



UNIVERSIDAD SIMÓN BOLÍVAR  
DECANATO DE ESTUDIOS PROFESIONALES  
COORDINACIÓN DE INGENIERÍA DE LA COMPUTACIÓN

**Desarrollo de un prototipo robot humanoide que busque, encuentre y patee una  
pelota**

Por:  
Jennifer Dos Reis De Dobrega  
Juliana Leon Quinteiro

Realizado con la asesoría de:  
Ivette Carolina Martinez

PROYECTO DE GRADO  
Presentado ante la Ilustre Universidad Simón Bolívar  
como requisito parcial para optar al título de  
Ingeniero en Computación

**Sartenejas, Noviembre 2014**

## **Resumen**

AQUI DEBE IR EL RESUMEN

## DEDICATORIA

# Agradecimientos

AGRADECIMIENTOS

# Índice general

Índice general	VII
Índice de cuadros	IX
Índice de figuras	X
<b>1. Marco teórico</b>	<b>3</b>
1.1. Robótica . . . . .	3
1.2. Robótica Inteligente (Agentes Inteligentes) . . . . .	4
1.2.1. Paradigmas de robótica . . . . .	4
1.2.1.1. Paradigma Jerárquico . . . . .	5
1.2.1.2. Paradigma Reactivo . . . . .	5
1.3. Inteligencia Artificial . . . . .	5
1.3.1. Aprendizaje de Máquinas . . . . .	6
1.3.2. Aprendizaje por reforzamiento . . . . .	6
1.3.3. Q- learning . . . . .	6
1.4. Visión Artificial . . . . .	7
1.4.1. Filtros . . . . .	7
1.4.2. Transformaciones Morfológicas . . . . .	7
1.4.2.1. Dilatación . . . . .	7
1.4.2.2. Erosión . . . . .	8
<b>2. Tecnologías utilizadas</b>	<b>10</b>
2.1. Herramientas de software . . . . .	10
2.2. Componentes de hardware . . . . .	11
<b>3. Integración de componentes</b>	<b>17</b>

<b>4. Movimientos</b>	<b>19</b>
4.1. Movimiento del cuerpo . . . . .	19
4.2. Movimiento de la cámara . . . . .	20
<b>5. Implementación y resultados</b>	<b>21</b>
<b>6. Conclusiones y recomendaciones</b>	<b>22</b>
<b>Bibliografía</b>	<b>23</b>
<b>A. Archivos intermedios</b>	<b>25</b>

# Índice de cuadros

# Índice de figuras

1.1. Dilatación . . . . .	8
1.2. Erosión . . . . .	9
2.1. Bioloid Kit . . . . .	12
2.2. Motores Dynamixel conectados en serie . . . . .	12
2.3. Sensor Gyro . . . . .	12
2.4. Chip FTDI conectado a la tarjeta Arbotix . . . . .	13
2.5. Extensor de puertos bioloid . . . . .	13
2.6. Servo motor analogico . . . . .	14
2.7. Tarjeta Raspberry Pi con descripción de los puertos . . . . .	14
2.8. Camara Raspberry Pi . . . . .	15
2.9. Lipo . . . . .	15



# Introducción

RobotCup [201a] es una competencia de fútbol iniciada desde 1997 donde contribuyen las áreas de robótica, investigación e inteligencia artificial. Entre sus categorías se encuentra RobotCup Soccer [201b], la cual consiste en la participación de pequeños robots humanoides que se enfrentan a otro equipo para jugar fútbol. El objetivo de esta competencia es lograr que en el año 2050 el equipo campeón logre vencer al ganador del año en la copa mundial de la FIFA (International Federation of Association Football). Las destrezas de robots con forma de humanos (como son caminar, percibir el mundo y tomar alguna acción sobre él) suelen ser más complejas de lo que se puede pensar. Una de las más avanzadas muestras en el área es el robot ASIMO [Co11], creado por la compañía Honda, cuyos últimos avances incluyen la predicción de trayectoria de objetos para poder esquivarlos.

En este proyecto se presenta un robot humanoide (Debupa) de tamaño pequeño (38 cm de altura) cuyos objetivos, basados en las reglas de la competencia RobotCup, son: detectar una pelota de un color único en el ambiente, buscarla y, al llegar hasta ella, patearla. Además en el proceso debe detectar si ha perdido su equilibrio de tal manera que ha caído al suelo y debe ser capaz de levantarse.

