

DESARROLLO DE UN OBJETO VIRTUAL DE APRENDIZAJE PARA LOS
ROBOTS BIOLOID CON FINALIDAD DE APLICACIONES EN ROBÓTICA.

INGRID NATALIA RODRIGUEZ OVALLE
BRIAN ALEXANDER CHACÓN HERNÁNDEZ

UNIVERSIDAD PILOTO DE COLOMBIA
PROGRAMA DE INGENIERÍA MECATRÓNICA
BOGOTA D.C., COLOMBIA
2017

DESARROLLO DE UN OBJETO VIRTUAL DE APRENDIZAJE PARA LOS
ROBOTS BIOLOID CON FINALIDAD DE APLICACIONES EN ROBÓTICA.

INGRID NATALIA RODRIGUEZ OVALLE
BRIAN ALEXANDER CHACÓN HERNÁNDEZ

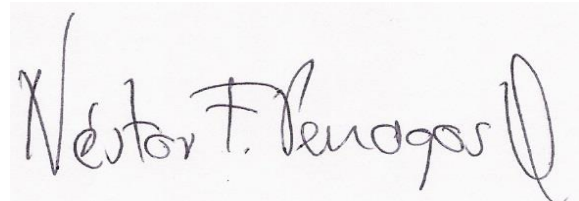
PROYECTO DE GRADO

DIRECTOR: NESTOR FERNANDO PENAGOS QUINTERO

UNIVERSIDAD PILOTO DE COLOMBIA
PROGRAMA DE INGENIERÍA MECATRÓNICA
BOGOTA D.C., COLOMBIA
2017

Nota de aceptación

El trabajo de grado, titulado “DESARROLLO DE UN OBJETO VIRTUAL DE APRENDIZAJE PARA LOS ROBOTS BIOLOID CON FINALIDAD DE APLICACIONES EN ROBÓTICA.” elaborado y presentado por los estudiantes Ingrid Natalia Rodriguez Ovalle y Brian Alexander Chacon Hernandez, como requisito parcial para optar al título de Ingeniero/a Mecatrónico/a, cumple el objetivo general y los específicos.

A handwritten signature in black ink on a light-colored background. The signature reads "Néstor F. Benegas" followed by a stylized flourish.

Firma del tutor

Bogotá, 13 de Febrero de 2017

CONTENIDO

1.INTRODUCCIÓN	11
1.1 RESUMEN	11
1.2 ABSTRACT	12
1.1 PLANTEAMIENTO DEL PROBLEMA	12
1.2.1 Antecedentes del problema	12
1.2.2 Descripción del problema	13
1.3 JUSTIFICACIÓN	13
1.4 OBJETIVOS	13
1.4.1 Objetivo general.	13
1.4.1 Objetivos específicos.	13
1.5 DELIMITACIÓN DEL PROYECTO	14
1.5.1 Alcances y limitaciones.	14
1.5.2 Línea de investigación del programa	14
1.6 MARCO REFERENCIAL	14
1.6.1 Marco teórico.	14
1.6.2 Estado del arte.	22
1.7 METODOLOGÍA	27
1.8 RECURSOS.	28
2. ANÁLISIS CINEMÁTICO DIRECTO, INVERSO Y DINÁMICO INVERSO ORIENTADO AL APRENDIZAJE PARA UNA EXTREMIDAD DE 3GDL	29
2.1 CINEMÁTICA DIRECTA	30
2.1.1 Cinemática directa por medio de las matrices de Transformación Homogénea para las extremidades de las piernas (12GDL)	33
2.1.2 Cinemática directa por medio de las matrices de Transformación Homogénea para un brazo (3GDL)	38
2.2 CINEMÁTICA INVERSA	41
2.3 CINÉTICA DEL BRAZO DE 3GDL	43
2.3.1 Energía cinética y potencial de eslabones.	46
2.3.2 Ecuación Lagrange.	49
3. PROCESAMIENTO DE IMAGEN PARA LA DETECCIÓN E IDENTIFICACIÓN DE OBJETOS DE INTERÉS	52

3.1 Fases para el procesamiento de imagen.....	58
3.1.1 Captura de imagen.....	58
3.1.2 Filtros de Color.....	59
3.1.3 Segmentación de color.....	62
3.1.4 Extracción de características.....	63
3.1.5 Caracterización de cancha.....	66
3.1.8 Caracterización de elementos detectados	66
4.COMUNICACIÓN	70
4.1 SELECCIÓN DE LA TECNOLOGÍA DE COMUNICACIÓN.....	70
4.2 EXTRACCIÓN DE DATOS (OSCILOSCOPIO Y VELOCIDAD)	72
4.3 PROTOCOLO RECEPCIÓN Y ENVÍO DE DATOS (DEMUX)	79
4.3.1 Comunicación con un módulo ZIG-100	80
4.4 DISEÑO Y MONTAJE DE CIRCUITO DE COMUNICACIÓN	83
5. ENSAMBLE Y PRUEBAS DE FUNCIONAMIENTO.....	85
5.1 CÁLCULO DE TRAYECTORIAS	85
5.2 DEFINICIÓN PARÁMETROS PARA DELANTERO Y ARQUERO ANTES DE REALIZAR ACCIONES.....	91
5.3 COMUNICACIÓN A PARTIR DEL PROCESAMIENTO DE IMAGEN	96
6. OBJETO VIRTUAL DE APRENDIZAJE (OVA).....	100
6.2 CONTENIDOS DEL OVA.....	101
6.2.3 Subcontenidos de secciones	103
7. CONCLUSIONES	107
8. RECOMENDACIONES Y TRABAJOS FUTUROS.....	108
9. ANEXOS	109
10. BIBLIOGRAFÍA	124

LISTA DE TABLAS

Tabla 1. Tabla de parámetros Denavit-Hartenberg para las extremidades inferiores	35
Tabla 2. Tabla de parámetros Denavit-Hartenberg para una extremidad superior.	39
Tabla 3. Notación usada en el cálculo de la dinámica.	46
Tabla 4 Tabla de cálculo de energías para eslabones.....	47
Tabla 5. Comparativa: costo-beneficio-rendimiento.....	55
Tabla 6. Componentes entorno controlado	57
Tabla 7. Comparativa de tecnologías de comunicación compatibles con el CM-530	71
Tabla 8. Extracción de datos para tecla U del control RC-100	75
Tabla 9. Final de extracción de datos para tecla U del control RC-100	76
Tabla 10. Comparativa Blog, página web y Aplicación.	100

LISTA DE FIGURAS

Ilustración 1. Áreas de la robótica.....	15
Ilustración 2. Robots humanoides, PACO-PLUS y Don Cuco el Guapo.....	16
Ilustración 3. Robot bípedo desarrollado en el Massachusetts Institute of Technology (EE.UU)	17
Ilustración 4. División de mecánica.....	18
Ilustración 5. Detección visual.....	21
Ilustración 6. Esquema general del robot y su representación en cadenas cinemáticas.....	30
Ilustración 7 Esquema cinemática directa e inversa	31
Ilustración 8. Render de extremidades inferiores Robot Bioloid.....	33
Ilustración 9. Esquema de sistema de ejes según método Denavit-Hartenberg para las extremidades inferiores.....	34
Ilustración 10. Renderizado de una extremidad superior robot Bioloid.....	38
Ilustración 11. Esquema de sistema de ejes según método Denavit-Hartenberg para una extremidad superior	39
Ilustración 12 Esquema de una extremidad superior para análisis cinemática inversa	41
Ilustración 13. Esquema posibilidades de posición eslabones brazo para la cinemática inversa	43
Ilustración 14. Dinámica inversa y directa.....	44
Ilustración 15. Modelado de brazo robot con ángulos correspondientes en cada eje para desarrollo de modelo dinámico.	45
Ilustración 16. Diagrama de Cuerpo libre para dinámica de un brazo de 3GDL	45
Ilustración 17. Robots de la Modalidad <i>Humanoid</i> de <i>Robocup</i>	52
Ilustración 18. Robots de la modalidad <i>Small Size</i> de <i>Robocup</i>	53
Ilustración 19. Entorno modalidad <i>Small Size</i> de <i>Robocup</i>	53
Ilustración 20. Medidas estandarizadas marcas de color modalidad <i>Small Size</i> de <i>Robocup</i>	54
Ilustración 21. Entorno Python 2.7.12	56
Ilustración 22. Entorno controlado real	56
Ilustración 23. Campo de visión cámara	57
Ilustración 24 Rutina de procesamiento de imagen.	58
Ilustración 25. Esquema de captura de imágenes	58
Ilustración 26. Esquema de filtros de color	60
Ilustración 27. Ejemplo de Filtro Gaussiano para una imagen aleatoria	60
Ilustración 28. Imagen de HSV aplicado a figuras triangulares azules	61
Ilustración 29. Escala de color HSV.....	61
Ilustración 30. Esquema de segmentación de color aplicado al proyecto.....	62
Ilustración 31. Ejemplo de prueba para las operaciones morfológicas de apertura y cierre.....	63
Ilustración 32. Secuencia extracción de características.....	63

Ilustración 33. Vector en la marca de cada robot.....	64
Ilustración 34. Ángulos formados entre los vectores de las marcas de los robots y el eje de coordenadas incorrecto.....	64
Ilustración 35. Ángulos formados entre los vectores de las marcas de los robots y el eje de coordenadas correcto.....	65
Ilustración 36. Distinción pelota naranja en campo.....	65
Ilustración 37. Caracterización de arquero, delantero y pelota	67
Ilustración 38. Marcas de robot y pelota naranja	67
Ilustración 39. Áreas de juego arquero y delantero.....	68
Ilustración 40. Esquema de objetos con su respectivo nombre en la cancha	68
Ilustración 41. Fotografía de objetos con su respectivo nombre en el campo de juego	69
Ilustración 42. Distribución de pines puerto de comunicaciones CM-530.....	72
Ilustración 43. Conexión del módulo Zig-110A con el controlador CM-530	72
Ilustración 44. Conexión del módulo Zig-100 con el control RC-100.	73
Ilustración 45. Algoritmo de conversión de datos ASCII a hexadecimal	74
Ilustración 46. Inicio de extracción de datos para tecla U del control RC-100	75
Ilustración 47. Finalización de extracción de datos para tecla U del control RC-100	76
Ilustración 48. Diagrama algoritmo de comunicación del control RC-100.....	77
Ilustración 49. Medición de la señal de salida del módulo receptor Zig-100A.....	77
Ilustración 50. Algoritmo de comunicación del control RC-100 con periodo	78
Ilustración 51. Paquetes de envío de datos al robot Bioloid	78
Ilustración 52. Esquema algoritmo de comunicación Arduino.....	80
Ilustración 53. Diagrama usando Arduino como DEMUX	81
Ilustración 54. Algoritmo usando Arduino como <i>demux</i>	82
Ilustración 55. Diseño de protocolo de comunicación de entrada a Arduino.....	83
Ilustración 56. Circuito de comunicaciones con módulos Zig-100.....	84
Ilustración 57. Montaje del circuito Shield con módulos Zig-100 en Arduino	84
Ilustración 58. Sistema 1, ejes coordenados iniciales.....	85
Ilustración 59. Sistema 2, ejes coordenados comunes	86
Ilustración 60. Marca de color en Robot	86
Ilustración 61. Representación ángulo robot delantero.....	87
Ilustración 62. Representación ángulo pelota	87
Ilustración 63. Representación ángulo Pelota delantero.....	88
Ilustración 64. Representación rango de error giro robot.....	88
Ilustración 65. Representación ángulo pelota y punto de gol.....	89
Ilustración 66. Representación ángulo total entre delantero, pelota y punto de gol	89
Ilustración 67. Trayectoria de gol, espacio brazo izquierdo y arco mayor.....	90
Ilustración 68. Trayectoria de gol, espacio brazo derecho y arco mayor	90
Ilustración 69. Posibles posiciones pelota-delantero	91
Ilustración 70. Robot girado derecha	92

Ilustración 71. Robot girado izquierda.....	92
Ilustración 72. Delantero-Pelota-Punto de Gol alineados 180°	93
Ilustración 73. Distancia delantero-pelota 33cm	93
Ilustración 74. Distancia delantero-pelota 30cm	94
Ilustración 75. Distancia delantero-pelota 24cm	94
Ilustración 76. Distancia delantero-pelota 22cm	94
Ilustración 77. Distancia delantero-pelota 19cm	95
Ilustración 78. Distancia de Tape arquero	95
Ilustración 79. Proceso de envío de datos a robot desde algoritmo principal para ejecutar acciones.	97
Ilustración 80. Secuencia del robot delantero enfocando punto de gol y pateando la pelota	98
Ilustración 81. Figura del robot arquero atajando pelota	99
Ilustración 82 Esquema general de página Web.....	102
Ilustración 83.Captura de pantalla de video introductorio a OVA.....	103
Ilustración 84.Captura de pantalla de tutorial Bioid en Solidworks	104
Ilustración 85. Archivo descargable algoritmo HSV detección objetos azules	104
Ilustración 86. Link de direccionamiento externo al recurso web de OpenCV, sección procesamiento de imagen.....	105
Ilustración 87. Esquema Demux Arduino, disponible en la sección comunicación página web.....	105
Ilustración 88.Reto-ejercicio capítulo generalidades.....	106
Ilustración 89.Captura de pantalla, pregunta evaluación capítulo mecánico	106

LISTA DE ANEXOS

ANEXO A. ENCUESTA ESTUDIANTES ACERCA DE KITS DE ROBÓTICA	109
ANEXO B. RESULTADOS PREGUNTA: ¿CONOCE USTED LOS KITS DE ROBÓTICA BIOLOID?.....	110
ANEXO C. RESULTADOS PREGUNTA: ¿SABE EN QUÉ CONSISTE UN KIT DE ROBÓTICA BIOLOID?.....	110
ANEXO D. RESULTADO PREGUNTA: ¿USTED CURSÓ LA ELECTIVA DE ROBÓTICA EN LA UNIVERSIDAD?	111
ANEXO E. RESULTADO PREGUNTA: ¿SABÍA USTED QUE LA UNIVERSIDAD ADQUIRIÓ VARIOS KITS DE ROBÓTICA?	111
ANEXO F. RESULTADO PREGUNTA: ¿ALGUNA VEZ HA EMPLEADO ÉSTOS KITS DE ROBÓTICA?	112
ANEXO G. RESULTADO PREGUNTA: ¿SABE CÓMO SE PONEN EN FUNCIONAMIENTO LOS ROBOTS?.....	112
ANEXO H. COMPATIBILIDAD CM-530.....	113
ANEXO I. DIAGRAMA DE FLUJOS FUNCIONAMIENTO ALGORITMO GENERAL DE ARDUINO	114
ANEXO J. SUBRUTINAS 1, 2 Y 3 DE ARDUINO.....	115
ANEXO K. SUBRUTINA 4 DE ARDUINO.....	116
ANEXO L. ALGORITMO PRINCIPAL DE COMUNICACIONES EN PYTHON	117
ANEXO M. OVA, PÁGINA PRINCIPAL, PÁGINA WEB.....	117
ANEXO N. OVA, INICIO PÁGINA WEB.....	118
ANEXO O. OVA, GENERALIDADES PÁGINA WEB.....	118
ANEXO P. OVA, MECÁNICO PAGINA WEB.....	119
ANEXO Q. OVA, PROCESAMIENTO DE IMAGEN PAGINA WEB	120
ANEXO R. OVA, COMUNICACIÓN PAGINA WEB	121
ANEXO S. OVA, RESULTADOS PAGINA WEB.....	122
ANEXO T. OVA, EXTENSION RESULTADOS PAGINA WEB	123

1.INTRODUCCIÓN

El auge de la robótica en los últimos años ha permitido una gran cantidad de aplicaciones en varios campos que eran inimaginables hace unas décadas, en estos momentos se han desarrollado plataformas o kit que facilitan la construcción de robots, entre ello se encuentra el kit “Bioloid Robotis Premium” que se desea utilizar ya que el mismo permite concentrarse en la programación de dicha plataforma y no en la construcción.

Este kit de robótica permite armar un robot tipo humanoide con el que queda un concepto abierto de aplicaciones entre las cuales se plantea programar dos robots humanoides, uno como arquero y otro como delantero de fútbol.

Recientemente la Universidad Piloto de Colombia adquirió una serie de kits de robótica Bioloid Robots Premium, con el propósito de incentivar la investigación en los estudiantes, y ampliar sus conocimientos, así como poner en práctica los mismos.

Para dar un aspecto más interesante a esta aplicación se implementará un objeto virtual de aprendizaje que permita a estudiantes y docentes aprender de aspectos importantes de robótica como el desarrollo y uso de los kits como jugadores de futbol.

1.1 RESUMEN

Este proyecto consiste en el desarrollo de un aplicativo de fútbol de robots para los Kits de robótica Bioloid y una página web para la divulgación del mismo, el cual se realiza empleando circuitos complementarios, múltiples plataformas y diferentes entornos de programación como Arduino, Python y Roboplus.

El objetivo principal de este proyecto es la realización de un Objeto Virtual de Aprendizaje para la comunidad estudiantil y docente de la Universidad de Mecatrónica, con el fin de difundir el conocimiento obtenido de la realización de este proyecto implementando los robots Bioloid en configuración humanoide y que se aproveche en un futuro para desarrollo de proyectos relacionados.

Se realizó un análisis del desarrollo de proyectos con este kit de robótica y si los estudiantes tenían noción de la existencia de éstos Kits en la Universidad, debido a los resultados obtenidos, se concluye que no existe proyecto alguno relacionado con éstos kits aparte de la introducción de éstos robots en las clases de robótica de la universidad que además es reducido debido a que es una electiva de la carrera.

El proyecto contempla el ensamble de varios procesos requeridos para que los robots sean autónomos y puedan jugar fútbol, entre estos procesos se encuentra el procesamiento de imagen para la visión por máquina, la comunicación para transmisión de datos a los robots desde el algoritmo principal y finalmente el desarrollo de la página web para el OVA.

1.2 ABSTRACT

This project it's about the development of a robot soccer application for the Bioloid Robotic Kit and a web page for itself's diffusion, which is produced applying complementary circuits, multiple platforms and different programming environments such as Arduino, Python and Roboplus. The main objective of this Project is the accomplishment of a Virtual Learning Object for the Mechatronics students and teachers, in order to spread the knowledge obtained from the realization of this Project implementing the Bioloid Robots in humanoid configuration and to be used in the future for the development of related projects. An analysis of the development of projects was carried out with this robotic's kit and if the students had notion of the existence of these Kits in the University, due to the results obtained, it is concluded that there is no project related to these only the introduction of these robots in robotic's class of the university that also is reduced because it is an elective of the career. The project involves the assembly of several processes required for robots to be autonomous and able to play football, among these processes is image processing for machine vision, communication for data transmission to robots from the main algorithm and finally the development of the website for the Virtual Learning Object.

1.1 PLANTEAMIENTO DEL PROBLEMA

1.2.1 Antecedentes del problema

La Universidad Piloto de Colombia (UPC) en el año 2015, adquirió 8 kits de robótica Bioloid Robotis Premium, y por adquisición de los mismos se dictaron capacitaciones que realizaron los fabricantes a un grupo específico de personas (entre estudiantes y docentes de la UPC), donde se abarcaron temas técnicos y prácticos de los kits de robótica Bioloid, de las que no quedaron memorias para su futuro desarrollo, las encuestas respecto al conocimiento y uso de los kits de robótica fueron realizadas a 25 estudiantes del programa y sus resultados se presentan en los Anexos A ,B ,C ,D ,E, F y G, se concluye que la mayoría de los estudiantes saben de la existencia de éstos kits de robótica, pero un muy bajo porcentaje conoce su funcionamiento. Por tal motivo el objetivo principal de este proyecto es ampliar el conocimiento teórico y práctico de estos Kits hacia la

comunidad Piloto por medio de un OVA, desarrollando un tema interesante (Futbol de robots) y completo, que permite usar el robot en su totalidad.

1.2.2 Descripción del problema

Debido al poco uso que se los da a los kits de robótica de la universidad, se realiza un aplicativo de los robots Bioloid en configuración A (humanoide), para que desempeñen roles de arquero y delantero de fútbol, como Objeto Virtual de Aprendizaje se implementa una página web donde se realiza una guía sobre estos kits de robótica encaminados al desarrollo del proyecto de robots de fútbol.

1.3 JUSTIFICACIÓN

Se piensa en un beneficio para la comunidad de estudiantes y docentes del programa de ingeniería mecatrónica de la Universidad, realizando un proyecto efectuado con una temática atractiva (fútbol de robots), ampliando las fuentes de información y aplicando conocimientos adquiridos a lo largo de la carrera.

Se realiza un OVA (Objeto Virtual de Aprendizaje), de manera que presente herramientas útiles y prácticas para el usuario, que sea de fácil acceso, visualmente llamativa y genere sentido de pertenencia a la institución.

1.4 OBJETIVOS

1.4.1 Objetivo general.

- Desarrollar un objeto virtual de aprendizaje con el cual se describa el manejo de los robots Bioloid tomando como temática de estudio el fútbol de robots.

1.4.1 Objetivos específicos.

- Determinar el modelo matemático para una extremidad del robot Bioloid en configuración Humanoide tipo A
- Mejorar la capacidad sensorica de cada robot para la ubicación y actuación del sistema.
- Implementar los sistemas de control de los robots para que puedan ejercer la función de delantero y arquero.
- Determinar qué tipo de OVA se ajusta al proyecto.
- Realizar las prácticas de aprendizaje aplicables al OVA.
- Realizar las pruebas de funcionamiento de los robots.

1.5 DELIMITACIÓN DEL PROYECTO

1.5.1 Alcances y limitaciones.

- Se emplean dos Kits de Robots Bioid de la Universidad Piloto de Colombia, como herramienta de trabajo.
- Se plantea el modelo cinemático directo para el sistema en general. El modelo cinemático directo, inverso y una breve introducción al modelo dinámico inverso para un brazo con 3GDL.
- El sistema de visión artificial tiene la capacidad de identificar los robots y la pelota para realizar las acciones pertinentes.
- El sistema se maneja en un entorno controlado.
- Se utilizan dos robots, uno como delantero y otro como arquero. El robot delantero tiene la capacidad de patear la pelota y en cuyo caso con probabilidad de insertarla en el arco, por su parte el arquero cumple su rol y tiene la probabilidad de tajar la pelota.
- Las medidas de la cancha de los robots están limitadas a la altura de la cámara a 1,56 metros y la capacidad de visión del campo a ésta altura.

1.5.2 Línea de investigación del programa

La línea de investigación que se aborda es referente a la Robótica y biomecatrónica puesto que el desarrollo del proyecto cuenta con robots reales. Además, se introducen nuevos dispositivos electrónicos y algoritmos requeridos para la comunicación entre controlador de fábrica de los kits y los dispositivos de control externo, igualmente la visión por máquina realizada influye directamente en el comportamiento de los robots.

1.6 MARCO REFERENCIAL

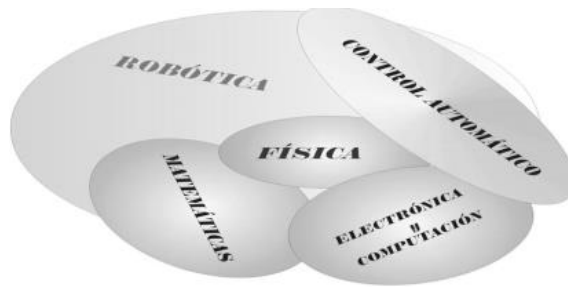
1.6.1 Marco teórico.

ROBÓTICA: “La robótica es un área multidisciplinaria que aborda el análisis, desarrollo de robots y sus potenciales aplicaciones.”¹

Así como se observa en la Ilustración 1.

¹ REYES Cortés Fernando. Robótica , control de robots manipuladores. 1ed. Alfaomega Grupo educativos. México. 2011. Pag 522. ISBN 978-607-707-190-7.

Ilustración 1. Áreas de la robótica



Fuente. Reyes Cortés Fernando. Robótica, Control de robots manipuladores

- **ROBOT**

Existen muchas definiciones del término robot, aunque sus diferencias son irrelevantes en este contexto. Para atender a una definición general de amplia aceptación se puede acudir a la ofrecida por la Asociación Americana de Robótica (Robotics Industries Association, RIA), que dice así:

<<...es un manipulador reprogramable, multifuncional, con control automático que puede estar fijo en un sitio o moverse y que está diseñado para mover piezas, herramientas o dispositivos especiales, por medio de movimientos variables reprogramados para la realización de tareas o trabajos. >>. ²

- **ROBOTS HUMANOIDES**

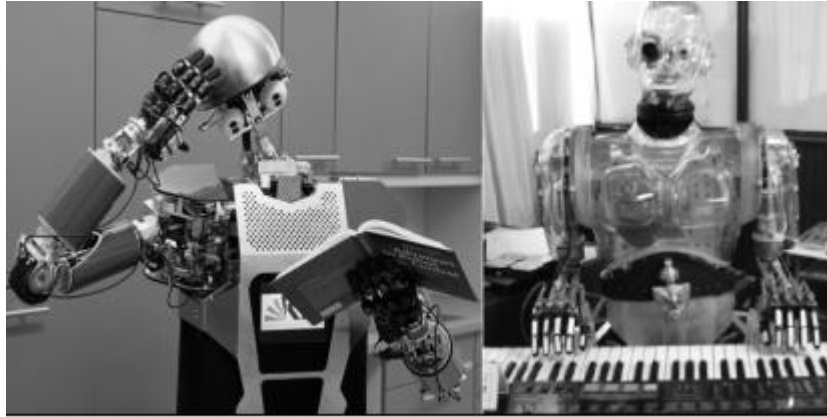
El campo de la robótica incluye el desarrollo de robots humanoides, también conocidos como androides, los cuales son máquinas antropomórficas capaces de imitar las funciones básicas del ser humano tales como caminar, hablar, ver, recolectar, limpiar y trasladar objetos (...) Los androides actuales poseen la capacidad de realizar actividades complicadas, por ejemplo, ejecutar actividades de danza, alcanzar velocidades de 6 km/hora (robot ASIMO Advanced Step in Innovative Mobility), y más aún, el principal potencial del androide es que puede ser empleado como auxiliar a personas en zonas de desastres, es decir, buscar víctimas atrapadas en lugares donde hay derrumbes, llevarles agua, inclusive rescatar y salvar vidas. ³

Se muestran algunos ejemplos de robots humanoides en la Ilustración 2.

² SALIDO Tercero Jesús .Cibernética Aplicada Robots educativos. 1ed.Afaomega Grupo Editor, S.A. de C.V. México. 2010.315 pag. ISBN 978-607-7854-43-2. 316.

³ REYES Cortés Fernando. Robótica , control de robots manipuladores. 1ed. Alfaomega Grupo educativos. México. 2011.522pag. ISBN 978-607-707-190-7.

Ilustración 2. Robots humanoides, PACO-PLUS y Don Cuco el Guapo



Fuente Salido Tercero Jesús. Cibernética Aplicada Robots educativos.

○ **LOCOMOCIÓN MEDIANTE PATAS**

Permiten aislar el cuerpo del terreno empleando únicamente puntos discretos de soporte. Es posible adaptar el polígono de soporte para mantener la estabilidad y pasar sobre obstáculos. Por consiguiente, tienen mejores propiedades que las ruedas para atravesar terrenos difíciles llenos de obstáculos. Asimismo, mediante patas, es posible conseguir la onmidireccionalidad y el deslizamiento en la locomoción es mucho menor.

En los robots con patas la complejidad de los mecanismos necesarios es mayor, así como el consumo de energía en la locomoción. En principio los problemas de planificación y control son más complejos que en los vehículos con ruedas. La configuración más común es la de seis patas. (...). Existen también robots cuadrúpedos y bípedos que mantienen el equilibrio en forma dinámica.⁴

● **ROBOTS BÍPEDOS**

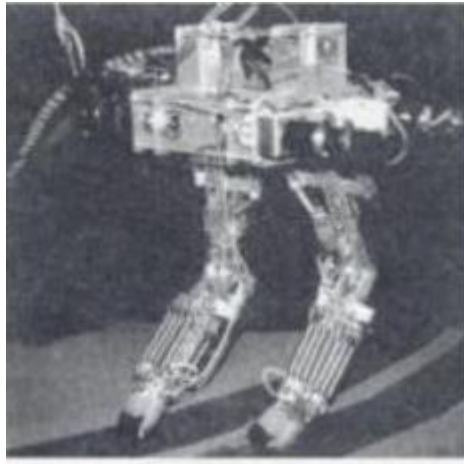
“Los Robots bípedos se diferencian de otros robots con piernas en que su estructura presenta un comportamiento inherentemente inestable, sin importar la forma en que han sido diseñados (Anderson et al., 2005). Ellos requieren un balance activo para moverse ya que su centro de masa se encuentra fuera del área de soporte en un ciclo de caminado (Raibert, 1986).”⁵

⁴ OLLERO Baturone Aníbal. ROBÓTICA Manipuladores y robots móviles. Publicado por Coedición Alfaomega y Marcombo. Barcelona, España. 2001. 447 pag. ISBN: 970-15-0758-4.

⁵ AGUILAR Diego, VIVAS Andrés. Modelado y control de un robot bípedo de nueve grados de libertad. En línea. Universidad del Cauca, Facultad de Ingeniería Electrónica y Telecomunicaciones, Popayán. 2009. Disponible en internet : <
http://www.unicauca.edu.co/ai/publicaciones/Aguilar_ACA09.pdf>.

En la Ilustración 3, se muestra un robot bípedo desarrollado en el MIT.

Ilustración 3. Robot bípedo desarrollado en el Massachusetts Institute of Technology (EE.UU)



Fuente KUMAR Saha Subir. Introducción a la robótica

- **DESCRIPCIÓN DEL SISTEMA**

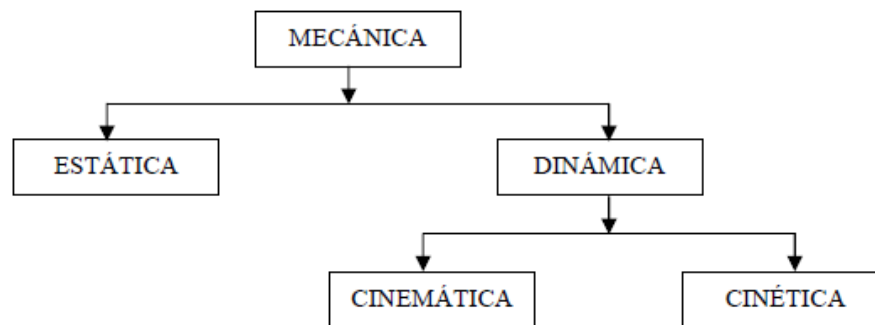
- **CINEMÁTICA**

“La cinemática es el estudio del movimiento independientemente de las fuerzas que lo producen. De manera más específica, la cinemática es el estudio de la posición, el desplazamiento, la rotación, la rapidez, la velocidad y la aceleración.”⁶

En la ilustración 4, se muestra la división de la mecánica como ciencia.

⁶ CASTILLO Ramirez Arturo Apuntes para la materia de Cinemática de las máquinas. Enero 2005. Universidad Autónoma San Luis Potosí, Facultad de ingeniería. 234pag. Pag 3.

Ilustración 4. División de mecánica



Fuente KUMAR Saha Subir. Introducción a la robótica

CINEMÁTICA INVERSA

“El problema de la cinemática inversa consiste en la determinación de las variables de articulaciones correspondientes a una orientación y posición específicas del efector final.”⁷

CINEMÁTICA DIRECTA

“En la cinemática directa para la posición, las posiciones de las articulaciones, es decir los ángulos de las articulaciones de revoluta y el desplazamiento de las articulaciones prismáticas se conocen. La tarea es encontrar la configuración del efector final, es decir su posición y orientación”.⁸

• **MODELO MATEMÁTICO**

Un modelo: es la representación abstracta de algún aspecto de la realidad. Su estructura está conformada por dos partes, la primera son todos aquellos aspectos que caracterizan la realidad modelizada, y la segunda no son más que las relaciones existentes entre los elementos antes mencionados.

