



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

TEMA:

**Diseño e implementación de controladores PID en robots
humanoides utilizando MatLab/Simulink**

AUTOR:

Ávila Vera, Jhon Alexander

Componente práctico del examen complejo previo a la
obtención del grado de **INGENIERO ELECTRÓNICO EN
CONTROL Y AUTOMATISMO**

REVISOR:

M.Sc. Romero Paz, Manuel de Jesús

Guayaquil, Ecuador

10 de marzo del 2022



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

CERTIFICACIÓN

Certificamos que el presente **Componente Práctico del Examen
Complejivo**, fue realizado en su totalidad por **Ávila Vera, Jhon Alexander**
como requerimiento para la obtención del título de **INGENIERO
ELECTRÓNICO EN CONTROL Y AUTOMATISMO**.

REVISOR

M. Sc. Romero Paz, Manuel de Jesús

DIRECTOR DE CARRERA

M. Sc. Héras Sánchez, Miguel Armando

Guayaquil, a los 10 días del mes de marzo del año 2022



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

DECLARACIÓN DE RESPONSABILIDAD

Yo, **Ávila Vera, Jhon Alexander**

DECLARÓ QUE:

El **Componente Práctico del Examen Complexivo, Diseño e implementación de controladores PID en robots humanoides utilizando MatLab/Simulink**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 10 días del mes de marzo del año 2022

EL AUTOR

ÁVILA VERA, JHON ALEXANDER



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

AUTORIZACIÓN

Yo, **Ávila Vera, Jhon Alexander**

Autorizó a la Universidad Católica de Santiago de Guayaquil a la **publicación** en la biblioteca de la institución del **Componente Práctico del Examen Complexivo, Diseño e implementación de controladores PID en robots humanoides utilizando MatLab/Simulink** cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 10 días del mes de marzo del año 2022

EL AUTOR

ÁVILA VERA, JHON ALEXANDER

REPORTE DE URKUND

The screenshot shows the URKUND interface with the following details:

- Documento:** Avila Jhon EC.docx (D130292050)
- Presentado:** 2022-03-13 23:09 (-05:00)
- Presentado por:** fernandopm23@hotmail.com
- Recibido:** edwin.palacios.ucsg@analysis.orkund.com
- Mensaje:** Revisión Examen Complejivo Jhon Avila. [Mostrar el mensaje completo](#). 1% de estas 17 páginas, se componen de texto presente en 1 fuentes.

The 'Lista de fuentes' (List of sources) panel on the right includes:

- Trabajo de Titulación-Zambrano Jose- Cabr...
- Trabajo de Titulación-Zambrano Jhse-Cabr...
- ProyectoFinalIP3.docx
- Fuentes alternativas:
 - <https://www.editores-srl.com.ar/CONTROL...>
 - <http://www.frir.uctn.edu.ar/archivos/alumno...>

UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA

ELECTRÓNICA EN CONTROL Y AUTOMATISMO

TEMA: Diseño e implementación de controladores PID en robots humanoides utilizando Matlab/Simulink

AUTOR: Ávila Vera, Jhon Alexander

Componente práctico del examen complejo previo a la obtención del grado de INGENIERO ELECTRONICO EN CONTROL Y AUTOMATISMO

REVISOR: M. Sc. Romero Paz, Manuel de Jesús

Guayaquil, Ecuador

31 de enero del 2022

REVISOR

M. Sc. Romero Paz, Manuel de Jesús

DEDICATORIA

Dedico este logro a mis padres Jhon Ávila y Elsa Vera por su apoyo brindado, por estar ahí conmigo aconsejándome, acompañándome en todos los momentos buenos y malos, y estar pendiente de mí a cada instante de mi vida. Son un apoyo fundamental para mí.

Muchas gracias los amo.

EL AUTOR

ÁVILA VERA, JHON ALEXANDER

AGRADECIMIENTO

En primer lugar, se lo dedico a Dios por haberme dado salud y sabiduría para poder cumplir este objetivo.

A mis padres John Ávila y Elsa Vera por apoyarme incondicionalmente en todas las etapas de mi vida, enseñándome buenos valores, para obtener con éxito este logro.

A mis hermanos Joao, John Alex, Joel, Daniela, y a mi cuñada Michelle que es como mi hermana y amiga fundamental, gracias por apoyarme, brindarme consejos y directrices.

Mi abuelita Norma Saltos, por sus oraciones, por brindarme su hogar cuando estaba por culminar mis estudios de secundaria, en donde aprendí buenas costumbres y valores.

Finalmente, a mis docentes, que gracias a ellos he aprendido lo que se, brindándome consejos y buenas enseñanzas para poder cumplir con esta etapa de mi vida.

EL AUTOR

ÁVILA VERA, JHON ALEXANDER



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

TRIBUNAL DE SUSTENTACIÓN

f.

M. Sc. ROMERO PAZ, MANUEL DE JESUS

DECANO

f.

M. Sc. VALLEJO SAMANIEGO, LUIS VICENTE

COORDINADOR DEL ÁREA

f.