En artículos relacionados de este mismo enfoque se puede encontrar el trabajo de Sven Behnke cuyo título es “See, walk, and kick: Humanoid robots start to play soccer” donde se describe la construcción del equipo de robots que participaron en la RobotCupSoccer en el año 2006. El artículo cubre el diseño mecánico y electrónico, además el software utilizado para

la percepción, control de comportamiento, comunicación y simulación de los robots. [BSS<sup>+</sup>].

Existen equipos que han participado durante varios años consecutivos en la competencia Robocup, logrando mejoras en sus diseños y técnicas; tal es el caso del equipo MRL que ha participado en los años 2011, 2012, 2013 y 2014 en la categoría “Humanoid League”, han iniciado con el hardware del robot DARwIn-OP y con el tiempo han modificado los componentes electrónicos para agregar eficiencia y estabilidad. Para el balance han utilizado un giróscopio y sensores de aceleración, y para la visión una cámara conectada por usb al CPU principal [SSA<sup>+</sup>].

En el desarrollo de habilidades más específicas con respecto a la competencia Robot-Cup Soccer, en el artículo de investigación de Seung-Joon Yi, Stephen McGill y Daniel D. Lee [YML], se refieren a dos posibles estrategias para el pateo de la pelota donde los factores fundamentales para un buen desempeño es la fuerza y la rapidez con que se patea, los investigadores ponen en práctica dos estrategias de pateo en distintas circunstancias del juego basado en la cinemática y dinámica de equilibrar el cuerpo al momento de realizar el pateo.

En la sección ?? se describen los componentes de hardware usados para construir el humanoide; luego en la sección ?? se explica cómo se unieron esas piezas. Con respecto a la parte de programación, en la sección ?? se describe cómo se logró constituir los movimientos necesarios para que el humanoide cumpla sus objetivos, mientras que en la sección ?? se muestran los resultados experimentales. Las herramientas y técnicas que permitieron lograr la detección de la pelota se detallan en la sección ?. También se describe la discretización del ambiente para reducir el número de estados. La comunicación de las tarjetas Arbotix y Raspberry Pi para que puedan trabajar en conjunto se explica en la sección ? y consideraciones especiales en la sección ?.

# Capítulo 1

## Marco teórico

En este capítulo se presentan los conceptos que conforman la base teórica para comprender el presente trabajo. Primero se brinda una descripción del término robótica y se definen algunos de los dispositivos utilizados en los robots. Posteriormente se describen algunos conceptos que tienen que ver con la robótica inteligente, por ejemplo el aprendizaje de máquinas. Finalmente se presentan los conceptos de visión artificial para detección de objetos.

### 1.1. Robótica

Para definir un lenguaje formal se requiere describir:

- **Robot:** Es un agente artificial, activo, cuyo entorno no es el mundo físico. El término activo descarta de esta definición a las piedras, el término artificial descarta a los animales, y el término físico descarta a los agentes de software puros o softbots, cuyo entorno lo constituyen los sistemas de archivos, bases de datos y redes de cómputo. [pet95]

- **Robótica:** Es la rama de la tecnología que se encarga del diseño, construcción, operación y aplicación de los robots. [Pre14]
- **Sensores:** Son los encargados de percibir el ambiente que rodea al robot. Según Murphy R.R son dispositivos que miden algún atributo del mundo. Un sensor recibe energía del entorno (sonido, luz, presión, temperatura, etc) y transmite una señal a una pantalla o computador ya sea de forma análoga o digital. [AiR00]
- **Actuador:** Es aquella parte del robot que convierte comandos de software en movimientos físicos. [pet95]
- **Servomotor:** Es un motor eléctrico, considerado como actuador, que permite ser controlado tanto en velocidad como en posición.
- **Giróscopio:** Es un sensor utilizado para medir y mantener la orientación, se mide a través del momento angular. [Con14]

## 1.2. Robótica Inteligente (Agentes Inteligentes)

### 1.2.1. Paradigmas de robótica

En la robótica inteligente, según Robin Murphy en [AiR00], existen tres paradigmas en los cuales se clasifica el diseño de un robot inteligente, estos paradigmas pueden ser descritos de dos maneras: la relación entre las primitivas básicas de la robótica percibir, planificar, actuar ,o de la forma en que los datos son percibidos y distribuidos en el sistema.

Percibir se refiere al procesamiento útil de la información de los sensores del robot. Planificar, cuando con información útil, se crea un conocimiento del mundo y se generan ciertas tareas que el robot podría realizar. Por último actuar consiste en realizar la acción correspondiente con los actuadores del robot para modificar el entorno.