Científicos e Ingenieros usan al menos alguna de las tres metodologías para obtener las ecuaciones de un modelo las cuales se describen a continuación:

⁷ KUMAR Saha Subir. Introducción a la robótica. De la primera edición en inglés de *Introduction to robotics*, by Subir Kumar Saha. Publicado por Tata McGraw-Hill Education Private Limited, NOIDA, India. 371pag. ISBN: 978-0-07-066900-0.p.52

⁸ Ibid., p. 52-53.

1. **Fundamental:** Usa la teoría aceptada de la ciencia fundamental para obtener ecuaciones. En este caso, las teorías que se aceptan son los axiomas básicos en el proceso lógico de construcción de un modelo.
2. **Empírica:** Hace uso de observación directa para desarrollar ecuaciones que describen los experimentos.
3. **Analogía:** Usar las ecuaciones que describen a un sistema análogo, con variables identificadas por analogía en una base uno a uno.

Además, un modelo matemático está basado en la lógica matemática, cuyos elementos son esencialmente variables y funciones, y las relaciones entre ellas, que vienen expresadas a través de relaciones matemáticas (ecuaciones, inecuaciones, operadores lógicos, etc.) que se empatan con las correspondientes relaciones del mundo real que modelizan (relaciones tecnológicas, leyes físicas, restricciones del mercado, etc.).

Una de las razones para obtener un modelo es la adecuación del cálculo del supuesto comportamiento de un proceso para determinadas condiciones, el cálculo depende de la aplicación; por ejemplo, un modelo de tratamiento de aguas debe ser usado para determinar la cantidad de contaminantes presentes para la limpieza parcial o total del agua tratada. De esta forma podemos mostrar que la importancia de los modelos matemáticos radica en que:

- Nos revela a veces relaciones que no son evidentes a primera vista.
- Una vez construido el modelo, es posible extraer de él propiedades y características de las relaciones que de otra forma permanecerían ocultas.
- En aquellas situaciones del mundo real en las que no es posible experimentar con la realidad, ofrecen un marco teórico para evaluar la toma de decisiones, así como sus consecuencias.

Los modelos pueden ser estáticos o dinámicos, en un modelo estático, la variable tiempo no desempeña un papel relevante, por el contrario, en un modelo dinámico, ya que alguno(s) de los elementos que intervienen en la modelización no permanecen constantes, sino que se consideran como funciones del tiempo, describiendo trayectorias temporales. El análisis de un modelo dinámico tiene por objeto el estudio de la trayectoria temporal específica de alguno (s) de sus elementos.⁹

➤ SISTEMA DE VISIÓN

⁹ MÉNDEZ Regalado Alejandro, PERALTA Reyes Ever, GOZALEZ Carlos Alberto. Ensayos. Cómo hacer un modelo matemático. Universidad Tecnológica de la Mixteca. Temas de Ciencia y Tecnología. Mayo - Agosto 2008. En línea. Disponible en <http://www.utm.mx/edi_antiores/temas035/2%20ensayo-35.pdf>

Los sensores de visión pueden clasificarse como de tipo externo sin contacto. Sin embargo, aquí se estudian en una sección separada debido a su complejidad, que requiere un tratamiento detallado. (...). Los sistemas de visión se utilizan exitosamente con robots para permitir que miren a su alrededor con el fin de encontrar las piezas que van a recoger y para colocarlas en los lugares apropiados.

❖ Elementos del sensor de visión

En los sistemas de visión, el componente principal de visión es una cámara completa, incluidos un equipo sensor, la electrónica asociada, el formato de señales de salida y un lente. En función de la aplicación, la cámara podría ser una RS-170/CCIR, NTSC/PAL (éstas son señales estándares de televisión americana RS-170 monocolor, de televisión europea/india CCIR monocolor, NTSC color y PAL producidas por las videocámaras, respectivamente), de escaneo progresivo, variable o de líneas. Los cinco parámetros de sistema más importantes que rigen la elección de la cámara son campo de vista, resolución, distancia de trabajo, profundidad de campo e índice de adquisición de datos de imágenes. (...)

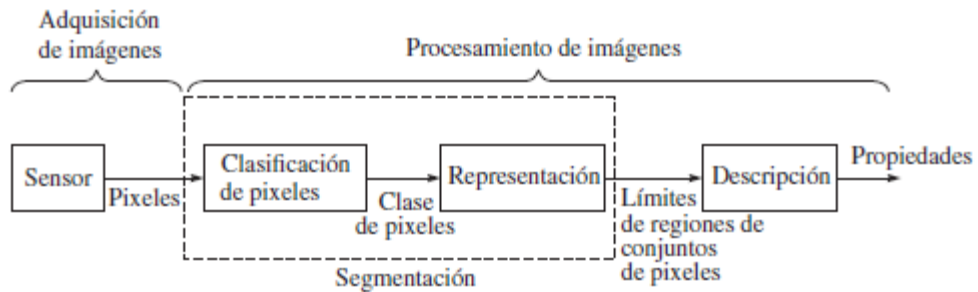
i) Sistemas de cámaras Los primeros sistemas de visión empleaban la cámara vidicón, que era un dispositivo voluminoso de tubos de vacío. Las cámaras vidicón son mucho más susceptibles a la interferencia de ruido electromagnético y requieren mucha energía.

Sus ventajas principales son una más alta resolución y mejor sensibilidad a la luz. A fin de reducir el tamaño, en la mayoría de los sistemas actuales se usan cámaras de estado sólido, basadas en técnicas de dispositivos de carga acoplada (CCD) o en dispositivos de inyección de carga (CID). Las cámaras de estado sólido son más pequeñas, más resistentes, duran más y tienen menos distorsión de imágenes inherente que las cámaras vidicón. También son levemente más caras, pero los precios están bajando. Los fabricantes frecuentemente recomiendan la operación de los sensores con filtros apropiados para eliminar longitudes de onda IR. Tanto los chips de CCD como de CID utilizan técnicas de transferencia grande para capturar una imagen. En una cámara CCD, la luz incide en el equivalente óptico de un chip de memoria de acceso aleatorio (RAM). La luz se absorbe en sustrato de silicón, donde la acumulación de carga es proporcional a la cantidad de luz que llega al equipo. Una vez que se haya recibido la cantidad suficiente de energía para proporcionar una imagen, las cargas se transmiten a través de los registros de control integrados.¹⁰

En el siguiente esquema (Ilustración 5), se describe el procedimiento general para la detección visual de objetos.

¹⁰ SUBIR KUMAR SAHA. Introducción a la robótica. De la primera edición en inglés de *Introduction to robotics*, by Subir Kumar Saha. Publicado por Tata McGraw-Hill Education Private Limited, NOIDA, India. 371 pag. ISBN: 978-0-07-066900-0.p. 66-67.

Ilustración 5. Detección visual



Fuente: Sparkfun. PcDuino3B. Página web. Disponible en <<https://www.sparkfun.com/products/13707>>

○ OVAS

Los OVA son componentes que permiten facilitar el aprendizaje por medio de interacción con un entorno virtual como se puede observar en el proyecto (ELABORACION DE UN OBJETO VIRTUAL DE APRENDIZAJE (OVA) SOBRE “TECNICAS DE FACILITACION EUROMUSCULAR PROPIOCEPTIVA”) realizado por –Isabel Cristina Gómez Díaz- lo que lleva a enseñar a profesionales de la salud sobre “Técnicas de Facilitación Neuromuscular Propioceptiva - TFNP”¹¹

Tipos de Objetos Virtuales de Aprendizaje

- Los Simuladores: Aplicativos en software y permiten interpretar la realidad y sus fenómenos facilitando la construcción de conocimiento en forma interactiva, permiten modificar variables y parámetros para experimentar nuevos resultados.
- La Multimedia: integra recursos de audio, video y datos que dispuestos de forma lógica, interactúan con el usuario promoviendo el aprendizaje autónomo.
- Tutoriales: se caracterizan por ser más instructivos describiendo en su gran mayoría procesos y procedimientos paso a paso relacionados para la consolidación de una actividad.
- Imágenes en movimiento: Esta categoría puede presentar: Animación, Video, Video-Clips, juegos interactivos.
- Imágenes fijas: Encontramos dentro de esta categoría:
 - Fotografías: Imágenes de diversas extensiones o formatos.
 - Ilustraciones: Donde se emplea el dibujo a mano alzada
 - Redes o Mapas conceptuales.

¹¹GÓMEZ DÍAZ ISABEL CRISTINA. Universidad de Santander, “UDES” Bucaramanga, . ELABORACION DE UN OBJETO VIRTUAL DE APRENDIZAJE (OVA) SOBRE “TECNICAS DE FACILITACION NEUROMUSCULAR PROPIOCEPTIVA” . [En línea]. [Citado el 24 de febrero de 2016]. Disponible en: <<http://www.bdigital.unal.edu.co/17477/2/13111-36454-1-PB.pdf>>.

- Documento texto, Tabla, gráfico.¹²

1.6.2 Estado del arte.

ASIMO The World's Most Advanced Humanoid Robot

Los ingenieros de Honda crearon ASIMO, un robot humanoide con 34 grados de libertad que le ayudan a caminar y realizar tareas semejantes a las de un ser humano. Un grado de libertad es la capacidad para moverse a la derecha y a la izquierda o arriba y abajo. Estos grados de libertad actúan como unas articulaciones para dar movimiento y flexibilidad. Posee materiales ligeros, además de potentes computadores y 34 servomotores a través de su cuerpo que ayudan a ASIMO a moverse con facilidad.¹³

Atlas - The Agile Anthropomorphic Robot

Atlas es un robot de alta movilidad, diseñado para tomar decisiones al aire libre y en terrenos rugosos, puede caminar bípedamente dejando las demás extremidades libres para levantar cosas, transportarlas y manipular su entorno. Atlas es fuerte y coordinado lo suficiente para escalar con sus manos y pies, puede escoger su camino en espacios congestionados. Las manos censadas y articuladas de Atlas permiten que use herramientas humanas. Atlas incluye 28 grados de libertad de actuadores hidráulicos, dos manos, brazos, piernas, pies y torso. Un sensor articulado en la cabeza incluye cámaras estéreo y un láser de rango buscador. Atlas está alimentado a través de una correa de sujeción flexible.¹⁴

SURENA III

Investigadores iraníes de la Universidad de Tehran, en una demostración el robot de tamaño adulto, donde caminaba, imitaba los gestos de los brazos de una persona, y se ponía de pie mientras se doblaba hacia atrás.

Con una carcasa de plástico, y ojos brillantes de led, Surena III es de 1.9m (6 ft y 3 in) de alto y pesa 98 Kg(216lb). Está equipado con una serie de sensores, incluyendo un módulo de visión 3D basado en Kinect y sus articulaciones son alimentadas por 31 servomotores. Un software de control es ejecutado en el robot

¹² GRUPOS JUVENILES CREATIVOS. Instituto Educativo Distrital Ciudadela 20 de Julio. Herramientas de comunicación virutales. [En línea]. [Citado el 18 de Marzo de 2016]. Disponible en: <<http://files.informaticainedicdelahoz.webnode.com.co/200000385-0c1410e084/Guia%20-%204%20InformaticaCiclo%20V.pdf>>

¹³ American Honda Motor Co., Inc. ASIMO The World's Most Advanced Humanoid Robot [ASIMO El robot humanoide más avanzado del mundo].s.f.[Base de datos en línea]. Disponible en la página Web: <<http://asimo.honda.com/asimo-specs/>> [citado en 28 Febrero de 2016]

¹⁴ BOSTON DYNAMICS. Atlas - The Agile Anthropomorphic Robot [Atlas- El ágil robot antropomórfico].[Base de datos en línea]. Propiedad Filial total de Google, Inc. Waltham, Massachusetts. Boston Dynamics.s.f.. Disponible en Internet: <http://www.bostondynamics.com/robot_Atlas.html>[citado en 28 Febrero de 2016]

y monitoreado utilizando operadores humanos para supervisar sus funciones, se basan en el sistema operativo Robot, o ROS.¹⁵

DARWIN-OP

(Dynamic Anthropomorphic Robot with Intelligence – Open Platform), es una plataforma asequible robot-miniatura-humanoide, con avanzado control computacional, sensores sofisticados, alta capacidad de carga útil, y habilidad de movimiento dinámico para la investigación, educación, y actividades de divulgación. Patrocinado por la (NSF), National Science Foundation en Estados Unidos, DARwIn- OP ha sido desarrollado por RoMeLa Virginia Tech, con la colaboración de University of Pennsylvania, Purdue University and Robotis Co. Posee un tamaño de 0.455m (17.9in), un peso de 2.8kg (6.2lb), baterías LiPo, y cámara Web por USB.¹⁶

ROBOT BÍPEDO UNIVERSIDAD NACIONAL

Se trata de uno de los logros del Grupo de Investigación Inteligencia Artificial, que consiste en un prototipo hecho por 27 servi-motores de tipo análogo, controlados con un microcontrolador, una tecnología cada vez más poderosa. También tiene implementadas partes rígidas en aluminio, entre otros materiales. El rostro, que se asemeja a la 'cara' de los personajes de la película Yo Robot, es fabricado en una resina de poliéster.¹⁷

CHARLI-L

Dennis Hong, profesor de ingeniería mecánica y director del Laboratorio de Robótica y Mecanismos de Virginia Tech, o RoMeLa, presentó un robot humanoide de tamaño normal llamado CHARLI, o un robot humanoide Autónomo cognitivo con inteligencia de aprendizaje.

El robot tiene de altura de 5 pies (1,52 metros), sin ataduras y autónomo, capaz de caminar y gesticular, su mayor innovación es que no utiliza articulaciones

¹⁵ EVAN ACKERMAN, Erico Guizzo, Angelica Lim, Jason Falconer. Iran Demonstrates New Humanoid Robot Surena III [Irán demuestra el nuevo robot humanoide Surena III. [Boletín informativo En Línea]. [eng]. New York, N.Y. .IEEE SPECTRUM. Fecha de publicación 17 Nov 2015 , 21:55 GMT. Disponible en Internet : < <http://spectrum.ieee.org/automaton/robotics/humanoids/iran-humanoid-robot-surena-iii>>. [Citado en 28 Febrero de 2016].

¹⁶ DR. DENNIS HONG, Daniel Lee. DARwIn OP: Open Platform Humanoid Robot for Research and Education. En línea. Los Ángeles. Septiembre 25 , 2011. < <http://www.romela.org/darwin-op-open-platform-humanoid-robot-for-research-and-education/>>. [Citado en 29 Febrero de 2016].

¹⁷ UNIMEDIOS, AGENDA DE NOTICIAS UN, CIENCIA & TECNOLOGÍA. En la UN construyen robot bípedo. Fecha de publicación 09 de marzo de 2011. Disponible en internet <<http://agenciadenoticias.unal.edu.co/nc/detalle/article/en-la-un-construyen-robot-bipedo/>> [Citado en 29 Febrero de 2016].

rotacionales, como la mayoría de robots humanoides, es por eso que él y su equipo de estudiantes diseñaron un sistema ingenioso donde acoplaron poleas y resortes, que son los que permiten el movimiento del robot.¹⁸

PROYECTOS RELACIONADOS

DESARROLLO DE ALGORITMO PARA DETECCIÓN Y COMANDO DE ROBOTS HUMANOIDES EN TAREAS DE RECOLECCIÓN

En este artículo se describe un algoritmo usado para controlar varios robots Bioloid en configuración humanoide, se trabaja un Sistema Multi-agente (SMA), que se trabaja para que los robots trabajen entre sí, igualmente el algoritmo trabaja simultáneamente en dos fases, la primera que se desempeña en visión de máquina y la ubicación espacial, y la segunda fase en la comunicación ZigBee con los robots, finalmente se realizaron las simulaciones virtuales pertinentes, empleando el programa Webots©.¹⁹

SIMULACIÓN DEL CICLO DE MARCHA DEL ROBOT BÍPEDO BIOLOID EN EL ENTORNO VIRTUAL V-REP

Este artículo presenta una nueva herramienta para para simular las trayectorias de marcha del robot Bioloid Kit Premium en configuración bípeda, se emplea el software VREP (Virtual Robot Experimentation Platform) para la simulación del movimiento del robot, y el lenguaje de Matlab para lograr una interfaz para el usuario, y que de allí pueda realizar las trayectorias de marcha al VREP. También se empleó Solid Edge para el modelo en 3D, debido a que éste incluye partes que posee el robot real, así mismo ondas sinusoidales para la marcha bípeda, de manera que se usa el algoritmo en el robot real con las simulaciones anteriormente mencionadas y se comprobó que es óptimo y funcional.²⁰

FÚTBOL DE ROBOTS: REVISIÓN DEL ESTADO DEL ARTE Y DESARROLLO DEL EQUIPO UBASOT DE SIMULACIÓN

Se realiza el estudio del problema fútbol de robots, teniendo en cuenta cada uno de los aspectos pertinentes al realizar un equipo de fútbol de robots, dividen el problema en modelado del entorno, acciones, comportamientos elementales,

¹⁸ EVAN ACKERMAN, Erico Guizzo, Angelica Lim, Jason Falconer. Virginia Tech's Humanoid Robot CHARLI Walks Tall . [Boletín informativo En Línea],[eng]. New York,N.Y. .IEEE SPECTRUM. Fecha de publicación 17 Nov 2015 , 21:55 GMT. Disponible en Internet : < <http://spectrum.ieee.org/automaton/robotics/humanoids/042810-virginia-tech-humanoid-robot-charli-walks-tall>>. [Citado el 5 de Marzo de 2016].

¹⁹ VARGAS TORRES, G. A., & CASTILLO ESTEPA, R. A. (2015). Desarrollo de algoritmo para detección y comando de robots humanoides en tareas de recolección. Revista Tecnura, 19(45), 127-139. doi:10.14483/udistrital.jour.tecnura.2015.3.a10

²⁰ TACUÉ GONZÁLEZ JEISON JAVIER, NARANJO MAGÉ HUGO FABIAN, RENGIFO RODAS CARLOS FELIPE.Simulación del ciclo de marcha del robot bípedo bioloid en el entorno virtual v-rep. Popayán, Cauca, Colombia. 2015.ResearchGate.En línea. Disponible en <https://www.researchgate.net/publication/284283402_Simulacion_del_Ciclo_de_Marcha_del_Robot_Bipedo_Bioloid_en_el_Entorno_Virtual_V-REP>

comportamientos complejos, jugadas, tácticas y estrategias, el objetivo principal es proveer un marco teórico que sea útil para proyectos y desarrollos futuros. Igualmente se presenta la implementación de un equipo de fútbol simulado usado en el Campeonato Mundial de Fútbol de Robots FIRA 2002. Para el desempeño de cada robot se tuvieron en cuenta parámetros como la comprensión del entorno, las decisiones tácticas y estratégicas, y la ejecución de acciones. El robot cuenta ruedas laterales que modifican la potencia para cambiar la dirección; cuenta con un algoritmo de planificación y para el desempeño en equipo cuentan con coordinación implícita.²¹

INVESTIGACIÓN EN FÚTBOL ROBÓTICO SOBRE PLATAFORMA LEGO MINDSTORM.

Se realiza la solución al problema fútbol de robots usando como elemento principal la plataforma lego Mindstorm, se busca la interacción entre los robots, y que cumplan su rol de arquero y delantero respectivamente. Para ello se usa como herramienta principal los sistemas multiagentes, para la ubicación espacial de los robots, se realizó el algoritmo adecuado con los sensores IR incluidos en el robot, así mismo luego de la ejecución del proyecto y de las pruebas pertinentes se concluye que éste tipo de sistemas hacen necesario la repetitividad de acciones mediante el anidamiento de ciclos, igualmente que se deben manejar en entornos cerrados, donde la luz, campos magnéticos o agentes contaminantes no realicen interferencia alguna; se realizaron los objetivos planteados pero se denotó limitación en cuanto al software, debido a la baja memoria y a la poca velocidad del procesamiento de los Lego MindStorm.²²

VISIÓN ARTIFICIAL ESTERO CON APLICACIÓN AL CONTROL DE UN BRAZO DE ROBOT

Se realiza un trabajo de tres años y medio (Enero 2000 a Mayo 2003), donde se diseñaron e implementaron varios métodos para la visión por computadora, que es posteriormente aplicada al control visual de robots manipuladores. Dentro de ellos se incluyen componentes necesarios para desarrollar el problema: robot manipulador A465; dos cámaras analógicas Pulnix; dos lentes Cosmimar C815b y una tarjeta digitalizadora de *National Instruments*; se desarrolla entonces un nuevo software (modular) que comunique los componentes y aplicar los algoritmos creados. Se diseña e implementa un nuevo grupo de algoritmos de visión por computadora, el cual consiste en extraer la información visual a partir de los valores de los centroides de las tres marcas visuales (entre ellas un círculo negro), tamaño y forma. Para la posición en espacio tridimensional se usan los lentes Cosmimar C815b; y finalmente se diseña e implementa un controlador difuso, pero se empleó

²¹ CLAUDIA C. CASTELO, Héctor R. Fassi, Flavio E. Scarpettini. Fútbol de Robots: Revisión del Estado del Arte y desarrollo del Equipo UBASot de Simulación. Tesis de Licenciatura. Buenos Aires. Departamento de Computación Facultad de Ciencias Exactas y Naturales Universidad de Buenos Aires. 2002. 326p

²² DIEGO MAURICIO GUTIERREZ RAMIREZ. Juan Pablo Ocampo Ramirez. Elvin Valdes Avila. Investigación en Fútbol Robótico sobre plataforma lego mindstorm. Tecnología en Electrónica. Bogotá. Corporación Universitaria Minuto de Dios. 2013. 169pag.

un controlador visual PD para la estabilidad, y redes neuronales multicapa para a compensación de las fuerzas de gravedad y fricción.²³

CONTROL DE ROBOTS AUTÓNOMOS MÓVILES BASADOS EN ARQUITECTURAS DE SISTEMAS EMBEBIDOS

El objetivo de éste documento es ampliar los conocimientos en cuanto a Ingeniería de Software, buscando, analizando y seleccionando algunos embebidos que cumplan la función de controlar Robots Autónomos Móviles, debido al poco énfasis funcional que le dan, se realizó un correcto análisis para seleccionar los embebidos adecuados.²⁴

HIERARCHICAL MOTION CONTROL FOR A TEAM OF HUMANOID SOCCER ROBOTS (Control jerárquico del movimiento para un equipo de fútbol de robots humanoides)

En este artículo, se describen los detalles de un controlador de movimiento para controlar un equipo de fútbol de robots humanoides, el cual consiste en una jerarquía de controladores con diferentes marcos de tiempo y niveles de abstracción. Un controlador de nivel bajo gobierna el control en tiempo real de cada ángulo de la articulación, ya sea utilizando ángulos de destino de las articulaciones. Un controlador de nivel medio se encarga de la locomoción bípeda y el equilibrio del robot. Un controlador de alto nivel decide el comportamiento a largo plazo del robot, y, finalmente, el controlador de nivel equipo coordina el comportamiento de un grupo de robots por medio de la comunicación asíncrona entre los robots. El sistema de movimiento usado en este artículo ha sido utilizado con éxito por muchos equipos de robots humanoides en la Robocup.²⁵

OVA DE COMPUERTAS DIGITALES

Así como se puede implementar a profesionales, esta herramienta también puede los OVA en la enseñanza enfocada ser utilizada en estudiantes de electrónica como en la práctica académica desarrollada por “OVA DE COMPUERTAS DIGITALES” de la universidad Católica de Pereira, en donde se analizan los OVA en varios de sus aspectos así como el objeto de estudio a desarrollar en la clase de electrónica

²³ M. EN C. MARCO ANTONIO MORENO ARMENDARIZ. Visión artificial estero con aplicación al control de un brazo robot. Doctor en Ciencias de en la Especialidad de Control Automático. México, D.F. Centro de investigación de estudios avanzados del Instituto Politécnico Nacional Departamento de Control Automático.2003.157pag.

²⁴ APU JONATAN MATÍAS PONZO. Control de robots autónomos móviles Basados en arquitecturas de sistemas embebidos, Propuesta de Modelo de Aplicación y Uso Potencial. Licenciado en sistemas.Buenos Aires, Argentina. Universidad Nacional de Lanús.2013.367 pag. Disponible en : <<http://sistemas.unla.edu.ar/sistemas/gisi/TFLS/Ponzo-TFL.pdf>>

²⁵ SEUNG-JOON YI, STEPHEN MCGILL, DENNIS HONG AND DANIEL LEE. Hierarchical Motion Control for a Team of Humanoid Soccer Robots. Int J Adv Robot Syst, 2016, 13:32. doi: 10.5772/62037. En línea. Disponible en: [tp://www.intechopen.com/journals/international_journal_of_advanced_robotic_systems/hierarchical-motion-control-for-a-team-of-humanoid-soccer-robots](http://www.intechopen.com/journals/international_journal_of_advanced_robotic_systems/hierarchical-motion-control-for-a-team-of-humanoid-soccer-robots)>

para enseñar un tema en específico en el que los aprendices están contextualizados con la terminología producto del autoaprendizaje llevado a cabo en el mismo OVA²⁶

DISPOSITIVO PAR LA ESTIMACIÓN DE FLUJO ÓPTICO EN IMÁGENES MEDIANTE FPGAS

La patente “WO2005050342A2” en la que quieren hacer un sistema de cómputo minimalista para procesamiento de “estimación de flujo óptico mediante fugas” y que se portable, implementan sistemas embebido de tipo FPGA (Siglas que corresponden a Field Programmable Gate Array) para desarrollar el respectivo hardware y software por medio de esta plataforma para la respectiva aplicación que en contraste con un DSP (siglas que corresponden a Digital Signal Processor) o sistema de cómputo tradicional no sería posible.²⁷

DISPOSITIVO ELECTRONICO DE ESTACIONES REMOTAS PARA CONTROL A DISTANCIA

La patente “WO2008074905A1” en la que a partir de un sistema embebido con periféricos inalámbricos se desea aplicar al campo agrícola una red de riego comunicada y reprogramable inalámbricamente y de bajo consumo para minimizar costos de mantenimiento siendo reubicable en la topología según sea requerido y teniendo en cuenta la limitante de distancia por los módulos de radiofrecuencia²⁸

1.7 METODOLOGÍA

La metodología llevada a cabo en el desarrollo del proyecto, conllevó los siguientes pasos:

1. Familiarización con kits de robótica, conocer el software y hardware de los kits de robótica Bioloid Premium, para realizar el aplicativo de fútbol de robots.
2. Cálculos mecánicos de cinemática y cinética para un brazo de 3GDL.
3. Selección de dispositivos y software para visión artificial de robots.
4. Desarrollo sistema de comunicación para robots con visión artificial inalámbrica.
5. Implementación visión artificial en robots y adecuación de entorno para los robots.

²⁶ Universidad Católica de Pereira. OVA DE COMPUERTAS DIGITALES. [En línea]. [Citado el 24 de febrero de 2016]. Disponible en:

<<http://ribuc.ucp.edu.co:8080/jspui/bitstream/handle/10785/1218/Proyecto%20OVA%20de%20COMPUERTAS%20DIGITALES.pdf?sequence=1>>.

²⁷ UNIV GRANADA. DISPOSITIVO PAR LA ESTIMACIÓN DE FLUJO ÓPTICO EN IMÁGENES MEDIANTE FPGAS. Inventores: Vidal, Fernando Ros; Fernandez, Sonia Mota; Alonso Antonio, Javier Diaz. Patente de aplicación. Numero de publicación: WO2005050342 A2. Fecha de publicación: 02 de junio de 2005

²⁸ Dracotic S L. Dispositivo electronico de estaciones remotas para control a distancia. Gomez Skarmeta, Antonio Fernand; Monsalve Tresaco, Enrique; Zamora, Miguel Angel. Patente de aplicación. Numero de publicación: WO2008074905A1. Fecha de publicación: 26 de junio de 2008.

6. Desarrollo de Algoritmo de vinculación entre visión de máquina y sistema de comunicación de robots.
7. Pruebas físicas del Algoritmo Visión artificial - comunicación inalámbrica, con los robots en el entorno controlado.
8. Selección, desarrollo y creación página Web como Objeto virtual de Aprendizaje, integrando el desarrollo del aplicativo fútbol de robots.

1.8 RECURSOS.

- Préstamo de dos kits de robótica “Biolid Robotis Premium” por parte de la Universidad Piloto de Colombia.
- El tiempo de trabajo de docentes que nos colaboran con la elaboración del proyecto son financiados por la universidad.
- El tiempo de trabajo de los estudiantes proponentes será financiado por recursos propios de los mismos.
- Embebidos, cámaras, sensores y demás dispositivos electrónicos requeridos financiados por la universidad y también por recursos propios de los autores.

2. ANÁLISIS CINEMÁTICO DIRECTO, INVERSO Y DINÁMICO INVERSO ORIENTADO AL APRENDIZAJE PARA UNA EXTREMIDAD DE 3GDL

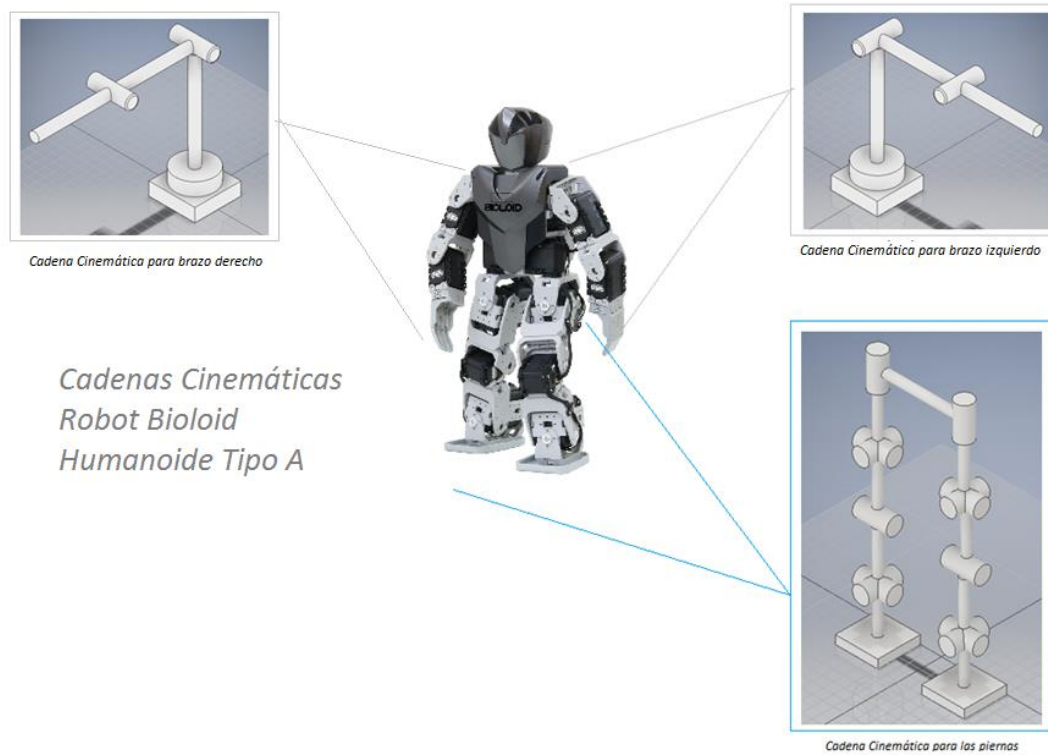
Cuando se realiza el diseño de un sistema, en este caso de un robot, se deben partir de parámetros físicos y de construcción para el desempeño óptimo del mismo, además de pronosticar el comportamiento y el consumo una vez realizado el montaje.

