

DESARROLLO DE ALGORITMOS PARA IDENTIFICACIÓN DE OBJETOS IMPLEMENTADOS CON MÉTODO DE MALLADO Y GRADIENTE EN UN ROBOT HUMANOIDE

DEVELOPMENT OF ALGORITHMS FOR IDENTIFICATION OF OBJECTS IMPLEMENTED WITH MESHING AND GRADIENT METHOD IN A HUMANOID ROBOT

José Guadalupe Zavala Villalpando

Tecnológico Nacional de México en Celaya, México
kg.zavala@itcelaya.edu.mx

Víctor Hugo Cacique Borrego

Tecnológico Nacional de México en Celaya, México
victor.cacique@itcelaya.edu.mx

Carolina Ruiz López

Tecnológico Nacional de México en Celaya, México
14031649@itcelaya.edu.mx

Abraham Morales González

Tecnológico Nacional de México en Celaya, México
14031730@itcelaya.edu.mx

Jesús Rafael Valdovinos Maldonado

Tecnológico Nacional de México en Celaya, México
14031732@itcelaya.edu.mx

Recepción: 16/septiembre/2019

Aceptación: 1/noviembre/2019

Resumen

El proyecto consistió en el desarrollo de algoritmos para la identificación de una pelota de tenis implementados con los métodos de mallado y gradiente con un módulo de procesamiento de imágenes (HaViMo 2.0) en un robot humanoide (Robot Bioloid Premium Tipo A) con veinte grados de libertad, el cual jugará la posición de un portero en competencias como la copa FIRA. Además, el robot fue capaz de posicionarse enfrente de la pelota. La programación fue realizada en lenguaje C usando el entorno de desarrollo integrado Eclipse y el controlador CM-530.

Palabras Clave: Método de gradiente, método de mallado, robot humanoide, visión.

Abstract

The project consisted in the development of algorithms for a tennis ball identification implemented with the methods of gridding and region growing with an image processing module (HaViMo 2.0) in a humanoid robot (Bioid Premium Type A Robot) with twenty degrees of freedom, which will play the position of a goalkeeper in competitions as FIRA RoboWorldCup. Also, was the robot able to sidewalk to position itself in front of the ball. The programming was made in C language using the integrated development environment Eclipse and the controller CM-530.

Keywords: *gradient method, meshing method, humanoid robot, vision.*

1. Introducción

La robótica busca la síntesis de las diversas funciones naturales que desarrolla el cuerpo humano de forma inherente, esto a través de la implementación de mecanismos, redes sensoriales y sistemas computacionales que permitan adquirir autonomía a cualquier sistema artificial creado [Vázquez, 2013].

Uno de los principales fines de la robótica es el de investigación, orientándose a crear nuevos algoritmos de inteligencia artificial que pueden ser extrapolados a diversas áreas. Es común simular tareas o comportamientos presentes en la naturaleza, tales como caminar, librar obstáculos, seguir líneas, realizar movimientos precisos, entre otros. Esto ha permitido desarrollar modelos y rutinas que han sido exitosamente empleados para ayudar al ser humano con sus tareas diarias [González, 2017].

Para lograr este cometido, alrededor del mundo se realizan una gran diversidad de eventos que buscan fomentar la investigación y la generación de nueva tecnología que permita llegar al fin ya mencionado. El evento de mayor renombre a nivel mundial, y que se celebra año con año, es el torneo de FIRA Roboworld Cup. En este torneo los robots utilizados para competir en la categoría de AndroSot deben cumplir con ciertas características como son que los tres robots (incluido el portero) sean bípedos, la longitud de sus pies no debe ser mayor al 70% de la altura del robot, la longitud de sus brazos no debe superar el 60% de la altura, así como tener un nivel de autonomía adecuado para el ambiente en el que los robots humanoides

interactúan, estos requisitos se dispusieron para lograr que en el año 2050 se pueda establecer un equipo de robots humanoides capaces de jugar contra un equipo compuesto de seres humanos [Vázquez, 2013].

Mediante estos eventos, FIRA espera ayudar a las mentes jóvenes a entender de una mejor manera y apreciar, con interés, los conceptos científicos y tecnológicos que se han desarrollado en estas competencias. FIRA cree que algunos de estos intereses serán el combustible para el desarrollo de habilidades científicas e ingenieriles, las cuales finalmente ayudarán a futuras investigaciones que sean de utilidad en diversas maneras a la humanidad.