#### **1.2.1.1. Paradigma Jerárquico**

Este paradigma es secuencial y ordenado. Primero el robot percibe el mundo y construye un mapa global. En base al mapa ya percibido y con “los ojos cerrados”, el robot planifica todas tareas necesarias para lograr la meta. Luego ejecuta la secuencia de actividades según la planificación realizada. Una vez culminada la secuencia se repite el ciclo percibiendo el mundo, planificando y actuando. [AiR00]

#### **1.2.1.2. Paradigma Reactivo**

El paradigma reactivo omite por completo el componente de la planificación y solo se basa en percibir y actuar. El robot puede mantener un conjunto de pares percibir-actuar, estos son llamados comportamientos y se ejecutan como procesos concurrentes. Un comportamiento toma datos de la percepción del mundo y los procesa para tomar la mejor acción independientemente de los otros procesos. [AiR00]

### **1.3. Inteligencia Artificial**

La inteligencia artificial es un término relacionado con la computación y la robótica que ha tenido varias definiciones, ocho de ellas, las cuales nacieron a finales del siglo XX, se encuentran organizadas en [pet95] bajo cuatro categorías: pensar y actuar de forma humana, pensar y actuar de forma racional. Con ello se puede entender que la inteligencia artificial tiene que ver con lograr que un robot resuelve problemas de manera inteligente, es decir, de manera que parezca que el razonamiento y comportamiento humano las ha resuelto.

### 1.3.1. Aprendizaje de Máquinas

El aprendizaje de máquinas es un área de la inteligencia artificial que está relacionada con la pregunta de cómo construir programas de computadora que automáticamente mejoren con la experiencia. Se dice que un programa aprende de la experiencia  $E$  con respecto a una tarea  $T$  y desempeño  $P$ . Si el desempeño en la tarea  $T$ , medido por  $P$ , mejora con la experiencia  $E$  [Mit06]

### 1.3.2. Aprendizaje por reforzamiento

El aprendizaje por reforzamiento es un tipo de aprendizaje de máquinas que se basa en un sistema de recompensas y penalizaciones. Las recompensas se pueden dar en cada estado o una sola vez al llegar al estado final.

El objetivo del agente es aprender de las recompensas para escoger la secuencia de acciones que produzca la mayor recompensa acumulada. [Mit06]

El agente existe en un entorno descrito por algunos estados  $S$ . Puede ejecutar un conjunto de acciones  $A$ . Cada vez que ejecuta una acción  $a$  en algún estado  $s$  el agente recibe una recompensa  $r$ . El objetivo es aprender una política  $\pi : S \rightarrow A$  que maximice la suma esperada de esas recompensas con descuento exponencial de las recompensas futuras. (Mitchell) El resultado de tomar las acciones puede ser determinista o no, en el caso de este proyecto no es determinista, es decir, existen porcentajes de probabilidad de pasar a un estado u otro al tomar una acción en un estado en particular.

### 1.3.3. Q- learning

Es un método de aprendizaje por reforzamiento

Compara las utilidades esperadas de acciones posibles sin necesidad de saber el resultado por tanto no se necesita un modelo del entorno [pet95]

## 1.4. Visión Artificial

Concepto Vision de computadoras (IMPORTANTE). En la inteligencia artificial es importante el reconocimiento de patrones y clasificación de objetos. El reconocimiento de objetos tiene como objetivo la extracción de información en una imagen y la interpretación de la misma, a través de un proceso algorítmico.

### 1.4.1. Filtros

El filtrado de imágenes es una técnica para la transformación de imágenes, que consiste en destacar sus características más relevantes orientadas a un propósito en particular.

Generalmente en la tarea de extracción de información de una imagen se utilizan filtros para descartar zonas o características que no son importantes para el patrón deseado y para determinar el área deseada ya sea por patrones de forma o color.

En la investigación, los algoritmos de filtrado aplicados a las imágenes fueron: Clausura Morfológica y Apertura Morfológica, filtros que aplican las técnicas de erosión y dilatación a las imágenes.

### 1.4.2. Transformaciones Morfológicas

Las transformaciones morfológicas básicas son llamadas dilatación y erosión, y si Image Morphology que se presentan en una amplia variedad de contextos como la eliminación del ruido, aislamiento de elementos individuales, elementos de unión disjuntos en una imagen. [Boo08]