Se deben tener en cuenta variables como los grados de libertad según las tareas articulares que realiza el robot, en este caso, 18DF (grados de libertad) , el tamaño del robot de altura de 35 centímetros , el material para realizar tareas de simulación de movimientos humanoides y desplazamiento de toda la estructura, debe ser resistente a posibles golpes, estéticamente atractivo y liviano, lo suficiente para poder desplazarse sin ningún inconveniente, lo anterior concerniente a la descripción física y de construcción, y éstos parámetros en el proyecto están definidos por el fabricante del kit de robótica.

Debido a que el diseño del robot, selección de materiales y construcción fueron realizados por el fabricante, el análisis matemático se plantea para pronosticar el comportamiento y el consumo del mismo, se realizan análisis cinemáticos directos e inversos y parte de la cinética.

El robot Bioloid es un sistema robótico de 18DF (grados de libertad) dentro del cual para el análisis planteado a continuación se toma como base fija el tronco y las extremidades se descomponen en cadenas cinemáticas abiertas como se muestra en la ilustración 6.

Ilustración 6. Esquema general del robot y su representación en cadenas cinemáticas



Fuente: propia y Robotshop, BIOLOID Premium Robot Kit, disponible en <<http://www.robotshop.com/en/robotis-bioloid-robot-construction-kit.html>>

Se extraen las matrices de Transformación homogénea, y posteriormente con el método de Denavit-Hartenberg se llega a la relación que existe entre los elementos del sistema, los brazos pueden representarse como dos cadenas cinemáticas abiertas, cada una con 3GDL el procedimiento es realizado para un solo brazo dado que son dos cadenas cinemáticas análogas.

Con el fin de desarrollar el objetivo específico “*Determinar el modelo matemático para una extremidad del robot Bioloid en configuración Humanoide tipo A*”, se plantea una descripción cinemática directa por el método de Denavit- Hartenberg²⁹, donde en objetivo la extremidad se refiere a un brazo. Se plantea entonces la descripción matemática de la cinemática inversa a partir de las ecuaciones y dinámica por el método de LaGrange³⁰. De igual forma se deja planteada la cinemática directa para las extremidades de las piernas.

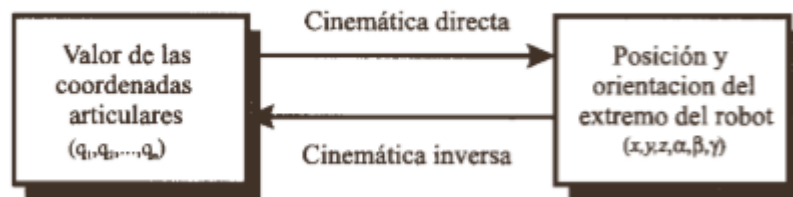
2.1 CINEMÁTICA DIRECTA

²⁹ BARRIENTOS Antonio, PEÑIN Luis Felipe, BALAGUER Carlos, ARACIL Rafael. 2 ed. McGraw-Hill, 2007. 512 p. ISBN 9788448156367.

³⁰ YEPEZ Mulia Enrique, YEPEZ Mitzli Martínez. 1 ed. Universidad Nacional Autónoma de México, 2007. 389p. ISBN 9786070219337.

Como establece Kumar³¹, La cinemática corresponde al estudio de la geometría del movimiento del sistema. Se utiliza para relacionar el desplazamiento, la velocidad, la aceleración y el tiempo, sin hacer referencia a la causa del movimiento como se muestra en la ilustración 7.

Ilustración 7 Esquema cinemática directa e inversa



Fuente: Barrientos Antonio. Fundamentos de robótica.

“La cinemática directa consiste en determinar cuál es la posición y orientación del extremo final del robot, con respecto a un sistema de coordenadas que se toma como referencia, conocidos los valores de las articulaciones y los parámetros geométricos de los elementos del robot.”³²

Existen diversos métodos para la solución de la misma, se optó por emplear el método de Denavit-Hartenberg para la extracción de matrices de transformada homogénea, ya que es el método más empleado en robótica y es sistemático lo cual reduce las posibilidades de error; además debido al gran número de grados de libertad, éste método es el más conveniente en comparación del método geométrico, dado que el método geométrico parte de una descripción trigonométrica básica respecto a diagramas de cuerpo libre, es decir que se plantean diversos diagramas de cuerpo libre respecto a cada eslabón de la cadena, para encontrar la posición de uno respecto al anterior; mientras el método de Denavit Hartenberg establece normas donde se puede plantear el análisis completo en un solo diagrama de cuerpo libre, siempre y cuando se realice el siguiente algoritmo.

ALGORITMO DE DENAVIT HARTEMBERG

Para la extracción de los parámetros de las matrices de transformación se utiliza el Algoritmo de Denavit-Hartenberg, se muestra a continuación:

³¹ KUMAR Subir Saha. Introducción a la robótica. Traducido por Dr. Thomas W. Bartenbach. 1 ed. México. D.F.: McGraw-Hill, 2010. 403 p. ISBN 13: 978-607-15-0313-8

³² BARRIENTOS Antonio, PEÑIN Luis Felipe, BALAGUER Carlos, ARACIL Rafael. 2 ed. McGraw-Hill, 2007. 512 p. ISBN 9788448156367.

1. "Numerar los eslabones comenzando con 1 (primer eslabón móvil de la cadena) y acabando con n (último eslabón móvil). Se numerará como eslabón 0 a la base fija del robot.
2. Numerar cada articulación comenzando por 1 (la correspondiente al primer grado de libertad y acabando en n).
3. Localizar el eje de cada articulación. Si esta es rotativa, el eje será su propio eje de giro. Si es prismática, será el eje a lo largo del cual se produce el desplazamiento.
4. Para i de 0 a $n - 1$, situar el eje Z_i , sobre el eje de la articulación $i + 1$.
5. Situar el origen del sistema de la base (S_0) en cualquier punto del eje Z_0 . Los ejes X_0 e Y_0 se situarán de modo que formen un sistema dextrógiro con Z_0 .
6. Para i de 1 a $n - 1$, situar el sistema (S_i) (solidario al eslabón i) en la intersección del eje Z_i con la línea normal común a Z_{i-1} y Z_i . Si ambos ejes se cortasen se situaría (S_i) en el punto de corte. Si fuesen paralelos (S_i) se situaría en la articulación $i+1$.
7. Situar X_i en la línea normal común a Z_{i-1} y Z_i .
8. Situar Y_i de modo que forme un sistema dextrógiro con X_i y Z_i .
9. Situar el sistema (S_n) en el extremo del robot de modo que Z_n coincida con la dirección de Z_{n-1} y X_n sea normal a Z_{n-1} y Z_n .
10. Obtener θ_i como el ángulo que hay que girar en torno a Z_{i-1} para que X_{i-1} y X_i queden paralelos.
11. Obtener D_i como la distancia, medida a lo largo de Z_{i-1} , que habría que desplazar S_{i-1} para que X_i y X_{i-1} quedasen alineados.
12. Obtener A_i como la distancia medida a lo largo de X_i (que ahora coincidiría con X_{i-1}), que habría que desplazar el nuevo S_{i-1} para que su origen coincidiese con (S_i).
13. Obtener a_i como el ángulo que habría que girar en torno a X_i (que ahora coincidiría con X_{i-1}), para que el nuevo (S_{i-1}) coincidiese totalmente con (S_i).
14. Obtener las matrices de transformación ${}^{i-1}A_i$.
15. Obtener la matriz de transformación que relaciona el sistema de la base con el del extremo del robot $T = {}^0A_1, {}^1A_2 \dots {}^{n-1}A_n$.
16. La matriz T define la orientación (submatriz de rotación) y posición (submatriz de traslación) del extremo referido a la base en función de las n coordenadas articulares.

Donde los parámetros importantes son

- θ . Es el ángulo que forman los eje X_{i-1} y X_i medido de un plano perpendicular al eje Z_{i-1} , utilizando la regla de la mano derecha.
- d . Es la distancia a lo largo del eje Z_{i-1} desde el origen del sistema de coordenadas $(i-1)$ -esimo hasta la intersección del eje Z_{i-1} con el eje X_i .
- a . Es la distancia a lo largo del eje X_i que va desde la intersección del eje Z_{i-1} con el eje X_i hasta el origen del sistema i -esimo, en el caso de articulaciones giratorias.

- α . Es el ángulo de separación del eje Z_{i-1} y el eje Z_i medido en un plano perpendicular al eje X_i , utilizando la regla de la mano derecha.”³³

2.1.1 Cinemática directa por medio de las matrices de Transformación Homogénea para las extremidades de las piernas (12GDL).

Para realizar el planteamiento de las matrices de transformadas de cada articulación, se supone un sistema donde una de las piernas es fija, y la cadena cinemática de ésta pierna y de la otra respectivamente son cadenas cinemáticas abiertas, tiendo este modelo, se plantea el sistema de coordenadas y planos según Denavit-Hartenberg

Dados los pasos del Algoritmo de Denavit-Hartenberg se obtuvo la Ilustración 9, en la Ilustración 8 se observa un Renderizado de las extremidades de las piernas, para su mejor visualización, los parámetros obtenidos se encuentran en la Tabla 1.

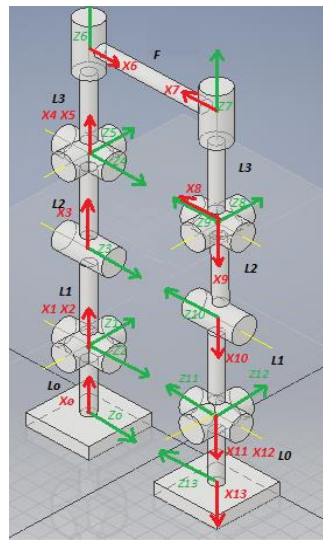
Ilustración 8. Render de extremidades inferiores Robot Bioloid



Fuente: Propia, Solidworks®

³³ BARRIENTOS Antonio, PEÑIN Luis Felipe, BALAGUER Carlos, ARACIL Rafael. Fundamentos de Robótica. 2 ed. McGraw-Hill, 2007. 512 p. ISBN 9788448156367.

Ilustración 9. Esquema de sistema de ejes según método Denavit-Hartenberg para las extremidades inferiores



Fuente: Propia, Autodesk Inventor Professional.

Donde:

q : Ángulo de articulación rotacional.

L: Longitudes entre articulaciones.

Tabla 1. Tabla de parámetros Denavit-Hartenberg para las extremidades inferiores

<i>Coordenada</i>	θ	<i>d</i>	<i>a</i>	α°
1	0	0	<i>L0</i>	90
2	<i>q1</i>	0	0	-90
3	<i>q2</i>	0	<i>L1</i>	0
4	<i>q3</i>	0	<i>L2</i>	0
5	<i>q4</i>	0	0	90
6	<i>q5</i>	0	<i>L3</i>	90
7	<i>q6</i>	0	<i>F</i>	180
8	<i>q7</i>	<i>L3</i>	0	90
9	<i>q8</i>	0	0	90
10	<i>q9</i>	0	<i>L2</i>	0
11	<i>q10</i>	0	<i>L1</i>	0
12	<i>q11</i>	0	0	-90
13	<i>q12</i>	0	<i>L0</i>	90

Fuente: Propia.

Una vez obtenida la tabla de parámetros, se utiliza la matriz de transformación general ${}^{i-1}A_1$ para obtener las matrices de cada articulación

$${}^{i-1}A_1 = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & \alpha_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & \alpha_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Se realiza una función en Matlab para la obtención éstas matrices, a la cual se le ingresan los datos sección por sección y se obtiene cada matriz, las matrices obtenidas son:

$${}^0_pA_1 = \begin{bmatrix} 1 & 0 & 0 & L_0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1_pA_2 = \begin{bmatrix} \cos(q_1) & 0 & -\sin(q_1) & 0 \\ \sin(q_1) & 0 & \cos(q_1) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2_pA_3 = \begin{bmatrix} \cos(q_2) & -\sin(q_2) & 0 & L_1 * \cos(q_2) \\ \sin(q_2) & \cos(q_2) & 0 & L_1 * \sin(q_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^3_pA_4 = \begin{bmatrix} \cos(q_3) & -\sin(q_3) & 0 & L_2 * \cos(q_3) \\ \sin(q_3) & \cos(q_3) & 0 & L_2 * \sin(q_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4_pA_5 = \begin{bmatrix} \cos(q_4) & 0 & \sin(q_4) & 0 \\ \sin(q_4) & 0 & -\cos(q_4) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^5_bA_6 = \begin{bmatrix} \cos(q_5) & 0 & \sin(q_5) & L_3 * \cos(q_5) \\ \sin(q_5) & 0 & -\cos(q_5) & L_3 * \sin(q_5) \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^6_pA_7 = \begin{bmatrix} \cos(q_6) & \sin(q_6) & 0 & F * \cos(q_6) \\ \sin(q_6) & -\cos(q_6) & 0 & F * \sin(q_6) \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^7_pA_8 = \begin{bmatrix} \cos(q_7) & 0 & \sin(q_7) & 0 \\ \sin(q_7) & 0 & -\cos(q_7) & 0 \\ 0 & 1 & 0 & L_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^8_pA_9 = \begin{bmatrix} \cos(q_8) & 0 & \sin(q_8) & 0 \\ \sin(q_7) & 0 & -\cos(q_8) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^9_pA_{10} = \begin{bmatrix} \cos(q_9) & -\sin(q_9) & 0 & L_2 * \cos(q_9) \\ \sin(q_9) & \cos(q_9) & 0 & L_2 * \sin(q_9) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0_pA_{11} = \begin{bmatrix} \cos(q_{10}) & -\sin(q_{10}) & 0 & L1 * \cos(q_{10}) \\ \sin(q_{10}) & \cos(q_{10}) & 0 & L1 * \sin(q_{10}) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{11}_pA_{12} = \begin{bmatrix} \cos(q_{11}) & 0 & -\sin(q_{11}) & 0 \\ \sin(q_{11}) & 0 & \cos(q_{11}) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{12}_pA_{13} = \begin{bmatrix} \cos(q_{12}) & 0 & -\sin(q_{12}) & L0 * \cos(q_{12}) \\ \sin(q_{12}) & 0 & \cos(q_{12}) & L0 * \cos(q_{12}) \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Dado que el sistema es simétrico y se analiza el sistema para una caminata, el robot siempre poseerá un pie apoyado en el suelo, mientras el otro realiza el movimiento pertinente, por ello se decide realizar la matriz de transformada de del eslabón 0 al 6, es decir la transformada A_6 .

La matriz de transformada A_6 viene dada por el producto entre todas las matrices anteriores, en este caso la de 0_pA_1 hasta 5_pA_6 , se tiene entonces la siguiente ecuación.

$$A_6 = {}^0_pA_1 * {}^1_pA_2 * {}^2_pA_3 * {}^3_pA_4 * {}^4_pA_5 * {}^5_pA_6$$

$$A_6 = H = \begin{bmatrix} R & d \\ 0^T & 1 \end{bmatrix} H = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Donde R corresponde a la matriz de rotación y d a la de translación.³⁴

El siguiente sistema de ecuaciones para una pierna de 6 DOF , se obtienen con Matlab.

Sistemas de Ecuaciones para una pierna de 6DOF

$$r_{11} = C_1 C_{234} C_5 - S_1 S_5$$

$$r_{21} = S_{234} C_5$$

$$r_{31} = S_1 C_{234} C_5 + C_1 S_5$$

$$r_{12} = C_1 C_{234}$$

³⁴ OLLERO Baturone Aníbal. ROBÓTICA Manipuladores y robots móviles. Publicado por Coedición Alfaomega y Marcombo. Barcelona , España. 2001 . 447 pag. ISBN: 970-15-0758-4.

$$r_{22} = -C_{234}$$

$$r_{32} = S_1 S_{234}$$

$$r_{13} = C_1 C_{234} S_5 + S_1 C_5$$

$$r_{23} = S_{234} S_5$$

$$r_{33} = S_1 C_{234} S_5 - C_1 C_5$$

$$d_x = L_3 C_1 C_{234} C_5 - L_3 S_1 S_5 + L_2 C_1 C_{23} + L_1 C_1 C_2 + L_0$$

$$d_y = L_3 S_{234} C_5 - L_2 S_{23} + L_1 S_2$$

$$d_z = L_3 S_1 C_{234} C_5 + L_3 C_1 S_5 + L_2 S_1 C_{23} + L_1 S_1 C_2$$

2.1.2 Cinemática directa por medio de las matrices de Transformación Homogénea para un brazo (3GDL).

Dado que en el robot Bioloid presenta movimiento de los brazos no dependientes (Brazo1 y Brazo2 por aparte), se realiza el análisis de un brazo con cadena cinemática abierta como se muestra en la figura, también se establece como base fija el hombro justo antes del origen del eje de rotación (Zo).

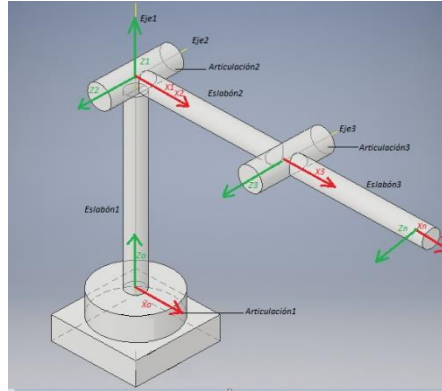
Dados los pasos del Algoritmo de Denavit-Hartenberg se obtuvo la ilustración 11, en la ilustración 10 se observa un Renderizado de una extremidad superior, para su mejor visualización, los parámetros obtenidos se encuentran en la Tabla 2

Ilustración 10. Renderizado de una extremidad superior robot Bioloid



Referencia: Autores, Solidworks.

Ilustración 11. Esquema de sistema de ejes según método Denavit-Hartenberg para una extremidad superior



Referencia: Autores, Autodesk Inventor Professional.

Tabla 2. Tabla de parámetros Denavit-Hartenberg para una extremidad superior.

Fuente: Propia.

<i>Coordenada</i>	θ	d	a	α°
1	$q1$	$L1$	0	0
2	$q2$	0	0	90
3	$q3$	0	$L2$	0

Una vez obtenida la tabla de parámetros, se utiliza la matriz de transformación general ${}^{i-1}A_1$ ³⁵ para obtener las matrices de cada articulación

$${}^{i-1}A_1 = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & \alpha_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & \alpha_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

³⁵ OLLERO Baturone Aníbal. ROBÓTICA Manipuladores y robots móviles. Publicado por Coedición Alfaomega y Marcombo. Barcelona, España. 2001. 447 pag. ISBN: 970-15-0758-4.

Se realiza una función en Matlab para la obtención éstas matrices, a la cual se le ingresan los datos sección por sección y se obtiene cada matriz, las matrices obtenidas son:

$${}^0_bA_1 = \begin{bmatrix} \cos(q1) & -\sin(q1) & 0 & 0 \\ \sin(q1) & \cos(q1) & 0 & 0 \\ 0 & 0 & 1 & L1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1_bA_2 = \begin{bmatrix} \cos(q2) & 0 & \sin(q2) & 0 \\ \sin(q2) & -\cos(q2) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2_bA_3 = \begin{bmatrix} \cos(q3) & -\sin(q3) & 0 & L2 * \cos(q3) \\ \sin(q3) & \cos(q3) & 0 & L2 * \sin(q3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

MATRIZ DE TRANSFORMADA

La matriz de transformada $A_3=$ viene dada por el producto entre todas las matrices anteriores, en este caso la de 0_bA_1 hasta 2_bA_3 , se tiene entonces la siguiente ecuación.

$$A_3 = {}^0_bA_1 * {}^1_bA_2 * {}^2_bA_3$$

$$A_3 = H = \begin{bmatrix} R & d \\ 0^T & 1 \end{bmatrix} H = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La matriz que relaciona el sistema de coordenadas de referencia con el efector final está dada por:

$$A_3 = \begin{bmatrix} \cos(q1 + q2) * \cos(q3) & -\cos(q1 + q2) * \sin(q3) & \sin(q1 + q2) & l2 * \cos(q1 + q2) * \cos(q3) \\ \sin(q1 + q2) * \cos(q3) & -\sin(q1 + q2) * \sin(q3) & -\cos(q1 + q2) & l2 * \sin(q1 + q2) * \cos(q3) \\ \sin(q3) & \cos(q3) & 0 & l1 + l2 * \sin(q3) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Sistemas de Ecuaciones para un brazo de 3DOF

$$r_{11} = \cos(q1 + q2) * \cos(q3)$$

$$r_{21} = \sin(q1 + q2) * \cos(q3)$$

$$r_{31} = \sin(q3)$$

$$r_{12} = -\cos(q1 + q2) * \sin(q3)$$

$$r_{22} = -\sin(q1 + q2) * \sin(q3)$$

$$r_{32} = \cos(q3)$$

$$r_{13} = \sin(q1 + q2)$$

$$r_{23} = -\cos(q1 + q2)$$

$$r_{33} = 0$$

$$d_x = l2 * \cos(q1 + q2) * \cos(q3)$$

$$d_y = l2 * \sin(q1 + q2) * \cos(q3)$$

$$d_z = l1 + l2 * \sin(q3)$$

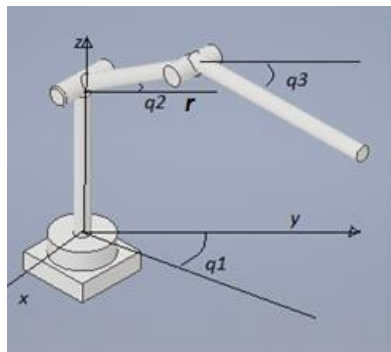
2.2 CINEMÁTICA INVERSA

Según Barrientos³⁶, La cinemática inversa permite obtener las variables articulares de posición y orientación para una configuración final conocida.

Para hallar los ángulos en una posición final deseada se tiene la cinemática inversa, se plantea el método geométrico, ya que este método es visualmente más entendible.

Se plantea la cinemática inversa para un brazo de 3 GDL como se muestra en la Ilustración 12.

Ilustración 12 Esquema de una extremidad superior para análisis cinemática inversa



Fuente: Autores, Solidworks.

³⁶ BARRIENTOS Antonio, PEÑIN Luis Felipe, BALAGUER Carlos, ARACIL Rafael. 2 ed. McGraw-Hill, 2007. 512 p. ISBN 9788448156367.

Para hallar los ángulos finales de un brazo de 3GDL por el método geométrico, se realizan operaciones trigonométricas, se supone k como la hipotenusa de un triángulo rectángulo, para q1 se tiene:

$$q1 = \arctg\left(\frac{py}{px}\right) \quad 2.2.c.1$$

Suponiendo que los eslabones q2 y q3 están situados en un plano utilizando teorema de coseno se tiene:

$$k^2 = px^2 + py^2 \quad (2.2.c.2)$$

$$k^2 + p_z^2 = l_2^2 + l_3^2 + 2 * l_2 * l_3 * \cos(q3) \quad (2.2.c.3)$$

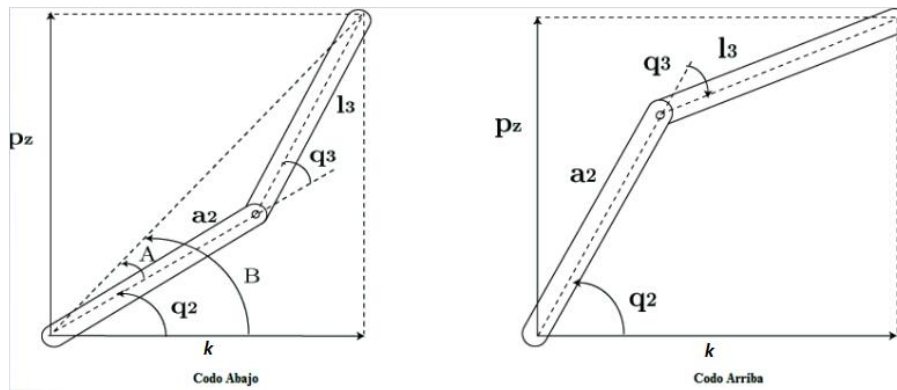
$$\text{Sen}(q3) = \frac{\mp\sqrt{1-\cos^2q_3}}{\cos q_3} \quad (2.2.c.4)$$

$$q3 = \arctg\left(\frac{\pm\sqrt{1-\cos(q_3)^2}}{\cos q_3}\right) \quad (2.2.c.5)$$

$$\text{Cos}(q3) = \frac{px^2 + py^2 + p_z^2 - l_2^2 - l_3^2}{2 * l_2 * l_3} \quad (2.2.c.6)$$

La ilustración 13 muestra las dos posibilidades para la segunda articulación del brazo para la cinemática inversa en la extremidad del segundo eslabón.

Ilustración 13. Esquema posibilidades de posición eslabones brazo para la cinemática inversa



Fuente: RAMIREZ Benavides Kryscia Daviana, Cinemática inversa del robot, disponible en <http://www.kramirez.net/wp-content/uploads/2012/04/CinematicaInversaRobot.pdf>

Las fórmulas obtenidas para el análisis fueron las siguientes:

$$q_2 = B - A \quad (2.2.c.7)$$

$$B = \arctg\left(\frac{p_z}{k}\right) = \arctg\left(\frac{p_z}{\pm\sqrt{(p_x)^2 + (p_y)^2}}\right) \quad (2.2.c.8)$$

$$A = \arctg\left(\frac{l_3 \cdot \sin(q_3)}{l_2 + l_3 \cdot \cos(q_3)}\right) \quad (2.2.c.9)$$

$$q_2 = \arctg\left(\frac{p_z}{\pm\sqrt{(p_x)^2 + (p_y)^2}}\right) - \arctg\left(\frac{l_3 \cdot \sin(q_3)}{l_2 + l_3 \cdot \cos(q_3)}\right) \quad (2.2.c.10)$$

El análisis anterior es el resultado del análisis por trigonometría del sistema, de igual manera se consultaron diferentes fuentes para su desarrollo.^{37 38}

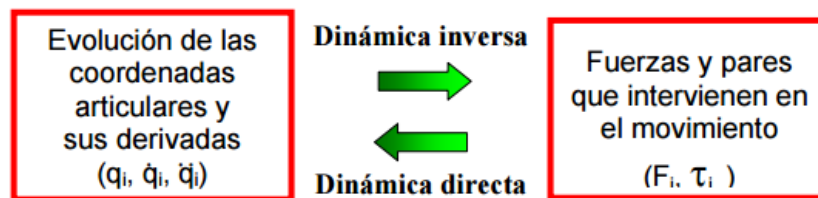
2.3 CINÉTICA DEL BRAZO DE 3GDL

³⁷RAMIREZ Benavides Kryscia Daviana. Cinemática inversa del robot. Escuela de Ciencias de la Computación e Informática. Costa Rica. 56 p. Disponible en < <http://www.kramirez.net/wp-content/uploads/2012/04/CinematicaInversaRobot.pdf>>

³⁸ RODRIGUEZ Zambrano. Modelo cinemático y control de un brazo robótico imprimible. Grado en ingeniería electrónica y automática industrial. España. Universidad Carlos III de Madrid. Disponible en < http://e-archivo.uc3m.es/bitstream/handle/10016/16924/TFG_Juan_Carlos_Rodriguez_Zambrana.pdf?sequence=3>

“Estudio de la relación que existe entre las fuerzas que actúan sobre un cuerpo, su masa y el movimiento de este mismo. La cinética se utiliza para predecir el movimiento ocasionado por fuerzas dadas, o para determinar las fuerzas que se requieren para producir un movimiento específico.”³⁹

Ilustración 14. Dinámica inversa y directa



Fuente: ABDALA Castillo Salomón, ÑECO Caberta Raúl. Caracterización de un robot manipulador articulado.

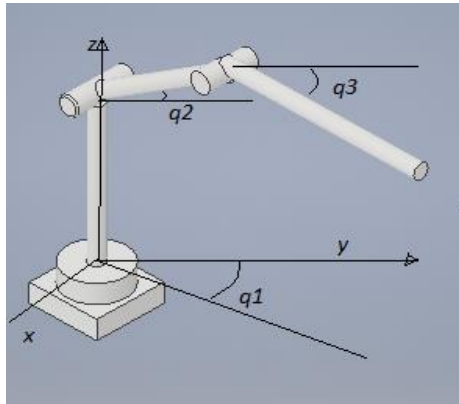
En este capítulo se aplica dinámica inversa para el conocimiento de fuerzas y pares que debe realizar un brazo para ejecución de movimientos determinados.

La dinámica se puede resolver por ecuaciones Newtonianas, pero al aumentar el número de GDL en un robot, la dificultad aumenta y algunos parámetros son más difíciles de visualizar. El método de LaGrange es el más adecuado si se quiere plantear la cinética de una cadena cinemática compleja o de numerosos grados de libertad. En esta sección se realiza una conceptualización general del procedimiento a seguir para la obtención de ecuaciones de torque respecto a la configuración de un brazo de 3GDL, teniendo en cuenta las consideraciones nombradas a continuación:

Como primer paso se plantea la ilustración 15 y la ilustración 16 donde se ilustran los ángulos pertinentes a cada eslabón respecto al eje del inmediatamente anterior, los eslabones y los ejes y los diagramas de cuerpo libre.

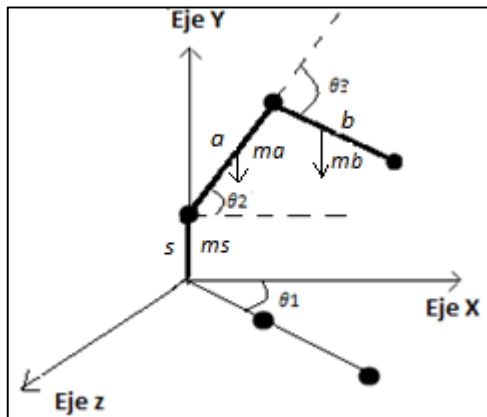
³⁹ BEER, JOHNSTON, CORNWELL. Mecánica vectorial para ingenieros dinámica. 9 ed. Mc Graw Hill. 2010. 1362 pag. ISBN: 978-607-15-0261-2

Ilustración 15. Modelado de brazo robot con ángulos correspondientes en cada eje para desarrollo de modelo dinámico.



Fuente: Autores, Autodesk Inventor Professional.

Ilustración 16. Diagrama de Cuerpo libre para dinámica de un brazo de 3GDL



Fuente: Autores.

Para realizar el análisis dinámico de un sistema se planteó anteriormente el diagrama de cuerpo libre del sistema y la notación a usar en la tabla 3.

Tabla 3. Notación usada en el cálculo de la dinámica.

Variable	Concepto
g	constante de gravedad
m_s	masa eslabón s (base)
m_a	masa eslabón a
m_b	masa eslabón b
s	Longitud eslabón s (base)
a	Longitud eslabón a
b	Longitud eslabón b
c_a	Distancia centro de masa s a a
c_b	Distancia centro de masa a a b
I_s	Inercia s
I_a	Inercia a
I_b	Inercia b
θ_1	Ángulo del eje x a z de la proyección de los eslabones
θ_2	Ángulo entre a y x
θ_3	Ángulo entre a y b
T	Energía Cinética
U	Energía Potencial

Fuente: Autores.

Quando se realiza el planteamiento de las ecuaciones dinámicas se tienen en cuenta las siguientes **consideraciones**:

- Sistema de estructura rígida con sus centros de masa concentrados en la mitad de cada eslabón, no se considera la deformación.
- Movimiento sin fricción: no se considera fricción en las articulaciones ni en los mecanismos, respecto al torque de entrada del servomotor.
- Se consideran solamente los parámetros de distancia, masa e inercias, visto desde un plano y sin considerar fuerzas centrífugas y de coriolis.

2.3.1 Energía cinética y potencial de eslabones.

En la Tabla 4, se calcula la energía cinética y potencial para cada eslabón de manera independiente, teniendo en cuenta la Ilustración 16.