M. Sc. PALACIOS MELÉNDEZ, EDWIN FERNANDO

OPONENTE

Índice General

Índice de Figuras	xi
Índice de Tablas.....	xiv
Resumen	xv
Capítulo 1: Descripción del componente práctico	2
1.1. Introducción.....	2
1.2. Objetivo General.	2
1.3. Objetivos Específicos.	2
Capítulo 2: Fundamentación teórica.	4
2.1. Introducción de la mecánica de robots.....	4
2.2. Fundamentos de sistemas de control en la robótica.	4
2.3. Controladores PID.....	5
2.4. Parámetros de controladores PID.	6
2.5. Acción P, I y D.....	7
2.5.1. Acción P (proporcional)	7
2.5.2. Acción I (integral).....	7
2.5.3. Acción D (derivador)	8
2.6. Formas del controlador PID.....	8
2.6.1. Forma estándar	8
2.6.2. Forma paralela.....	8
2.6.3. Forma serie.....	9
2.7. Servocontrol visual	9
2.7.1. Posicionamiento de la cámara.....	10
Capítulo 3: Diseño, Simulación y Resultados obtenidos.....	12
3.1. Descripción del componente práctico.....	12
3.2. Comunicación del robot con la computadora.	12

3.3.	Algoritmos de control aplicados en el robot.....	14
3.3.1.	Diseño de controlador PID aplicados al robot.....	15
3.3.2.	Aplicaciones de servo visuales basadas en posición	22
	Conclusiones	35
	Recomendaciones.....	36
	Bibliografía.....	37

Índice de Figuras

Capítulo 2:

Figura 2. 1: Diagrama de bloques del controlador de bucle cerrado.....	4
Figura 2. 2: Diagrama de bloques del sistema de control PID.	5
Figura 2. 3: configuraciones de cámara-robot más comunes.	10
Figura 2. 4: Marcos de coordenadas relevantes; mundo, efector final, cámara y objetivo.....	10
Figura 2. 5: Estructura dinámica de ver y mover basada en la posición.	11

Capítulo 3:

Figura 3. 1: Servomotor Dynamixel modelo AX-18A utilizado en Robotis Bioloid GP.....	12
Figura 3. 2: Representación de la conexión en serie de los servomotores AX-18A.	13
Figura 3. 3: Minicomputadora Gigabyte con memoria 32 GB.	13
Figura 3. 4: Generación de la señal de entrada PRBS.	13
Figura 3. 5: Estructura del paquete de comunicación RS-485.....	14
Figura 3. 6: Diagrama de bloques del controlador PID para robot humanoide.	16
Figura 3. 7: Diagrama de bloques del controlador PID de 2 DoF para robot humanoide.	16
Figura 3. 8: Gráficos del controlador PID y PID de 2 DoF obtenidos para $x=26$	19
Figura 3. 9: Gráficos del controlador PID y PID de 2 DoF obtenidos para $y=27$	19
Figura 3. 10: Gráficos del controlador PID y PID de 2 DoF obtenidos para $z=-1$	20
Figura 3. 11: Gráficos del controlador PID y PID de 2 DoF obtenidos para $x=15$	20

Figura 3. 12: Gráficos del controlador PID y PID de 2 DoF obtenidos para $y=31$	21
Figura 3. 13: Gráficos del controlador PID y PID de 2 DoF obtenidos para $z=-3$	21
Figura 3. 14: Bloque de dispositivo de vídeo Matlab/Simulink.	23
Figura 3. 15: Obtención de una imagen en B/N a partir de una imagen a color	23
Figura 3. 16: Descripción gráfica de la proyección de imagen del objeto en la cámara.....	24
Figura 3. 17: Conjuntos de ejes de objeto-cámara-brazo.	25
Figura 3. 18: Imagen que ha sido distorsionada respecto a la imagen original.	26
Figura 3. 19: Imagen que ha sido distorsionada respecto a la imagen original.	27
Figura 3. 20: Diseño del controlador del servomotor basada en la posición.	28
Figura 3. 21: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $x=1.73$	30
Figura 3. 22: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $y=29.73$	30
Figura 3. 23: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $z=-10.23$	31
Figura 3. 24: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $x=10.06$	31
Figura 3. 25: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $y=27.68$	32
Figura 3. 26: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $z=-2.35$	32
Figura 3. 27: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $x=8.67$	33
Figura 3. 28: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $y=27.68$	33

Figura 3. 29: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $z=-2.85$ 34

Índice de Tablas

Capítulo 3:

Tabla 3. 1: Valores de posición del robot humanoide.	17
Tabla 3. 2: Valores para brazo izquierdo del robot humanoide.....	29

Resumen

El presente documento presenta el desarrollo del componente práctico del examen complejo denominado “Diseño e implementación de controladores PID en robots humanoides utilizando MatLab/Simulink”. En el capítulo 1, se presenta la parte descriptiva de este componente práctico. En el capítulo 2 se discute los fundamentos teóricos de sistemas de control en la robótica y del servocontrol visual. Se ha visto que los servosistemas pueden tener defectos, una precisión insuficiente, una estabilidad demasiado relativa, un tiempo de reacción demasiado lento, un rebasamiento excesivo. En el capítulo 3 se presenta el diseño, simulación y análisis de resultados. Por ejemplo, para la determinación del controlador se basó en las especificaciones que se pueden establecer ya sea en velocidad, precisión, tiempo de reacción, entre otras. En este caso el controlador PID (Proporcional-Integral-Derivativo) se encarga de regular el servomotor utilizado en un robot humanoide. También, se realizó el posicionamiento de objetos mediante una cámara integrada al robot humanoide utilizando el servocontrol visual.