La primera referencia considerada presenta una visión general de un robot humanoide futbolista en la Copa Mundial de FIRA 2007, construido a partir de hardware de fácil acceso, en el cual se emplea un algoritmo de visión para detectar los postes de las porterías, para la auto-localización del robot.

Los primeros resultados de la conexión de un módulo de visión HaViMo 2.0 en un Bioloid pertenecen al equipo DotMX, teniendo el objetivo de desarrollar un sistema de visión capaz de ejecutar tareas de búsqueda y seguimiento [Santoyo, 2013].

En la Copa FIRA 2012, se desarrolló el control por medio de visión de un robot portero que, por primera vez en la historia de las copas de robótica, mantuvo una racha de cero goles recibidos. El sistema de visión global de control requiere un procesamiento de imágenes en una computadora externa a cualquier robot. Con base a esto, se procedió a realizar la identificación de los elementos básicos en un partido, algunas características de la cancha y la pelota. Para este caso se hizo un análisis de los diferentes métodos empleados para la identificación de cada elemento tal como es mencionado por Xiaochuan Zhang [Vázquez, 2013].

A pesar de los avances tecnológicos implementados en los últimos años en este tipo de competencias, la mayoría de los equipos participantes en la categoría AndroSot no han implementado un sistema de visión individual a cada uno de los robots participantes; lo cual será un requisito indispensable para la Copa FIRA 2019, ya que la cámara superior con la que cada equipo ubicaba a sus jugadores dentro de la cancha será removida. Es por ello que en este proyecto se plantea el desarrollo de un sistema de visión individual para un Robot Bioloid Premium Tipo A con función

de portero, el cual no sólo podrá detectar la pelota oficial de dicha competencia (pelota oficial de tenis), sino que contará con dos grados de libertad (movimientos de cabeceo y guiñada) que, aunado a un método de procesamiento de imágenes (gradiente o mallado), le permitirá tomar decisiones exactas acerca de la posición de la pelota, y los movimientos que tendrá que realizar para seguirla lateralmente.

2. Métodos

Estructura del Robot

Se utilizó un Robot Bioloid Premium Tipo A de la marca Robotis, el cual cuenta con 18 grados de libertad, cada grado de libertad corresponde a un actuador Dynamixel AX-12 (Figura1) que le permiten hacer movimientos semejantes a los humanos. Los actuadores ya mencionados poseen una resolución de 0.35° , con un rango de 0 a 1024 pasos (0 a 300°). Es posible modificar el torque aplicado en ellos, así como la velocidad de movimiento; además cuentan con un protocolo de comunicación sencillo para aplicaciones en diferentes lenguajes de programación, como lenguajes C y C++.

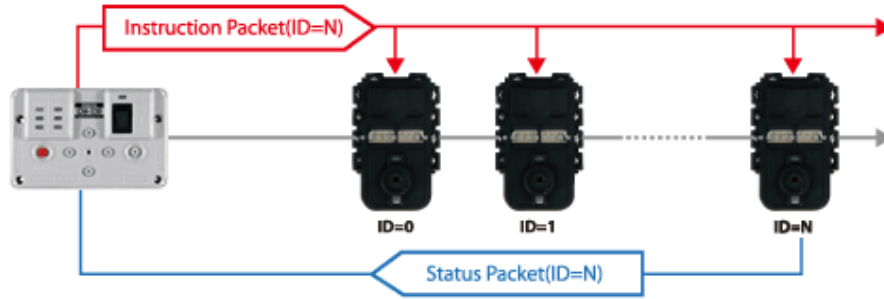


Fuente: User manual Dynamixel AX-12

Figura 1 Dynamixel AX-12.

El protocolo de comunicación empleado entre los actuadores Dynamixel AX-12 y el controlador CM530 es el Half-Duplex UART que consiste en el envío y recepción de paquetes de datos. Existen dos tipos de paquetes (Figura 2): de instrucción y de estado:

- Paquete de instrucción: Es enviado desde el controlador hacia los actuadores y se constituye de los elementos de la tabla 1.



Fuente: User manual Dynamixel AX-12

Figura 2 Protocolo de comunicación.

Tabla 1 Elementos de paquete de instrucciones.