Tabla 4 Tabla de cálculo de energías para eslabones

Cálculo de energías para los eslabones	
Energía Cinética	Energía potencial
<p>Eslabón S</p> <p>Debido al movimiento rotacional de la misma, la formula sería la siguiente</p> $K_s = \frac{1}{2} I_s \dot{\theta}_1^2 \quad (2.3.c.1)$ $I_s = m_s * s^2 \quad (2.3.c.2)$ $K_s = \frac{1}{2} * m_s * s^2 * \dot{\theta}_1^2 \quad (2.3.c.3)$	<p>Eslabón S</p> <p>La energía potencial en el eslabón s es igual a cero, esto es debido a que por más desplazamiento del eslabón en θ_s, la altura de la masa respecto al eje de coordenadas en el origen nunca cambiará, es decir la altura es igual a cero.</p> $P_s = 0 \quad (2.3.c.4)$
<p>Eslabón a</p> <p>El eslabón L_a presenta energía cinética rotacional y a su vez traslacional, debido a que la articulación de éste gira sobre su propio eje, mientras al mismo tiempo está conectado al eslabón inmediatamente anterior y estará afectado por el movimiento del eslabón L_s donde éste hará desplazar el eslabón.</p> $K_a = \frac{1}{2} m_a \left(\frac{a}{2}\right)^2 \dot{\theta}_2^2 + \frac{1}{2} m_a \left(\frac{a}{2} \cos \theta_2\right)^2 \dot{\theta}_1^2 \quad (2.3.c.5)$ $K_a = \frac{1}{2} m_a \frac{a^2}{4} \dot{\theta}_2^2 + \frac{1}{2} m_a \frac{a^2}{4} \cos^2 \theta_2 \dot{\theta}_1^2 \quad (2.3.c.6)$ <p>Se simplifica y se aplica la identidad $\cos^2 \theta = \left[\frac{1}{2} + \frac{1}{2} \cos 2\theta\right]$ en (2.3.c.6) así</p> $K_a = \frac{1}{8} m_a a^2 \dot{\theta}_2^2 + \frac{1}{8} m_a a^2 \left[\frac{1}{2} + \frac{1}{2} \cos 2\theta_2\right] \dot{\theta}_1^2 \quad (2.3.c.7)$ $K_a = \frac{1}{8} m_a a^2 \dot{\theta}_2^2 + \frac{1}{16} m_a a^2 \dot{\theta}_1 + \frac{1}{16} m_a a^2 \cos 2\theta_2 \dot{\theta}_1^2 \quad (2.3.c.8)$	<p>Eslabón a</p> <p>La energía potencial va a depender de la altura en 'y' del centroide del eslabón respecto al origen. Se tendría la longitud de altura del eslabón s, y la altura del centroide de a.</p> $P_a = m_a g \left(\frac{a}{2} \text{sen} \theta_2 + s\right) \quad (2.3.c.9)$

Eslabón b

Al igual que el Eslabón a, presenta energía cinética rotacional y traslacional, ya que está girando sobre su propio eje, y estará afectado por el movimiento del primer eslabón.

$$K_b = \frac{1}{2} m_b V_b^2 + \frac{1}{2} I_b \dot{\theta}_1^2 \quad (2.3.c.10)$$

La velocidad es equivalente a la suma de las posiciones derivadas, por tanto, se tiene:

$$V_b^2 = \dot{x}_b^2 + \dot{y}_b^2 \quad (2.3.c.11)$$

Se halla la posición en X de b respecto al origen

$$x_b = a \cos \theta_2 + \frac{1}{2} b \cos(\theta_2 + \theta_3) \quad (2.3.c.12)$$

Se deriva b de la ecuación (2.3.c.12)

$$\dot{x}_b = -a \sin \theta_2 \dot{\theta}_2 - \frac{1}{2} b \sin(\theta_2 + \theta_3) (\dot{\theta}_2 + \dot{\theta}_3) \quad (2.3.c.13)$$

Se eleva al cuadrado la ecuación (2.3.c.13)

$$\dot{x}_b^2 = a^2 \sin^2 \theta_2 \dot{\theta}_2^2 - ab \sin \theta_2 \sin(\theta_2 + \theta_3) (\dot{\theta}_2^2 + \dot{\theta}_2 \dot{\theta}_3) + \frac{1}{4} b^2 \sin^2(\theta_2 + \theta_3) (\dot{\theta}_2 + \dot{\theta}_3)^2 \quad (2.3.c.14)$$

Se halla la posición en Y de b respecto al origen

$$y_b = s + a \sin \theta_2 + \frac{1}{2} b \sin(\theta_2 + \theta_3) \quad (2.3.c.15)$$

$$\dot{y}_b = a \cos \theta_2 \dot{\theta}_2 + \frac{1}{2} b \sin(\theta_2 + \theta_3) (\dot{\theta}_2 + \dot{\theta}_3) \quad (2.3.c.16)$$

Elevando al cuadrado se tiene

$$\dot{y}_b^2 = a^2 \cos^2 \theta_2 \dot{\theta}_2^2 + ab \cos \theta_2 \cos(\theta_2 + \theta_3) (\dot{\theta}_2^2 + \dot{\theta}_2 \dot{\theta}_3) + \frac{1}{4} b^2 \cos^2(\theta_2 + \theta_3) (\dot{\theta}_2 + \dot{\theta}_3)^2 \quad (2.3.c.17)$$

Sustituyendo (2.3.c.14) y (2.3.c.17) en (2.3.c.11)

$$V_b^2 = a^2 \sin^2(\theta_2) \dot{\theta}_2^2 + ab \sin \theta_2 \sin(\theta_2 + \theta_3) (\dot{\theta}_2^2 + \dot{\theta}_2 \dot{\theta}_3) + \frac{1}{4} b^2 \sin^2(\theta_2 + \theta_3) (\dot{\theta}_2 + \dot{\theta}_3)^2 + a^2 \cos^2(\theta_2) \dot{\theta}_2^2 + ab \cos \theta_2 \cos(\theta_2 + \theta_3) (\dot{\theta}_2^2 + \dot{\theta}_2 \dot{\theta}_3) + \frac{1}{4} b^2 \cos^2(\theta_2 + \theta_3) (\dot{\theta}_2 + \dot{\theta}_3)^2 \quad (2.3.c.18)$$

Aplicando identidades trigonométricas se obtiene:

Eslabón b

$$P_2 = m_b g a \sin \theta_2 +$$

$$\frac{1}{2} m_b g b \sin(\theta_2 + \theta_3) + m_b g s \quad (2.3.c.26)$$

$\dot{x}_b^2 + \dot{y}_b^2 = a^2 \dot{\theta}_2^2 + ab(\dot{\theta}_2^2 + \dot{\theta}_2 \dot{\theta}_3) \cos \theta_3 + \frac{1}{4} b^2 (\dot{\theta}_2 + \dot{\theta}_3)^2$ <p style="text-align: right;">(2.3.c.19)</p> <p>Se tiene que la inercia en b es</p> $I_b = m_b X_b^2$ <p style="text-align: right;">(2.3.c.20)</p> <p>De la ecuación (2.3.c.12) se eleva al cuadrado</p> $X_b^2 = \left[a \cos \theta_2 + \frac{1}{2} b \cos(\theta_2 + \theta_3) \right]^2$ <p style="text-align: right;">(2.3.c.21)</p> <p>Aplicando álgebra se resuelve la ecuación (2.3.c.21)</p> $X_b^2 = a^2 \cos^2 \theta_2 + ab \cos \theta_2 \cos(\theta_2 + \theta_3) + \frac{1}{4} b^2 \cos^2(\theta_2 + \theta_3)$ <p style="text-align: right;">(2.3.c.22)</p> <p>Reemplazando (2.3.c.22) en (2.3.c.20)</p> $I_b = m_b \left(a^2 \cos^2 \theta_2 + ab \cos \theta_2 \cos(\theta_2 + \theta_3) + \frac{1}{4} b^2 \cos^2(\theta_2 + \theta_3) \right)$ <p style="text-align: right;">(2.3.c.23)</p> <p>Se reemplaza (2.3.c.23) y (2.3.c.19) en (2.3.c.10)</p> $K_b = \frac{1}{2} m_b \left[a^2 \dot{\theta}_2^2 + ab(\dot{\theta}_2^2 + \dot{\theta}_2 \dot{\theta}_3) \cos \theta_3 + \frac{1}{4} b^2 (\dot{\theta}_2 + \dot{\theta}_3)^2 \right] +$ $\frac{1}{2} m_b \dot{\theta}_1^2 \left(a^2 \cos^2 \theta_2 + ab \cos \theta_2 \cos(\theta_2 + \theta_3) + \frac{1}{4} b^2 \cos^2(\theta_2 + \theta_3) \right)$ <p style="text-align: right;">(2.3.c.24)</p> <p>Operando se tiene</p> $K_b = \frac{1}{2} m_b a^2 \dot{\theta}_2^2 + \frac{1}{2} m_b ab(\dot{\theta}_2^2 + \dot{\theta}_2 \dot{\theta}_3) \cos \theta_3 + \frac{1}{8} m_b b^2 (\dot{\theta}_2 + \dot{\theta}_3)^2 +$ $\frac{1}{4} m_b a^2 \dot{\theta}_1^2 + \frac{1}{4} m_b a^2 \cos 2\theta_2 \dot{\theta}_1^2 + \frac{1}{2} m_b ab \cos \theta_2 \cos(\theta_2 + \theta_3) \dot{\theta}_1^2 +$ $\frac{1}{16} m_b b^2 \dot{\theta}_1^2 + \frac{1}{16} m_b b^2 \cos 2(\theta_2 + \theta_3) \dot{\theta}_1^2$ <p style="text-align: right;">(2.3.c.25)</p>	
--	--

Fuente: Autores.

2.3.2 Ecuación Lagrange.

Una vez obtenidas las ecuaciones de energía de los eslabones, se procede a realizar el cálculo de Lagrange.

La ecuación de Lagrange es la derivada de la diferencia de la suma de las energías que actúan en el sistema así:

(2.3.c.27)

$$L = \Sigma K_T - \Sigma P_T$$

(2.3.c.28)

$$L = (K_{s+} K_a + K_b) - (P_s + P_a + P_b)$$

$$\begin{aligned} \Sigma K_T = & \frac{1}{2} m_s s^2 \dot{\theta}_1^2 + \frac{1}{8} m_a a^2 \dot{\theta}_2^2 + \frac{1}{16} m_a a^2 \dot{\theta}_1^2 + \frac{1}{16} m_a a^2 \cos 2\theta_2 \dot{\theta}_1^2 + \\ & \frac{1}{2} m_b a^2 \dot{\theta}_2^2 + \frac{1}{2} m_b ab (\dot{\theta}_2^2 + \dot{\theta}_2 \dot{\theta}_3) \cos \theta_3 + \frac{1}{8} m_b b^2 (\dot{\theta}_2 + \dot{\theta}_3)^2 + \\ & \frac{1}{4} m_b a^2 \dot{\theta}_1^2 + \frac{1}{4} m_b a^2 \cos 2\theta_2 \dot{\theta}_1^2 + \frac{1}{2} m_b ab \cos \theta_2 \cos(\theta_2 + \theta_3) \dot{\theta}_1^2 + \\ & \frac{1}{16} m_b b^2 \dot{\theta}_1^2 + \frac{1}{16} m_b b^2 \cos 2(\theta_2 + \theta_3) \dot{\theta}_1^2 \end{aligned}$$

(2.3.c.29)

$$\Sigma P_T = m_a g \left(\frac{a}{2} \text{sen} \theta_2 + s \right) + m_b g a \text{sen} \theta_2 + \frac{1}{2} m_b g b \text{sen}(\theta_2 + \theta_3) + m_b g s \quad (2.3.c.30)$$

Ecuación de Lagrange

$$L = \Sigma K_T - \Sigma P_T$$

$$\begin{aligned} L = & \frac{1}{2} m_s s^2 \dot{\theta}_1^2 + \frac{1}{8} m_a a^2 \dot{\theta}_2^2 + \frac{1}{16} m_a a^2 \dot{\theta}_1^2 + \frac{1}{16} m_a a^2 \cos 2\theta_2 \dot{\theta}_1^2 + \\ & \frac{1}{2} m_b a^2 \dot{\theta}_2^2 + \frac{1}{2} m_b ab (\dot{\theta}_2^2 + \dot{\theta}_2 \dot{\theta}_3) \cos \theta_3 + \frac{1}{8} m_b b^2 (\dot{\theta}_2 + \dot{\theta}_3)^2 + \\ & \frac{1}{4} m_b a^2 \dot{\theta}_1^2 + \frac{1}{4} m_b a^2 \cos 2\theta_2 \dot{\theta}_1^2 + \frac{1}{2} m_b ab \cos \theta_2 \cos(\theta_2 + \theta_3) \dot{\theta}_1^2 + \\ & \frac{1}{16} m_b b^2 \dot{\theta}_1^2 + \frac{1}{16} m_b b^2 \cos 2(\theta_2 + \theta_3) \dot{\theta}_1^2 \\ & - m_a g \left(\frac{a}{2} \text{sen} \theta_2 + s \right) - m_b g a \text{sen} \theta_2 - \frac{1}{2} m_b g b \text{sen}(\theta_2 + \theta_3) - m_b g s \end{aligned}$$

(2.3.c.31)

Ecuación Euler-LaGrange

$$\tau = \frac{d}{dt} \frac{\delta \mathcal{L}}{\delta \dot{q}} - \frac{\delta \mathcal{L}}{\delta q} \text{ para } \theta_1, \theta_2, \theta_3 \quad (2.3.c.32)$$

El análisis anterior se realiza respecto a los conocimientos básicos de energías y diagramas de cuerpo libre, y en base a diferentes documentos consultados. ^{40 41}

(**) Robocup evento para la promover la robótica y la investigación, ofreciendo un desafío público, pero un reto formidable. Para más información visite < <http://www.robocup.org/>>.

⁴⁰ ABDALA Castillo Salomón, ÑECO Caberta Raúl . Caracterización de un robot manipulador articulado. Maestro en ciencias en Ingeniería Mecatrónica. México. Centro Nacional de Investigación y Desarrollo Tecnológico. 2003.196 p. disponible en < <https://www.cenidet.edu.mx/subaca/web-mktr/submenus/investigacion/tesis/3-4%20Salomon%20Abdala%20Castillo%20-%20Raul%20%C3%91eco%20Caberta.pdf>>

⁴¹ CIRERA Eduardo A, BRAVO Gerardo A, FILIPIGH Jorge E. Modelado de un brazo robótico mediante el método de Lagrange. República Argentina. Universidad Nacional del Nordeste comunicaciones científicas y tecnológicas. 2005.Disponible en < <http://www.unne.edu.ar/unnevieja/Web/cyt/com2005/7-Tecnologia/T-027.pdf>>

3. PROCESAMIENTO DE IMAGEN PARA LA DETECCIÓN E IDENTIFICACIÓN DE OBJETOS DE INTERÉS

El objetivo principal de este proyecto requiere que los robots desempeñen un rol de delantero y arquero respectivamente y este desarrollo sea implementado un OVA, por ello se plantea hacer un análisis del espacio sobre el cual se encuentran los robots. Como posible solución inicialmente se plantea trabajar con una tarjeta-embebido debido que en la modalidad *humanoid* de la *Robocup* (**), los robots deben ser autónomos y contar con habilidad para correr, caminar, patear el balón, además de poseer equilibrio, tener una visión del campo y de los demás jugadores, los procesos de percepción y control se deben realizar de manera remota, para ello se les implementa de manera local sobre ellos mismos como se muestra en la ilustración 17.

Las tareas nombradas anteriormente requieren alto procesamiento, por ende, componentes individuales para cada robot, que se convierten en un problema de limitación de procesamiento y de costos. Ya que se están controlando dos robots (no cooperativos) e igualmente sobre un campo de visión controlado, se plantea otra modalidad de fútbol de robots, también de la *Robocup* llamada *Small Size*.

Ilustración 17. Robots de la Modalidad *Humanoid* de *Robocup*



Fuente: Robocup sitio web oficial. Disponible en < <http://www.robocup.org/>>

Small Size, consiste en un sistema de visión estandarizada que provee dos cámaras, éstas están ubicadas sobre el campo de juego y los robots en el campo controlados por computadores, que a su vez poseen una comunicación inalámbrica directa con los robots.

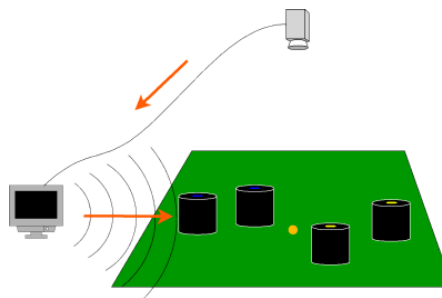
Los robots utilizados en la competencia *Small Size* se muestran en la ilustración 18, y el entorno dentro del cual los robots se desenvuelven en dicha modalidad se muestra en la ilustración 19.

Ilustración 18. Robots de la modalidad *Small Size* de *Robocup*



Fuente: Página web Carnegie Mellon University, disponible en: <https://www.cmu.edu/news/stories/archives/2015/july/robocup-win.html>

Ilustración 19. Entorno modalidad *Small Size* de *Robocup*



Fuente: Small size Robot league página web, disponible en http://robocupssl.cpe.ku.ac.th/_media/dataflow.png?cache=

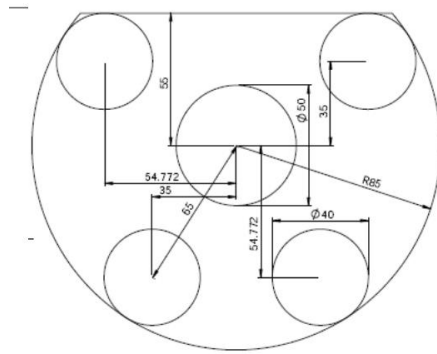
El proyecto está basado en el entorno y el tipo de visión por máquina de esta modalidad. Los robots poseen una visión global del terreno y son controlados inalámbricamente por medio de un computador, como se muestra en la ilustración 19.

El tipo de visión por máquina parte también de la modalidad *Small Size*, el reconocimiento de objetos se realiza por medio de marcas de color que se encuentran la parte superior del robot como muestra la ilustración 19, los robots para esta modalidad se encuentran estandarizados, según el manual de reglas oficial de la robocup “Small Size”, se reglamenta que el color para el robot debe ser negro o gris oscuro, cualquiera de los dos mate. Deben existir cinco círculos en la superficie del robot, donde van los colores. El círculo del centro posee el color

distintivo del equipo de 50 mm , el cual puede ser azul o amarillo. Los demás círculos son identificativos para cada robot y son clave para la ubicación espacial de cada robot, el gráfico correspondiente a la reglamentación de marcas para 2015 se encuentra en la ilustración 20.

Para este proyecto se utilizaron identificadores de robot de dos colores: azul y verde. Con lo que se puede determinar la ubicación y orientación para cada robot.

Ilustración 20. Medidas estandarizadas marcas de color modalidad *Small Size* de *Robocup*.



Fuente: Small size página web, disponible en http://robocupssl.cpe.ku.ac.th/_media/rules:ssl-rules-2015.pdf

Ya que en la modalidad *Small Size* es necesario utilizar un computador a continuación, se realiza una comparativa costo-beneficio-rendimiento de los posibles dispositivos a para las dos modalidades como se muestra en la tabla 5.

Tabla 5. Comparativa: costo-beneficio-rendimiento.

Característica	Modalidad	
	Humanoid	Small Size
Procesamiento	Raspberry Pi3 (1.2GHz)	Compute Stick (up 1.83 GHz)
Ubicación espacial	Si	Si
Número de Cámaras	2	1
RAM	1G LPDDR2 (900)MHZ	2G LPDDR3L (1333) MHZ
Almacenamiento	0 Gb+ Micro SD	32 Gb + Micro SD
Número de equipos	Raspberry pi3 (x2)	Compute Stick Intel (x1)
Precio equipos	Raspberry Pi3 \$ 160.000 Cámara PiCam \$ 80.000	Compute Stick \$ 300.000 Cámara web Logitech c170 \$ 70.000
Precio total	\$ 410.000	\$ 370.000

Fuente: Autores y mercado libre.

Dada la comparativa en la tabla 5, se opta por el *compute stick* de la modalidad *Small Size* para aplicar al proyecto, ya que dentro del planteamiento presenta las siguientes características.

- ✓ Costo bajo, posee un costo menor a su comparativa.
- ✓ Eficaz, el software posee más procesamiento por ende es mejor para algoritmos multitarea, en este caso es necesario, debido a que se procesará la imagen, y al mismo tiempo se están enviando datos a los robots, el desempeño del software para éstas tareas debe ser alto si se quiere una reacción rápida y exacta de los robots.
- ✓ Eficiente, Se emplean menos recursos para cumplir las tareas asignadas.

- **Software**

Para el procesamiento de imagen de la detección e identificación de objetos, se utiliza lenguaje de programación Python, y se trabaja en el ambiente de Python 2.7.12 para Windows como se observa en la ilustración 21, ya que es una de las versiones más utilizadas por los desarrolladores y presenta mayor compatibilidad con versiones anteriores de OpenCV 3.1 y otras librerías. En este caso se hace un enfoque a OpenCV, que es una librería de fuente abierta siendo usada para visión artificial, se emplea la versión 2.4.13, y otras librerías adicionales de procesamiento de imagen, graficadoras y de operaciones matemáticas.

Ilustración 21. Entorno Python 2.7.12

```
Principal.py - C:\Users\innacroft\Dropbox\Proyecto de grado\Capitulo2.PROCESAMIENTO L...
File Edit Format Run Options Window Help
#librerias
import cv2
import math
import serial, time
#librerias propias
import AuxiliarCentroides #Importa archivo de codigo *.py
import Gira_Robot
#def codigoAdicional():

def principal():
    global Delantero
    global AngPelado
    global AngPeladoI
    global ArqueroDeltanero
    global AnguloIzq
    global Histeresis
    global CONTADOR
    CONTADOR=0
    ArqueroDeltanero = [(0,0),(0,0)]
    global ArqueroDeltaneroAngulo
    ArqueroDeltaneroAngulo = [0,0]
    global pelota
    pelota = (0,0)
    MedicionAng = [(0,0),(0,0)]
    arduino = serial.Serial("COM3", 97600)
    #####
    captura = cv2.VideoCapture(0)
    captura.set(3,640) #creamos que de ancho se podia 720 pero en realidad maxim
    captura.set(4,360)
    while(1):
        for i in range(0,2):
            valido = captura.read()
            #cv2.imshow('Camera', imagen)
            #imagen = imagen[0:360,0:720]
            if valido: #Para esperar que la camera se inicialice y tenga frames va
                centroides, centroides, masks, mask, mask=AuxiliarCentroides.m
                if ((len(centroides)>0) & (len(centroides)>0) & (len(centroides)==1)):
                    Le:1 Col:0
```

Fuente: Autores.

- **Hardware**

El entorno de los robots se realiza en un entorno controlado, los componentes se muestran en la siguiente ilustración 22.

Ilustración 22. Entorno controlado real



Fuente: Autores.

En la tabla 6 se listan los componentes del entorno controlado de la ilustración 22.

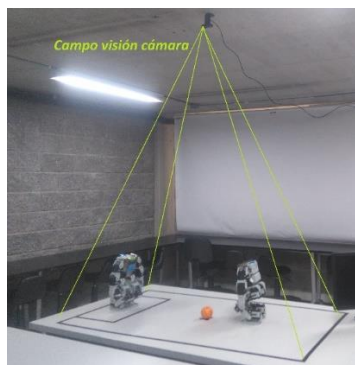
Tabla 6. Componentes entorno controlado

No.	Componente
1	Compute Stick
2	Arduino
3	Cámara
4	Robot Arquero
5	Robot Delantero
6	Periféricos de visualización y control
7	Pelota
8	Demarcación campo de Juego

El desarrollo del programa se realiza en un *compute stick Intel* con características ya planteadas en la Tabla 5. Las imágenes se extraen mediante una cámara web Logitech c170, ubicada sobre el campo de los robots, captando así la totalidad del área como se ilustra en la ilustración 23.

El área de juego de los robots se encuentra en un entorno controlado, la cámara siempre estará a una altura de 1,56m perpendicular a la superficie como se muestra en la Ilustración 23. El campo del robot es de color **blanco**, las marcas de delimitación de la cancha son de color negro, las dimensiones de la cancha son de 135cm x 76cm, que corresponden a la apertura de la cámara en “x” “y” “y” respectivamente. La luz ambiental es proporcionada por 8 bombillos de tubo fluorescente.

Ilustración 23. Campo de visión cámara

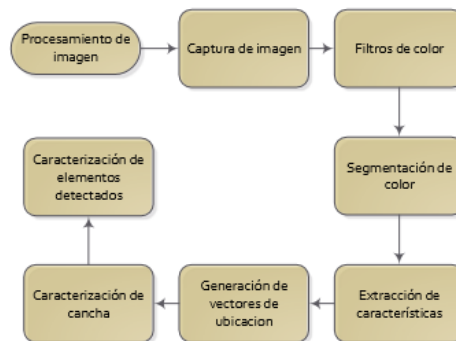


Fuente: Autores.

3.1 Fases para el procesamiento de imagen

Una vez seleccionado el Software y Hardware a utilizar, se plantea en la ilustración 24, la rutina de procesamiento de imagen con el fin de detectar y ubicar espacialmente los robots y la pelota en la cancha.

Ilustración 24 Rutina de procesamiento de imagen.

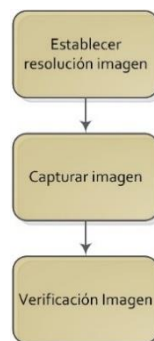


Fuente: Autores. Microsoft Office Visio®

3.1.1 Captura de imagen.

La captura de imágenes con la cámara en Opencv de Python se realiza siguiendo el esquema de la ilustración 25.

Ilustración 25. Esquema de captura de imágenes



Fuente: Autores. Microsoft Office Visio®

Como primer paso, para realizar la detección de los objetos se necesita obtener la imagen donde se encuentra el objeto de interés, antes de capturar la imagen se debe establecer una configuración que se ajuste al propósito.

Por ello se configura la resolución de 640x 360 píxeles, debido a que se requiere un formato 16:9, y ésta resolución cumple con el ancho máximo de píxeles, y el alto se adecúa de acuerdo al siguiente cálculo:

$$\text{Factor de Formato} = \frac{16}{9} = 1.7777 \quad 3.1.1.c.1$$

Para aplicar el anterior formato a la captura de video, se tiene como ancho máximo 640 píxeles, y en relación con la fórmula anterior, se calcula el alto, de la siguiente manera

$$\text{Altura de fotograma} = \frac{\text{Ancho de fotograma}}{\text{Factor de Formato}} \quad 3.1.1.c.2$$

$$\text{Altura de fotograma} = \frac{640}{1.7777} = 360 \quad 3.1.1.c.3$$

Por lo tanto, la resolución utilizada para la captura de video es de 640x 360.

La primera línea del código recibe el video en la cámara asignada, y se configura la resolución anterior. Debido a que en algunos casos la cámara necesita unos segundos para iniciar se establece un ciclo *while*, que establezca una condición y evite que los primeros *frames* entren vacíos al programa.

3.1.2 Filtros de Color.

La imagen de entrada debe ser pre-procesada aplicando filtros de color como se muestra en la Ilustración 26.

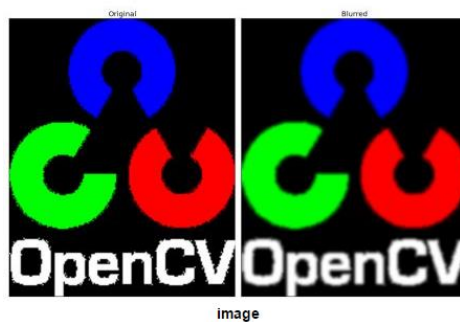
Ilustración 26. Esquema de filtros de color



Fuente: Autores. Microsoft Office Visio®

Una vez obtenida la imagen en la ...sección 3.1.1..., se obtiene una imagen en escala RGB, la cual es filtrada mediante un filtro Gaussiano para eliminar posible ruido ocasionado por variaciones como ruido en digitalización y variación de luz, la ilustración 27 muestra un ejemplo de este tipo de filtro.

Ilustración 27. Ejemplo de Filtro Gaussiano para una imagen aleatoria

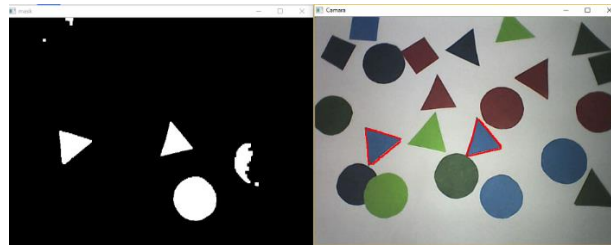


Fuente: OpenCv 3.1.0 docs, disponible en <
http://docs.opencv.org/3.1.0/d4/d13/tutorial_py_filtering.html>

Posteriormente, se realiza la conversión a modelo de color HSV, se declaran los rangos máximos y mínimos dentro de los cuales se va a seleccionar el color a

procesar, en este caso, se opta por los colores azul, verde y tonos rojizos; se elige HSV debido a que posee menos variabilidad en cuanto a otros modelos de color, aunque es dependiente al cambio de luz, es menos sensible y ocupa un error más bajo a la hora de seleccionar diferentes matices, en este caso se eligen colores que dentro de sus matices no presentan similitud y lograr así una mejor distinción. En la ilustración 28, se muestra el modelo HSV aplicado al color azul.

Ilustración 28. Imagen de HSV aplicado a figuras triangulares azules



Fuente: Autores. Python 2.7.12.

Se seleccionan los colores azul y verde para las marcas de color en los robot y rojo para la pelota.

Debido a que el color rojo presenta dos rangos en la escala de HSV como se muestra en la Ilustración 29. Se debe realizar dos máscaras independientes de color para rojo: una para valores mínimos y otra para valores máximos, se suman con la función de Opencv "cv2.add". para obtener finalmente toda la gama de rojos.

Ilustración 29. Escala de color HSV



Fuente: Answer Opencv <<http://answers.opencv.org/question/28899/correct-hsv-inrange-values-for-red-objects/>>

3.1.3 Segmentación de color.

Ilustración 30. Esquema de segmentación de color aplicado al proyecto



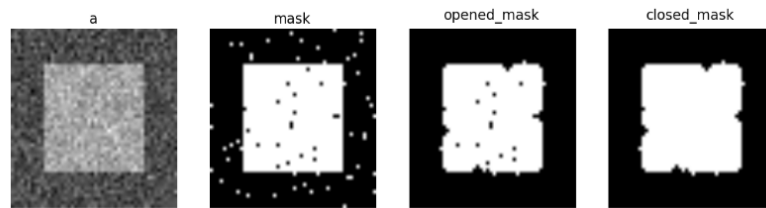
Fuente: Autores. Microsoft Office Visio®

Cuando se seleccionan los colores a identificar, el siguiente paso es realizar operadores para segmentar la imagen y eliminar ruido.

En este proceso se utilizan las máscaras de los colores obtenidas anteriormente (HSV para tonos rojizos, HSV para el azul y HSV para el verde), de manera que se puedan manipular y mejorar.

Se aplica un comando dentro de OpenCv, conocido como *threshold*, para binarizar la imagen, de manera que el objeto de interés sea blanco y el fondo que se discrimina, sea negro; continuamente se realiza una operación de apertura, una operación morfológica que contiene una erosión seguida de dilatación, esto es para eliminar ruido en la imagen. Con la dilatación se eliminan los pixeles con pocos vecinos, para eliminar puntos o defectos en la imagen que pueden suponer ruido, pero la imagen se aprecia más delgada. La dilatación por su parte agrega pixeles a los sobrantes, volviéndola a su estado natural sin ruido, pero con una imagen poco parecida a la original; el cierre por su parte realiza una dilatación seguida de erosión para lograr una morfología más semejante a la inicial. En la ilustración 31, se posee la imagen inicial, una máscara binarizada y las operaciones de apertura y cierre.