Palabras claves: ROBÓTICA, CONTROLADORES, SERVOCONTROL, POSICIONAMIENTO, PARÁMETROS, SIMULACIÓN.

Capítulo 1

Descripción del componente práctico

1.1. Introducción.

El aumento de los productos que las personas utilizan en su vida diaria ha aumentado la necesidad de robots. Se han adaptado a la producción robots con diversas características para producir los productos deseados en las fábricas de forma seriada y rápida.

El uso exitoso de robots en la industria ha llevado al inicio de estudios sobre robots que pueden imitar los movimientos humanos en la comunidad académica, especialmente después de la década de 1980. Estos robots, que se asemejan a los humanos en términos de apariencia y movilidad, se denominan robots humanoides en la literatura. Hoy en día, los robots se utilizan en una amplia gama de áreas, desde la industria del juguete hasta la industria del cine, desde los sistemas de automatización hasta la industria de defensa.

1.2. Definición del Problema.

La movilidad de los servomotores del robot humanoide disponible en el club de Robótica es controlado de manera remota y de no autónoma. Por esto, surge la necesidad de realizar el diseño e implementación de controladores PID en robots humanoides para controlar el movimiento de servomotores y procesando imágenes a través del software MatLab/Simulink

1.3. Objetivo General.

Realizar el diseño e implementación de controladores PID en robots humanoides utilizando MatLab/Simulink.

1.4. Objetivos Específicos.

- a. Describir los fundamentos teóricos de los sistemas de control en la robótica, controladores PID y servocontrol.
- b. Diseñar controladores PID para robot humanoide y del procesamiento de imágenes mediante cámara.

- c. Analizar los resultados obtenidos del posicionamiento del objeto mediante servocontrol en MatLab.

Capítulo 2

Fundamentación teórica.

2.1. Introducción de la mecánica de robots.

La mecánica del robot se divide en dos partes principales, la cinemática del robot y la dinámica del robot. La cinemática de robots; es una ciencia que examina el cambio de posición, velocidad y aceleración dependiente del tiempo que ocurre durante el movimiento de los objetos, sin considerar las fuerzas que crean este movimiento. La dinámica del robot, por otro lado, es una rama de la ciencia que considera los momentos de fuerza que crean estos movimientos mientras examina el movimiento de las extremidades del robot. Gracias a la dinámica del robot, se puede examinar el comportamiento dinámico del robot y se puede diseñar un controlador.

2.2. Fundamentos de sistemas de control en la robótica.

El control de robots es un tema importante en la robótica. El controlador debe usarse para que el procesador final del robot pueda seguir la trayectoria deseada. Los métodos de control como la posición, la velocidad y la fuerza se utilizan en el control de robots. En el control de los robots se utilizan métodos de control como la posición, la velocidad y la fuerza. En el control de posición, se pide al responsable del robot que siga la trayectoria deseada. En el control de velocidad, se realiza el control de velocidad de los motores pertenecientes a cada libertad. El control de par, por su parte, se utiliza para eliminar las perturbaciones en el robot.

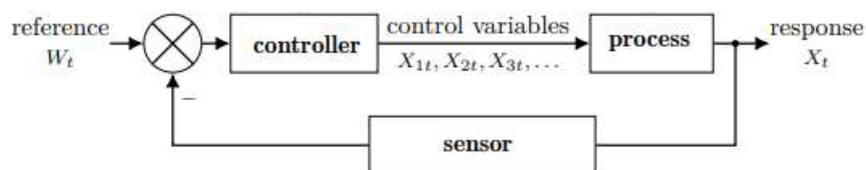


Figura 2. 1: Diagrama de bloques del controlador de bucle cerrado.
Fuente: (Schöning et al., 2022)

Para que el robot realice el movimiento deseado, se requiere un control de circuito cerrado. El valor real del parámetro a controlar se lee con el sensor apropiado y se toma como señal de retroalimentación. Este valor obtenido se compara con el valor de referencia y se aplica al controlador

relevante como una señal de error. El controlador se encarga del control del sistema generando la señal de control adecuada para el parámetro a controlar, tal como se muestra en la figura 2.1.

2.3. Controladores PID.

Como resultado del creciente desarrollo de la informática, los métodos de optimización son cada vez más populares. Por otro lado, los controladores PID son suficientes para resolver un gran número de problemas de control, especialmente cuando la dinámica del sistema es sencilla y los requisitos de rendimiento son limitados. (Kadu & Patil, 2016)

En los sistemas clásicos de control de lazo cerrado (Classical Closed Loop Control Systems, CLCS), el controlador más común utilizado es el controlador proporcional-integral-derivativo (PID). Este amplio uso se debe a que la mayoría de los ingenieros entienden cómo variar los tres parámetros, basándose en el error de control pasado (I), presente (P) y futuro (D) para lograr el comportamiento previsto del CLCS. (Schöning et al., 2022)

El controlador PID (véase la figura 2.2), también llamado regulador PID es un sistema de control, está compuesto por un comparador para observar la diferencia (error) entre la medida y la consigna, y un corrector cuyo algoritmo permite obtener una ley de evolución de la medida del proceso conforme a las especificaciones. Según, Mpanza & Pedro, (2021) permite el control en bucle cerrado de un sistema industrial o de "proceso".