0xFF	0xFF	ID	LENGTH	INSTRUCTION	PARAMETER 1...	PARAMETER N	CHECKSUM
------	------	----	--------	-------------	----------------	-------------	----------

Fuente: User manual Dynamixel AX-12

- Paquete de estado: Es enviado desde los actuadores hacia el controlador y se constituyen de los elementos de la tabla 2:
 - ✓ 0xFF: Los dos bytes indican el inicio de un paquete.
 - ✓ ID: Es el número identificador del actuador.
 - ✓ LENGTH: Longitud del paquete cuyo valor es el número de parámetros más 2.
 - ✓ INSTRUCTION: Es la instrucción que el actuador ejecutará (Posición, velocidad, torque, entre otros).
 - ✓ ERROR: Es un byte que representa los errores enviados por el actuador.
 - ✓ PARAMETER 1...N: Usado si es necesaria información adicional.
 - ✓ CHECKSUM: Es el cálculo para comprobar que el paquete es correcto. Se calcula sumando $\sim (ID + LENGTH + INSTRUCTION + PARAMETER 1 + \dots + PARAMETER N)$. \sim representa la operación lógica NOT.

Tabla 2 Elementos de paquete de estado.

0xFF	0xFF	ID	LENGTH	ERROR	PARAMETER 1...	PARAMETER N	CHECKSUM
------	------	----	--------	-------	----------------	-------------	----------

Fuente: User manual Dynamixel AX-12

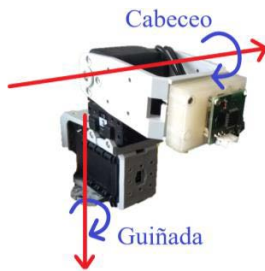
Al Robot Bioloid Premium Tipo A se le añadieron dos grados de libertad para los movimientos de la cabeza, la cual fue armada con piezas del propio kit de Robotis, dos actuadores extras y una cámara HaViMo 2.0 (Figura 3).



Fuente: User manual Dynamixel AX-12

Figura 3 Robot Bioloid Premium Tipo A.

Estos dos actuadores extras le permiten realizar los movimientos de guiñada y cabeceo (Figuras 4, 5 y 6), con los cuales podrá localizar la pelota mediante los algoritmos de visión.



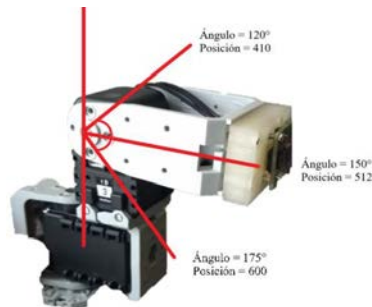
Fuente: User manual Dynamixel AX-12.

Figura 4 Rotaciones de la cabeza.



Fuente: User manual Dynamixel AX-12

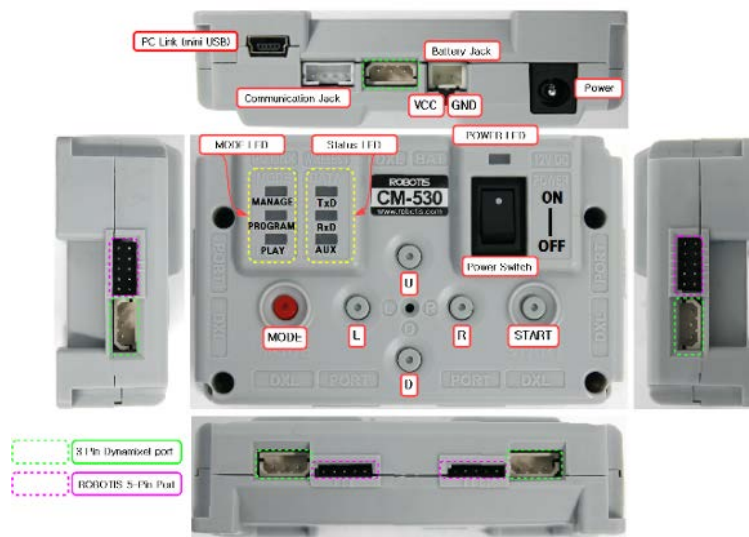
Figura 5 Límites de la guiñada.



Fuente: User manual Dynamixel AX-12

Figura 6 Límites de cabeceo.