Ilustración 31. Ejemplo de prueba para las operaciones morfológicas de apertura y cierre



Fuente: scipy-lectures, disponible en < <http://www.scipy-lectures.org/intro/scipy.html>>

Una vez y teniendo en cuenta los bordes suavizados, se recurre a un filtro que convierte la imagen en una imagen binaria teniendo en cuenta la diferencia de gradientes entre pixeles vecinos, ésta línea de código convierte la imagen en una imagen Binaria, con fondo negro y el objeto de interés en blanco.

3.1.4 Extracción de características.

Posteriormente a la segmentación del objeto, se extraen las características distintivas de cada uno. Principalmente se extraen los centroides de las figuras verdes y las figuras azules, lo que permite realizar operaciones matemáticas con las coordenadas de los mismos, se encuentra este primer paso en la ilustración 32.

Ilustración 32. Secuencia extracción de características



Fuente: Autores. Microsoft Office Visio®

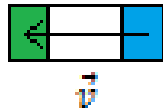
Para lograr una ubicación espacial distintiva para el robot y la pelota, se operan estos centroides, de manera que se obtiene la ubicación espacial de cada objeto así:

- **Delantero y arquero**

Para la ubicación de cualquiera de los dos robots, se emplean marcas de color verde y azul para cada uno.

Las marcas de los robots se encuentran en la parte superior de los mismos, se traza un vector desde el centroide azul al verde, el cual comprenderá un vector de dirección y orientación correspondiente a la ubicación del robot como se ilustra en la ilustración 33.

Ilustración 33. Vector en la marca de cada robot



Fuente: Autores.

Se debe tener en cuenta que para la identificación de arquero y delantero las marcas deben cumplir con una distancia entre centroides comprendida en un rango de 32 a 40 píxeles, es decir la Magnitud del vector \vec{v} , inmediatamente el programa realiza la identificación, traza una línea azul desde el centroide azul al verde, como se muestra en la ilustración 34.

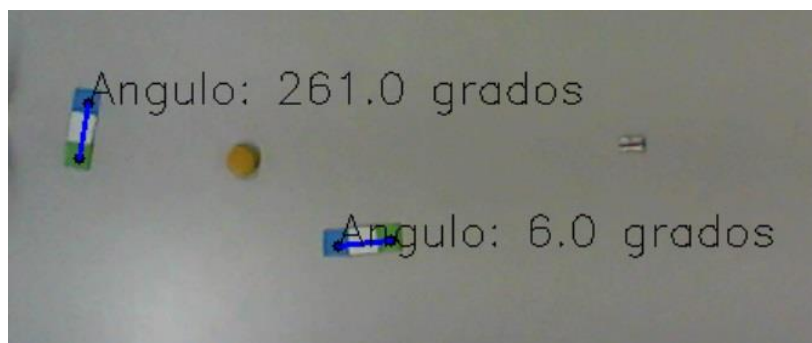
Ilustración 34. Ángulos formados entre los vectores de las marcas de los robots y el eje de coordenadas incorrecto.



Fuente: Autores. Python 2.7.12.

Las dos imágenes en la ilustración 34, muestran el ángulo de la magnitud entre los dos centroides; al realizar este proceso, se puede observar que un eje normal donde el eje “x” se encuentra en el eje horizontal hacia el este, y “y” en dirección vertical hacia el norte, la medida de los ángulos no corresponde, es por este motivo que se deduce que el eje “Y” está invertido, por este motivo se cambian los cuadrantes en que está el ángulo y así se obtiene una medición ajustada visualmente a los ángulos deseados, la medición de los nuevos ángulos se ilustra en la ilustración 35.

Ilustración 35. Ángulos formados entre los vectores de las marcas de los robots y el eje de coordenadas correcto.

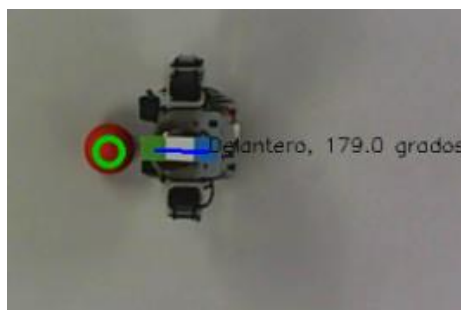


Fuente: Autores. Python 2.7.12.

- **Pelota**

La distinción de la pelota se hace sólo con el color naranja de la misma y se le ordena al programa que trace un círculo en el área detectada, como se muestra en la ilustración 36.

Ilustración 36. Distinción pelota naranja en campo



Fuente: Autores. Python 2.7.12.

Una de las restricciones del algoritmo es la presencia de la pelota dentro de la cancha para el funcionamiento del programa, es decir que únicamente los robots jugarán fútbol con la presencia de la pelota.

3.1.5 Caracterización de cancha.

Para realizar la cancha, es necesario realizar la caracterización del entorno virtual con el real, en este caso la conversión de pixeles a centímetros, con una altura de la cámara a 1,56m respecto a la superficie de la cancha. Inicialmente se mide la apertura de la cámara en el eje “x”, y en el eje “y”, las medidas y sus respectivos cálculos se presentan a continuación:

$$135cm = 640px$$

$$? cm = 1px$$

$$\frac{135cm}{640px} * 1px = 0.21cm$$

3.1.6.c.1

$$76cm = 360px$$

$$? cm = 1px$$

$$\frac{76cm}{360px} * 1px = 0.21cm$$

3.1.6.c.2

La equivalencia por tanto será

$$1px = 0.21cm$$

$$1cm = 4.76px$$

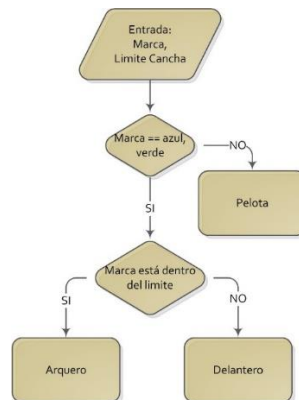
3.1.6.c.3

Se traza una cancha virtual de línea azul con medidas de 33,6cm por 56cm, al costado lateral izquierdo.

3.1.8 Caracterización de elementos detectados

La caracterización de los elementos dentro del desarrollo del proyecto está dada por el siguiente flujograma en la ilustración 37.

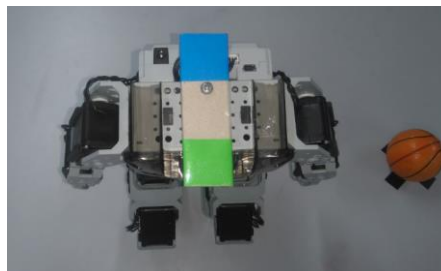
Ilustración 37. Caracterización de arquero, delantero y pelota



Fuente: Autores. Microsoft Office Visio®

Por medio de marcas de color es posible detectar e identificar elementos como en este caso, los robots, la marca de los robots es de color azul y verde y la de la pelota es de color naranja como se ilustra en la ilustración 38, De igual manera la marca siempre supondrá la dirección del robot, por tal motivo siempre estará fija teniendo en cuenta que el centroide verde es la parte delantera del robot.

Ilustración 38. Marcas de robot y pelota naranja

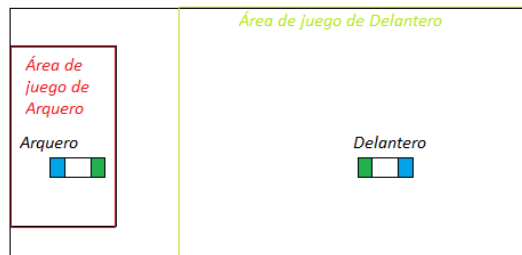


Fuente: Autores.

Para lograr la distinción entre arquero y delantero se toma el área de arco como distintivo, como se muestra en la ilustración 39, el arquero siempre debe estar dentro del área del arco, mientras el delantero va a encontrarse en cualquier lugar

del campo que no sea el arco, de la misma manera se le ordena al programa que muestre el nombre de Delantero o Arquero respectivamente.

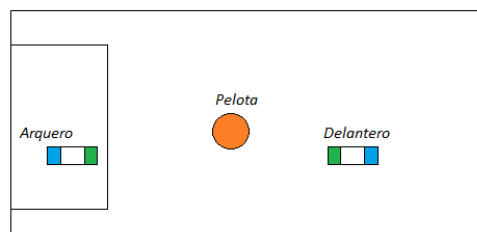
Ilustración 39. Áreas de juego arquero y delantero



Fuente: Autores.

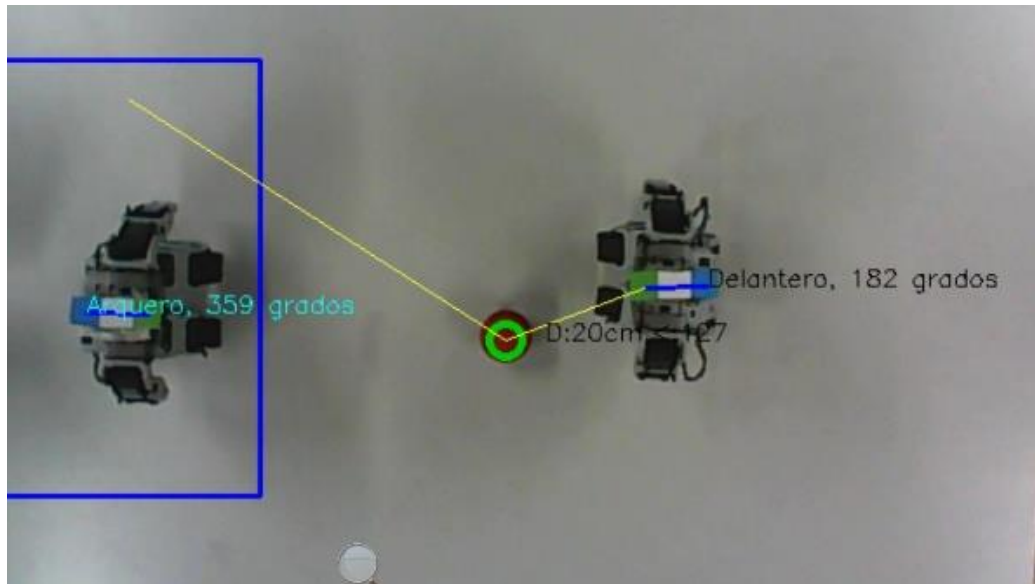
De ésta forma se tendría la distinción de cada uno de los objetos visto gráficamente en la ilustración 40 y realmente en la ilustración 41.

Ilustración 40. Esquema de objetos con su respectivo nombre en la cancha



Fuente: Autores.

Ilustración 41. Fotografía de objetos con su respectivo nombre en el campo de juego



Fuente: Autores, Python 2.7.3.12

Como se ilustra en la fotografía anterior, ilustración 41, se realiza la identificación de tres objetos: el arquero, el delantero y la pelota en el campo de juego.

4.COMUNICACIÓN

Para lograr que los robots ejecuten las tareas deseadas (arquero y delantero de futbol), debe existir un protocolo de comunicación definido y estructurado entre los robots y un computador.

Por tal motivo se plantean los siguientes ítems:

- a. Selección de tecnología de comunicación.
- b. Análisis del protocolo de transmisión de datos.
- c. Diseño de un circuito impreso el cual establece la relación física entre los módulos de comunicación y el controlador.
- d. Diseño de algoritmo de envío de datos hacia los robots; de esa forma los robots pueden ejecutar los movimientos al mismo tiempo.

Los ítems se desarrollan a continuación.

4.1 SELECCIÓN DE LA TECNOLOGÍA DE COMUNICACIÓN

Los kits Bioloid cuentan con el CM-530, este soporta varias tecnologías de comunicación, así como se muestran en el ANEXO H.

El CM-530 internamente maneja la alimentación y las señales de comunicación a 3.3v y tiene un circuito con FTDI, es decir, un circuito que convierte señales TTL 3.3v a USB. Lo que permite conectar los diferentes módulos compatibles:

Como se puede observar en el ANEXO H el CM-530 cuenta con las siguientes tecnologías:

- Comunicación serial por mini cable USB.
- Comunicación Bluetooth
- Comunicación Zig-bee
- Comunicación infrarrojos

En la Tabla 7 se comparan las tecnologías de comunicación compatibles con el CM-530 de esa forma realizar la selección de la tecnología de comunicación más conveniente.

Tabla 7. Comparativa de tecnologías de comunicación compatibles con el CM-530

Tecnología de comunicación	Referencia	Ventajas	Desventajas
Comunicación serial por mini cable USB.	Cable mini USB	Comunica CM-530 con Windows para programación	El cable inhabilita la posibilidad del libre movimiento de los robot
Comunicación Bluetooth	BT-100/110A, BT-210, BT-410	Compatible con IOS, Android, Windows	Bluetooth no compatible con control RC-100A incluidos en los kits de robótica.
Comunicación Zig-bee	ZIG 100/110A	Compatible con micro-controladores. Por medio de un circuito se puede controlar múltiples robots. No susceptible a obstáculos	Programables con un circuito que solo el fabricante proporciona
Comunicación infrarrojos	IR receiver	Fácil utilización por medio del control RC-100	El espectro infrarrojo presenta ruidos por interferencia en el ambiente y es susceptible a obstáculos.

Fuente: Autores.

Los criterios de selección de la tecnología y la referencia son los siguientes:

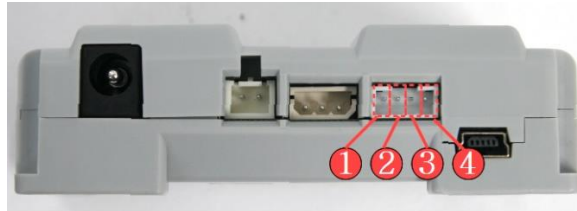
- Comunicación de tipo inalámbrica.
- Fidelidad de la transmisión.
- Comunicación con múltiples robots simultanea e independientemente.
- Accesible a los estudiantes de la universidad.

En referencia al primer criterio se descarta la tecnología USB; teniendo en cuenta el segundo criterio se descarta la tecnología infrarroja debido a que puede presentar interferencia con obstáculos y ruido debido a los espectros infrarrojos presentes en el ambiente provenientes de la luz solar; teniendo en cuenta el tercer criterio tanto los módulos bluetooth como los módulos Zig-Bee pueden comunicar los robot simultanea e independientemente, pero a partir del cuarto criterio debido a que los kits de robótica en la universidad cuentan con módulos Zig-Bee son la opción que cumple con todos los criterios.

4.2 EXTRACCIÓN DE DATOS (OSCILOSCOPIO Y VELOCIDAD)

La distribución de pines para la conexión del CM-530 en donde se conecta el módulo ZIG-110A se muestra en la ilustración 42.

Ilustración 42. Distribución de pines puerto de comunicaciones CM-530.



Fuente Robotis, disponible

en<<http://support.robotis.com/en/product/controller/cm530.htm>>

- 1- GND: Tierra (0v).
- 2- VDD: Alimentación de voltaje (2.7-3.3V).
- 3- RXD: Recepción de datos.
- 4- TXD: Transmisión de datos.

El fabricante propone una forma de conexión para el correcto funcionamiento de la comunicación utilizando estos módulos. Esto se refiere a que se deberían conectar de la siguiente forma:

La conexión del módulo Zig-110A con el controlador como viene por defecto se conecta como se enseña en la ilustración 43.

Ilustración 43. Conexión del módulo Zig-110A con el controlador CM-530



Fuente

Robotis,

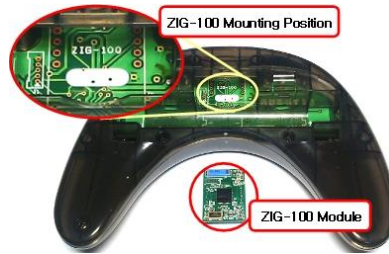
disponible

en<

<http://support.robotis.com/en/product/controller/cm530.htm>>

La conexión del módulo Zig-100 con el control RC-100 se realiza como se muestra en la ilustración 44.

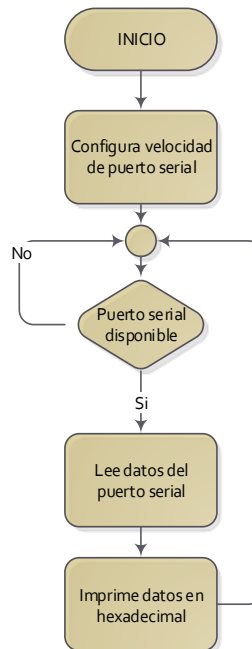
Ilustración 44. Conexión del módulo Zig-100 con el control RC-100.



Fuente Robotis, disponible en http://support.robotis.com/en/product/auxdevice/communication/rc100_manual.htm

Puesto que en las especificaciones del módulo no se hace mención a la velocidad de transmisión de datos es necesario realizar pruebas de captura de datos para lograr encontrar dicha velocidad. Se utiliza Arduino como medio receptor de señales TTL y por medio de un computador con Windows se realiza la visualización de datos que el puerto COM genera. Para ello es necesario realizar un algoritmo de conversión de datos ASCII a hexadecimal como se muestra en la ilustración 45, para capturarlos y visualizarlos en un archivo de texto para el posterior análisis.

Ilustración 45. Algoritmo de conversión de datos ASCII a hexadecimal



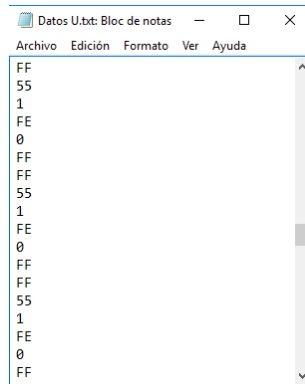
Fuente: Autores. Microsoft Office Visio®

Ahora se requiere hacer la conexión cableada para extraer los datos que el control envía al robot, por lo que se procede a conectar los pines de alimentación del módulo ZIG-110A a los pines 3.3v y GND del Arduino y el TX con el RX respectivamente.

La velocidad de transmisión es encontrada probando el código con cada una de las siguientes velocidades en baudios: 300, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400, 57600 y 115200. Finalmente, la velocidad con la que funciona correctamente y se notan patrones es 57600 baudios.

Los datos extraídos con esta velocidad de transmisión se observan en la ilustración 46.

Ilustración 46. Inicio de extracción de datos para tecla U del control RC-100



Fuente: Autores.

De la Ilustración 46 se puede diferenciar el patrón que se muestra a continuación en la Tabla 8, que corresponde cuando se presiona la tecla U del control remoto continuamente.

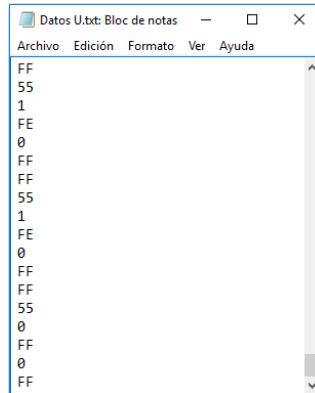
Tabla 8. Extracción de datos para tecla U del control RC-100

DATOS
FF
55
01
FE
00
FF

Fuente: Autores.

Adicionalmente a este patrón siempre al finalizar la comunicación se obtienen los datos que se muestran en la ilustración 47.

Ilustración 47. Finalización de extracción de datos para tecla U del control RC-100



Fuente: Autores.

De la ilustración 47, se puede diferenciar el patrón que se muestra a continuación en la Tabla 9, que corresponde cuando un botón del control se deja de presionar.

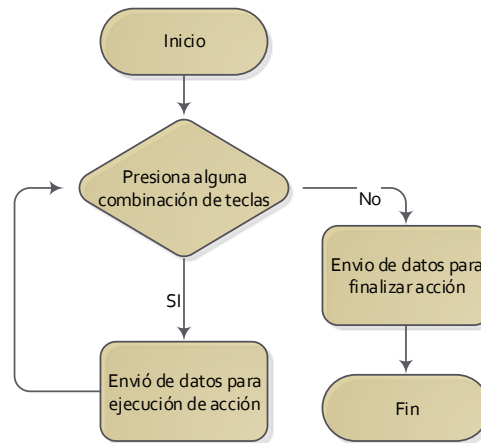
Tabla 9. Final de extracción de datos para tecla U del control RC-100

DATOS
FF
55
00
FF
00
FF

Fuente: Autores.

Con estos datos se puede deducir que el robot es estimulado por los datos de comunicación de la forma en que se muestra en la ilustración 48.

Ilustración 48. Diagrama algoritmo de comunicación del control RC-100

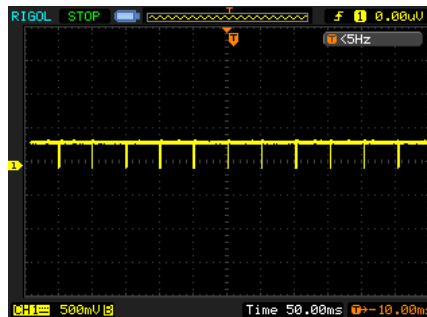


Fuente: Autores. Microsoft Office Visio®

La ilustración 48 se toma como algoritmo fundamental de funcionamiento de la comunicación.

Por otro lado, se realiza una medición de los datos de entrada del robot en un osciloscopio para determinar periodos de envío de los paquetes del que se puede observar el resultado en la ilustración 49.

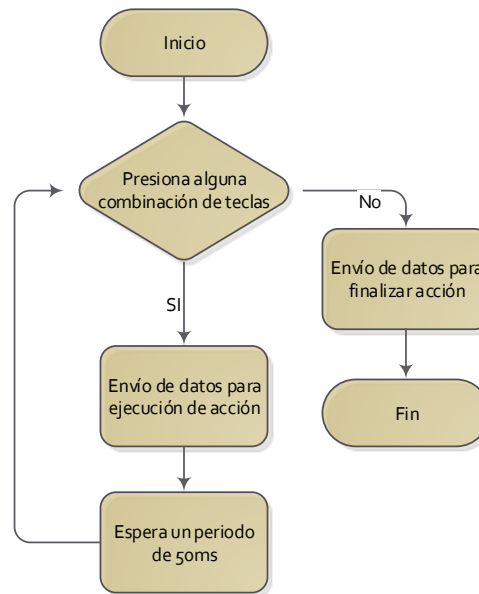
Ilustración 49. Medición de la señal de salida del módulo receptor Zig-100A



Fuente: Autores.

Como se puede observar, la imagen tiene 50ms de periodo por cuadro y el comportamiento de la señal es el mismo, esto quiere decir que se está enviando un paquete de datos cada 50mS, pieza clave para comunicar los Robots adecuadamente. A la luz de esta medición se recompone el algoritmo de funcionamiento de la comunicación con los robots como se muestra en la ilustración 50.

Ilustración 50. Algoritmo de comunicación del control RC-100 con periodo



Fuente: Autores. Microsoft Office Visio®

Para encontrar más comandos es necesario realizar un cálculo de lógica binaria para obtener el paquete de datos que se debe enviar al robot como se muestra en la ilustración 51.

Ilustración 51. Paquetes de envío de datos al robot Bioloid

FF	55	Data_L	~Data_L	Data_H	~Data_H
----	----	--------	---------	--------	---------

Fuente: Autores.

4.3 PROTOCOLO RECEPCIÓN Y ENVÍO DE DATOS (DEMUX)

Independientemente de la tecnología de comunicación se debe tener en cuenta los datos que se envían al robot para la respectiva acción. Debido a que se debe cumplir los roles de delantero y arquero es necesario tener control de los datos que se envían por separado para la actuación del sistema, esto, se puede llevar a cabo por medio de la codificación y decodificación de la información necesaria.

Robotis presenta como ejemplos de funcionamiento archivos de movimiento y tareas que juntos hacen cumplir una serie de acciones específicas. Para el humanoide tipo A se presentan dos tipos de archivos para ser cargados directamente en RoboPlus:

- **Archivos de Humanoide tipo A** presenta archivos de movimiento (*.mtn), para los servomotores.
- **Archivos de Soccer tipo A** presenta el tipo de movimiento a realizar (*.tsk), para realizar movimientos de fútbol de robot, se les conoce como archivos de tarea.

Estos archivos están disponibles en la página web oficial de Robotis⁴²

Con los archivos de Soccer tipo A se pueden efectuar los movimientos respectivos de un jugador de futbol, por medio del control RC-100 e implementando comunicación inalámbrica es posible realizar una serie de diversos comandos definidos por medio de combinaciones de botones del control.

En la página oficial de Robotis existe un manual que define según la combinación de botones del control, acciones definidas y programadas previamente.

En el control RC-100 por cada combinación de datos envía un paquete de datos codificado al robot, el robot lo decodifica y lo procesa.

El CM-530 maneja registros de 16-bit que definen una probabilidad e 65536 posibles comandos procesables por el mismo. Cuando se desea transmitir esta información la comunicación UART maneja paquetes de datos de 8-bit por lo que es necesario romper la cadena de datos de 16-bit en 2 y transmitir dos cadenas de 8-bit que corresponden a la parte alta y baja de la cadena de datos respectivamente. Para diferenciar entre los datos alto-bajo es necesario transmitir los datos de forma conveniente y organizada.

En los manuales virtuales proporcionados por el fabricante se encuentra la especificación técnica del control RC-100 que presenta la forma en que se transmiten los datos. Los paquetes de comunicación se transmiten como se observa en la ilustración 51, donde los datos necesarios como lo es un dato binario de 16-bit es codificado en formato hexadecimal y como se menciona anteriormente

⁴²Robotis. Página oficial de los robots Bioloid. Disponible en <http://support.robotis.com/en/product/bioloid/premiumkit/download/bioloid_premium_demo.htm>

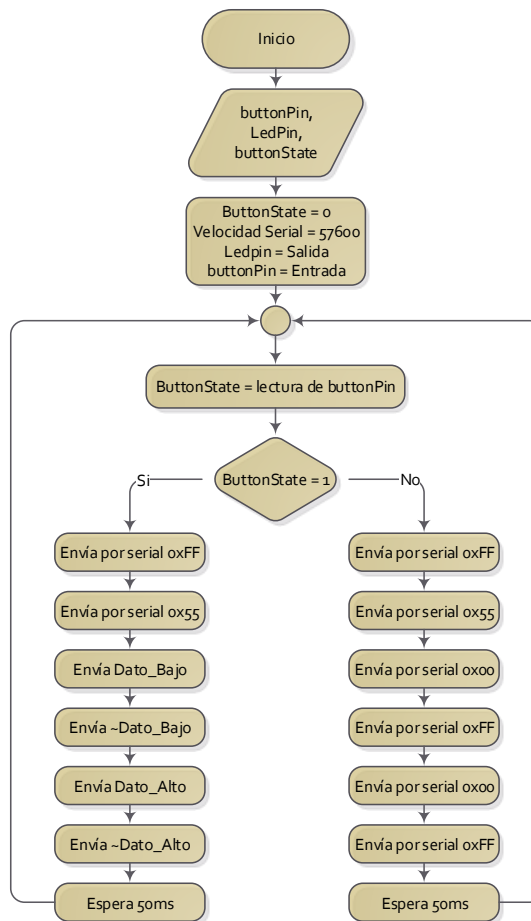
organizados de forma conveniente para verificar la validez del paquete. Los datos son organizados de la siguiente manera:

- Los datos 1 y 2 que son a 'FF' y '55' respectivamente, corresponden a la cabecera de un paquete de datos que viene.
- Los datos 'Data_L' y Data_H son la división en bytes del dato binario de 16-bit y que corresponden a la parte baja (Low) y alta (High) de dicho dato.
- Los datos '~Data_L' y '~Data_High' corresponden al complemento binario de los datos 'Data_L' y 'Data_H' respectivamente.

4.3.1 Comunicación con un módulo ZIG-100

Con el protocolo definido, se diseña un algoritmo de envío de datos por medio de un micro-controlador, como se muestra en la ilustración 52, para ello se usa la herramienta de desarrollo Arduino al implementar el algoritmo.

Ilustración 52. Esquema algoritmo de comunicación Arduino

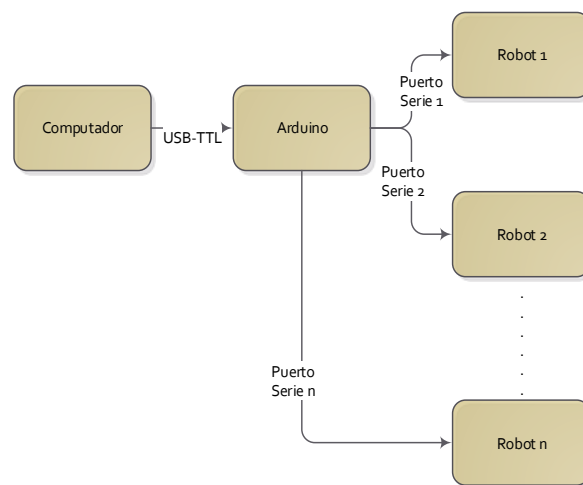


Fuente: Autores. Microsoft Office Visio®

El algoritmo de la Ilustración 52, sirve para comunicar un robot por medio del puerto serial principal de Arduino.

4.3.2 Comunicación con dos módulos ZIG-100 La comunicación con dos módulos requiere tener más de dos puertos seriales de comunicación lo que es un problema al momento de comunicar pues se requiere diseñar un *multiplexor* (*MUX*) de datos y un *demultiplexor* (*demux*) de datos como se puede observar en la Ilustración 53.

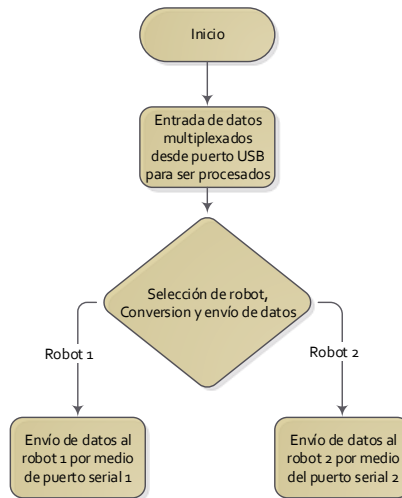
Ilustración 53. Diagrama usando Arduino como DEMUX



Fuente: Autores. Microsoft Office Visio®

Aplicando el funcionamiento de Arduino como *demux* para un sistema con dos robots se obtiene un diagrama de flujos como se muestra en la Ilustración 54.

Ilustración 54. Algoritmo usando Arduino como *demux*



Fuente: Autores. Microsoft Office Visio®

Es posible comunicar dos o más dispositivos por medio de puerto serial con Arduino utilizando la librería incluida e importable en el software Arduino 1.6.12 *SoftwareSerial.h* que permite crear varios puertos seriales y configurar los pines digitales como salidas de puerto serial lo que permite crear tantos puertos seriales como pines digitales tenga el Arduino. En este caso se trabaja con Arduino Uno, ya que cada puerto serial requiere la utilización de dos pines digitales. Teniendo en cuenta que se necesita usar dos robots, se requieren solo 4 pines digitales adicionando los dos pines digitales de comunicación que ya trae por defecto que son el 0 y el 1.