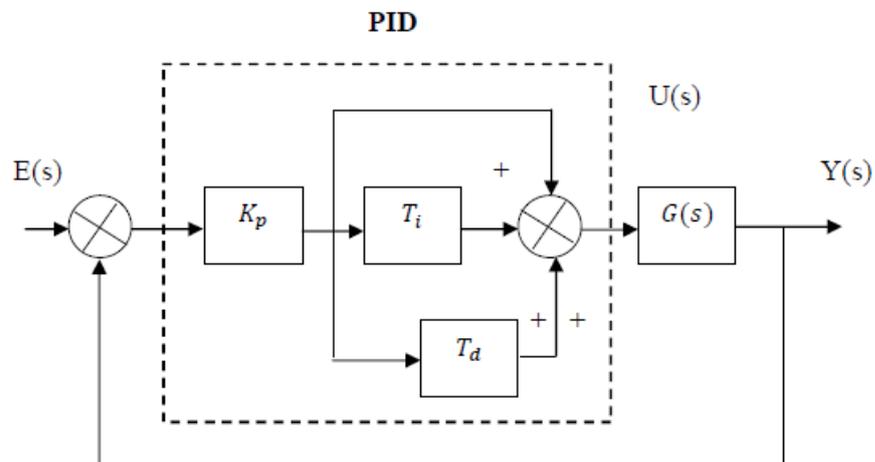


Figura 2. 2: Diagrama de bloques del sistema de control PID.

Fuente: (Kumar et al., 2011)

La ventaja del controlador PID es integrar los efectos positivos de los tres controladores anteriores. La determinación de los coeficientes K_p , T_i y T_d del controlador PID permite mejorar tanto la precisión (T_i como K_p); estabilidad (T_d); la rapidez (T_d, K_p). El ajuste de un PID es generalmente bastante complejo, los métodos prácticos de ajuste permiten obtener buenos resultados.

Sin embargo, los controladores PID no funcionan bien en procesos no lineales ni en procesos con tiempos muertos pequeños. Además, dependiendo de su parámetro, los controladores PID son propensos a sobrepasar y subestimar el punto de consigna o llevarlo a cabo muy lentamente. Como contramedidas, se han optimizado las estrategias de giro de los parámetros, y el diseño del controlador PID simple se ha descompuesto en diseños más complejos, como el controlador PID-P, PI-PD y PI-D. (Schöning et al., 2022)

2.4. Parámetros de controladores PID.

La idea básica de este controlador es generar un comando $u(t)$ dado por el regulador PID, en su forma clásica se describe mediante la ecuación:

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right]$$

Se compone de la suma de tres términos:

- ✓ El término proporcional “P” (proporcional al error): K_p es la ganancia proporcional.

$$P = K_p e(t)$$

- ✓ El término integral “I” (proporcional a la integral del error): T_i es la constante de tiempo de acción integral, integral, en segundos o minutos. Se puede configurar por su inversa (repeticiones por segundo o por minuto).

$$I = K_p \frac{1}{T_i} \int_0^t e(t) dt$$

- ✓ El término derivado "D" (proporcional a la deriva del error): T_d es la constante de tiempo de la parte derivada.

$$D = K_p T_d \frac{de(t)}{dt}$$

Finalmente, los parámetros del controlador PID son la ganancia proporcional K_p , el tiempo integral T_i y el tiempo derivativo T_d , expresados los tiempos en segundos.

2.5. Acción P, I y D.

2.5.1. Acción P (proporcional)

El propósito de la corrección proporcional es reducir significativamente el tiempo de respuesta. Por otro lado, se introduce un overshoot (excederse) y sobre todo la precisión no es excelente ya que hay un error de posición.

a) Ventajas:

- ❖ Corrige al instante, así de rápido.
- ❖ Permite vencer las grandes inercias del sistema.
- ❖ Aumentar la ganancia.

b) Inconvenientes:

- ❖ Amplificación en toda la banda de frecuencia.
- ❖ La precisión no es importante.

2.5.2. Acción I (integral)

El objetivo de esta corrección es cancelar el error de posición, es decir, obtener en régimen permanente la salida igual al setpoint o punto de ajuste (error de posición cero $s=0$). Sin embargo, el tiempo de respuesta es largo.

a) Ventajas:

- ❖ Cancelar error estático (precisión perfecta).
- ❖ Ajuste de variables ruidosas.

b) Inconvenientes:

- ❖ Retraso de fase, la acción integral que puede conducir a la inestabilidad si el corrector está mal colocado.

2.5.3. Acción D (derivador)

El efecto derivado no tiene influencia en el estado estacionario (la derivada es cero), pero no hay un efecto significativo en el estado transitorio.

a) Ventajas:

- ❖ Acelera la respuesta y mejora la estabilidad del sistema.
- ❖ Permite en particular una rápida amortiguación de las oscilaciones debidas a la aparición de una perturbación o a una variación brusca del punto de consigna.

b) Inconvenientes:

- ❖ Aumento de la sensibilidad al ruido.