#### 1.4.2.1. Dilatación

La dilatación es una convulsión de alguna imagen (o región de una imagen) , que llamaremos A, con un núcleo que llamaremos B, el núcleo que puede ser de cualquier forma o

tamaño, tiene un solo punto de anclaje definido. Muy a menudo, el núcleo es un pequeño cuadrado o disco sólido con el punto de anclaje en el centro. El núcleo puede ser pensado como una plantilla o mascarilla, y su efecto es que para la dilatación de un operador de máximo local sobre la imagen, se calcula el valor de píxel máximo común a B y reemplazamos el píxel de la imagen en el punto de anclaje con ese valor máximo. Esto causa regiones brillantes dentro de una imagen y la hacen crecer. Este crecimiento es el origen del término “operador de dilatación”. [Boo08]

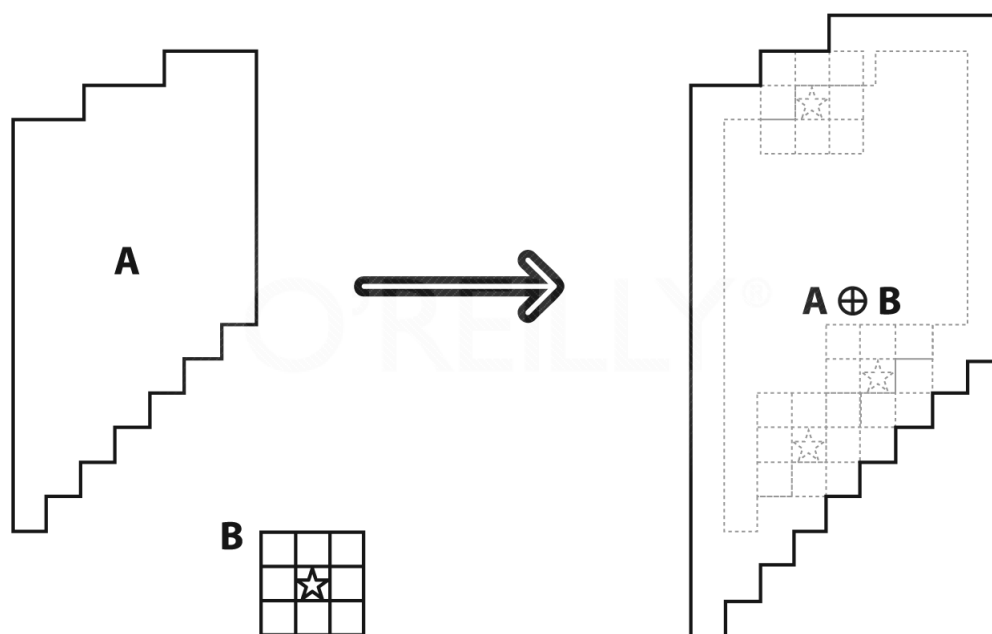


Figura 1.1: Dilatación

#### 1.4.2.2. Erosión

La erosión es la operación inversa a la dilatación. Esta acción del operador es equivalente a la erosión el cálculo de un mínimo local sobre el área del núcleo. La erosión genera una nueva imagen desde la original, utilizando el siguiente algoritmo: como el núcleo B es analizado sobre la imagen, se calcula el mínimo valor del píxel superpuesto por B y se reemplaza el píxel de la imagen con un punto de anclaje de valor mínimo. [Boo08]