Todos los datos que se obtienen de la cámara son procesados por el controlador CM-530 (Figura 7), el cual contiene un ARM Cortex STM32F103RE de 32 bits. El controlador posee una arquitectura de botones, leds indicadores, así como 6 puertos auxiliares de 5 pines compatibles con OLLO para la expansión de sensores y dispositivos terceros. A su vez contiene 5 puertos de bus TTL de bioloid para controlar un número grande de actuadores de la serie AX, dichos puertos son compatibles con el módulo de visión implementado en este proyecto.



Fuente: ROBOTIS e-manual

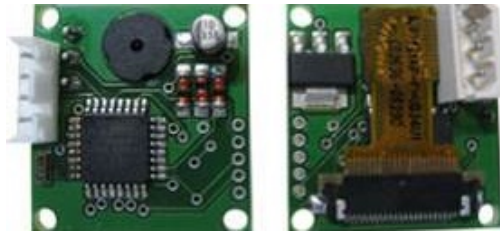
Figura 7 Controlador CM-530.

Módulo de Visión

Actualmente se busca que los robots humanoides se asemejen lo mejor posible a la anatomía y comportamiento humano, esto conlleva a que sean capaces de detectar objetos a su alrededor y tomar decisiones de manera eficaz. Para ello deben contar con un sistema de visión. HaViMo es una pequeña cámara que con dos métodos básicos de procesamiento de imágenes precargados (gradiente y mallado) es capaz de identificar colores en el espacio RGB (Figura 8). Cuenta con una resolución de 160x120 píxeles y una velocidad de captura de 19 fps.

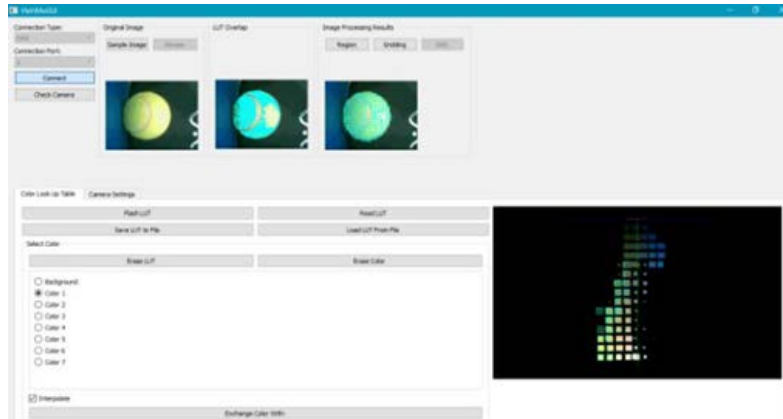
Para que la cámara pueda identificar los colores y aplicar los dos métodos de procesamiento de imagen, es necesario hacer un proceso de calibración previo, para esto la cámara cuenta con una interfaz gráfica de usuario llamado HaViMo GUI

(Figuras 9 y 10), en el que se toma una captura del objeto a identificar y se seleccionan los colores que conforman al objeto. Se recomienda hacer varias capturas del objeto en diferentes posiciones para que la gama de tonos del color sea más amplia y se pueda identificar el objeto, aunque esté muy lejos o muy cerca.



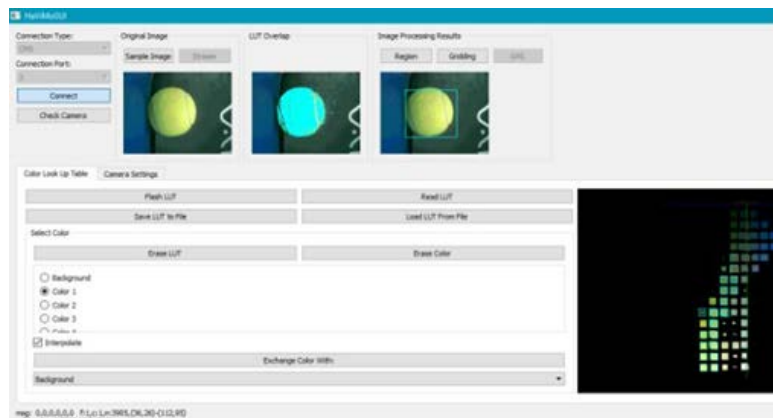
Fuente: Elaboración propia

Figura 8 HaViMo 2.0.



Fuente: Elaboración propia

Figura 9 HaViMo GUI, calibración y método de mallado.