2.6. Formas del controlador PID.

2.6.1. Forma estándar

La forma estándar de la función de transferencia del controlador PID es:

$$C(p) = K_p \left(1 + \frac{1}{pT_i} + \frac{pT_d}{1 + p\frac{T_d}{N}} \right)$$

La acción derivativa se implementa con un filtro de primer orden, cuya constante de tiempo es N veces menor que T_d . Este filtro permite atenuar el efecto del ruido de alta frecuencia en el control PID, siendo la elección típica entre 10 y 20. La ecuación de $C(p)$ representa la forma convencional comercializada por la mayoría de los fabricantes de reguladores PID.

2.6.2. Forma paralela

La forma paralela de la función de transferencia del controlador PID es:

$$C(p) = K + \frac{K_i}{p} + \frac{pK_d}{1 + \left(\frac{K_d}{K}\right)p}$$

Donde: K , K_i , K_d son constantes. La ventaja de esta forma proviene del hecho de que se puede obtener una acción derivada proporcional, integral o

pura con parámetros finitos del controlador. Por otro lado, los parámetros no tienen una interpretación física obvia.

2.6.3. Forma serie

La forma de la función de transferencia del regulador es:

$$C(p) = \frac{(1 + \tau_1 p)(1 + \tau_2 p)}{\tau_i(1 + \tau_n p)}$$

Donde, τ_1 , τ_2 , τ_i , τ_n constantes de tiempo. Esta forma facilita la correlación de los parámetros del controlador con las constantes de tiempo del proceso. El regulador PID en la forma serie siempre se puede representar como:

$$K_p = \frac{\tau_1 + \tau_2 - \tau_n}{\tau_i}$$

$$T_i = \tau_1 + \tau_2 - \tau_n$$

$$T_d = \frac{\tau_1 \tau_2 - \tau_n \tau_1 - \tau_2 \tau_n + \tau_n^2}{(\tau_1 + \tau_2 - \tau_n)}$$

$$N = \frac{\tau_1 \tau_2 - \tau_n \tau_1 - \tau_2 \tau_n + \tau_n^2}{\tau_n(\tau_1 + \tau_2 - \tau_n)}$$

2.7. Servocontrol visual

Los sensores tienen un lugar importante para que los robots realicen los movimientos deseados. Las personas perciben el mundo exterior con sus cinco órganos de los sentidos y responden a las reacciones de su entorno con estos órganos. Los robots pueden comunicarse con el mundo exterior y los sensores. Hay diferentes tipos y diferentes características de sensores relacionados con esto. El uso de cámaras como sensores en robots se ha incrementado en los últimos años. Especialmente después de la década de 1980, los estudios académicos sobre el uso de la cámara en el control del robot cobraron impulso.

El método de utilizar la cámara como sensor en robots se denomina servocontrol visual (Visual Servo-Control, VSC) en la literatura. En el sentido más general, En el sentido más general, el servocontrol visual consiste en utilizar los datos obtenidos al procesar la imagen obtenida de la cámara con

la ayuda de un ordenador para determinar la señal de retroalimentación necesaria para el control del movimiento del robot.

2.7.1. Posicionamiento de la cámara.

En los sistemas de servocontrol visual, generalmente se utiliza una de las dos posiciones de cámara básicas o combinadas, tal como se muestra en la figura 2.3.

- ❖ Posición de la cámara fija en el extremo funcional
- ❖ Posición fija de la cámara en el espacio de trabajo.
- ❖ Posición de cámara híbrida, que es una combinación de estas dos posiciones de cámara.



Figura 2. 3: configuraciones de cámara-robot más comunes.

Fuente: (Corke, 2020)

En el primer modelo de posición de cámara, la cámara está montada en el extremo funcional. Aquí hay una relación fija conocida entre la pose de la cámara y la pose del funcional final. Esta relación se define como e_{x_c} . La exposición del objeto de destino en relación con el marco de la cámara se indica mediante c_{x_t} . Estas relaciones se muestran en la figura 2.4.

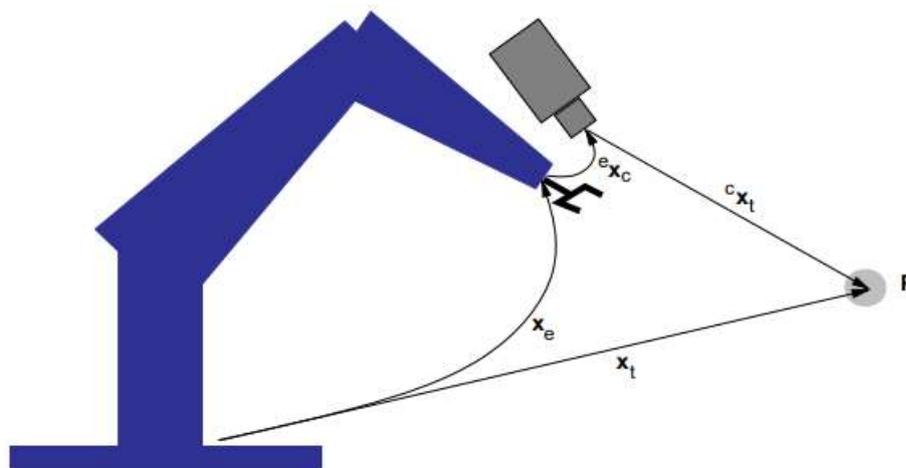


Figura 2. 4: Marcos de coordenadas relevantes; mundo, efector final, cámara y objetivo.

