot, las más comunes fueron las siguientes:

Pregunta 1. ¿Qué es?

Pregunta 2. ¿Cómo se llama?

Pregunta 3. ¿Qué hace?

Pregunta 4. ¿Puedo jugar con él?

Cabe mencionar que uno de los niños no interactuó fácilmente con el prototipo, notándose rasgos de timidez en su personalidad. Según lo descrito por Martha Vásquez en la web www.consumer.es el niño tímido suele mostrar las siguientes actitudes:

- El niño intenta evitar a las personas que no le resultan familiares.
- El niño prefiere estar solo. Antes que integrarse a un grupo.
- Al estar con otros niños, suelen estar callados y son poco participativos.
- Se muestran recelosos con todo lo que desconocen.

En la **Tabla 3-3.** se observa los resultados cuantitativos de las pruebas realizadas de interacción Niño-Robot.

Tabla 3-3. Parámetros tomados en cuenta para las pruebas de interacción Niño-Robot.

Parámetros de Evaluación Rango de edad del niño	NIVEL DE INTERÉS DEL NIÑO EN EL JUEGO.	TIEMPO DE INTERACCIÓN VOLUNTARIA DEL NIÑO
1 a 2 años	Bajo	15 minutos
2 a 4 años	Medio	20 minutos
4 a 6 años	Alto	25 minutos

Realizado por: Cayambe, Diana; Castañeda, Juan, 2016.

Fuente: Cayambe, Diana; Castañeda, Juan, 2016.

3.5. Resultados

Los resultados obtenidos en el presente trabajo de titulación se determinaron a través de pruebas realizadas a los elemento del Sistema de VA. La realización de dichas pruebas tiene como finalidad dar respuesta a las preguntas de Sistematización del Problema formuladas en el anteproyecto y descritas en el Planteamiento del Problema del presente escrito.

A continuación se dará respuesta a las preguntas planteadas:

1. ¿Qué tecnología a utilizar de los diferentes sistemas embebidos existentes será eficiente en este proyecto para lograr los objetivos propuestos?

El sistema embebido elegido para el proyecto fue Raspberry Pi B+, ya que al ser microcomputador posee en una sola placa los circuitos que permiten controlar los componentes de un computador habitual, como son: puertos USB, controlador Ethernet, conector para cámara, puerto de video, puerto audio y puertos GPIO.

En el sistema de Visión Artificial implementado se hizo uso de los puertos mencionados para controlar los actuadores, reproducir los mensajes, capturar las imágenes en tiempo real y realizar el tracking de la pelota

La placa raspberry Pi B+ permite instalar diversos software en su Sistema Operativo Raspbian, facilitando así el uso de herramientas y bibliotecas al momento de realizar la programación del Sistema de VA del robot.

2. ¿Qué tipo de locomoción será la más óptima para el tipo de movimiento que realizará el prototipo?

La locomoción triciclo es la más óptima, ya que el robot necesita realizar movimientos en línea recta para dirigirse a la pelota una vez localizado el centro del área de la misma, realizar giros para localizar el centro de la pelota, dar la vuelta en caso de encontrarse con obstáculos.

Además se tuvo en cuenta el tamaño y peso del robot, pues al implementar el tipo de locomoción mencionado se logró mantener el equilibrio de la toda la estructura, distribuyendo el peso de todos los elementos; con la ayuda de las ruedas controladas colocadas en las partes laterales inferiores del robot y una rueda no controlada ubicada en la base del robot.

3. ¿Ayudará la Visión Artificial a que el prototipo localice una pelota en el espacio; dando así una ubicación más exacta?

Sí, porque los recursos ocupados en el proyecto fueron escogidos precisamente para aprovechar sus características de precisión y visión real de la escena sensada; una vez conocidas las propiedades que nos ofrece la Visión Artificial en cuanto a la posición de un objeto que se desea

seguir es posible interactuar con diferentes dispositivos y actuadores para que así el robot se desplace hasta la ubicación del objeto.

La Visión Artificial permite descomponer la escena capturada por el módulo de la cámara de la placa Raspberry PI B+. Al segmentar la escena es posible realizar el procesamiento del objeto segmentado en base a su color y forma, permite también conocer la ubicación de la pelota en coordenadas X e Y.

4. ¿Qué herramientas de OPEN-SOURCE serán las más indicadas para resolver los problemas propuestos?

El sistema operativo Raspbian Wheezy es un software de código abierto que permite la instalación de paquetes y bibliotecas utilizadas en sistema de Visión Artificial del robot.

Las herramientas que ofrece OpenCV permitieron el desarrollo del sistema de Visión Artificial en tiempo real del robot mediante parámetros como: la estructura de datos, procesamiento y análisis de imágenes, reconocimiento de patrones, calibración de la cámara, reconocimiento y tracking de la pelota.

Mediante el uso de las herramientas que ofrece Python es posible importar las bibliotecas al código de programación para controlar: sensores, servomotores, motores reductores, switch, salida de audio; a través de los puertos GPIO a los que se encuentran conectadas las señales de cada elemento.

5. ¿Cómo influirá el prototipo en el desarrollo intelectual y psicomotriz del niño?

En la etapa de la infancia el juego es el instrumento de enseñanza de mayor relevancia, por ello al interactuar el robot con el niño en un juego, influye en el desarrollo de sus habilidades tanto intelectuales como psicomotrices.

En cuanto al desarrollo intelectual del niño el robot ayudará a: estimular la imaginación, mejorar la concentración y memoria, perfeccionar la vocalización de palabras; ya que el niño observará como el robot se desplace para buscar la pelota hasta encontrarla y también escuchará los mensajes que reproducirá el robot a medida que avance el juego.