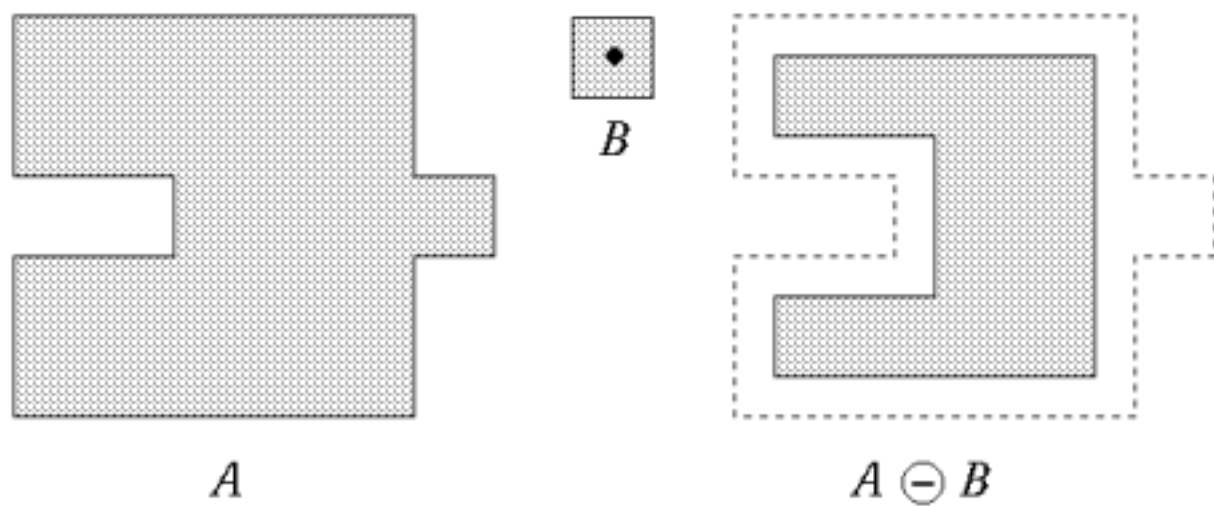


Figura 1.2: Erosión

# Capítulo 2

## Tecnologías utilizadas

PODRIA IR ACA LA PRESENTACION DEL PROBLEMA  
OBJETIVOS

### 2.1. Herramientas de software

En esta sección se describen las herramientas de software utilizadas para la programación del proyecto.

- Pypose: Software especializado en el control de los servomotores Dynamixel Ax-12. Una de las más importantes características es que, luego de haber fijado a mano las posiciones de los motores, permite la lectura simultánea de esas posiciones para captar la pose del robot. Con esta herramienta es posible formar una secuencia de poses que generen un movimiento, por ejemplo, caminar. [Fer10]
- ROS: ROS (Robot Operating System) es un framework que proporciona bibliotecas y herramientas para ayudar a los desarrolladores de software a crear aplicaciones robóticas. Proporciona abstracción de hardware, controladores de dispositivos, bibliotecas,

visualizadores, paso de mensajes, gestión de paquetes y más. ROS se encuentra bajo licencia de código abierto, la licencia BSD.

- OpenCv (Open Source Computer Vision Library): Es una librería de visión de computadoras y aprendizaje de maquinas de código abierto. Ha sido diseñada para acelerar el uso de la percepción de maquinas y para proveer una estructura común en las aplicaciones de visión de computadoras. Registrada bajo la licencia BSD, de código abierto. [Tea]
- IDE Arduino: Es un entorno de desarrollo para escribir y cargar código en la tarjeta Arduino. Otras tarjetas con microcontroladores AVR también son compatibles, como la Arbotix. El lenguaje de programación del IDE de Arduino es una implementación de Wiring el cual esta basado en Processing. [Ard14]

## 2.2. Componentes de hardware

En esta sección se describen los principales componentes utilizados para armar la estructura del robot.

- Bioloid Premium kit: Es un kit de robótica con piezas modulares que permite armar diferentes tipos de robot pero principalmente humanoides. El fabricante, ROBOTIS, incluye un manual con varios modelos de robots con instrucciones de ensamblaje. Provee una tarjeta controladora, CM-530, a la que se conectan los motores Dynamixel y algunos sensores que se programan a través de la interfaz de ‘RoboPlus’.
- Motores Dynamixel Ax-12+: Son actuadores inteligentes y modulares que incorporan un reductor de engranajes, un motor DC de presión y un circuito de control con funcionalidad de red, todo en un solo paquete. (R. INC, Dynamixel AX-12, 2006.)



Figura 2.1: Bioloid Kit

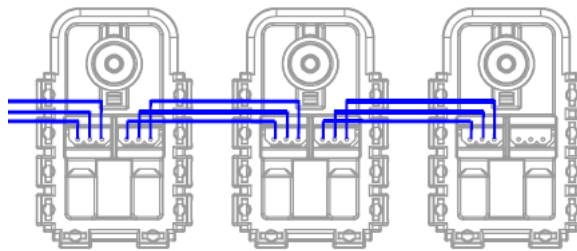


Figura 2.2: Motores Dynamixel conectados en serie

- Gyro: Es un giroscopio de la marca Robotis que mide la velocidad angular, diseñado para mantener el balance del robot y ser usado para otras aplicaciones de movimiento.

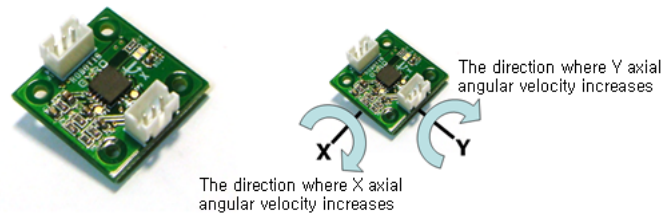


Figura 2.3: Sensor Gyro

- Arbotix: El controlador ArbotiX es una solución de control avanzado para manejar servos Dynamixel AX/MX/RX/EX y robots basados en Bioloid. Incorpora un potente microcontrolador AVR, radio inalámbrica XBEE, conductores de motor dual, y cabezas de estilo servo de 3 pines para E/S digital y analógica. [LLC]

- FTDI (Future Technology Devices International) : Es una tarjeta controladora que ofrece el servicio de conversión de datos de USB a UART. Permite la comunicación entre diferentes dispositivos.