Fuente: (Corke, 2020)

El método basado en la posición dinámica tiene una estructura jerárquica, tal como se muestra en la figura 2.5. En este método de control, la imagen proporcionada por la cámara proporciona puntos objetivo para controlar las variables articulares, y la retroalimentación del sensor de cada articulación se utiliza para estabilizar el robot.

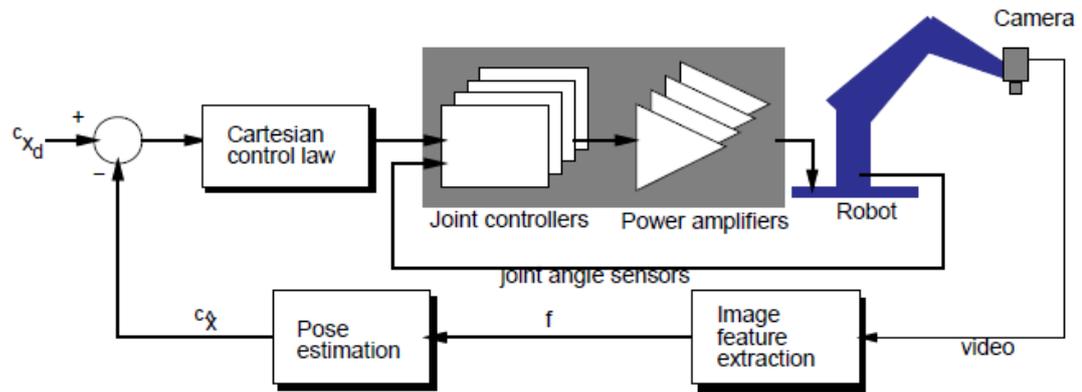


Figura 2. 5: Estructura dinámica de ver y mover basada en la posición.
Fuente: (Corke, 2020)

Capítulo 3

Diseño, Simulación y Resultados obtenidos.

3.1. Descripción del componente práctico.

En este capítulo se dará información sobre el equipamiento del robot humanoide móvil, cuya parte superior es similar a la parte superior del cuerpo de un humano, y cuya parte inferior cuenta con un mecanismo de oruga para asegurar su movimiento en el suelo. Luego, se presentarán los algoritmos de control aplicados al robot y se darán los datos obtenidos en tablas y gráficos.

3.2. Comunicación del robot con la computadora.

Cuando se examina el robot en términos de cinemática y dinámica, se deben usar diferentes tipos de motores en cada articulación. Se ha optado por utilizar un servomotor ya que dispone de su propio microcontrolador y sistema de engranaje (caja de cambios). Se utilizaron servomotores de la marca Dynamixel debido a la capacidad de controlar datos como velocidad, posición, aceleración, etc., para conectar el chasis al cuerpo principal de diferentes maneras y tener una alta relación par-velocidad, tal como se muestra en la figura 3.1.



Figura 3. 1: Servomotor Dynamixel modelo AX-18A utilizado en Robotis Bioloid GP.
Fuente: (Robotis, 2022)

Una de las características más importantes de los motores de la marca Dynamixel es que los motores se pueden conectar entre sí en serie, tal como se muestra en figura 3.2. Se escogió una minicomputadora de escritorio modelo Gigabyte GB-BKi7HA-7500 para realizar todo el software del robot. Esta computadora consume menos energía que otras computadoras de

escritorio. Posee un procesador Intel Core i7-7500, 32 GB de RAM y disco SSD, tal como se muestra en la figura 3.3.

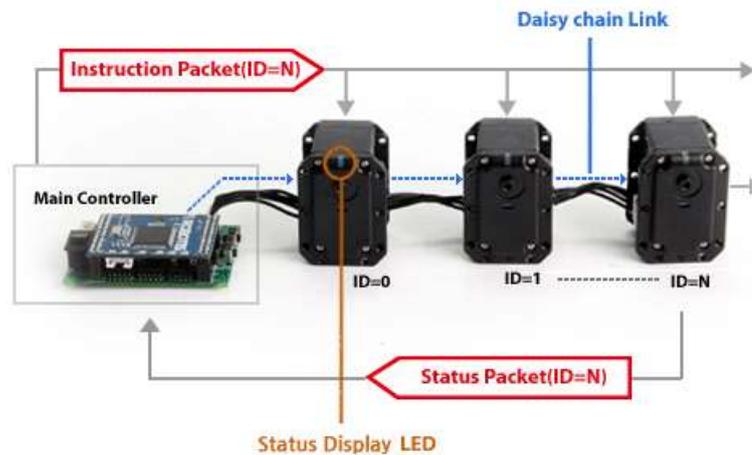


Figura 3. 2: Representación de la conexión en serie de los servomotores AX-18A.
Elaborado por: Autor.