En el desarrollo psicomotriz el niño mejorará su equilibrio y realizará movimientos más precisos al patear la pelota, además mejorará sus capacidades sensoriales. Al moverse en un espacio el niño adquirirá una mayor fuerza en sus músculos.

CONCLUSIONES

1. En el presente proyecto se ha implementado un prototipo robótico al que se le ha incorporado los elementos básicos de un Sistema de Visión Artificial. El funcionamiento del robot radica en el accionamiento de actuadores en base a la ubicación de la pelota obtenida mediante el tracking de la misma, la placa Raspberry Pi B+ sirvió de plataforma para la programación que controla los actuadores del robot.
2. Existen diversos algoritmos aplicados al tracking de objetos, basándonos en estudios ya realizados y en la aplicación del robot se utilizó el Algoritmo de Umbralización, apoyado en el modelo HSV para definir el color del objeto y en las operaciones morfológicas para eliminar el ruido. Con el uso del Algoritmo mencionado se consiguieron resultados fiables y concretos del color y forma de la pelota; además de una respuesta rápida del procesamiento de la escena.
3. En la actualidad se dispone de una gran variedad de software Open Source; para la programación del Sistema de Visión Artificial del robot se han utilizado herramientas de programación de código que ofrece el lenguaje C++ con lo que obtuvo un mayor control y manejo de recursos en el procesamiento y análisis de la imagen. La librería OpenCV facilita al sistema de VA la interpretación de la información y el reconocimiento de patrones de la escena.
4. El software SolidWorks cuenta con herramientas de diseño en 3D permite el modelamiento de la estructura externa del robot de una forma versátil. El diseño de la estructura se dividió en piezas para su posterior ensamblaje facilitando el acople de motores y disposición de sensores y demás elementos en la estructura tanto interna como externa del robot. El diseño realizado se procesó en el software de la impresora 3D, para finalmente imprimir las piezas.
5. El uso de algoritmos y elementos que proporciona la Visión Artificial ha permitido al robot deducir de forma automática la estructura y propiedades tridimensionales del objeto a seguir en tiempo real. Las características tridimensionales procesadas incluyen: las propiedades geométricas que son la forma, tamaño y localización de la pelota; así como también las propiedades físicas que son el color, iluminación y textura de la pelota.
6. El uso de diversas librerías previamente instaladas en las placa Raspberry Pi han permitido el control de los actuadores y demás elementos que conforman el robot-; con la ayuda de bibliotecas dedicadas al procesamiento de archivo de audio se adicionó al sistema de VA del

robot mensajes pregrabados que serán reproducidos en diferentes momentos del juego, para así evitar que el juego se torne aburrido y mantener la atención del niño durante el juego.

7. El niño interactúa con el robot a través de un juego de pelota, en la que tanto el niño como el robot buscarán la pelota para patearla o empujarla. El juego tiene como finalidad motivar al niño en su desarrollo intelectual y psicomotriz; pues como se conoce en la etapa de la infancia es donde el ser humano adquiere habilidades que le permitirán un desenvolverse mejor en la sociedad.

RECOMENDACIONES

1. Al implementar un sistema de VA se debe tener en cuenta la selección adecuada de cada uno de los elementos que intervienen en el proceso para así evitar errores en el procesamiento de imágenes y conseguir un control óptimo de los actuadores que intervienen en sistema. La velocidad de respuesta el sistema de VA dependerá en gran medida del módulo de proceso elegido.
2. En la implementación del Algoritmo de Umbralización se debe prestar atención a las operaciones morfológicas que se les aplicarán a la imagen para eliminar el ruido, pues cada operación morfológica se encarga de resaltar características en concreto de un objeto. La selección de las operaciones morfológicas a aplicar a la imagen dependerá de las características que se deseen procesar por el sistema de VA.
3. Para el uso de las herramientas y bibliotecas que brinda el software Open Source, se recomienda tener un conocimiento de la sintaxis o reglas de escritura dadas por el Lenguaje C++; pues así evitar errores de compilación ocasionados por una sintaxis incorrecta en el código de programación.
4. Al momento de modelar la estructura de un robot en un software diseño 3D se recomienda un diseño versátil para así tener facilidad al momento de acoplar actuadores como: motores, sensores, circuitos, entre otros; además se debe tener cuidado en cuanto al tamaño y volumen de cada pieza modelada para no añadir peso innecesario a la estructura del robot.
5. Es conveniente realizar siempre una calibración de la cámara antes de ejecutar el tracking de un objeto en espacios abiertos, ya que la variación de la iluminación suele afectar la veracidad de los resultados que obtiene una misma cámara probada en laboratorios que en campo abierto.
6. La secuencia de audio para reproducir en la placa Raspberry Pi B+ es recomendable que posea una calidad de audio media ya que de ser alta su reproducción será entrecortada y notablemente débil.
7. Previo al inicio del juego es recomendable explicarle al niño sobre el desenvolvimiento del robot, así como también las secuencia del juego; para así conseguir que el niño se vaya familiarizando con el robot y pueda interactuar con el sin mayores problemas.

BIBLIOGRAFÍA

ASHWIN PAJANKAR. Raspberry Pi Computer Vision Programming [web]. Birmingham-Reino Unido: Packt Publishing, 2015, pp. 1-68.

[Consulta: 22 de septiembre 2015].