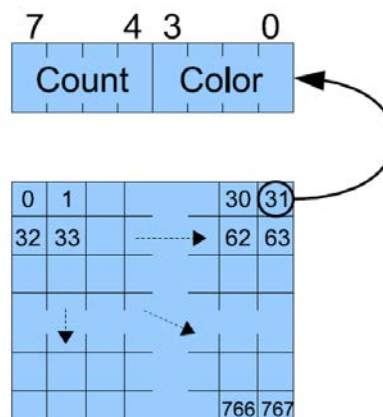
Fuente: Elaboración propia

Figura 10 HaViMo GUI, calibración y método de gradiente.

Una de las características de la cámara es la capacidad de calibrar 7 colores diferentes, los cuales se quedan guardados en la memoria EEPROM de la cámara y así no es necesario recalibrar en caso de apagarse y volverse a encender.

Método de mallado

El método de mallado comprime la imagen en una cuadrícula de 32x24, después hace un escaneo de cuadro por cuadro en busca de los colores calibrados, cabe mencionar que cada cuadro contiene 25 píxeles, es decir que tienen una dimensión de 5x5. El método va guardando en arreglos de 8 bits el color predominante y su número de píxeles, esto con cada uno de los recuadros, donde los 4 bits menos significativos son el color predominante y los 4 más significativos son el número de píxeles (Figura 11).



Fuente: *Embedded Vision Module for Bioloid Quick Start. Technical Report.*

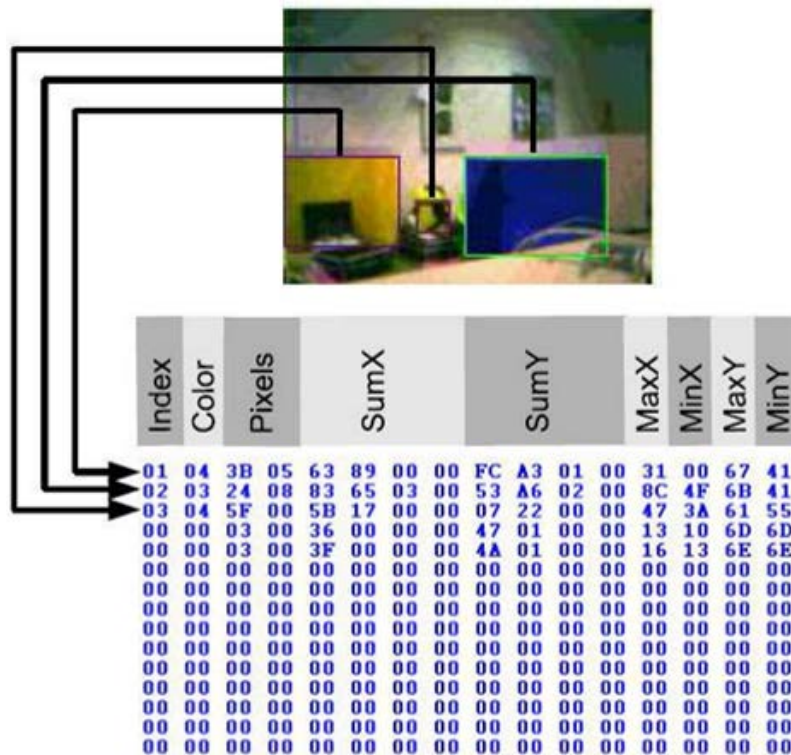
Figura 11 Matriz resultante del método de mallado.

Método de gradiente

El método de gradiente escanea toda la imagen y busca regiones de color continuas, se necesita al menos 4 píxeles vecinos para establecer si hay una conexión entre regiones del mismo color, al finalizar el algoritmo, todas las regiones continuas de un mismo color son sumadas y encerradas en un recuadro. La cámara puede detectar 15 regiones y va guardando en una matriz de 16x15 los datos de cada región (Figura 12). Entre los datos se encuentran:

- Index: Corresponde al número de región.

- Color: Color predominante de la región previamente calibrado.
- Píxeles: Número de píxeles del color dentro de la región.
- SumX: Sumatoria de las coordenadas en X de los píxeles detectados.
- SumY: Sumatoria de las coordenadas en Y de los píxeles detectados.
- MaxX: Coordenada máxima en X de la región.
- MinX: Coordenada mínima en X de la región.
- MaxY: Coordenada máxima en Y de la región.
- MinY: Coordenada mínima en Y de la región.