Para controlar el manejo de datos se tiene una variable de entrada que corresponde al número de robot que se va a usar para crear los algoritmos y enviar datos a cada uno de los robots por separado. De donde se puede definir lo siguiente:

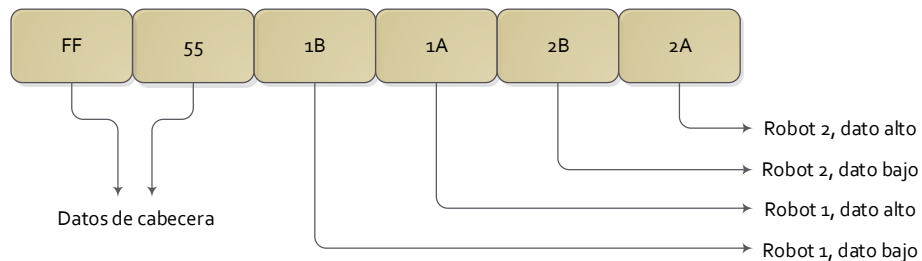
- Robot 1: Delantero
- Robot 2: Arquero

Se diseña un protocolo propio de entrada de datos a Arduino para seleccionar el robot y a la vez enviar los datos. Este, consiste en utilizar una cabecera de dos bytes al inicio y después de eso los datos de los distintos robots. Puesto que los números necesarios para recepción y envío se recomponen por medio de un número binario de 16-bit por cada dato enviado corresponde a dos bytes. Por lo tanto:

2 datos de cabecera + 2 datos de robot1 + 2 datos de robot 2 = 6 datos por paquete

El diseño del protocolo y reconstrucción de datos se muestra en la Ilustración 55.

Ilustración 55. Diseño de protocolo de comunicación de entrada a Arduino.



Fuente: Autores. Microsoft Office Visio®

Por lo que se hace necesario crear un vector 6x1 de entrada de datos seriales en el que se debe cumplir que los dos primeros datos de cabecera sean iguales a FF,55 y a partir de ahí selecciona los datos del robot 1 y los datos del robot 2 para ser procesados. Cuando el algoritmo cumple los datos de cabecera funciona como un filtro, esto se hace para tener fidelidad en los datos de entrada, pues en el ambiente puede haber datos no deseados.

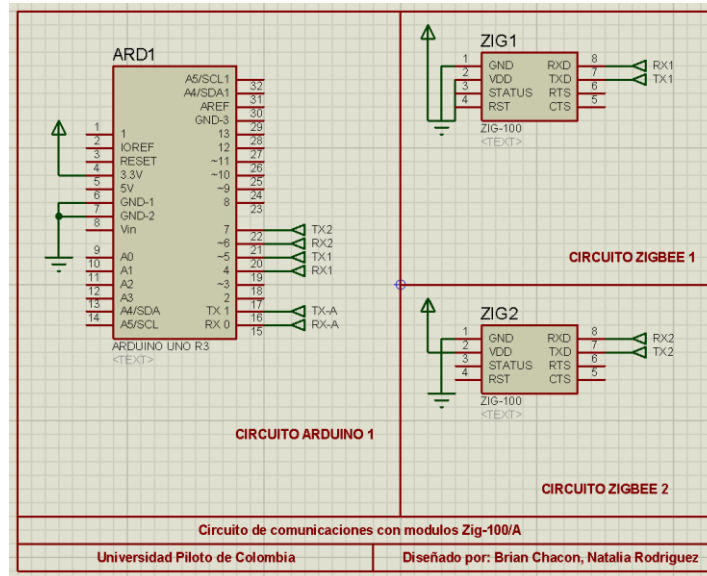
El diagrama de funcionamiento general que define la recepción y selección de los datos se muestra en el ANEXO I, que a su vez para el completo funcionamiento se diseñaron las subrutinas que se pueden observar en el ANEXO J y ANEXO K.

Debido a que el control de movimientos del robot se desarrolla en el lenguaje Python se usa el mismo para crear los scripts necesarios que ejecutan las acciones pertinentes y envían los datos de la forma adecuada a Arduino por puerto serial. Para implementar comunicación por puerto serial en Python se utiliza la librería pyserial 2.7 en Windows y se diseña el algoritmo que se muestra en el ANEXO L.

4.4 DISEÑO Y MONTAJE DE CIRCUITO DE COMUNICACIÓN

Para implementar los módulos ZIG-100 de Robotis se diseña un circuito de acople tipo *Shield* de Arduino que permita comunicar TX y RX para dos módulos como se puede observar en la ilustración 56.

Ilustración 56. Circuito de comunicaciones con módulos Zig-100



Fuente: Autores., PROTEUS 8.1

Al diseñar, fabricar y montar la tarjeta de circuito impreso con los 2 módulos Zig-100 y conectado el *Shield* a Arduino se obtiene la siguiente tarjeta alternativa y el circuito físico mostrado en la ilustración 57.

Ilustración 57. Montaje del circuito Shield con módulos Zig-100 en Arduino



Fuente: Autores.

5. ENSAMBLE Y PRUEBAS DE FUNCIONAMIENTO

En este capítulo se realiza la integración de dos de los capítulos anteriores, es decir Comunicación y procesamiento de imagen.

Para que los Robots ejerzan los papeles de delantero y arquero se plantea un algoritmo propio, que calcula los ángulos entre los objetos de la cancha para posteriormente definir la trayectoria computacional que debe seguir el jugador delantero para hacer gol; la trayectoria física que el robot debe ejecutar se define a partir del envío de acciones al mismo por medio del sistema de comunicaciones diseñado. Para cada jugador (arquero y delantero) se define el algoritmo de acciones respecto a la posición de la pelota.

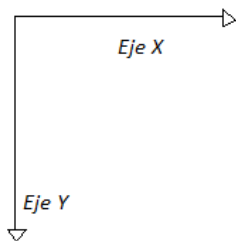
Establecido el algoritmo de acciones para cada jugador se procede a realizar pruebas de funcionamiento y de ser necesario modificar el algoritmo.

5.1 CÁLCULO DE TRAYECTORIAS

Calculo de trayectoria para jugador delantero.

Se plantea un algoritmo que calcula los ángulos de cada jugador y de la pelota dentro de la cancha y entre ellos respectivamente, inicialmente debido a que las imágenes de una cámara tienen el sistema coordenado como se muestra en la ilustración 58, los ángulos son calculados a partir de esos ejes.

Ilustración 58. Sistema 1, ejes coordenados iniciales

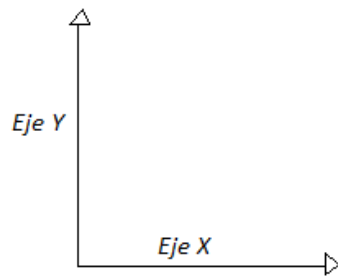


Fuente: Autores.

Por lo tanto, en primer lugar, se realiza un algoritmo que convierte los ángulos medidos desde el sistema coordenado de la ilustración 58 a ángulo medidos desde

el sistema coordenado de la ilustración 59, que corresponde a un sistema conocido y fácil de entender.

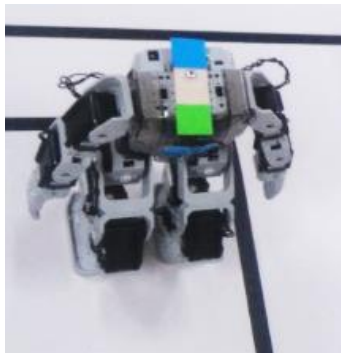
Ilustración 59. Sistema 2, ejes coordenados comunes



Fuente: Autores.

El análisis de la posición del robot delantero es a partir del centroide verde, como se explicó en capítulos anteriores, las marcas están predispuestas de manera para que el verde sea la posición delantera del robot como se muestra en la ilustración 60.

Ilustración 60. Marca de color en Robot

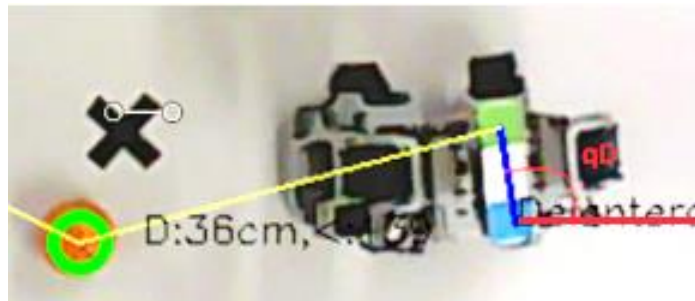


Fuente: Autores.

Dados los sistemas coordenados anteriores (ilustración 58 y 59) se plantean a continuación los ángulos requeridos y su respectiva ilustración, para la ubicación espacial de los robots respecto a la pelota y al ángulo de lanzamiento.

- **Ángulo de Delantero (qD)** ángulo entre la Magnitud del vector del centroide verde delantero y el sistema coordenado, la ilustración 61 demuestra el ángulo.

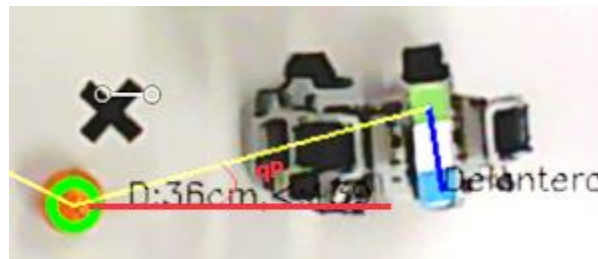
Ilustración 61. Representación ángulo robot delantero



Fuente: Autores., Python 2.7.12

- **Ángulo de Pelota (qP)** ángulo entre el vector que parte del centroide de la pelota al centroide verde del delantero y el sistema coordenado 2.

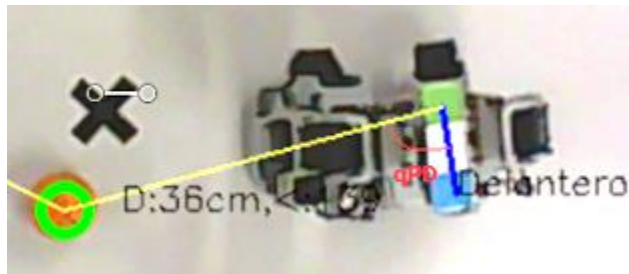
Ilustración 62. Representación ángulo pelota



Fuente: Autores., Python 2.7.12

- **Ángulo entre Pelota y delantero (qPD):** el ángulo es el resultante entre el ángulo del vector del centroide verde del delantero y el ángulo de la pelota

Ilustración 63. Representación ángulo Pelota delantero

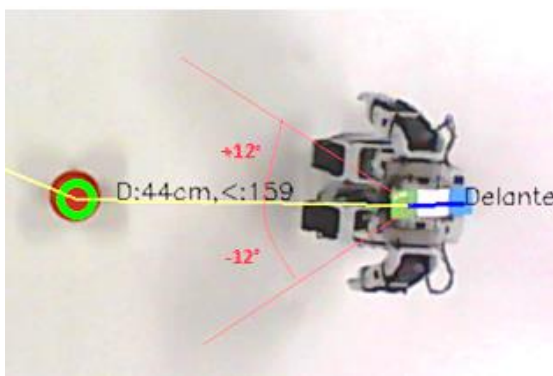


Fuente: Autores., Python 2.7.12

Realizando cálculos entre los ángulos descritos anteriormente, se emplea como condición de orientación que la pelota respecto al jugador delantero siempre se encuentre en dirección norte (en frente) como se observa en la ilustración 64, El ángulo qPD de la ilustración 63, corresponderá a 180° en la imagen siguiente.

Teniendo en cuenta un margen de error de $\pm 12^\circ$ de alineación, debido a que el ángulo de giro mínimo del robot es de 12.5° aproximadamente, para determinar el ángulo mínimo se realizan pruebas físicas de giro, enviando por medio de un computador la acción de girar a la izquierda o a la derecha, cada tres segundos, visualizando la medida en ángulos en la cámara.

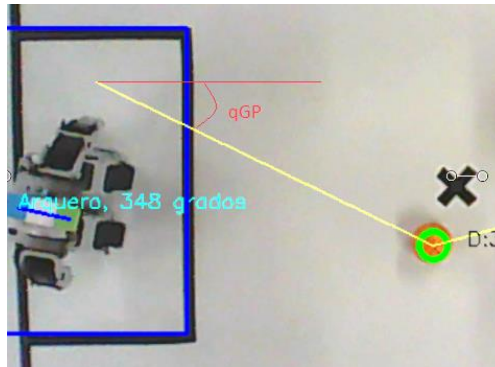
Ilustración 64. Representación rango de error giro robot



Fuente: Autores., Python 2.7.12

- **Angulo entre pelota y punto de gol (qGP)** el ángulo es el resultante entre el ángulo del vector del centroide de la pelota y el punto de gol.

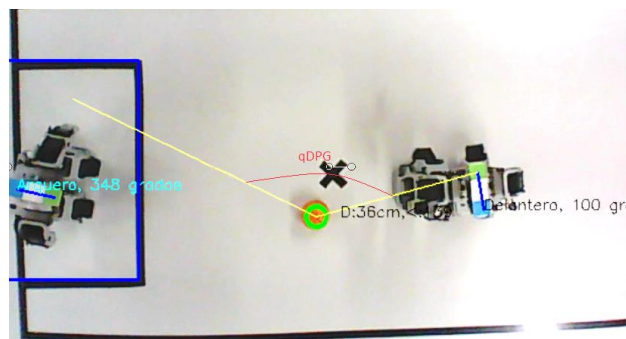
Ilustración 65. Representación ángulo pelota y punto de gol



Fuente: Autores., Python 2.7.12

- **Angulo entre el delantero, la pelota y el punto de gol (qDPG)** ángulo que está establecido entre los 3 puntos de interés: la pelota, el robot y el punto de gol, éste ángulo debe ser 180° para que el robot ejecute el lanzamiento.

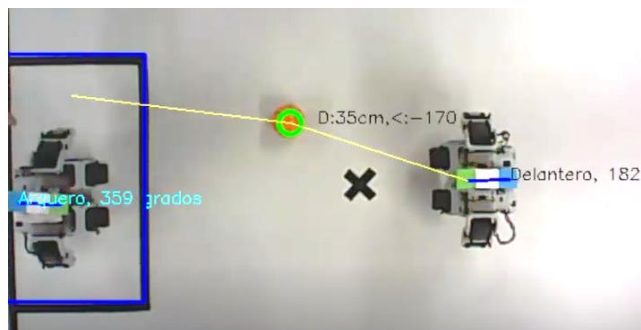
Ilustración 66. Representación ángulo total entre delantero, pelota y punto de gol



Fuente: Autores., Python 2.7.12

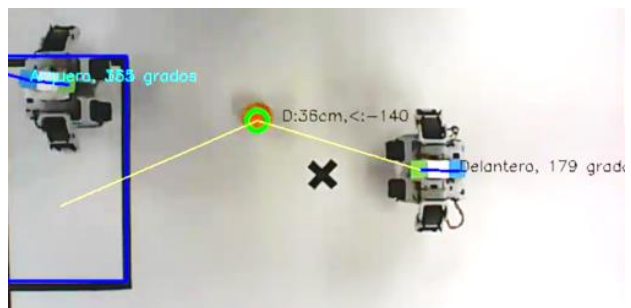
El ángulo de interés es (qDPG) delantero, pelota y punto de gol , ya que con éste ángulo se ejecutan los movimientos del robot, establece la trayectoria que debe alcanzar la pelota, que en este caso se estableció un cálculo entre el brazo y la demarcación de lateral de la cancha correspondiente, se calcula la mitad, éste es el punto de gol, es decir que siempre el gol va a apuntar hacia donde se encuentre mayor espacio disponible, habrán dos posibles alternativas como se muestran en las Ilustraciones 67 y 68.

Ilustración 67. Trayectoria de gol, espacio brazo izquierdo y arco mayor



Fuente: Autores., Python 2.7.12

Ilustración 68. Trayectoria de gol, espacio brazo derecho y arco mayor



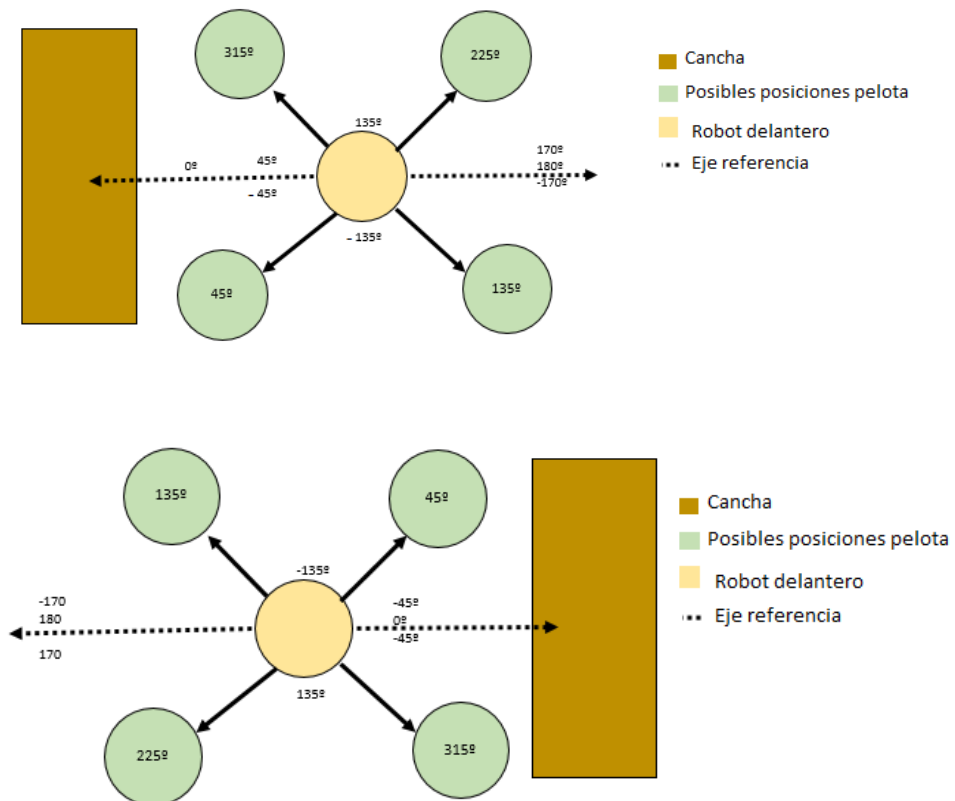
Fuente: Autores., Python 2.7.12

5.2 DEFINICIÓN PARÁMETROS PARA DELANTERO Y ARQUERO ANTES DE REALIZAR ACCIONES

Delantero

Se realizó un análisis de los posibles ángulos y los cuadrantes sobre los cuales era posible que la pelota y el delantero se encontraran como se muestra en la ilustración 69.

Ilustración 69. Posibles posiciones pelota-delantero

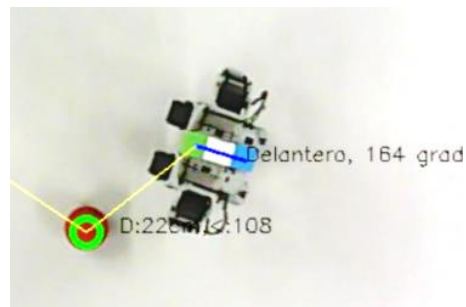


Fuente: Autores.

Se realizaron varios condicionales y se operaron los ángulos junto con el error de giro para la posición Delantero-Robot, existen dos casos diferentes dentro de los cuales el robot se encuentra en una posición diferente a 180° respecto a la pelota:

si está girado hacia la derecha el ángulo es negativo como se ilustra a continuación:

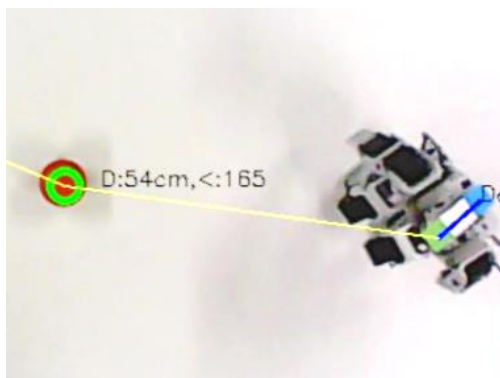
Ilustración 70. Robot girado derecha



Fuente: Autores., Python 2.7.12

Por otro lado, si se encuentra girado hacia la izquierda, el ángulo es positivo, se muestra en la ilustración 71.

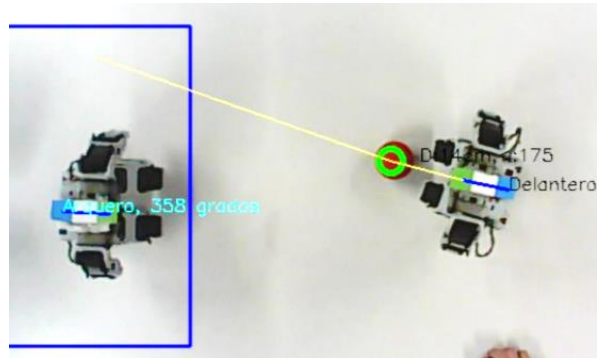
Ilustración 71. Robot girado izquierda



Fuente: Autores., Python 2.7.12

Una vez alcanzado el ángulo con la pelota, el robot delantero tiene que cumplir los 180° entre el ángulo de Delantero-Pelota y Punto de Gol. Para ello se cumple la misma condición del caso Delantero-Pelota, se plantea que, si el robot está a la derecha, implicará un ángulo negativo, y si está a la izquierda un ángulo positivo. De esta manera el ángulo final es 180° como se muestra en la siguiente imagen.

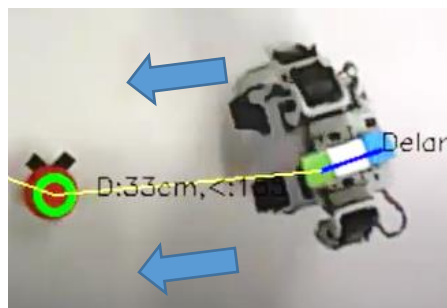
Ilustración 72. Delantero-Pelota-Punto de Gol alineados 180°



Fuente: Autores., Python 2.7.12

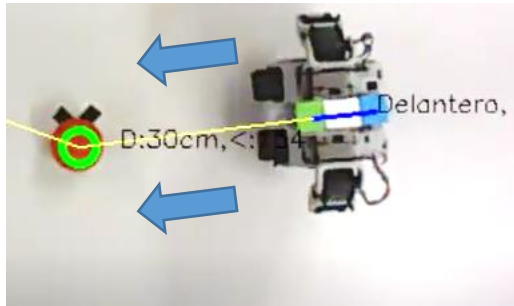
Después de cumplir las condiciones establecidas anteriormente, la última condición implementada es la distancia a la cual el robot debe patear el balón, en el programa se establece un rango de distancia de 20 centímetros o menos como se muestra en la transición de las imágenes a continuación.

Ilustración 73. Distancia delantero-pelota 33cm



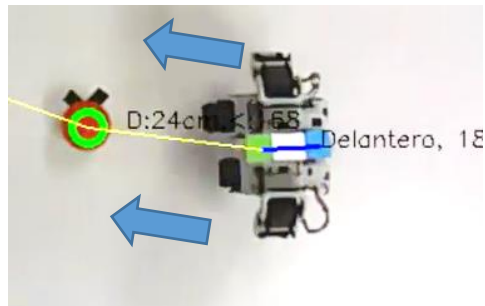
Fuente: Autores., Python 2.7.12

Ilustración 74. Distancia delantero-pelota 30cm



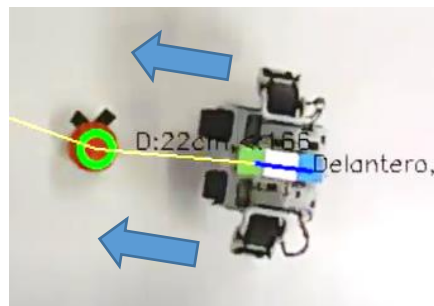
Fuente: Autores., Python 2.7.12

Ilustración 75. Distancia delantero-pelota 24cm



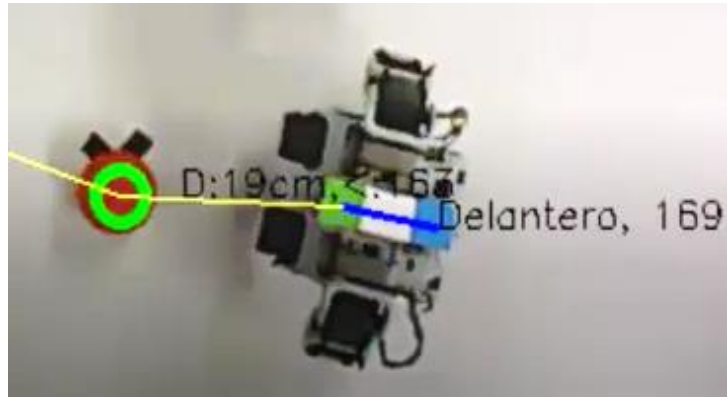
Fuente: Autores., Python 2.7.12

Ilustración 76. Distancia delantero-pelota 22cm



Fuente: Autores., Python 2.7.12

Ilustración 77. Distancia delantero-pelota 19cm

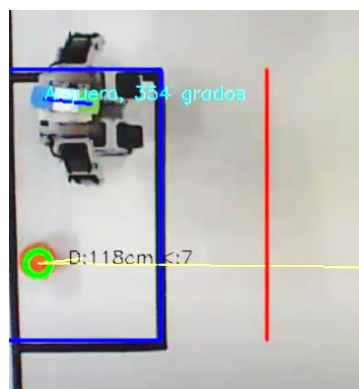


Fuente: Autores., Python 2.7.12

Arquero

Para el cálculo de trayectoria de la pelota en el arquero, el programa traza una línea roja a 45 centímetros del origen en "x", es decir aproximadamente 20 centímetros del arquero, cuando la pelota esté a 45 centímetros el arquero ejecutará determinada acción.

Ilustración 78. Distancia de Tape arquero



Fuente: Autores., Python 2.7.12

5.3 COMUNICACIÓN A PARTIR DEL PROCESAMIENTO DE IMAGEN

Para que los robots ejecuten movimientos, se vincula el algoritmo de procesamiento de digital de imágenes a el de comunicación que es el encargado de enviar las secuencias de movimiento a los robots.

Dentro del algoritmo principal, se crea una función que llama los archivos del algoritmo de comunicación.

El algoritmo principal indica las instrucciones que debe ejecutar el robot desde el procesamiento de imagen hasta el controlador principal del robot, éste controlador se encuentra programado con una secuencia preestablecida del kit de robótica (secuencia de fútbol). La función principal de éste algoritmo es la unión directa con el procesamiento de imagen para así por medio de software de Arduino, Python y Hardware como los módulos de comunicación inalámbricos *Zigbee* y la placa de Arduino, sean enviados los datos correctamente al controlador principal *CM-530*, encargado de convertirlos en el movimiento pertinentes para robots jugadores de fútbol arquero y delantero.

Para más detalles de los controladores *CM-530* y la vinculación directa de éstos con los módulos inalámbricos *Zigbee* diríjase a ...4. COMUNICACIÓN...

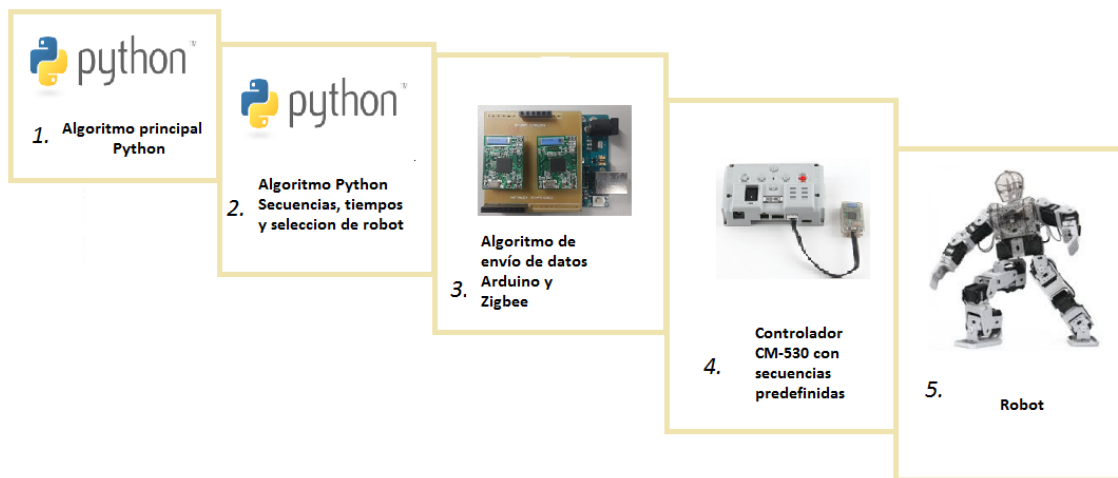
El programa principal en *Python*, establece las condiciones para el movimiento de los robots, éste se vincula con el programa de comunicación que posee las cadenas de datos correspondientes a cada acción del robot, así mismo proporciona información como cadenas de datos, y el tiempo entre datos que se deben enviar al Arduino. El Arduino selecciona los datos y los envía a los módulos *Zigbee* que éstos a su vez funcionan como un puente inalámbrico hacia el controlador *CM-530* de los robots.

El controlador recibe los datos y al tener programada la secuencia de fútbol de robots, finalmente el robot ejecuta la acción determinada.

El esquema del proceso de vinculación de comunicación al algoritmo principal se describe a continuación.

Es decir que la secuencia se representaría el vínculo entre la comunicación y el procesamiento de imagen, se muestra en la Ilustración 79 así.

Ilustración 79. Proceso de envío de datos a robot desde algoritmo principal para ejecutar acciones.



Fuente: Autores.

Donde cada Secuencia de la Ilustración anterior se explica mejor a continuación:

1. Algoritmo Python principal: dependiendo de la condición específica envía dato.

2. Algoritmo Python secuencias, tiempos y selección del robot (arquero o delantero): recibe datos del algoritmo principal seleccionando datos y tiempos de datos de funciones.

3. Algoritmo de envío de datos Arduino y Zigbee: El programa de Arduino recibe cadenas de datos y tiempos para enviarlos a los 2 Zigbee conectados físicamente sobre la tarjeta.

4. Controlador con secuencias predefinidas: El controlador principal del robot CM-530 recibe los datos enviados por los Zigbee y envía secuencia determinada a los motores.

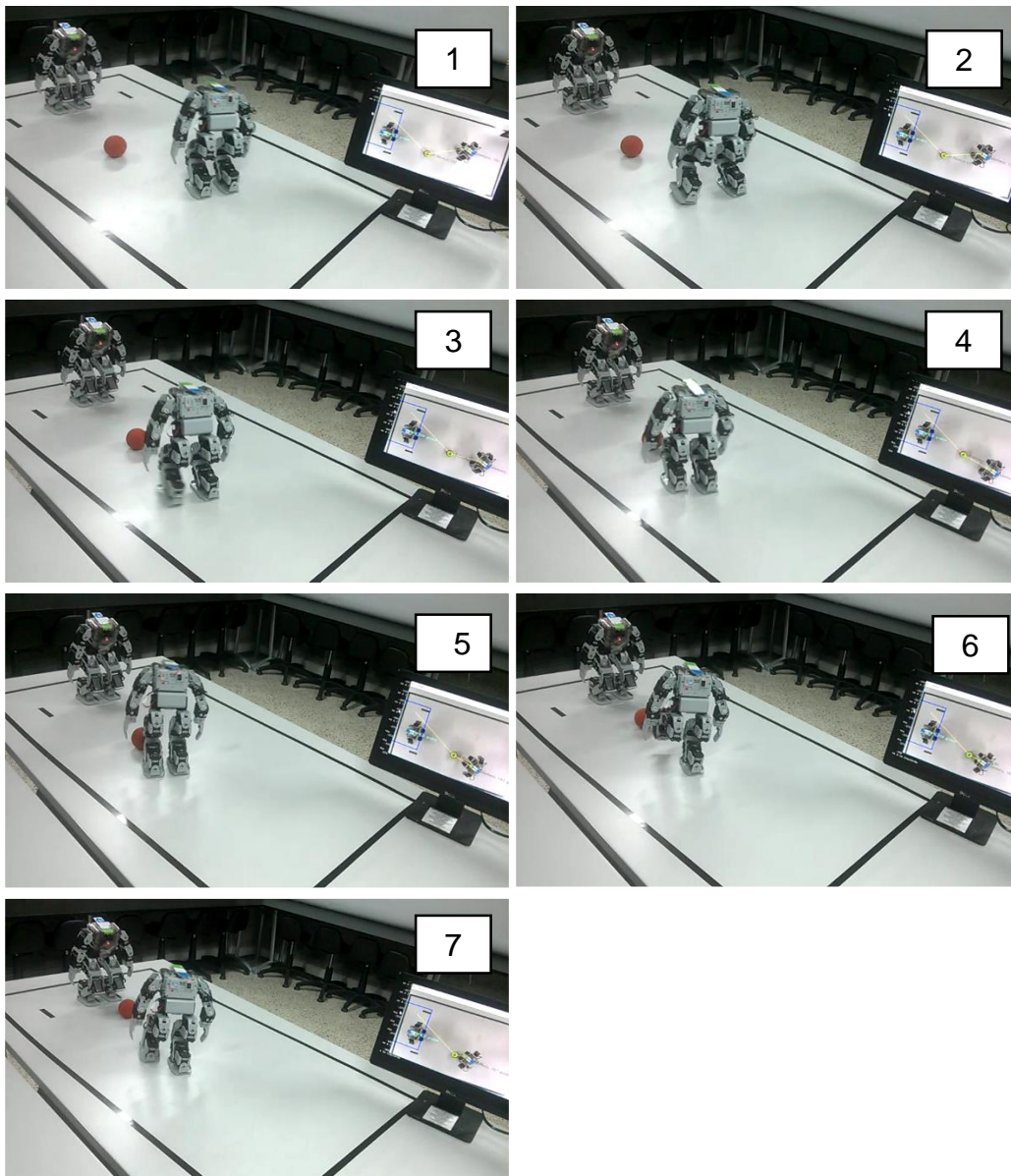
5. Robot: finalmente cada robot ejecuta los movimientos requeridos.