Disponible en: <http://PacktLib.PacktPub.com>

BRAHMBHATT Samarth. Practical OpenCV [web]. New York-Estados Unidos: Clearance Center, 2013, pp. 7-95.

[Consulta: 22 de septiembre 2015].

Disponible en: <http://www.springer.com/la/>

ETOLOCKA Hernesto. *Módulo de 4 relés para Arduino* [blog]. Lima-Perú: Profetoloka, 9 de mayo del 2015.

[Consulta: 15 de enero 2016].

Disponible en: <http://www.profetoloka.com.ar/>

GÓMEZ GONZÁLES, Sergio. El Gran Libro de SolidWorks. 2ª. ed. Barcelona-España: Marcombo, 2014, pp. 10-69.

GRIMMETT, Richard Raspberry Pi Robotic Projects [web]. Birmingham-Reino Unido: Packt Publishing, 2014, pp. 83-119.

[Consulta: 20 de septiembre 2015].

Disponible en: <http://PacktLib.PacktPub.com>

ARGENTINA, GRUPO DE INTELIGENCIA ARTIFICIAL Y ROBÓTICA., Algoritmo de seguimiento de objetos basado en visión asistida por computador en tiempo real utilizando CAMShift e histogramas ponderados., Buenos Aires - Argentina, Universidad Tecnológica Regional Buenos Aires. 2014, pp. 1-6.

LA RIOJA, GRUPO DE INVESTIGACIÓN EDMANS. Técnicas y algoritmos básicos de visión artificial. La Rioja- España, Universidad de La Rioja, Servicio de Publicaciones., 2006. pp. 1-91.

ARGENTINA, GRUPO PROYECTO BIOPUS. Software Open Source y el desarrollo de Nuevas Interfaces en el Arte Interactivo [en línea]. Cordova-Argentina. Jornadas de Artes y Medios Digitales, 2006, pp. 1-5.

[Consulta: 15 de octubre 2015].

Disponible en: <http://www.proyectobiopus.com>

HIEMENZ, Joe. La impresión 3D con FDM. Stratasys, 2011, pp. 1-5.

[Consulta: 05 de enero 2016].

Disponible en: www.stratasys.com

ILBAY LLANGARI, Luis Guido. Evaluación de algoritmos de tracking 3d para la simulación de un brazo robótico, mediante kinect [En línea], (TESIS). Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Escuela de Ingeniería Electrónica en Control y Redes Industriales Riobamba- Ecuador. 2015. pp. 24-127.

[Consulta: 05 de enero 2016].

Disponible en: <http://dspace.esPOCH.edu.ec/bitstream/123456789/4402/1/108T0124.docx.pdf>

MACHUCA MENDOZA, Cristian Rodrigo; & LÓPEZ AYALA, Luis Fernando. Locomoción de un robot cuadrúpedo basada en redes neuronales artificiales (TESIS). Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Escuela de Ingeniería Electrónica en Control y Redes Industriales Riobamba- Ecuador. 2015. pp. 20-106.

[Consulta: 05 de enero 2016].

Disponible en: <http://dspace.esPOCH.edu.ec/bitstream/123456789/3803/1/108T0113.pdf>

MAFLA MEDINA, Gabriela Mercedes; & ORTIZ ESPINOSA, Aldiniver José. Estudio de los algoritmos de reconocimiento de patrones para la automatización de un semáforo inteligente mediante fpgas (TESIS). Escuela Superior Politécnica de Chimborazo, Facultad de Informática

y Electrónica, Escuela de Ingeniería Electrónica en Control y Redes Industriales Riobamba-Ecuador. 2014., pp. 50-60.

[Consulta: 05 de enero 2016].

Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/3567/1/108T0099.pdf>

MCMANUS Sean; & COOK Mike. Raspberry Pi For Dummies [allitebooks]. Nueva Jersey-Estados Unidos: Packt Publishing, 2013, pp. 107-337.

[Consulta: 15 de octubre 2015].

Disponible en: www.allitebooks.com

MUÑOZ, Carlos. Cómo elaborar y asesorar una investigación de tesis. 2a ed. Juárez-México: PEARSON EDUCACIÓN, 2010, pp. 122-297.

[Consulta: 02 de febrero 2015].

Disponible en: <http://esPOCH.librositio.net/libroImprime.php?libroId=9>.

POLOLU ROBOTICS & ELECTRONICS. A-Star 32U4 Micro. [Pdf].

[Consulta: 20 de noviembre 2015].

Disponible en: <https://www.pololu.com/product/3101/resources>.

RASPBERRY PI. [Pdf].

[Consulta: 20 de agosto 2015].

Disponible en: <https://www.raspberrypi.org/>

SALCEDO TOVAR, Marco Luis. Minicomputador educacional de bajo costo Raspberry Pi. *Revista Ethos Venezolana* [web]. Vol. 7 No. 1, Caracas- Venezuela, 2015, pp. 28 - 45

ISSN 1856-9862 ~ Depósito legal pp 200902ZU3258

[Consulta: 20 de noviembre 2015].

Disponible en: http://www.uniojeda.edu.ve/files/upload/ethos/volumen-7-n1/Revista-Ethos-Venezolana_Vol.-7-No.-1.pdf

SERVOCITY. HS-311 STANDARD. [Pdf].

[Consulta: 25 de septiembre 2015].

Disponible en: https://www.servocity.com/html/hs-311_standard.html#.ViQOwyu-njA.

DEPARTAMENTO DE ELECTRÓNICA. *Sistema de locomoción de robots móviles* [web]. Sevilla-España. Automatización y Robótica Industrial, 2010 pp. 1-20.