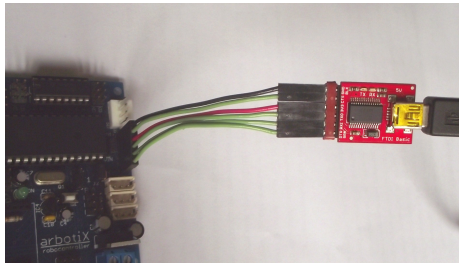


Figura 2.4: Chip FTDI conectado a la tarjeta Arbotix

- Extensor de puertos bioloid : Permite aumentar el número de cadenas de servos conectados a la tarjeta.

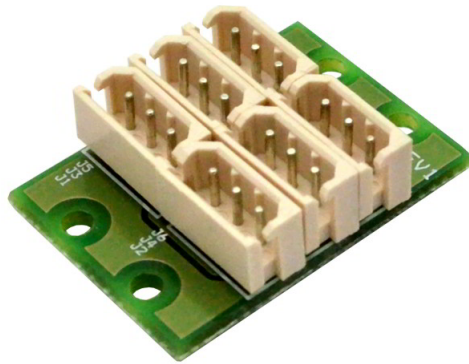


Figura 2.5: Extensor de puertos bioloid

- Servo motor analogico micro TG9 e: Es un pequeño servomotor cuyo torque alcanza 1.50 kg-cm y una velocidad de 60° por segundo. Permite ser controlado en posición en un rango de 180°.



Figura 2.6: Servo motor analogico

- Raspberry Pi: La Raspberry Pi es un ordenador del tamaño de una tarjeta de crédito a la que se puede conectar un televisor y un teclado. Se trata de un pequeño ordenador capaz de ser utilizado en proyectos de electrónica, y para muchas de las tareas que una PC de escritorio hace, como hojas de cálculo, procesadores de texto y juegos. <http://www.raspberrypi.org/>

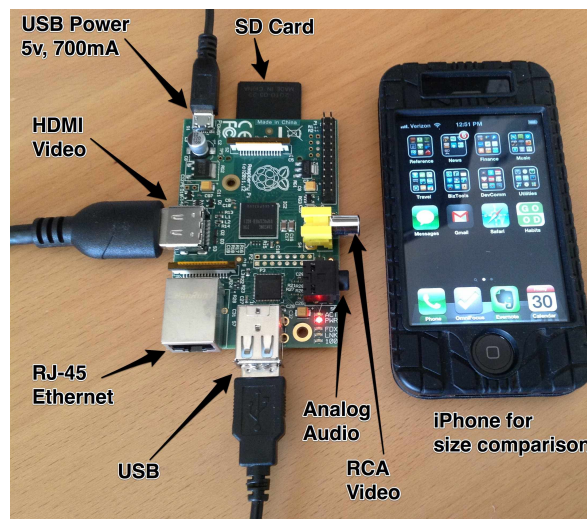


Figura 2.7: Tarjeta Raspberry Pi con descripción de los puertos

- Camara Raspberry Pi: Es un sensor encargado de captar imagenes y grabar videos de alta definicion. Se conecta a la Raspberry Pi con un cable de cinta plana de 15 cm en el puerto CSI. Tiene 5 megapíxeles de foco fijo que soporta los modos de vídeo de

1080x30, 720x60 y VGA90. Puede ser manejada con las librerías MMAL, V4L u otras librerías de terceros como la de Python.

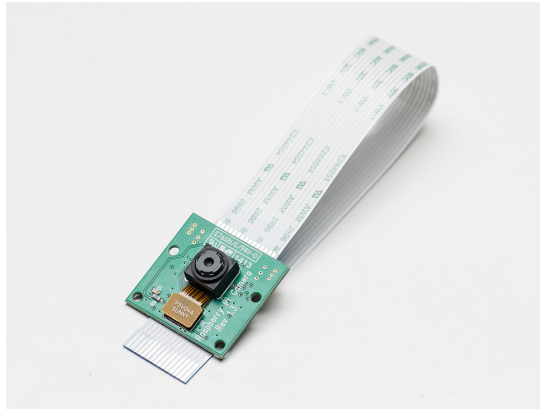


Figura 2.8: Camara Raspberry Pi

- Batería de polímero de litio (Lipo): Es la fuente de poder usada para que los motores y componentes electronicos funcionen. La batería usada es de 11.1 voltios y 1 amperio.