5.3.1 Acciones de los robots. En el punto 5.2 se especifican los parámetros que poseen los robots cuando éstos se encuentran en el campo de juego, pero sin ejecutar acción alguna.

Delantero

Se realiza el algoritmo de manera que si el programa arroja valores positivos el robot ejecute acción de girar hacia la derecha; si el programa arroja valores negativos, el robot ejecute acción de girar hacia la izquierda, y al mismo tiempo cumpla con una norma menor o igual a veinte, mientras esto no se cumpla el robot caminará hacia adelante. En la siguiente ilustración se muestra el robot ejecutando el programa completo.

Ilustración 80. Secuencia del robot delantero enfocando punto de gol y pateando la pelota



Fuente: Autores.

Arquero

Para el arquero el algoritmo es más sencillo, en el... punto 5.2 ... se describe que a una distancia de veinte centímetros del centroide verde del arquero, éste debe ejecutar la acción de atajar el balón, ésta distancia es calculada debido a la velocidad de pateo del arquero. En todo caso no se realiza una exactitud alta para el arquero debido que no tendría sentido hacer un delantero que lanzara la pelota y el arquero siempre la tapara, éste fue un error que no se tuvo en cuenta al comienzo del proyecto, pero que de igual manera se puede realizar. En la siguiente imagen se muestra el robot arquero ejecutando la acción de atajar cuando se cumple la condición de la distancia de la pelota de veinte centímetros o menos.

Ilustración 81. Figura del robot arquero atajando pelota



Fuente: Autores.

6. OBJETO VIRTUAL DE APRENDIZAJE (OVA)

“Un objeto de aprendizaje es un conjunto de recursos digitales, auto contenible y reutilizable; con un propósito educativo y constituido por al menos tres componentes internos: Contenidos, actividades de aprendizaje y elementos de contextualización”.⁴³

El objetivo principal del proyecto es el desarrollo de un OVA, el significado se explica anteriormente, se plantea la realización de un OVA virtual pero que contiene interacción física con sus estudiantes, en este caso, se emplean los kits de robótica como canal entre el estudiante, el OVA y el conocimiento.

6.1 Selección de OVA

Existen diferentes recursos digitales para enseñar, los cuales pueden ser blogs, aplicaciones digitales y páginas web; para que el aprendiz recurra directamente.

Se realiza una tabla de comparación de características entre blog, página web y aplicación.

Tabla 10. Comparativa Blog, página web y Aplicación.

Tipo	Blog	Página Web	Aplicación
Característica			
Interacción	X	X	X
Animaciones		X	X
Actualización constante	X		X
Comentarios directos con usuarios	X	X	X
Costo		X	X
Portabilidad	X	X	

Fuente: Autores.

Dada la comparativa anterior, se establece más pertinente la implementación de una página web con el contenido desarrollado en éste documento, esto es debido a las características:

- ✓ Portabilidad: ya que al encontrarse en línea el usuario tiene acceso en cualquier dispositivo y en cualquier lugar
- ✓ Interacción: incluye una interacción directa con el usuario debido a que está realizada para el mismo.

⁴³ FUNDACION I+D+I. Desarrollo de objetos virtuales de aprendizaje-OVA. En la web, disponible en <<http://www.fundacionidi.org/index.php/unidad-ti/desarrollo-de-objetos-de-aprendizaje-ova>>

- ✓ Material didáctico: Se pueden incluir animaciones, imágenes, videos, audios, y cualquier otra fuente electrónica que sirva de apoyo.
- ✓ Interacción con usuarios: aunque las tres poseen esta característica en común, la aplicación no es portable, ya que va a depender del sistema operativo de la aplicación, y el blog está diseñado para ser escrito en su mayor parte.
- ✓ Costo: aunque la página web presenta un costo de la persona que es la encargada de introducir los contenidos, es un costo inicial ya que la página se encuentra en un lugar de desarrollo libre.

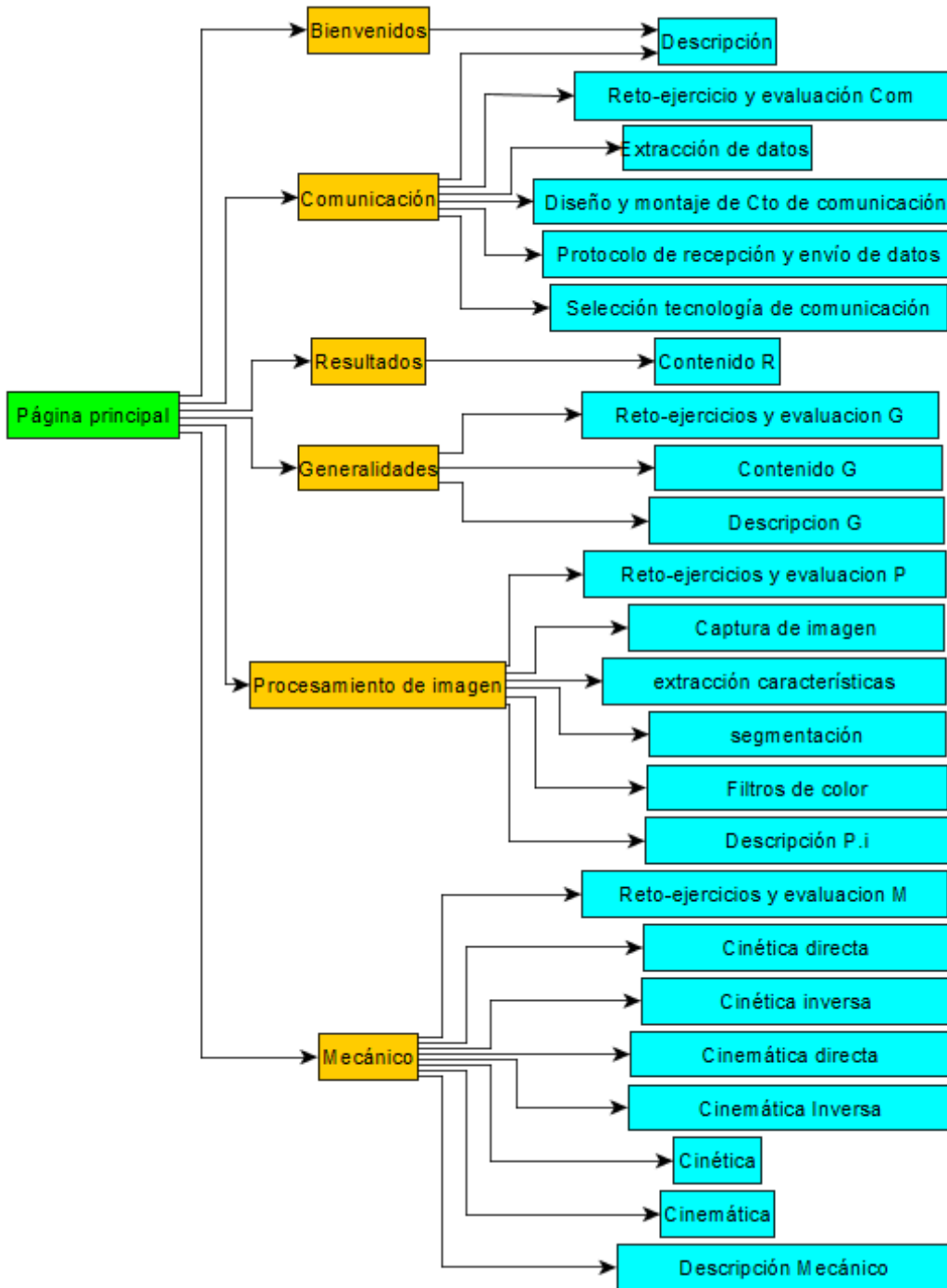
Dados los conceptos anteriores, en la página web se incluyen contenidos básicos para el usuario, creando un ambiente físico y virtual didáctico mediante la página web y los kits de robótica.

6.2 CONTENIDOS DEL OVA

El OVA, se encuentra en el link <http://roboticabiolooidpiloto.tk/>, ver ANEXO M.

La meta principal de la OVA, es presentar el aplicativo de fútbol de robots para motivar a los usuarios (estudiantes y docentes), a realizar proyectos en base a éste, brindándoles un espacio con herramientas y nociones básicas para la manipulación de los robots y de algunos conceptos de robótica fomentando el autoaprendizaje. Para ello se plantea el siguiente esquema donde se encuentran las partes de la página web:

Ilustración 82 Esquema general de página Web



Fuente: Autores.

Dada la ilustración 82, a continuación, se explica cada sección principal de la página, en el cuadro conceptual están representados con color azul.

- **Inicio (Bienvenidos):** concepto general de la página donde se orienta al usuario acerca de la página, qué es y qué se espera, ver ANEXO N.
- **Generalidades:** Descripción general del hardware y el software de los robots Bioloid, ver ANEXO O.
- **Mecánico:** breve introducción a mecánica en robots específicamente para un brazo, ver ANEXO P.
- **Procesamiento de imagen:** Descripción de visión de máquina empleado en el proyecto fútbol de robots, ver ANEXO Q.
- **Comunicación:** Descripción de comunicación entre el controlador CM-530 del robot y módulos Zigbee por medio de Arduino, ver ANEXO R
- **Resultados:** Multimedia de los resultados obtenidos a partir del desarrollo del proyecto, ver ANEXO S y ANEXO T.

6.2.3 Subcontenidos de secciones

- **Videos introductorios** Videos disponibles en el inicio de cada sección, con el fin de orientar al estudiante, sobre los contenidos de cada capítulo y las herramientas presentes en los mismos. En la página se encuentran cuatro videos de este tipo al inicio de cada capítulo. En la ilustración 83, se muestra la captura de el video introductorio de la página web.

Ilustración 83. Captura de pantalla de video introductorio a OVA

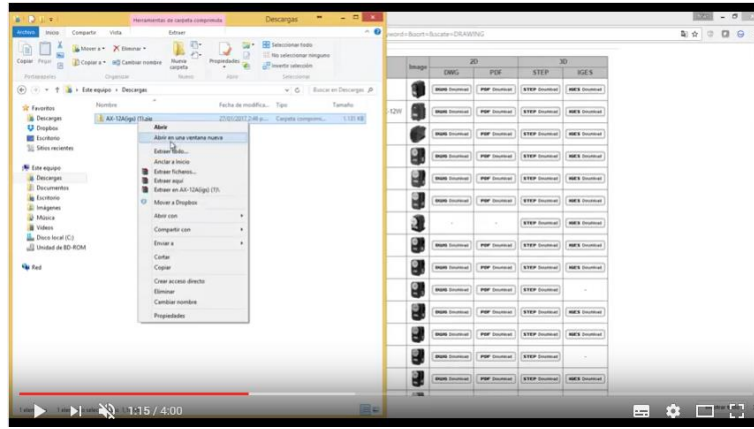


Fuente: autores, Youtube.

- **Videos tutoriales** Videos presentes en algunas de las secciones, explicando el paso a paso para enseñar un tema en concreto, el objetivo principal de éstos videos es que el aprendiz realice la práctica en tanto observe el video. En la página web se pueden encontrar tres videos de este

tipo. En la ilustración 84, se muestra una captura de pantalla de un tutorial de Solidworks de los Bioloid, disponible en la página web y en Youtube.

Ilustración 84. Captura de pantalla de tutorial Bioloid en Solidworks



Fuente: autores, Youtube.

- **Material descargable** Archivos disponibles para que el estudiante pueda descargar y recibir apoyo de los mismos. Un ejemplo se muestra en la ilustración 85.

Ilustración 85. Archivo descargable algoritmo HSV detección objetos azules

Se puede descargar algoritmo de HSV para detección de objetos azules en Python usando la cámara en tiempo real a continuación



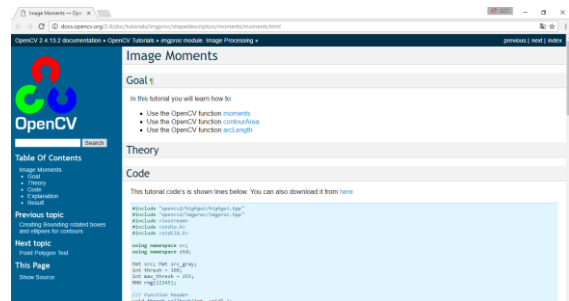
Fuente: autores.

- **Links externos** Contenido externo relacionado con cada uno de los ítems expuestos, éstos son usados para ampliar la información o que hacen parte del proyecto y no hacen parte de fuente propia. También se implementó un link de descarga de software para que el aprendiz. Como se muestra en la ilustración 86.

Ilustración 86. Link de direccionamiento externo al recurso web de OpenCV, sección procesamiento de imagen.

Puede encontrarse un código fuente de Moments en el siguiente link:

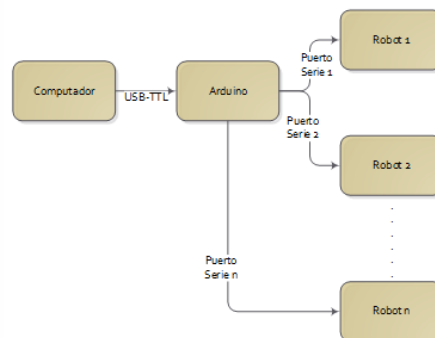
[Ver Link](#)



Fuente: autores.

- **Esquemas e imágenes explicativas** Representación gráfica de procedimientos, conceptos y algoritmos, que se le brinda al estudiante para mejorar su comprensión.

Ilustración 87. Esquema Demux Arduino, disponible en la sección comunicación página web.



Fuente: Autores.

- **Reto-ejercicios** Ejercicios planteados para que el estudiante ponga en práctica sus conocimientos, y obtenga un resultado de lo aprendido, se muestra en la ilustración 88, uno de los retos de la página.

Ilustración 88. Reto-ejercicio capítulo generalidades

RETO- EJERCICIO -CAPÍTULO GENERALIDADES

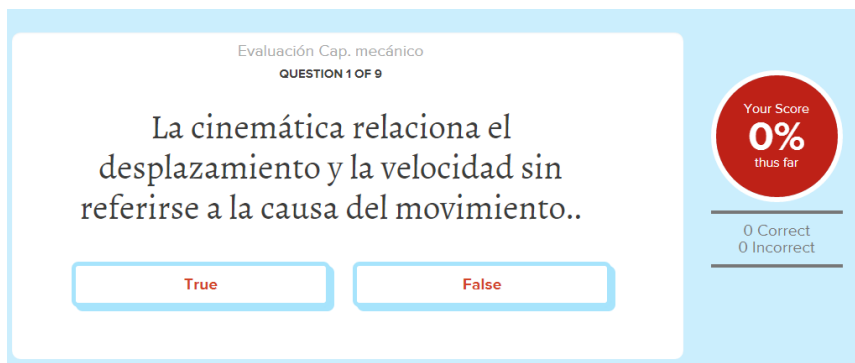
1. En el siguiente video, encontrarás el tutorial de cómo compilar el programa de fútbol de robots, la misión, es seguir paso a paso y lograr el movimiento del robot con el control remoto.

<https://www.youtube.com/watch?v=V9laWAgCawM>

Fuente: autores.

- **Evaluación** Sección de preguntas con múltiple respuesta para que el estudiante ponga a prueba sus conocimientos, y realice una retroalimentación. En la siguiente imagen (ilustración 89), se muestra una pregunta de la evaluación propuesta para la sección mecánica de la página web.

Ilustración 89. Captura de pantalla, pregunta evaluación capítulo mecánico



The screenshot shows a quiz interface with a light blue background. At the top, it says "Evaluación Cap. mecánico" and "QUESTION 1 OF 9". The question text is "La cinemática relaciona el desplazamiento y la velocidad sin referirse a la causa del movimiento..". Below the question are two buttons: "True" and "False". On the right side, there is a red circular progress indicator showing "Your Score 0% thus far" and a score summary: "0 Correct" and "0 Incorrect".

Fuente: autores, quizbean.

Inicialmente no se plantea sección mecánica en la página web, ya que no existe un vínculo directo entre los cálculos matemáticos y el desarrollo del proyecto, esto es debido a que no se realiza diseño y construcción del robot, sin embargo, se considera pertinente realizar una introducción a la mecánica para de un brazo, de manera que sea entendible y útil para diseños y construcciones futuras.

7. CONCLUSIONES

- Para realizar el análisis de un sistema robótico con múltiples grados de libertad, se debe descomponer el sistema y realizar suposiciones de las cuales se puede realizar el posible planteamiento matemático.
- En la vida real cuando se realice un diseño de un robot, el análisis matemático de un sistema se deben considerar otros aspectos físicos no incluidos en este proyecto, como lo son fuerzas centrípetas, fuerzas de rozamiento, efecto coriolis, entre otros.
- Para realizar un sistema de visión por máquina, es importante tener una idea clara de la aplicación que se va a realizar, para seleccionar correctamente el tipo de visión de máquina a desarrollar.
- Para segmentar el color rojo en HSV es necesario realizar 2 segmentaciones y sumarlas para obtener la gama completa de rojos.
- Las fases de procesamiento de imagen por color son captura de imagen, filtro de color, segmentación de color, extracción de características, y caracterización de elementos. Se debe tener en cuenta que entre estas fases existen otros procedimientos aplicados como la reducción de ruido.
- El controlador del kit de robótica CM-530 presenta una gran ventaja, ya que posee la capacidad de conectar múltiples periféricos externos, entre éstos periféricos existe la posibilidad de realizar varios tipos de comunicación, el tipo de comunicación se puede elegir según el tipo de aplicación.
- La implementación de los módulos Zig-Bee para la aplicación de futbol es clave implementar la librería SoftwareSerial.h incluida en Arduino que permite conexión de múltiples módulos inalámbricos.
- Se aplican conocimientos teóricos a la hora de manejar los robots, pero es importante siempre realizar pruebas de funcionamiento y hacer retroalimentación para mejorar el desempeño de los mismos.
- Utilizar una página web como objeto virtual de aprendizaje es una herramienta muy versátil debido a que las personas tienen acceso en cualquier momento y en dispositivos con acceso a internet.
- El rendimiento del procesamiento de imágenes en el futbol de robots se ve disminuido debido a los tiempos de envío de datos de comunicación incluidos dentro del mismo programa.
- El OVA por medio de sus herramientas de retos y evaluaciones permite al estudiante comprender en que se ha fallado para tener retroalimentación fomentando el autoaprendizaje.

8. RECOMENDACIONES Y TRABAJOS FUTUROS

- Se recomienda al estudiante seguir los tutoriales de Solidworks del OVA, como base para considerar aspectos físicos en diseño mecánico de los robots como lo son momentos de inercia que permiten el complementar los cálculos de fuerzas centrípetas, fuerzas de rozamiento, efecto coriolis, entre otros.
- El OVA está orientado a estudiantes que posean información básica acerca de robótica.
- Se deja abierta la propuesta de éste proyecto para implementación de robótica cooperativa y aplicarla en futuros desarrollos.
- Se propone la realización de un equipo completo de fútbol, debido a que se desarrolló un delantero y un arquero, se podría realizar la duplicación del circuito de comunicación y de esta manera incluir más robots.
- El procesamiento de imágenes puede verse mejorado considerablemente implementando un computador con un procesador más rápido y realizando las tareas multi-hilo entre procesamiento de imagen y comunicaciones.

9. ANEXOS

ANEXO A. ENCUESTA ESTUDIANTES ACERCA DE KITS DE ROBÓTICA

¿Conoce usted de los kit de robótica Bioloid? *

- SI
- NO

¿Sabe en que consiste un kit de robótica bioloid? *

- SI
- NO

¿Usted cursó la electiva de robótica en la universidad? *

- SI
- NO

¿Sabia usted que la universidad adquirió varios kit de robótica? *

- SI
- NO

¿Alguna vez ha empleado estos kits de robótica? *

- SI
- NO

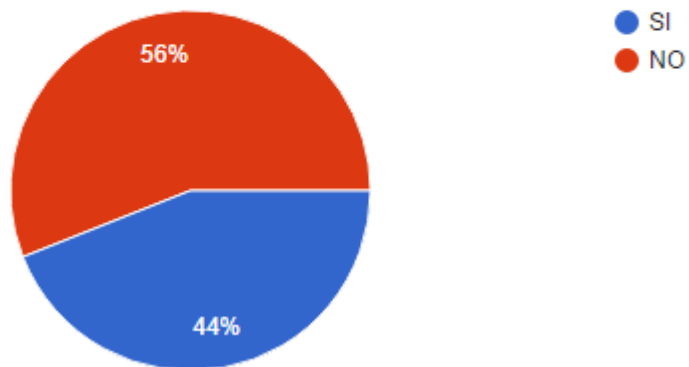
¿Sabe como se ponen en funcionamiento los robot? *

- SI
- NO

Fuente: Autores.

ANEXO B. RESULTADOS PREGUNTA: ¿CONOCE USTED LOS KITS DE ROBÓTICA BIOLOID?

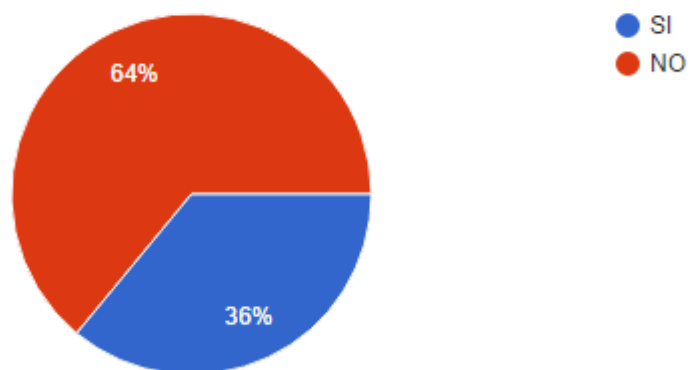
¿Conoce usted de los kit de robótica Bioloid? (25 respuestas)



Fuente: Autores.

ANEXO C. RESULTADOS PREGUNTA: ¿SABE EN QUÉ CONSISTE UN KIT DE ROBÓTICA BIOLOID?

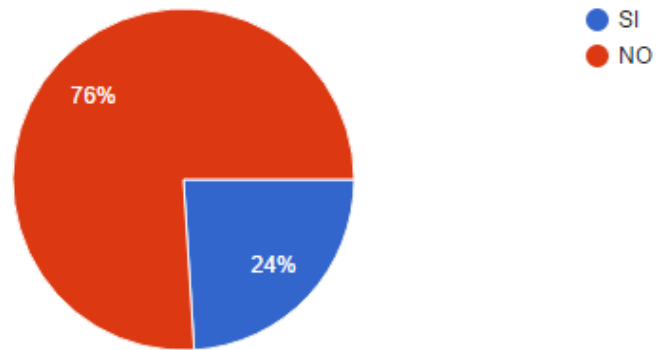
¿Sabe en que consiste un kit de robótica bioloid? (25 respuestas)



Fuente: Autores.

ANEXO D. RESULTADO PREGUNTA: ¿USTED CURSÓ LA ELECTIVA DE ROBÓTICA EN LA UNIVERSIDAD?

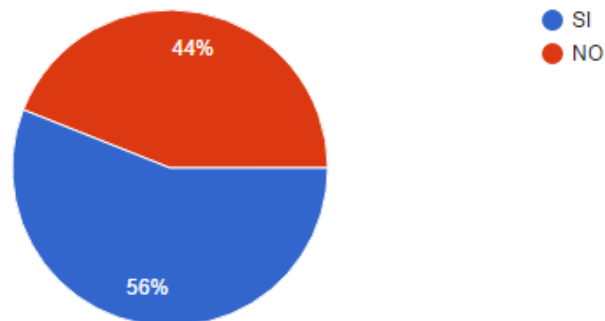
¿Usted cursó la electiva de robótica en la universidad? (25 respuestas)



Fuente: Autores.

ANEXO E. RESULTADO PREGUNTA: ¿SABÍA USTED QUE LA UNIVERSIDAD ADQUIRIÓ VARIOS KITS DE ROBÓTICA?

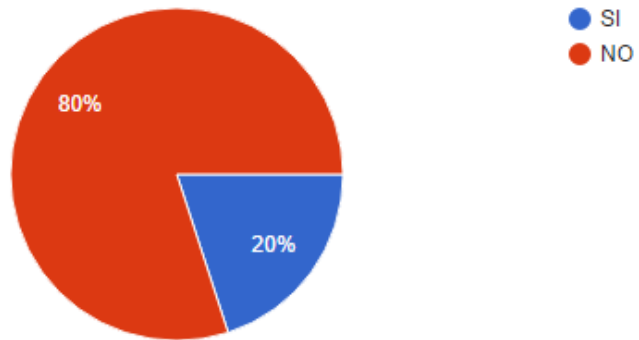
¿Sabía usted que la universidad adquirió varios kit de robótica? (25 respuestas)



Fuente: Autores.

ANEXO F. RESULTADO PREGUNTA: ¿ALGUNA VEZ HA EMPLEADO ÉSTOS KITS DE ROBÓTICA?

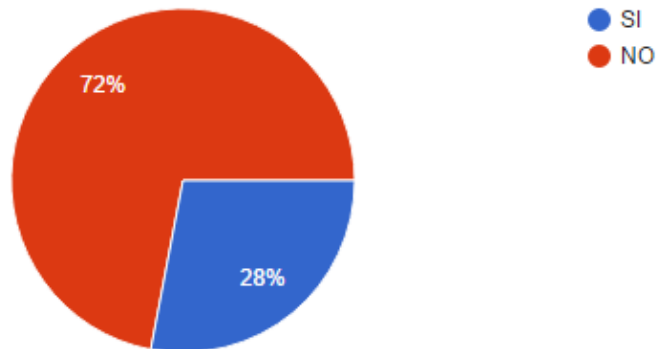
¿Alguna vez ha empleado estos kits de robótica? (25 respuestas)



Fuente: Autores.

ANEXO G. RESULTADO PREGUNTA: ¿SABE CÓMO SE PONEN EN FUNCIONAMIENTO LOS ROBOTS?

¿Sabe como se ponen en funcionamiento los robot? (25 respuestas)



Fuente: Autores.