VÁZQUEZ Martha. *Niños tímidos, seis consejos para que superen su timidez* [blog]. Vasco-España, 13 de septiembre de 2013.

[Consulta: 15 de agosto 2015].

Disponible en: <http://www.consumer.es/>

VIGUERAS, Ángel Soriano. *Diseño y Programación de Algoritmos para Robot Móviles. Aplicación al robot LEGO-NXT (TESIS).* Universidad Politécnica de Valencia, Ingeniería Informática. Valencia-España, 2015, pp. 7-92.

[Consulta: 15 de enero 2016].

Disponible en: <https://riunet.upv.es/bitstream/handle/10251/11167/Memoria.pdf>

VILLEGAS, Jaime. *Que es un detector de movimiento* [blog]. Medellín-Colombia: TecnoSeguro, 02 de febrero del 2012.

[Consulta: 20 de enero 2016].

Disponible en: <http://www.tecnoseguro.com/faqs/alarma/que-es-un-detector-de-movimiento-pasivo-o-pir.html>

YAMANOOR Sai; & YAMANOOR Srihari. *Raspberry Pi Mechatronics Projects HOTSH T* [web]. Birmingham: Packt Publishing, 2015, pp. 10-47.

[Consulta: 15 de noviembre 2015].

Disponible en: <http://PacktLib.PacktPub.com>



MODEL B+

Product Name	Raspberry Pi Model B+
Product Description	The Raspberry Pi Model B+ incorporates a number of enhancements and new features. Improved power consumption, increased connectivity and greater IO are among the improvements to this powerful, small and lightweight ARM based computer.
RS Part Number	811-1284

Specifications

Chip	Broadcom BCM2835 SoC
Core architecture	ARM11
CPU	700 MHz Low Power ARM1176JZF5 Applications Processor
GPU	Dual Core VideoCore IVB Multimedia Co-Processor Provides Open GL ES 2.0, hardware-accelerated OpenVG, and 1080p30 H.264 high-profile decode Capable of 1Gpixels/s, 1.5Gtexels/s or 24GFLOPs with texture filtering and DMA infrastructure
Memory	512MB SDRAM
Operating System	Boots from Micro SD card, running a version of the Linux operating system
Dimensions	85 x 56 x 17mm
Power	Micro USB socket 5V, 2A

Connectors:

Ethernet	10/100 BaseT Ethernet socket
Video Output	HDMI (rev 1.3 & 1.4) Composite RCA (PAL and NTSC)
Audio Output	3.5mm Jack, HDMI
USB	4 x USB 2.0 Connector
GPIO Connector	40-pin 2.54 mm (100 mil) expansion header: 2x20 strip Providing 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines
Camera Connector	15-pin MIPI Camera Serial Interface (CSI-2)
JTAG	Not populated
Display Connector	Display Serial Interface (DSI) 15 way flat flex cable connector with two data lanes and a clock lane
Memory Card Slot	SDIO

Accessories



▲ Camera Module
776-7731



▲ International power supply
766-3311



▲ 8GB SD card pre-programmed with NOOBS - **796-1603**



▲ HDMI - VGA
778-1882



▲ WiFi dongle
760-3621



▲ 10400mAh Li-Ion battery pack
775-7517



▲ Raspberry Pi user guide
794-7183

Applications

- Mobile Phones
- Digital Still Cameras
- PC Multimedia

Product Features

- 1.4 μm x 1.4 μm pixel with OmniBSI technology for high performance (high sensitivity, low crosstalk, low noise)
- optical size of 1/4"
- automatic image control functions:
 - automatic exposure control (AEC)
 - automatic white balance (AWB)
 - automatic band filter (ABF)
 - automatic 50/60 Hz luminance detection
 - automatic black level calibration (ABLC)
- programmable controls for frame rate, AEC/AGC 16-zone size/position/weight control, mirror and flip, cropping, windowing, and panning
- image quality controls: lens correction, defective pixel canceling
- support for output formats: 8-/10-bit raw RGB data
- support for video or snapshot operations
- support for LED and flash strobe mode
- support for internal and external frame synchronization for frame exposure mode
- support for 2x2 binning for better SNR in low light conditions
- post binning resampling filter to minimize spatial/aliasing artifacts on 2x2 binned image
- support for horizontal and vertical sub-sampling
- standard serial SCCB interface
- digital video port (DVP) parallel output interface
- MIPI interface (two lanes)
- 32 bytes of embedded one-time programmable (OTP) memory
- on-chip phase lock loop (PLL)
- embedded 1.5V regulator for core power
- programmable I/O drive capability, I/O tri-state configurability
- support for black sun cancellation