Figura 3. 3: Minicomputadora Gigabyte con memoria 32 GB.
Elaborado por: Autor.

Cada motor realiza comunicación serial rs-485 con la computadora. Dado que la computadora no tiene un puerto rs-485, la comunicación entre motores y motores se realiza a través de USB mediante un convertidor (USB2Dynam-xel) tal como se muestra en la figura 3.3.

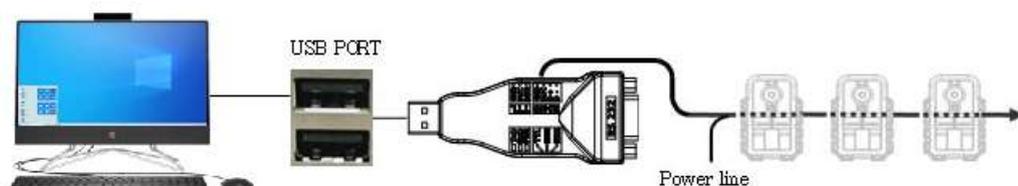


Figura 3. 4: Generación de la señal de entrada PRBS.
Elaborado por: Autor.

El servomotor utilizado tiene un microcontrolador con interfaz RS-485. Por lo tanto, se utiliza el protocolo RS-485 para que cada motor se comunique con la computadora. Los datos a procesar entre la computadora y los servomotores deben enviarse en paquetes. En este protocolo, cada motor tiene un número id (número de identidad). El paquete de datos enviado desde la computadora se procesa en el motor al que pertenece.

La estructura del paquete de datos se muestra en la figura 3.5. Aquí, las dos primeras direcciones (cabeceras) del paquete son estándar y siempre iguales. La tercera dirección del paquete consta del número de identificación del motor, las siguientes direcciones son el parámetro y el último dato es la dirección (suma de verificación) donde se verifica el paquete en busca de defectos.

0xff	0xff	id	Length	Instruction	Parameter1	...	Parameter N	Check Sum
------	------	----	--------	-------------	------------	-----	-------------	-----------

Figura 3. 5: Estructura del paquete de comunicación RS-485.

Elaborado por: Autor.

Los algoritmos de control y movimiento en tiempo real del robot se implementaron utilizando el programa Matlab-Simulink. La comunicación con los motores se proporciona mediante el uso de la biblioteca RapidSTM32 del programa Matlab-Simulink.

3.3. Algoritmos de control aplicados en el robot.

Se han realizado aplicaciones para demostrar la capacidad de trabajo del robot humanoide. Aquí, en primer lugar, se determinó el algoritmo de control básico del robot y luego se llevó a cabo la aplicación de servocontrol visual. En primer lugar, se aplica un controlador PID de controlador clásico y un controlador PID de 2 grados de libertad (DoF) a cada brazo del robot.

Los resultados obtenidos como resultado de las aplicaciones de ambos controladores se tabulan en tablas, se grafican los resultados y finalmente se comparan. En la sección 3.3.1. se proporciona información detallada sobre este estudio. En la sección 3.3.2., se dan expresiones matemáticas básicas del algoritmo del servocontrol visual basado en la posición del robot.

Para eliminar las distorsiones generales en el procesamiento de imágenes causadas por el movimiento de la cabeza del robot, se apoyaron las redes neuronales artificiales. Se proporciona información detallada en la sección 3.3.2. En la misma sección, se sostienen en la mano objetos de destino de color verde con forma rectangular de diferentes alturas y la mano derecha o izquierda del robot captura el objeto. Los resultados obtenidos se dan en tablas y gráficos.

3.3.1. Diseño de controlador PID aplicados al robot

En este estudio, para determinar el algoritmo básico de control del robot, se aplicaron controladores PID y PID de 2DoF a cada brazo del robot y los resultados obtenidos se compararon numérica y gráficamente. La posición angular actual de los servomotores requerida para el control del robot se puede obtener en tiempo real con el sensor dentro del motor. Las posiciones angulares obtenidas de cada motor se insertan en la cinemática de avance del brazo robótico para calcular la posición real $(x_g - y_g - z_g)$ del punto final del brazo.

Se comparan los valores de la posición real calculada y la posición de referencia $(x_r - y_r - z_r)$ proporcionados por el usuario desde el exterior y se obtiene el valor de error (e). Este valor de error obtenido se inserta en el controlador y se obtiene la señal de control adecuada (u). Los valores de control se insertan en las ecuaciones cinemáticas inversas del brazo del robot y se calcula el valor angular de cada articulación correspondiente a este valor de control. Estos valores calculados también se utilizan para accionar los motores. El diagrama de bloques del controlador PID se muestra en la figura 3.6.