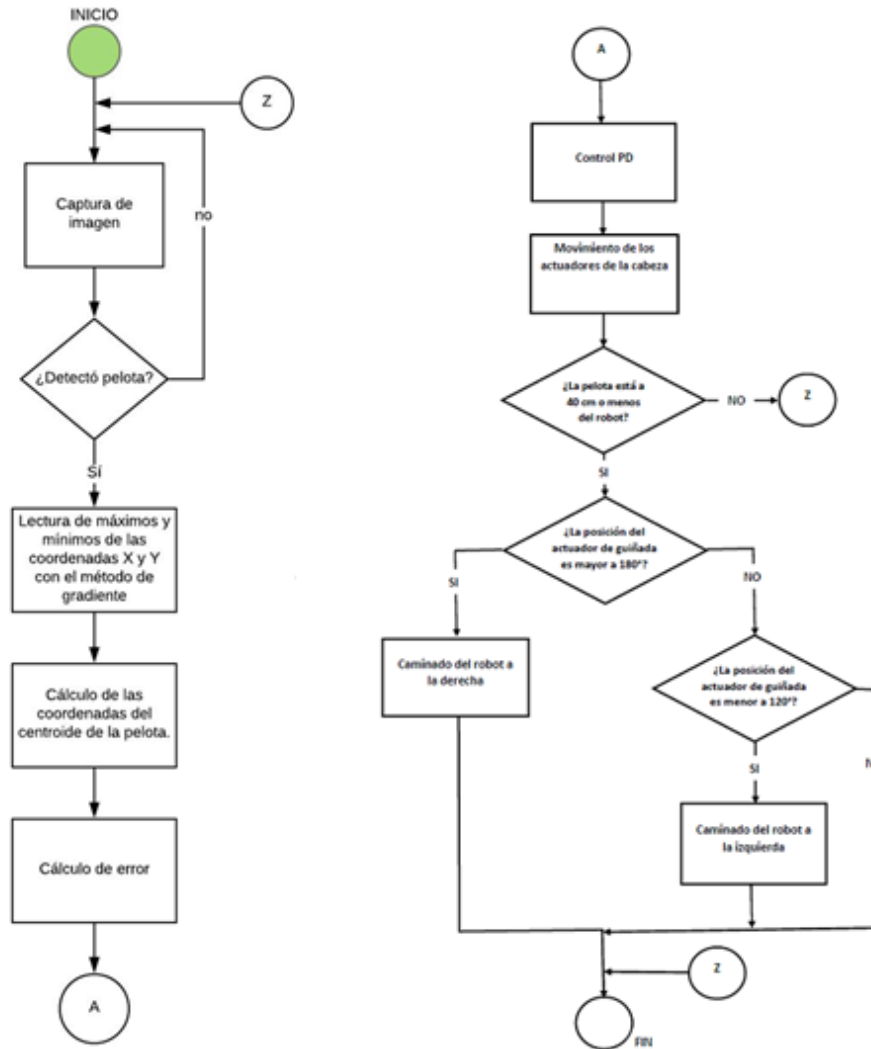


Fuente: Embedded Vision Module for Bioloid Quick Start. Technical Report.

Figura 12 Matriz resultante del método de gradiente.

Programación

Se desarrolló el algoritmo de visión y movimientos del robot en lenguaje C en el software Eclipse, el cual es una plataforma de código abierto compatible con el controlador CM-530. La lógica de programación se muestra en el diagrama de flujo (Figura 13).



Fuente: Elaboración propia

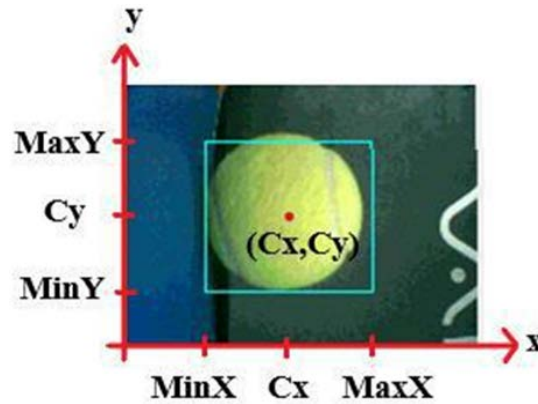
Figura 13 Diagrama de flujo.