OV5647

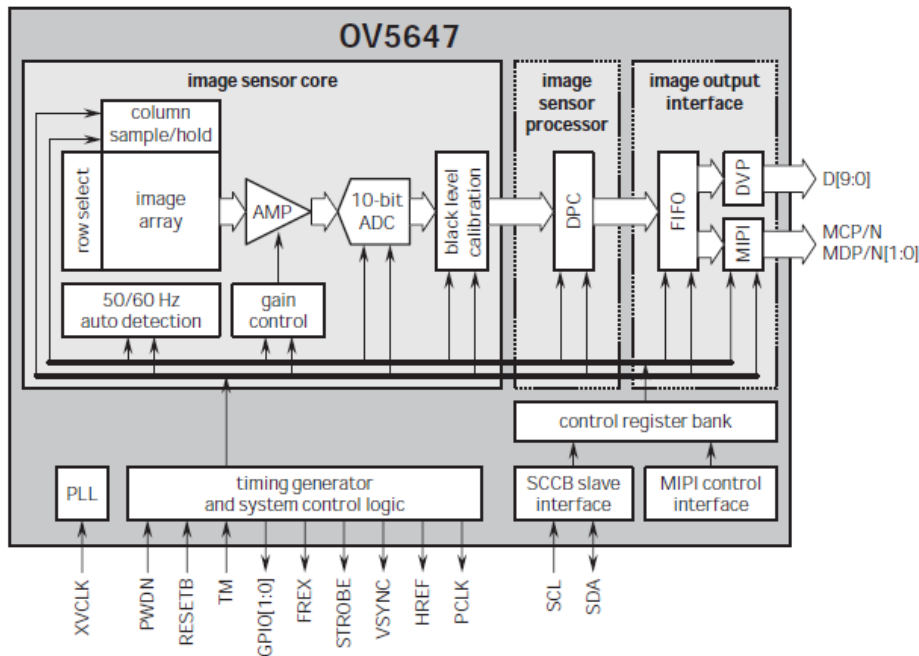
Ordering Information

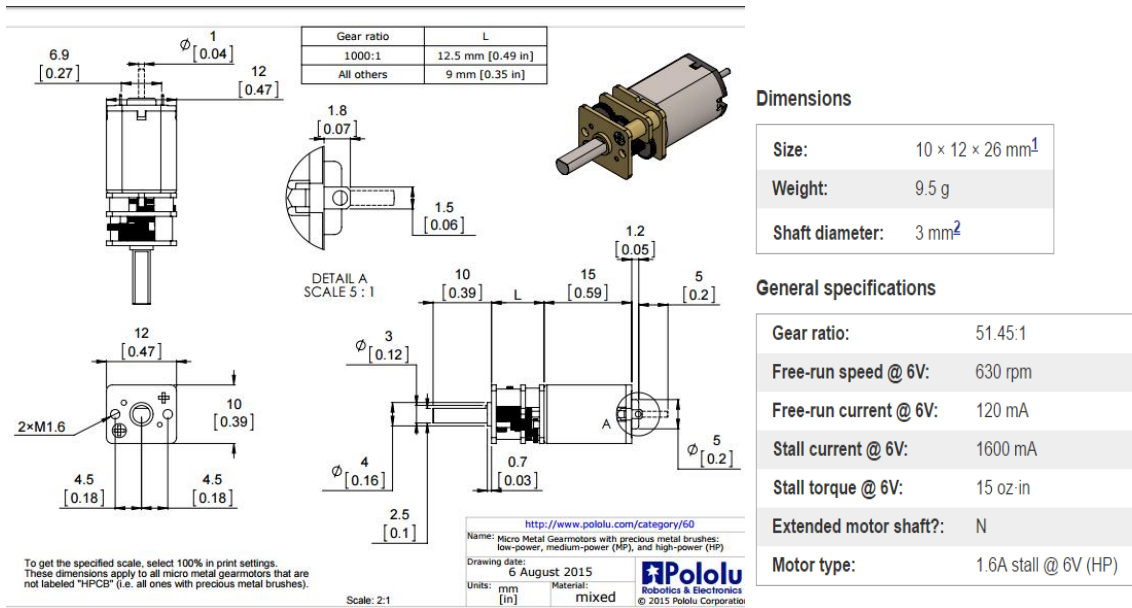
- OV05647-G04A (color, chip probing, 200 μm backgrinding, reconstructed wafer)

Product Specifications

- active array size: 2592 x 1944
- power supply:
 - core: 1.5 V \pm 5% (internal regulator)
 - analog: 2.6 - 3.0 V
 - I/O: 1.7 - 3.0 V
- temperature range:
 - operating: -30°C to 70°C
 - stable image: 0°C to 50°C
- output formats: 8-/10-bit raw RGB data
- lens size: 1/4"
- lens chief ray angle: 24°
- input clock frequency: 6 - 27 MHz
- S/N ratio: 36 dB
- dynamic range: 68 dB
- sensitivity: 680 mV/(lux-sec)
- maximum image transfer rate:
 - QSXGA (2592x1944): 15 fps
 - 1080p: 30 fps
 - 960p: 45 fps
 - 720p: 60 fps
 - VGA (640x480): 90 fps
 - QVGA (320x240): 120 fps
- shutter: rolling shutter/global shutter
- maximum exposure interval: 1968 x t_{ACW}
- pixel size: 1.4 μm x 1.4 μm
- well capacity: 4.3 Ke-
- dark current: 16 mV/sec @ 60°C
- fixed pattern noise: < 1% of $V_{\text{PEAK-TO-PEAK}}$
- image area: 3673.6 μm x 2738.4 μm
- die dimensions: 5520 μm x 4700 μm