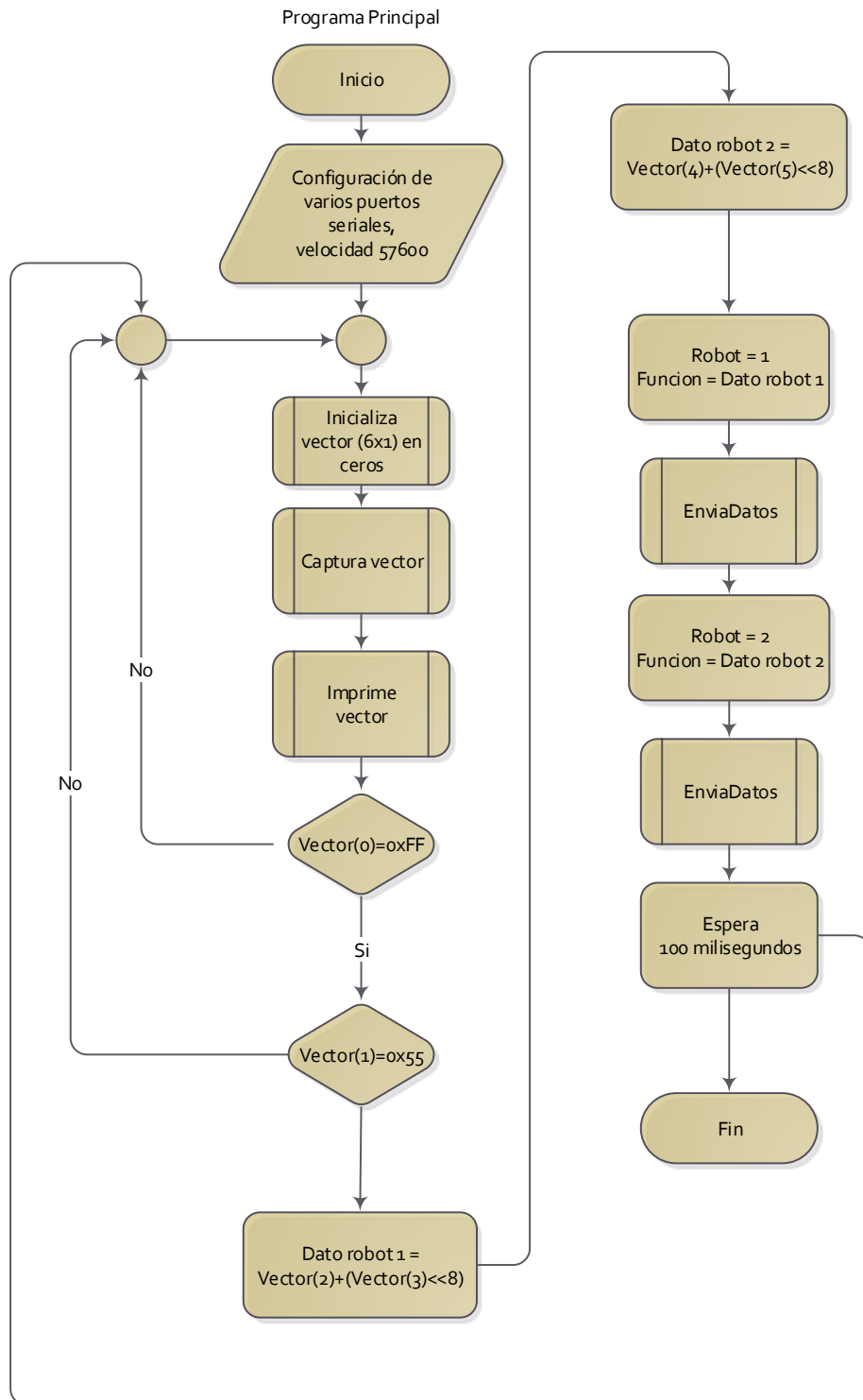
ANEXO H. COMPATIBILIDAD CM-530

Controller Function Analysis

function	CM-100A	CM-150	CM-200	CM-5	CM-510	CM-530	CM-700	CM-900	OpenCM9.04	OpenCM9.04+485 Exp.Board
Robo Plus	✓	✓	✓	✓	✓	✓	✓			
USB Downloader (LN-101)	✓	✓	✓				✓	✓	✓	✓
Bioid Serial Cable				✓	✓					
Mini USB Cable						✓				
Micro USB Cable(Type B)		✓						✓	✓	✓
Dynamixel AX-Series				✓	✓	✓	✓	✓	✓	✓
Dynamixel DX-Series							✓	✓		
Dynamixel RX-Series							✓	✓		
Dynamixel EX-Series							✓	✓		
Dynamixel MX-Series_TTL				✓	✓	✓	✓	✓	✓	✓
Dynamixel MX-Series_RS485							✓	✓		✓
Dynamixel XM-Series										✓
Dynamixel XH-Series										✓
XL-320									✓	✓
IR Sensor IRSS-10	✓	✓	✓		✓	✓	✓		✓	✓
Touch Sensor TS-10	✓	✓	✓		✓	✓	✓		✓	✓
DMS-80					✓	✓	✓		✓	✓
AX-S1				✓	✓	✓	✓	✓	✓	✓
IR Sensor Array				✓	✓	✓	✓	✓	✓	✓
Gyro Sensor GS-12					✓	✓	✓		✓	✓
Color Sensor CS-10		✓	✓			✓			✓	✓
Magnets Sensor MGSS-10		✓	✓			✓			✓	✓
Passive IR Sensor PIR-10		✓	✓		✓	✓			✓	✓
temperature Sensor TPS-10		✓	✓		✓	✓			✓	✓
Servo Motor SM-10	✓	✓	✓			✓				
Geared Motor GM-10A	✓	✓	✓							
High speed Geared Motor	✓	✓								
Low speed Geared Motor	✓	✓								
LED Module LM-10	✓	✓	✓			✓			✓	✓
IR Receiver	✓	✓			✓	✓	✓			
ZIG-100				✓						
ZIG-110A	✓	✓	✓		✓	✓	✓	✓	✓	✓
BT-110A	✓	✓	✓		✓	✓	✓	✓	✓	✓
BT-210	✓	✓	✓		✓	✓	✓	✓	✓	✓
BT-410	✓	✓	✓		✓	✓	✓	✓	✓	✓
LI-ion Battery LBS-040		✓	✓						✓	✓
LIPO 11.1V Battery					✓	✓	✓			
NIMH 9.6V Battery Pack				✓						
Olio Battery BOX OBB-10A	✓									
Battery BOX ABB-030			✓							
LittleFuse 125V 10A LFU-10					✓	✓	✓			
Fuse 250V 5A FU-5				✓						
User Device					✓	✓	✓	✓	✓	✓

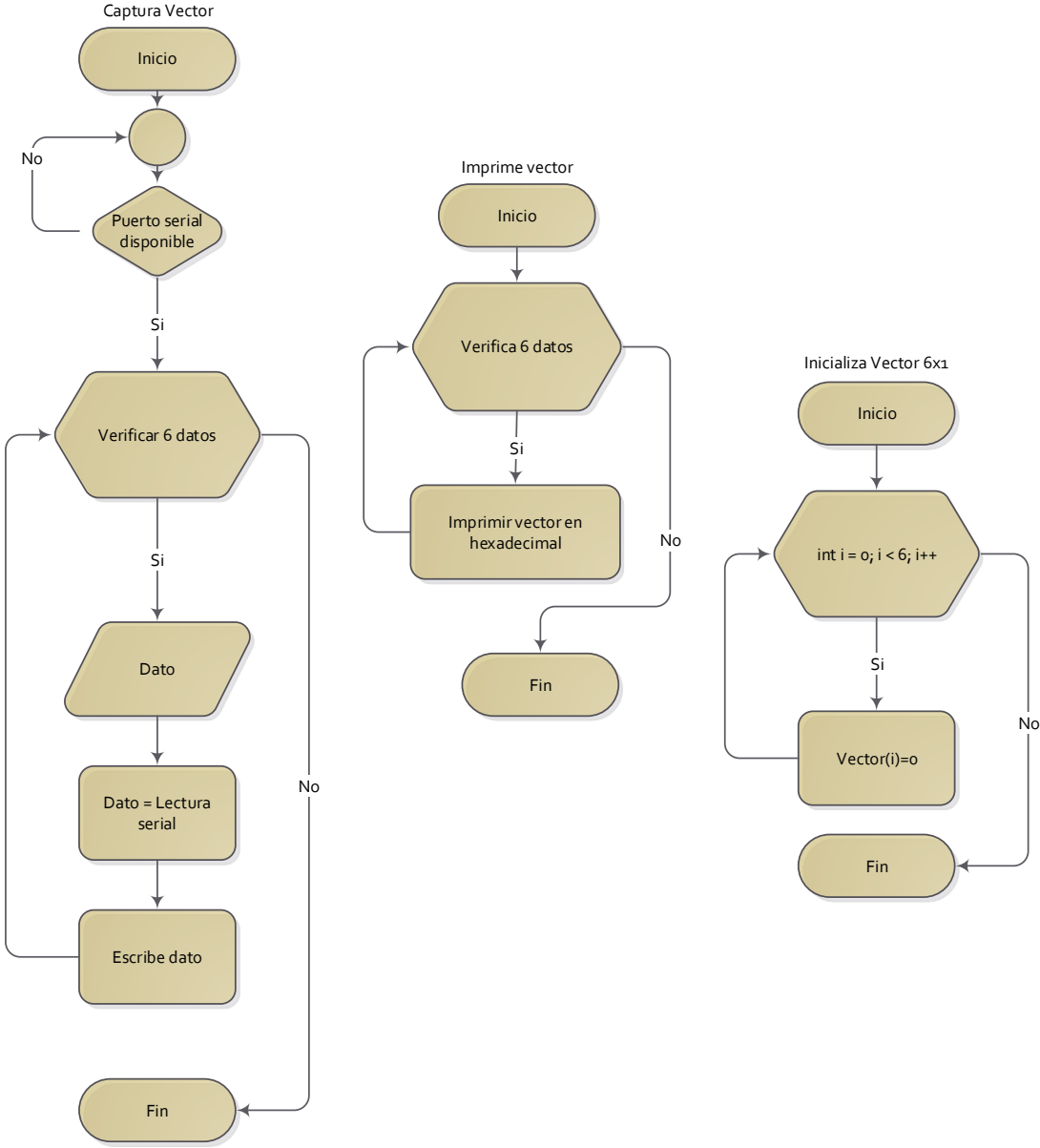
Imagen tomada de: http://support.robotis.com/en/product/controller_main.htm

ANEXO I. DIAGRAMA DE FLUJOS FUNCIONAMIENTO ALGORITMO GENERAL DE ARDUINO



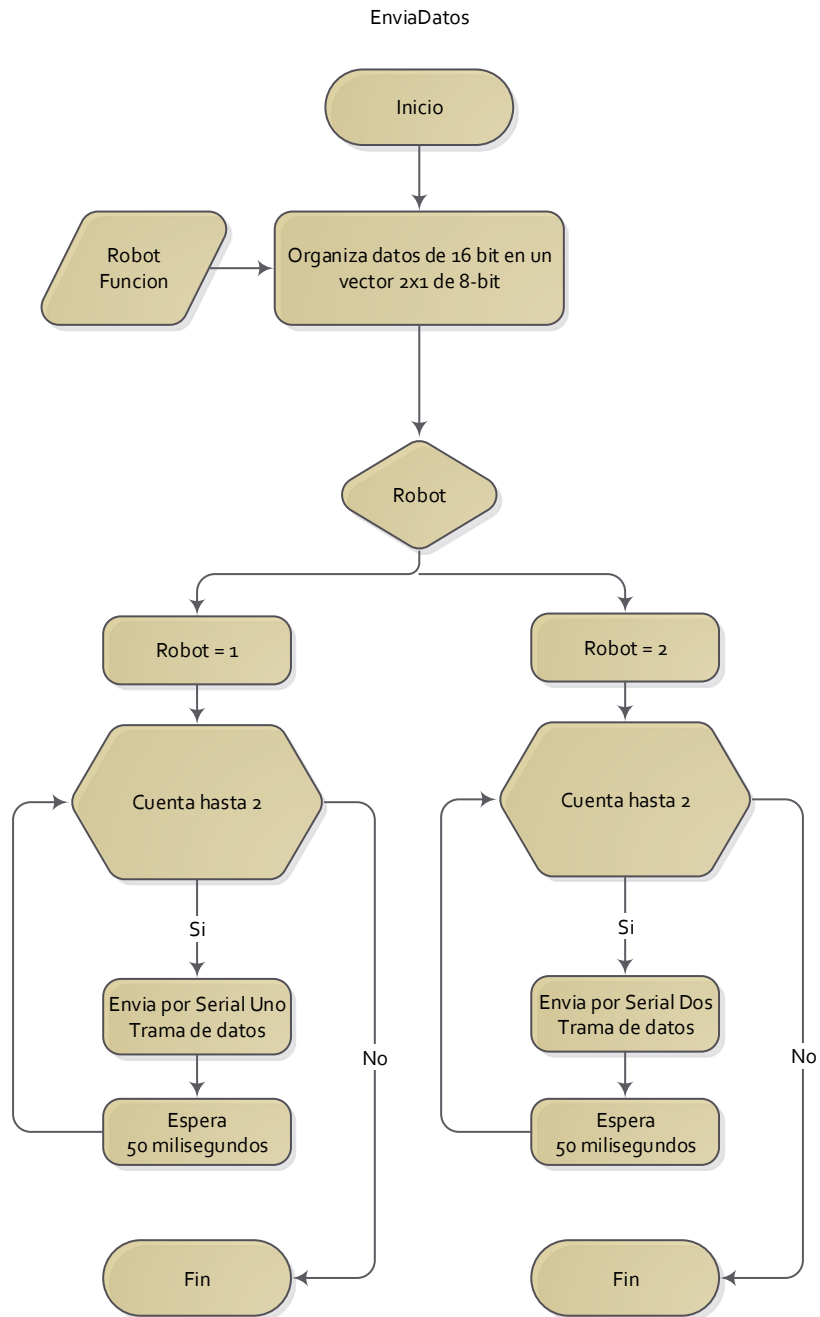
Fuente: Autores.

ANEXO J. SUBROUTINAS 1, 2 Y 3 DE ARDUINO



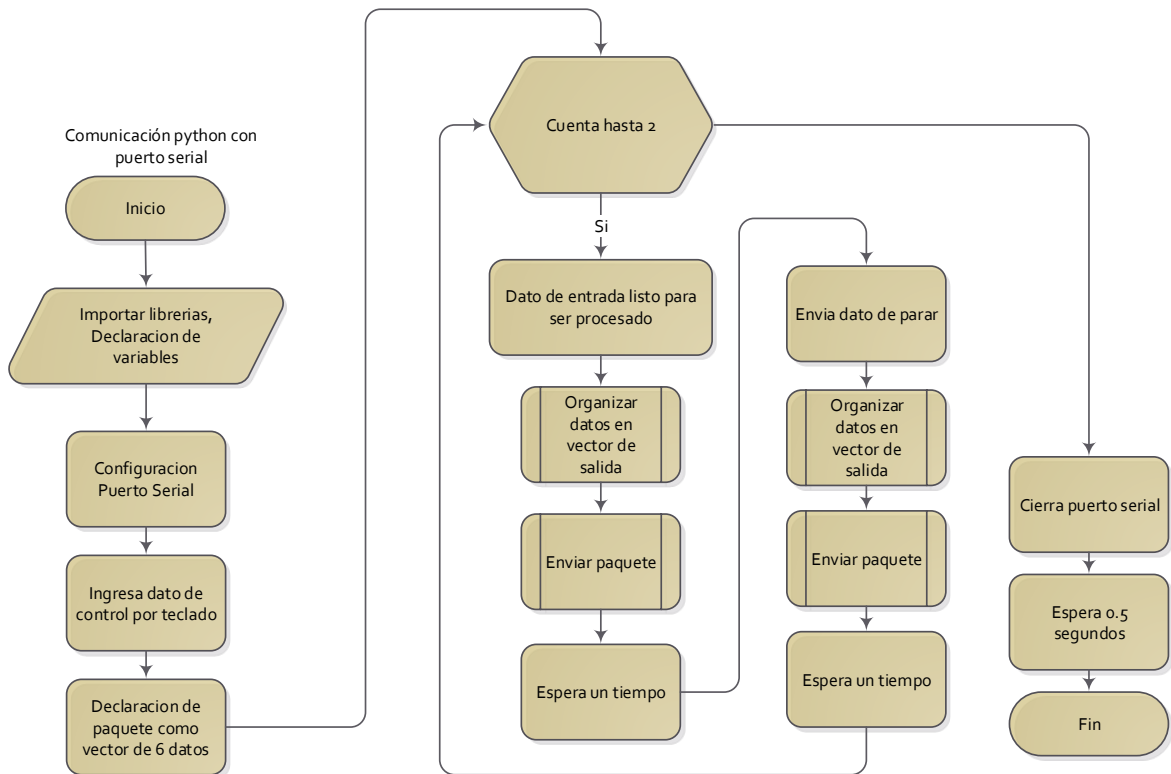
Fuente: Autores.

ANEXO K. SUBRUTINA 4 DE ARDUINO



Fuente: Autores.

ANEXO L. ALGORITMO PRINCIPAL DE COMUNICACIONES EN PYTHON



Fuente: Autores.

ANEXO M. OVA, PÁGINA PRINCIPAL, PÁGINA WEB.



Fuente: Autores.

ANEXO N. OVA, INICIO PÁGINA WEB



Fuente: Autores.

ANEXO O. OVA, GENERALIDADES PÁGINA WEB.




Fuente: Autores.

ANEXO P. OVA, MECÁNICO PAGINA WEB



Bienvenidos Generalidades **Mecánico** Procesamiento de imagen Comunicación Resultados

Descripción	Una de las partes importantes de realizar el diseño de un robot, es conocer cómo se mueve y bueno.. cómo realizar su diseño. La mecánica es importante para el diseño, el consumo y el correcto desempeño del robot, además para poder reducir el porcentaje de falla en cuanto a el movimiento del mismo.	
Cinemática		
Cinemática directa		
Cinemática inversa		
Denavit-Hartenberg		
Cinética	En este caso para un análisis claro de la mecánica de un robot tan complejo, se plantea el análisis de casos sencillos para si se decide hacer un diseño propio no se empiece desde cero, sino teniendo algunos conceptos claros. Se debe tener en cuenta que en todos los casos se plantea un sistema rígido, es decir se van a despreciar las deformaciones. Tampoco se toman en cuenta fuerzas	
Cinética inversa		
Cinética directa		

Fuente: Autores.

ANEXO Q. OVA, PROCESAMIENTO DE IMAGEN PAGINA WEB



Bienvenidos Generalidades Mecánico **Procesamiento de imagen** Comunicación Resultados

Para que el robot sea autónomo o realice la función que se le plantea, se recurrió a la implementación de visión por máquina para la identificación de los robots y de la pelota a patear.



Para el procesamiento de imagen de la detección e identificación de objetos, se utiliza lenguaje de

Fuente: Autores.

ANEXO R. OVA, COMUNICACIÓN PAGINA WEB



[Bienvenidos](#) [Generalidades](#) [Mecánico](#) [Procesamiento de imagen](#) **[Comunicación](#)** [Resultados](#)

COMUNICACIÓN

Para lograr que los robots ejecuten las tareas deseadas (arquero y delantero de futbol), debe existir un protocolo de comunicación definido y estructurado entre los robots y un computador.

Por tal motivo se plantean los siguientes ítems:

- Selección de tecnología de comunicación.
- Análisis del protocolo de transmisión de datos.
- Diseño de un circuito impreso el cual establece la relación física entre los módulos de comunicación y el controlador.
- Diseño de algoritmo de envío de datos hacia los robots; de esa forma los robots pueden ejecutar los movimientos al mismo tiempo.

Fuente: Autores.

ANEXO S. OVA, RESULTADOS PAGINA WEB



[Bienvenidos](#)

[Generalidades](#)

[Mecánico](#)

[Procesamiento de imagen](#)

[Comunicación](#)

[Resultados](#)



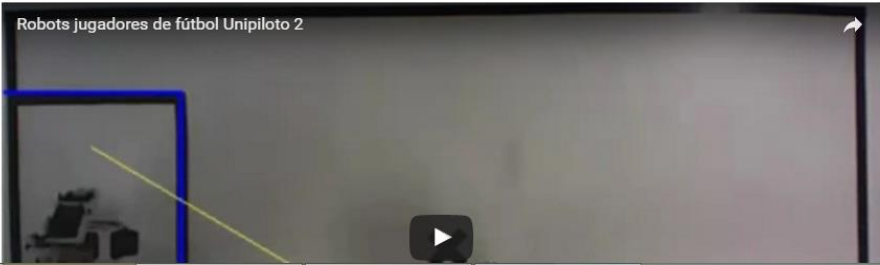
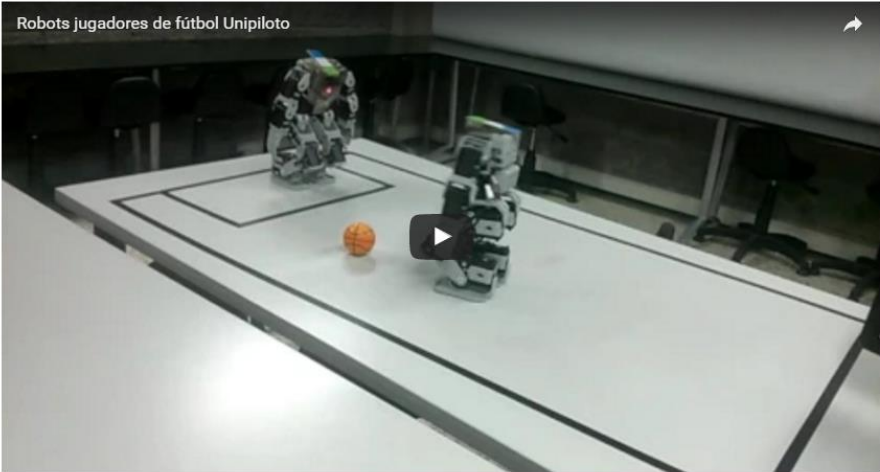
LOS RESULTADOS OBTENIDOS SE ENCUENTRAN A CONTINUACIÓN:

Fuente: Autores.

ANEXO T. OVA, EXTENSION RESULTADOS PAGINA WEB



LOS RESULTADOS OBTENIDOS SE ENCUENTRAN A CONTINUACIÓN:



Fuente: Autores.

10. BIBLIOGRAFÍA

1. SALIDO Tercero Jesús .Cibernética Aplicada Robots educativos. 1ed.Alfaomega Grupo Editor, S.A. de C.V. México. 2010.315 pag. ISBN 978-607-7854-43-2. 316.
2. REYES Cortés Fernando. Robótica , control de robots manipuladores. 1ed. Alfaomega Grupo educativos. México. 2011.522pag. ISBN 978-607-707-190-7.
3. OLLERO Baturone Aníbal.ROBÓTICA Manipuladores y robots móviles. Publicado por Coedición Alfaomega y Marcombo. Barcelona , España. 2001 . 447 pag. ISBN: 970-15-0758-4.
4. AGUILAR Diego, VIVAS Andrés. Modelado y control de un robot bípedo de nueve grados de libertad. En línea. Universidad del Cauca, Facultad de Ingeniería Electrónica y Telecomunicaciones, Popayán. 2009.Disponible en internet : <http://www.unicauca.edu.co/ai/publicaciones/Aguiar_ACA09.pdf>.
5. CASTILLO Ramirez ArturoApuntes para la materia de Cinemática de las máquinas. Enero 2005. Universidad Autónoma San Luis Potosí, Facultad de ingeniería. 234pag. Pag 3.
6. KUMAR Saha Subir. Introducción a la robótica.De la primera edición en inglés de *Introduction to robotics*, by Subir Kumar Saha. Publicado por Tata McGraw-Hill Education Private Limited, NOIDA, India. 371pag. ISBN: 978-0-07-066900-0.p.52
7. Ibid., p. 52-53.
8. MÉNDEZ Regalado Alejandro, PERALTA Reyes Ever, GOZALEZ Carlos Alberto. Ensayos. Cómo hacer un modelo matemático. Universidad Tecnológica de la Mixteca. Temas de Ciencia y Tecnología. Mayo - Agosto 2008. En línea. Disponible en <http://www.utm.mx/edi_anteriores/temas035/2%20ensayo-35.pdf>
9. SUBIR KUMAR SAHA. Introducción a la robótica.De la primera edición en inglés de *Introduction to robotics*, by Subir Kumar Saha. Publicado por Tata McGraw-Hill Education Private Limited, NOIDA, India. 371pag. ISBN: 978-0-07-066900-0.p. 66-67.
10. GÓMEZ DÍAZ ISABEL CRISTINA. Universidad de Santander, “UDES” Bucaramanga, . ELABORACION DE UN OBJETO VIRTUAL DE APRENDIZAJE (OVA) SOBRE “TECNICAS DE FACILITACION NEUROMUSCULAR PROPIOCEPTIVA” . [En línea]. [Citado el 24 de febrero de 2016]. Disponible en: <<http://www.bdigital.unal.edu.co/17477/2/13111-36454-1-PB.pdf>>.
11. GRUPOS JUVENILES CREATIVOS. Instituto Educativo Distrital Ciudadela 20 de Julio. Herramientas de comunicación virutales. [En línea]. [Citado el 18 de Marzo de 2016]. Disponible en: <<http://files.informaticainedicdelahoz.webnode.com.co/200000385-0c1410e084/Guia%20-%204%20InformaticaCiclo%20V.pdf>>
12. American Honda Motor Co., Inc. ASIMO The World’s Most Advanced Humanoid Robot [ASIMO El robot humanoide más avanzado del mundo].s.f.[Base de datos en línea]. Disponible en la página Web: < <http://asimo.honda.com/asimo-specs/>> [citado en 28 Febrero de 2016]
13. BOSTON DYNAMICS. Atlas - The Agile Anthropomorphic Robot [Atlas- El ágil robot antropomórfico].[Base de datos en línea]. Propiedad Filial total de Google, Inc. Waltham, Massachusetts. Boston Dynamics.s.f.. Disponible en Internet: <http://www.bostondynamics.com/robot_Atlas.html>[citado en 28 Febrero de 2016]
14. EVAN ACKERMAN, Erico Guizzo, Angelica Lim, Jason Falconer. Iran Demonstrates New Humanoid Robot Surena III [Irán demuestra el nuevo robot humanoide Surena III. [Boletín informativo En Línea].[eng]. New York,N.Y. .IEEE SPECTRUM. Fecha de publicación 17 Nov 2015 , 21:55 GMT. Disponible en Internet : < <http://spectrum.ieee.org/automaton/robotics/humanoids/iran-humanoid-robot-surena-iii>>. [Citado en 28 Febrero de 2016].
15. DR. DENNIS HONG, Daniel Lee. DARwIn OP:Open Platform Humanoid Robot for Research and Education. En línea. Los Ángeles. Septiembre 25 , 2011. < <http://www.romela.org/darwin-op-open-platform-humanoid-robot-for-research-and-education/>>. [Citado en 29 Febrero de 2016].
16. UNIMEDIOS, AGENDA DE NOTICIAS UN,CIENCIA & TECNOLOGÍA.En la UN construyen robot bípedo. Fecha de publicación 09 de marzo de 2011. Disponible en internet

<<http://agenciadenoticias.unal.edu.co/nc/detalle/article/en-la-un-construyen-robot-bipedo/>>
[Citado en 29 Febrero de 2016].

17. EVAN ACKERMAN, Erico Guizzo, Angelica Lim, Jason Falconer. Virginia Tech's Humanoid Robot CHARLI Walks Tall . [Boletín informativo En Línea],[eng]. New York,N.Y. .IEEE SPECTRUM. Fecha de publicación 17 Nov 2015 , 21:55 GMT. Disponible en Internet : < <http://spectrum.ieee.org/automaton/robotics/humanoids/042810-virginia-tech-humanoid-robot-charli-walks-tall>>. [Citado el 5 de Marzo de 2016].
18. VARGAS TORRES, G. A., & CASTILLO ESTEPA, R. A. (2015). Desarrollo de algoritmo para detección y comando de robots humanoides en tareas de recolección. Revista Tecnura, 19(45), 127-139. doi:10.14483/udistrital.jour.tecnura.2015.3.a10
19. TACUÉ GONZÁLEZ JEISON JAVIER, NARANJO MAGÉ HUGO FABIAN, RENGIFO RODAS CARLOS FELIPE.Simulación del ciclo de marcha del robot bípedo bioloid en el entorno virtual v-rep. Popayán, Cauca, Colombia. 2015.ResearchGate.En línea. Disponible en <https://www.researchgate.net/publication/284283402_Simulacion_del_Ciclo_de_Marcha_del_Robot_Bipedo_Bioloid_en_el_Entorno_Virtual_V-REP>
20. CLAUDIA C. CASTELO, Héctor R. Fassi, Flavio E. Scarpettini. Fútbol de Robots: Revisión del Estado del Arte y desarrollo del Equipo UBASot de Simulación.Tesis de Licenciatura. Buenos Aires.Departamento de Computación Facultad de Ciencias Exactas y Naturales Universidad de Buenos Aires.2002.326p
21. DIEGO MAURICIO GUTIERREZ RAMIREZ.Juan Pablo Ocampo Ramirez. Elvin Valdes Avila. Investigación en Fútbol Robótico sobre plataforma lego mindstorm. Tecnología en Electrónica. Bogotá. Corporación Universitaria Minuto de Dios. 2013.169pag.
22. M. EN C. MARCO ANTONIO MORENO ARMENDARIZ. Visión artificial estero con aplicación al control de un brazo robot. Doctor en Ciencias de en la Especialidad de Control Automático. México, D.F. Centro de investigación de estudios avanzados del Instituto Politécnico Nacional Departamento de Control Automático.2003.157pag.
23. APU JONATAN MATÍAS PONZO. Control de robots autónomos móviles Basados en arquitecturas de sistemas embebidos, Propuesta de Modelo de Aplicación y Uso Potencial. Licenciado en sistemas.Buenos Aires, Argentina. Universidad Nacional de Lanús.2013.367 pag. Disponible en : <<http://sistemas.unla.edu.ar/sistemas/gisi/TFLS/Ponzo-TFL.pdf>>
24. SEUNG-JOON YI, STEPHEN MCGILL, DENNIS HONG AND DANIEL LEE. Hierarchical Motion Control for a Team of Humanoid Soccer Robots. Int J Adv Robot Syst, 2016, 13:32. doi: 10.5772/62037. En línea. Disponible en: tp://www.intechopen.com/journals/international_journal_of_advanced_robotic_systems/hierarchical-motion-control-for-a-team-of-humanoid-soccer-robots>
25. Universidad Católica de Pereira. OVA DE COMPUERTAS DIGITALES. [En línea]. [Citado el 24 de febrero de 2016]. Disponible en: <<http://ribuc.ucp.edu.co:8080/jspui/bitstream/handle/10785/1218/Proyecto%20OVA%20de%20COMPUERTAS%20DIGITALES.pdf?sequence=1>>.
26. UNIV GRANADA. DISPOSITIVO PAR LA ESTIMACIÓN DE FLUJO ÓPTICO EN IMÁGENES MEDIANTE FPGAS. Inventores: Vidal, Fernando Ros; Fernandez, Sonia Mota; Alonso Antonio, Javier Diaz. Patente de aplicación. Numero de publicación: WO2005050342 A2. Fecha de publicación: 02 de junio de 2005
27. Dracotic S L. Dispositivo electronico de estaciones remotas para control a distancia. Gomez Skarmeta, Antonio Fernand; Monsalve Tresaco, Enrique; Zamora, Miguel Angel. Patente de aplicación. Numero de publicación: WO2008074905A1. Fecha de publicación: 26 de junio de 2008.
28. BARRIENTOS Antonio, PEÑIN Luis Felipe, BALAGUER Carlos, ARACIL Rafael. 2 ed. MGrav-Hill,2007.512 p. ISBN 9788448156367.
29. YEPEZ Mulia Enrique, YEPEZ Miztli Martínez. 1 ed. Universidad Nacional Autónoma de México,2007.389p. ISBN 9786070219337.
30. KUMAR Subir Saha. Introducción a la robótica.Traducido por Dr.Thomas W.Bartenbach. 1 ed.México.D.F.: MGrav-Hill,2010.403 p. ISBN 13: 978-607-15-0313-8

31. BARRIENTOS Antonio, PEÑIN Luis Felipe, BALAGUER Carlos, ARACIL Rafael. 2 ed. McGraw-Hill,2007.512 p. ISBN 9788448156367.
32. BARRIENTOS Antonio, PEÑIN Luis Felipe, BALAGUER Carlos, ARACIL Rafael. Fundamentos de Robótica. 2 ed. McGraw-Hill,2007.512 p. ISBN 9788448156367.
33. OLLERO Baturone Aníbal. ROBÓTICA Manipuladores y robots móviles. Publicado por Coedición Alfaomega y Marcombo. Barcelona , España. 2001 . 447 pag. ISBN: 970-15-0758-4.
34. OLLERO Baturone Aníbal. ROBÓTICA Manipuladores y robots móviles. Publicado por Coedición Alfaomega y Marcombo. Barcelona , España. 2001 . 447 pag. ISBN: 970-15-0758-4.
35. BARRIENTOS Antonio, PEÑIN Luis Felipe, BALAGUER Carlos, ARACIL Rafael. 2 ed. McGraw-Hill,2007.512 p. ISBN 9788448156367.
36. RAMIREZ Benavides Kryscia Daviana. Cinemática inversa del robot. Escuela de Ciencias de la Computación e Informática. Costa Rica. 56 p. Disponible en < <http://www.kramirez.net/wp-content/uploads/2012/04/CinematicaInversaRobot.pdf>>
37. RODRIGUEZ Zambrano. Modelo cinemático y control de un brazo robótico imprimible. Grado en ingeniería electrónica y automática industrial. España. Universidad Carlos III de Madrid. Disponible en < http://e-archivo.uc3m.es/bitstream/handle/10016/16924/TFG_Juan_Carlos_Rodriguez_Zambrana.pdf?sequence=3>
38. BEER, JOHNSTON, CORNWELL. Mecánica vectorial para ingenieros dinámica. 9 ed. McGraw Hill. 2010.1362 pag. ISBN: 978-607-15-0261-2
39. (**) Robocup evento para la promover la robótica y la investigación, ofreciendo un desafío público, pero un reto formidable. Para más información visite < <http://www.robocup.org/>>.
40. ABDALA Castillo Salomón, ÑECO Caberta Raúl . Caracterización de un robot manipulador articulado. Maestro en ciencias en Ingeniería Mecatrónica. México. Centro Nacional de Investigación y Desarrollo Tecnológico. 2003.196 p. disponible en < <https://www.cenidet.edu.mx/subaca/web-mktro/submenus/investigacion/tesis/3-4%20Salomon%20Abdala%20Castillo%20-%20Raul%20%C3%91eco%20Caberta.pdf>>
41. CIRERA Eduardo A, BRAVO Gerardo A, FILIPIGH Jorge E. Modelado de un brazo robótico mediante el método de Lagrange. República Argentina. Universidad Nacional del Nordeste comunicaciones científicas y tecnológicas. 2005.Disponible en < <http://www.unne.edu.ar/unnevieja/Web/cyt/com2005/7-Tecnologia/T-027.pdf>>
42. Robotis.Página oficial de los robots Bioloid.Disponible en<http://support.robotis.com/en/product/bioloid/premiumkit/download/bioloid_premium_demo.htm>
43. FUNDACION I+D+I. Desarrollo de objetos virtuales de aprendizaje-OVA.En la web, disponible en <<http://www.fundacionidi.org/index.php/unidad-ti/desarrollo-de-objetos-de-aprendizaje-ova>>