Localización del centroide

Para poder seguir la pelota es necesario conocer su ubicación actual. Gracias al método de gradiente se puede calcular y localizar el centroide del objeto. El centroide está conformado por dos coordenadas, una coordenada en X (C_x) y una coordenada en Y (C_y) (Figura 14). Una forma de obtenerlo es mediante ecuaciones 1 y 2.

$$C_x = \left(\frac{MaxX - MinX}{2} \right) + MinX \quad (1)$$

$$C_y = \left(\frac{MaxY - MinY}{2} \right) + MinY \quad (2)$$

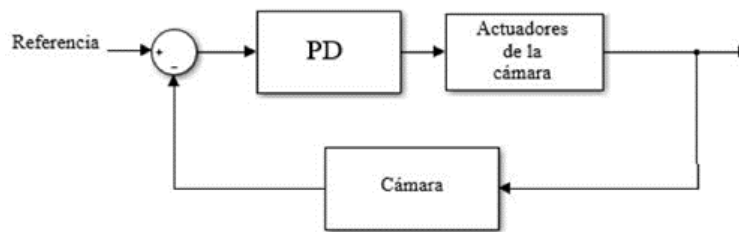


Fuente: Elaboración propia

Figura 14 Coordenadas del centroide del objeto.

Control

Se implementó un control proporcional y derivativo en los actuadores de la cabeza, con el fin de tener un movimiento preciso (Figura 15).



Fuente: Elaboración propia

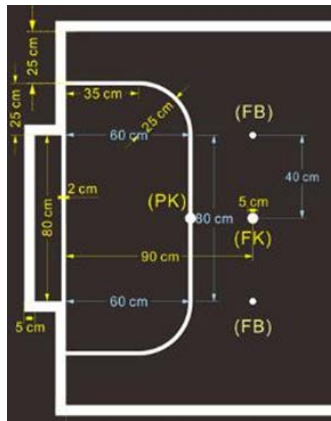
Figura 15 Esquema del control PD.

Es importante señalar que se implementaron 2 controles PD, uno para el actuador correspondiente al eje de las abscisas, y otro para el eje de las ordenadas. La referencia del sistema es el centro de la captura dada en unidades de píxeles: la coordenada 80 para el eje X y la coordenada 60 para el eje Y. Se calculó un error mediante la referencia y la posición actual de la pelota; este ingresa al control PD el cual ejecuta una acción para que los actuadores se muevan a una posición. Posteriormente por medio de la cámara se retroalimenta la nueva posición de la pelota y así sucesivamente, hasta que el error sea cero.

La sintonización de ambos controladores se realizó proponiendo una constante arbitraria K_c con la cual el sistema oscilaba y mediante el segundo método de Ziegler-Nichols se determinaron las constantes K_p y K_d .

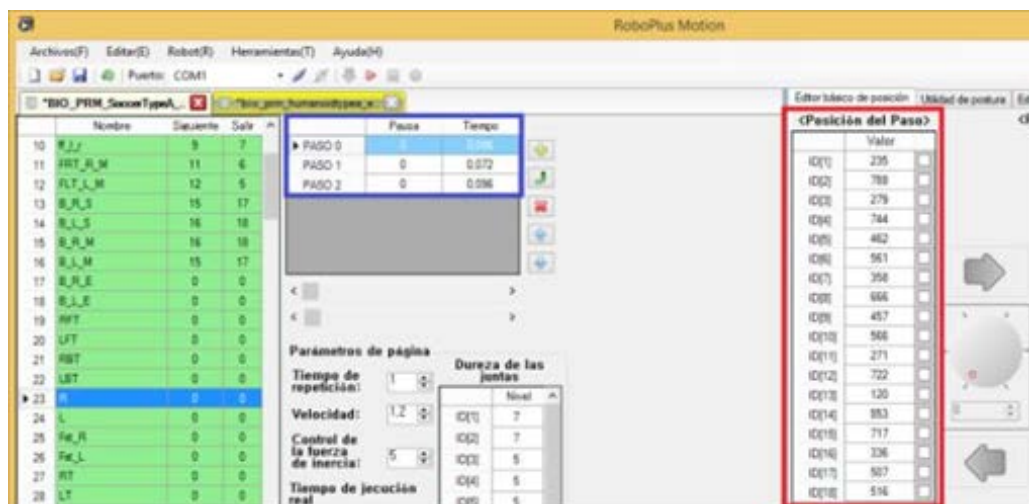
Algoritmos de giro del robot

El desarrollo de los algoritmos de giro se basó en programas ya preestablecidos y proporcionados por la compañía ROBOTIS, utilizando el software RoboPlus en conjunto con la herramienta RoboPlus Motion. Al tratarse de un portero este debe estar dentro del área chica de la portería con dimensiones de 130 x 60 cm (Figura 16) de acuerdo con la competencia FIRA en la categoría AndroSot.