Functional Block Diagram





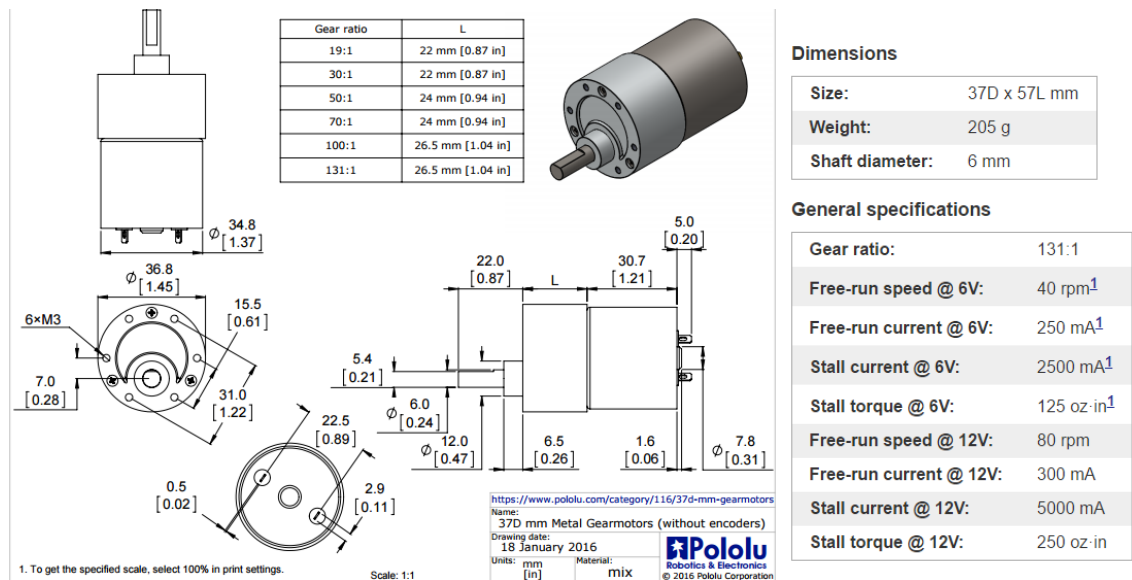
Dimensions

Size:	10 × 12 × 26 mm ¹
Weight:	9.5 g
Shaft diameter:	3 mm ²

General specifications

Gear ratio:	51.45:1
Free-run speed @ 6V:	630 rpm
Free-run current @ 6V:	120 mA
Stall current @ 6V:	1600 mA
Stall torque @ 6V:	15 oz-in
Extended motor shaft?:	N
Motor type:	1.6A stall @ 6V (HP)

ANEXO C. DATASHEET MICRO MOTOR REDUCTOR 50:1



Dimensions

Size:	37D x 57L mm
Weight:	205 g
Shaft diameter:	6 mm

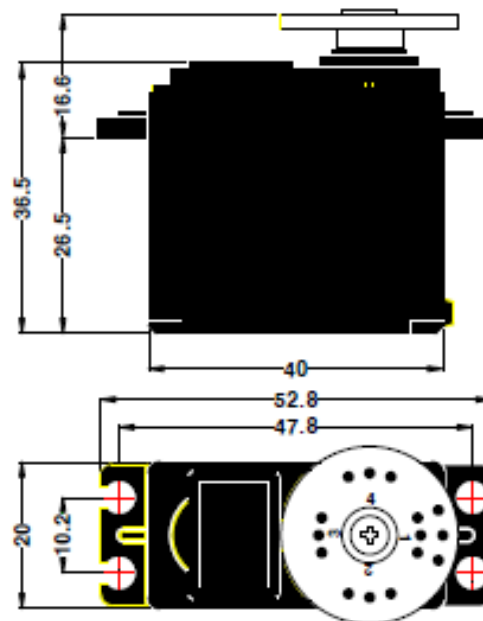
General specifications

Gear ratio:	131:1
Free-run speed @ 6V:	40 rpm ¹
Free-run current @ 6V:	250 mA ¹
Stall current @ 6V:	2500 mA ¹
Stall torque @ 6V:	125 oz-in ¹
Free-run speed @ 12V:	80 rpm
Free-run current @ 12V:	300 mA
Stall current @ 12V:	5000 mA
Stall torque @ 12V:	250 oz-in

ANEXO D. DATASHEET MOTOR REDUCTOR 131:1

ANNOUNCED SPECIFICATION OF HS-311 STANDARD SERVO

1. TECHNICAL VALUE		
CONTROL SYSTEM	: +PULSE WIDTH CONTROL 1500usec NEUTRAL	
OPERATING VOLTAGE RANGE	: 4.8V TO 6.0V	
TEST VOLTAGE	: AT 4.8V	AT 6.0V
OPERATING SPEED	: 0.19sec/60° AT NO LOAD	0.15sec/60° AT NO LOAD
STALL TORQUE	: 3.0kg.cm(42oz.in)	3.5kg.cm(48.60oz.in)
IDLE CURRENT	: 7.4mA AT STOPPED	7.7mA AT STOPPED
RUNNING CURRENT	: 160mA/60° AT NO LOAD	180mA/60° AT NO LOAD
STALL CURRENT	: 700mA	800mA
DEAD BAND WIDTH	: 5usec	5usec
OPERATING TRAVEL	: 40°/ONE SIDE PULSE TRAVELING 400usec	
DIRECTION	: CLOCK WISE/PULSE TRAVELING 1500 TO 1900usec	
MOTOR TYPE	: CORED METAL BRUSH	
POTENTIOMETER TYPE	: 4 SLIDER/DIRECT DRIVE	
AMPLIFIER TYPE	: ANALOG CONTROLLER & TRANSISTOR DRIVER	
DIMENSIONS	: 40x20x36.5mm(1.57x0.78x1.43in)	
WEIGHT	: 43g(1.51oz)	
BALL BEARING	: TOP/RESIN BUSHING	
GEAR MATERIAL	: RESIN	
HORN GEAR SPLINE	: 24 SEGMENTS/°5.76	
SPLINED HORNS	: SUPER/R-XA	
CONNECTOR WIRE LENGTH	: 300mm(11.81in)	
CONNECTOR WIRE STRAND COUNTER	: 40EA	
CONNECTOR WIRE GAUGE		



ANEXO E. DATASHEET SERVOMOTOR HS-311



Tech Support: services@electfreaks.com

Ultrasonic Ranging Module HC - SR04

Product features:

Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:

- (1) Using IO trigger for at least 10us high level signal,
- (2) The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.
- (3) IF the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to returning.