Figura 2.9: Lipo

- Circuito con regulador de 5v: Es un circuito diseñado y construido para este proyecto cuya finalidad es regular la entrada de la corriente. Por una de las salidas se expulsa 5v y por la otra se mantiene el mismo voltaje de entrada.



# Capítulo 3

## Integración de componentes

Para la construcción del robot se ha utilizado el kit de piezas Bioloid Premium de marca Robotis el cual incluye motores Dynamixel Ax-12+, una tarjeta controladora CM-510, un sensor Gyro, un manual, entre otros elementos. El manual incluye las instrucciones de como armar varios modelos de humanoide, el utilizado en este proyecto es el tipo B, haciendo uso de 16 motores. En la figura X y Y se puede observar la estructura del robot que aparece en el manual del kit.

Figura X vista frontal del robot. Se puede apreciar la identificación 'ID' de cada motor Dynamixel Ax-12+. Nota: los motores 9 y 10 no se utilizan

Figura Y. Vista trasera del robot

Manual de instrucciones y piezas del robot

En lugar de la utilización de la tarjeta CM-510 se ha decidido usar la tarjeta controladora Arbotix debido a que la controladora CM-510 no acepta la incorporación actuadores o dispositivos adicionales. La Arbotix permite la incorporación de nuevos actuadores y más dispositivos con sencillez. En la figura Z se puede observar la estructura del robot con la Arbotix incorporada. En la parte interna del tronco del robot se sitúa el sensor Gyro.

Vista trasera del robot con la Arbotix

Para el movimiento de la cámara se ha incorporado dos servomotores, uno para el movimiento horizontal y otro para el vertical. La conexión es pin a pin en los puertos especiales para ese tipo de motores ('Hobby servos') fuente (ver figura y). La cámara ha sido conectada a la Raspberry Pi en el puerto CSI (ver la figura). El resultado de estas tres piezas instaladas en el robot se puede apreciar en la figura x.

Figura y Ilustración de los puertos Hobby de la Arbotix

figura x vista delantera del robot con la cámara y servomotores instalados

Los motores Dynamixel se conectan a la controladora Arbotix por medio de los puertos bioloid de la tarjeta como se muestra en la figura BLA. Sin embargo como la tarjeta solo cuenta con tres puertos y el robot posee cuatro extremidades, se ha optado por agregar un expansor de puertos bioloid y así conectar cada extremidad en un puerto diferente.

La comunicación de la tarjeta de Arbotix con la computadora, incluso con la Raspberry Pi, se realiza a través del puerto FTDI por medio un chip conectado como lo ilustra la figura BLA.

Camara Raspberry Pi conectada al puerto CSI de la tarjeta

Figura BLA. Componentes conectados a la Arbotix, cable FTDI y un motor dynamixel conectados Imagen tomada de :

Como fuente de poder se ha utilizado una batería de polímero de litio de 11 V y 1 amp. Debido a que no todos los componentes poseen las mismas exigencias con respecto a voltaje y amperaje, se realizó un regulador (ver figura tal) con salida de 5 voltios para la tarjeta Raspberry Pi y los dos micro servomotores, y otra salida de 11v para la tarjeta Arbotix que a su vez alimenta a los componentes conectados en ella (motores Dynamixel y Giroscopio).

figura tal. Circuito con entrada de 11 V. Una salida de 5 v para los micro servomotores analogicos y tarjeta Raspberry Pi. Otra salida de 11 v para alimentar la controladora Arbotix.

# Capítulo 4

## Movimientos

### 4.1. Movimiento del cuerpo

Con fines explicativos, en este proyecto, la palabra ‘pose’ se referirá a la posición específica de los 16 motores que constituyen el esqueleto del robot. Un conjunto de poses ejecutadas en secuencia se denominará ‘acción de movimiento’.

Las acciones de movimiento establecidas son:

- Caminar hacia adelante
- Girar a la izquierda
- Girar a la derecha
- Levantarse cuando ha caído boca abajo
- Levantarse cuando ha caído boca arriba
- Patear con la pierna derecha
- Patear con la pierna izquierda

Existen también dos acciones de movimiento que no se encuentran relacionadas con la posición de los motores del esqueleto del robot sino a la posición de los motores que controlan la posición de la cámara. Estas se explicarán en la sección de movimiento de la cámara.