Fuente: Federation of International Robot-soccer Association
Figura 16 Dimensiones de la portería.

En el software (Figura 17) se pueden visualizar varios parámetros como: la duración del paso, la posición de cada actuador, entre otros.



Fuente: Elaboración propia
Figura 17 Software RoboPlus Motion.

Los pasos de los movimientos de RoboPlus Motion se migraron al lenguaje C, con el objetivo de unirlos con el algoritmo de visión. Cabe mencionar que fue necesario modificar velocidades y torques de los actuadores con el fin de obtener un caminado semejante al presentado en RoboPlus Motion.

3. Resultados

De manera cualitativa se determinó que se necesita realizar un procesamiento extra en el método de gradiente para obtener información necesaria para el cálculo del centroide. Además, se obtuvieron los siguientes datos cuantitativos (Tabla 3).

Tabla 3 Resultados cuantitativos de parámetros para identificación de pelota.

Parámetros	Resultados obtenidos
Velocidad máxima de la pelota para ser seguida con la cabeza.	25 cm/s
Alcance máximo para identificación de la pelota.	1.5 m
Rango de ángulos de la cabeza en donde el robot empezará a caminar.	60-120° 180- 240°
Velocidad de desplazamiento lateral.	2 cm/s

Fuente: Elaboración propia

Se determinó que el mejor método de procesamiento de imagen para este proyecto fue el de gradiente, ya que la información que podemos obtener de dicho método nos brinda todo lo necesario para calcular el centroide de la pelota, a diferencia del método de mallado, en donde se requeriría cálculos extras para determinar el centroide, utilizando más recursos del procesador.

El seguimiento de la pelota es correcto siempre y cuando la pelota no sobrepase una velocidad de 25 cm/s. Así mismo el robot puede moverse lateralmente a una velocidad no mayor a 2 cm/s para posicionarse en frente de la pelota, si ésta se encuentra a una distancia menor a 40 cm y mayor a 15 cm del robot.

4. Discusión

Las mejoras realizadas con respecto al procesamiento, fue la utilización del controlador CM-530, el cual cuenta con mayor procesamiento de datos en

comparación con el CM-510; y mejor rendimiento de la batería en comparación con la Raspberry Pi 3. Ambos han sido usados con anterioridad en las competencias de la copa FIRA. Se determinó que el mejor método para el procesamiento de imágenes fue el método de gradiente debido a la facilidad que brinda para poder localizar la pelota usando menos recursos del procesador del controlador, debido a esto se pudo detectar la pelota a una velocidad de 25 cm/s. Con respecto al control usado para mover los actuadores de la cabeza, se obtuvieron mejores resultados con un control PD en comparación con un control PID, debido a que se buscaba que el sistema fuera de respuesta rápida y no se requería reducir el error en estado estable por la razón de que la pelota rara vez estaría estática.

5. Bibliografía y Referencias

- [1] González Segura, C. M., García García, M.L., Narváez Díaz, E. & Bacab Pech, R. (2017). Robot Bioloid Premium Jaranero Controlado Remotamente Por Voz, México: Pistas Educativas, vol. 39, no. 126, pp. 99-117.
- [2] Vázquez Vera, J. L., Santoyo Mora, M., Martínez Robles, E. E., Camarillo Gómez, K. A., Pérez Soto, G. I., Padilla Medina, J. A. & Cacique Borrego, V. H. (2013). Control por Visión de Robots Humanoides para FIRA Roboworld Cup 2013. México: XV COMROB.
- [3] Santoyo Mora, M., Cacique Borrego, V. H., Camarillo Gómez, K. A. & Ibarra Zannatha, J. M. (2013). Programming the Goalkeeper of 20 DOF for FIRA Cup. Malasia: 16th FIRA Roboworld Cup.
- [4] Computer Science Department of Freie Universität Berlin: Embedded Vision Module for Bioloid Quick Start. Technical Report.
- [5] ROBOTIS: User manual Dynamixel AX-12. Technical Report.
- [6] ROBOTIS, e-Manual, <http://support.robotis.com/en/>
- [7] Federation of International Robot-soccer Association, Sitemap, <http://www.fira.net/?mid=firaoverview>