Test distance = (high level time×velocity of sound (340M/S) / 2,

Wire connecting direct as following:

- 5V Supply
- Trigger Pulse Input
- Echo Pulse Output
- 0V Ground

Electric Parameter

Working Voltage	DC 5 V
Working Current	15mA
Working Frequency	40Hz
Max Range	4m
Min Range	2cm
Measuring Angle	15 degree
Trigger Input Signal	10us TTL pulse
Echo Output Signal	Input TTL lever signal and the range in proportion
Dimension	45*20*15mm

ANEXO F. DATASHEET SENSOR ULTRASÓNICO HC-SR04

PIR Sensor (#555-28027)

General Description

The PIR (Passive Infra-Red) Sensor is a pyroelectric device that detects motion by measuring changes in the infrared levels emitted by surrounding objects. This motion can be detected by checking for a high signal on a single I/O pin.

Features

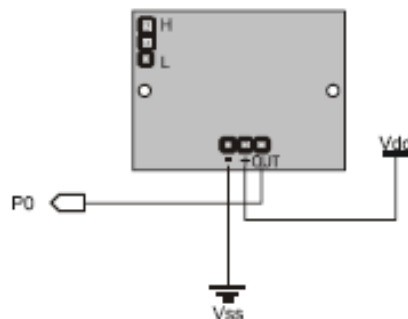
- Single bit output
- Small size makes it easy to conceal
- Compatible with all Parallax microcontrollers
- 3.3V & 5V operation with <100uA current draw

Application Ideas

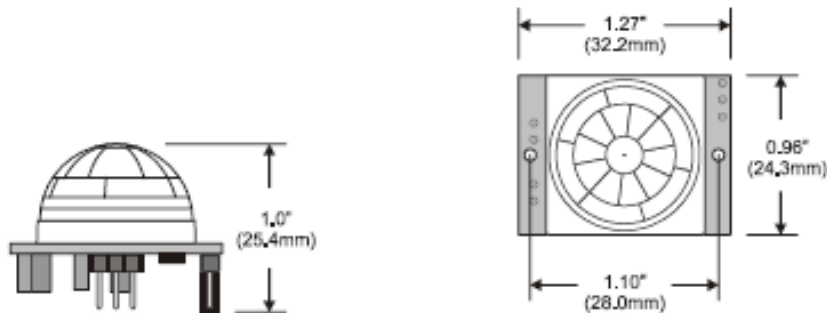
- Alarm Systems
- Halloween Props

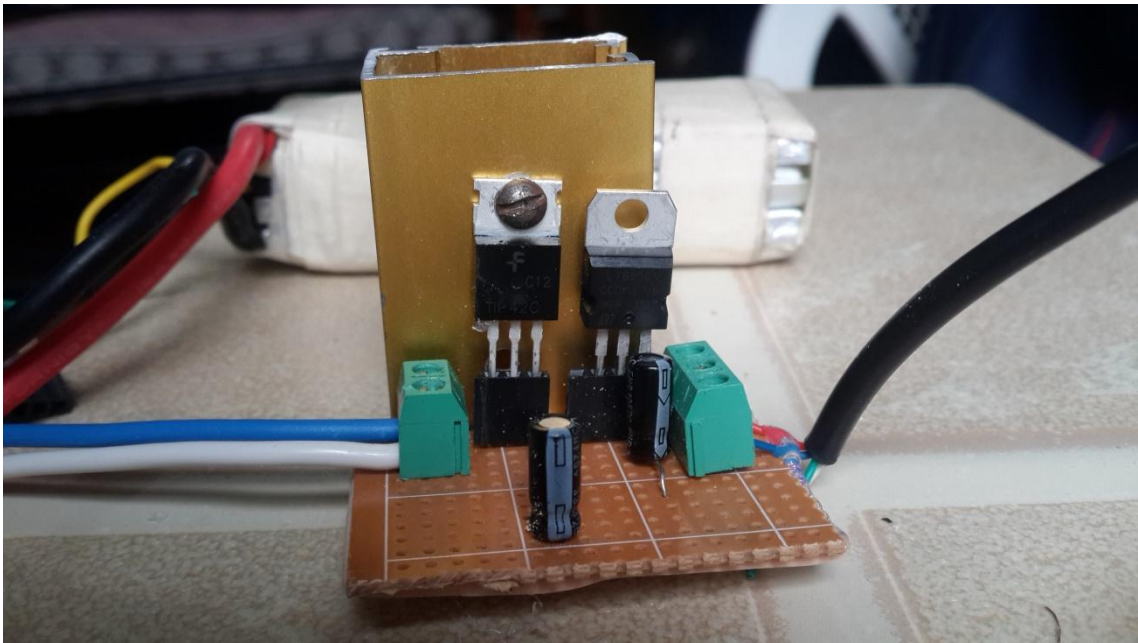
Quick Start Circuit

Note: The sensor is active high when the jumper (shown in the upper left) is in either position.