Las poses han sido fijadas a través de la tarjeta controladora Arbotix y el software Pypose. De esta manera se ha fijado y guardado un conjunto de poses para cada acción de movimiento. Luego se ha exportado el archivo de las poses y acciones de movimiento para ser utilizado en el programa, en lenguaje Wiring, a ser ejecutado en Arbotix. La programación en Arbotix se ha realizado bajo el ambiente del IDE de Arduino.

DESCRIBIR CADA POSE??? Creo q solo vale la pena si ponemos algunas secuencias de fotos. MAS ESPECIFICO COMO ES PYPOSE – ¿ en la definicion de pypose (pienso q es mejor)

## 4.2. Movimiento de la cámara

La cámara ha sido instalada sobre dos micro servomotores analógicos, otorgándole dos grados de libertad. El servomotor ubicado en la parte inferior se encarga del movimiento horizontal y el superior del movimiento vertical (ver figura x). Las acciones de movimiento relacionadas con el movimiento de la cámara se reduce a 9 posiciones fijas (ver figura C) cuya distribución obedece al objetivo de que la cámara obtenga una amplia visión, sin dejar espacios no visibles.

Estos motores se controlan desde la Arbotix usando la librería HServo. Esta librería solo puede ser usada para los motores conectados en los puertos Hobby A y B (pines 12 y 13) (ver la figura de la arbotix-puertosHobby ). Brinda la ventaja de un control más preciso, evitando que los motores tiemblen ya que los pulsos son generados por temporizadores de hardware.

# Capítulo 5

## Implementación y resultados

En el presente capítulo se explicará la implementación realizada según la especificación presentada los capítulo y . Así como también se presentarán los resultados obtenidos para los ejemplos dispuestos en el apendice .

## Capítulo 6

# Conclusiones y recomendaciones

La presente investigación ha nacido de la motivación por hacer que en Venezuela se incurriese en proyectos que involucren humanoides autónomos e inteligentes. Se ha inspirado especialmente en la categoría Robocup soccer de la competencia internacional Robocup. Desde 1997, fecha en la que inició la competencia, Venezuela nunca ha participado en categorías con humanoides, mientras que países latinoamericanos como México, Brasil y Colombia sí han tenido avances en este campo. Si bien este proyecto no cumple con todas las reglas de la competencia se espera que éste pueda dar pie a continuar investigaciones dentro de Venezuela.

Los componentes utilizados en este proyecto son relativamente económicos comparados con otros en el mercado. La integración del kit Bioloid Premium con la Arbotix y la Raspberry Pi, ha hecho posible construir un humanoide inteligente sin tener que invertir exorbitantes cantidades de dinero. Una de las contribuciones más importantes es la coordinación y paralelismo exitoso entre todos los componentes utilizados.

Las mejoras que se pueden incorporar al proyecto podrían ser: la inclusión de aprendizaje por reforzamiento para patear de forma exitosa, incluir que la patada sea en dirección al arco e incluso añadir aprendizaje por reforzamiento para hacer que el robot pueda predecir la posición de una pelota en movimiento para que pueda patearla en el momento indicado.

# Bibliografía

- [201a] RoboCup 2014. Robocup 2014.
- [201b] RoboCup 2014. Robocup soccer.
- [AiR00] *Introduction to AI Robotics*. 2000.
- [Ard14] Arduino. Arduino, 2014.
- [Boo08] *Learning OpenCV*. 2008.
- [BSS<sup>+</sup>] Sven Behnke, Michael Schreiber, Jörg Stückler, Reimund Renner, and Hauke Strasdat. See, walk, and kick: humanoid robots start to play soccer.
- [Co11] Honda Motor Co. Honda unveils all-new asimo with significant advancements, 2011.
- [Con14] Wolfram Demonstrations Project Contributors. Gyrocospe, 2014.
- [Fer10] M Fergs. Pypose, 2010.
- [LLC] Trossen Robotics LLC. arbotix robocontrollers.
- [Mit06] Tom M. Mitchell. The discipline of machine learning. 2006.
- [pet95] *Artificial Intelligence A Modern Approach*. 1995.
- [Pre14] Oxford University Press. Robotics, 2014.

- [SSA<sup>+</sup>] Mostafa E. Salehi<sup>1</sup>, Reza Safdari, Erfan Abedi, Bahareh Foroughi, Amir Salimi, Emad Farokhi, Meisam Teimouri, and Roham Shakiba. Mrl team description paper for humanoid kidsize league of robocup 2014.
- [Tea] OpenCV Developers Team. About.
- [YML] Seung-Joon Yi, Stephen McGill, and Daniel D. Lee. Improved online kick generation method for humanoid soccer robots.



# Apéndice A

## Archivos intermedios

APENDICE