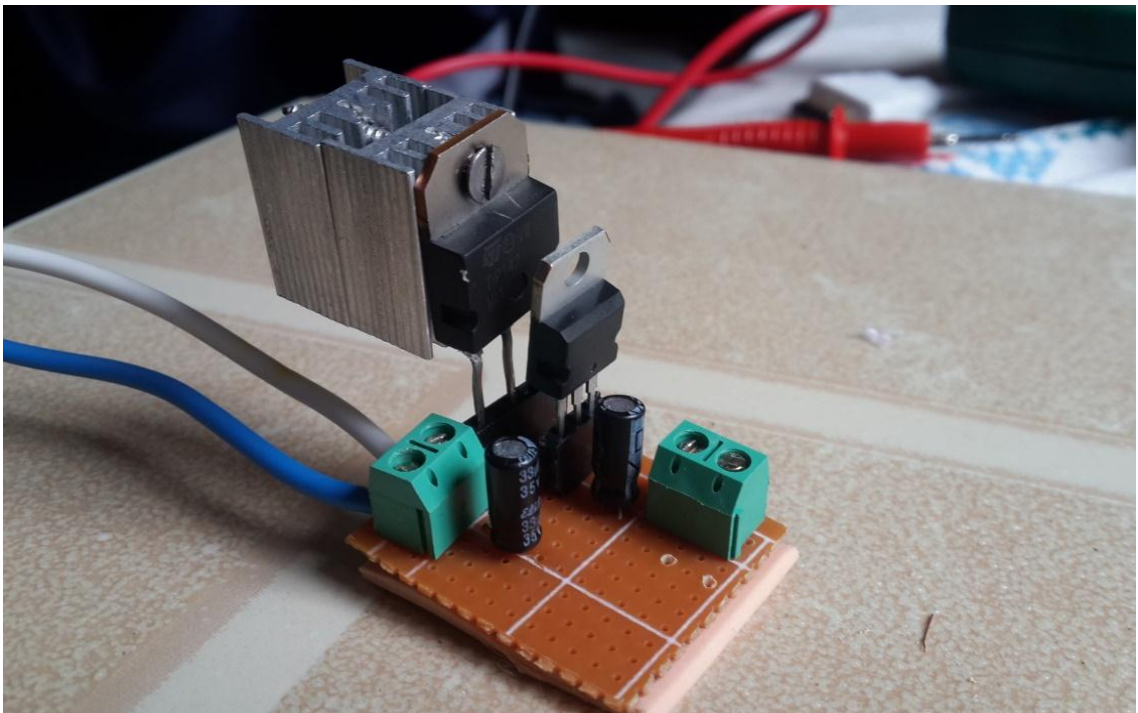


Module Dimensions

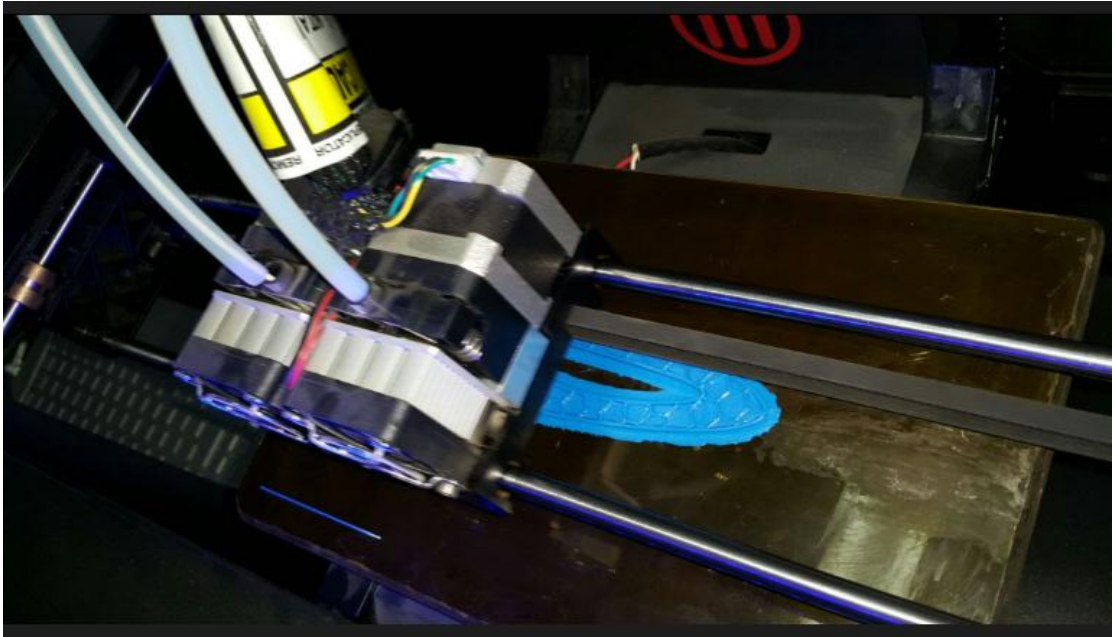




ANEXO H. CIRCUITO IMPLEMENTADO DEL DIAGRAMA DE CONEXIÓN DEL REGULADOR DE VOLTAJE DE 5V DC.



ANEXO I. CIRCUITO IMPLEMENTADO DEL DIAGRAMA DE CONEXIÓN DEL REGULADOR DE VOLTAJE DE 5V DC.



ANEXO J. IMPRESIÓN DE UNA DE LAS PIEZAS DEL ROBOT

Parámetros para reconocer patrones de figuras	NÚMERO DE VÉRTICE	ÁNGULOS	DIÁMETRO
Figura			
Triángulo	3	Varía de acuerdo al tipo de triángulo	---
Rectángulo	4	90°	---
Hexágono	6	120°	---
Círculo	Más de 6		El mismo en todas las direcciones

ANEXO K. PARÁMETROS PARA RECONOCER PATRONES DE FIGURAS