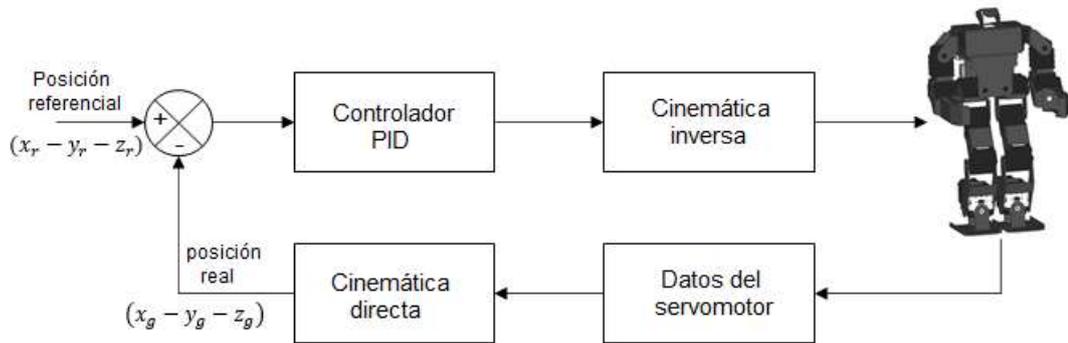


Figura 3. 6: Diagrama de bloques del controlador PID para robot humanoide.

Elaborado por: Autor.

El controlador PID de 2DoF tiene el lazo de control PID descrito anteriormente, el brazo de alimentación directa del controlador es el controlador PD y el brazo de retroalimentación consiste en el controlador PID, tal como se muestra en la figura 3.7. La información detallada se puede encontrar en la sección 3.5.

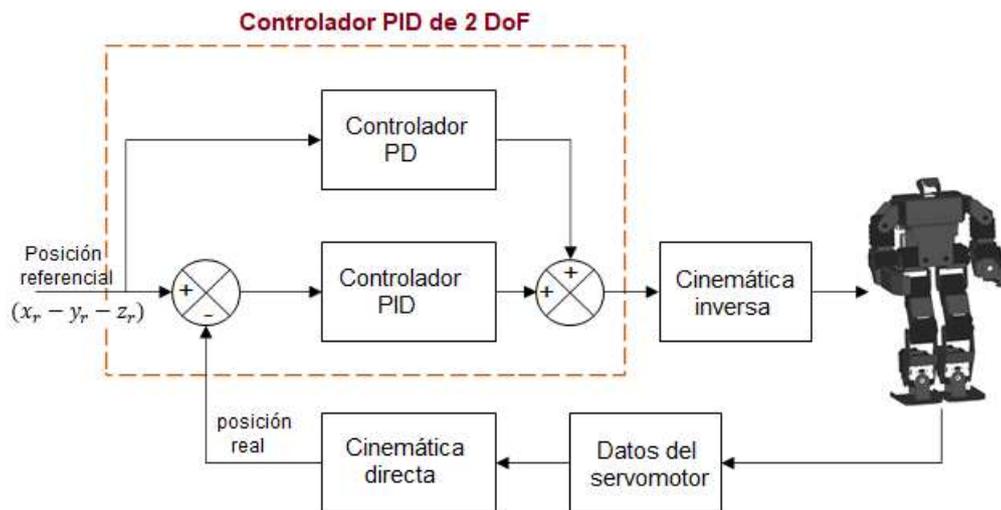


Figura 3. 7: Diagrama de bloques del controlador PID de 2 DoF para robot humanoide.

Elaborado por: Autor.

En ambas aplicaciones, los puntos de referencia $(x_r - y_r - z_r)$, que son los puntos finales del brazo robótico, se seleccionan aleatoriamente. El tiempo de muestreo fue de 10 ms y el software implementado se ejecutó durante 60 s. El brazo del robot avanza solo $\pi/8$ radianes desde la articulación del codo durante los primeros 3 s. Después de 3 s, se obtuvo la señal de control adecuada utilizando los valores de posición real y de referencia. Esta señal de control obtenida se puso en cinemática inversa y se calcularon los valores necesarios para accionar los motores.

Los valores de posición de referencia se seleccionan de la siguiente manera.

$$x_r = [15 \ 21 \ 26]$$

$$y_r = [24 \ 27 \ 31]$$

$$z_r = [-1 \ -3 \ -6 \ -9 \ -12]$$

Se realizó trabajo experimental tanto para algoritmos de control como de error relativo y se calcularon valores RMS de acuerdo con el valor de referencia del punto final. Estos valores calculados se presentan los más relevantes en la tabla 3.1.

Tabla 3. 1: Valores de posición del robot humanoide.

	Valores de posición referencial	PID	% error del PID	PID de 2 DoF	% error del PID 2 DoF	Valor eficaz del PID	Valor eficaz del PID de 2 DoF
x	15	14.844	1.039	15.014	-0.096	7.226	4.609
y	24	24.094	-0.391	24.002	-0.007	3.701	1.959
z	-12	-12.083	-0.689	-12.061	-0.510	2.697	1.532
x	15	14.824	1.170	14.851	0.995	7.197	4.692
y	27	27.045	-0.166	27.029	-0.108	2.912	2.044
z	-12	-12.062	-0.513	-12.059	-0.495	1.899	1.507
x	15	14.847	1.023	14.983	0.116	7.261	5.059
y	31	30.991	0.028	31.021	-0.067	2.868	2.250
z	-12	-12.030	-0.252	-12.018	-0.153	1.829	1.374
x	21	20.911	0.425	20.969	0.145	7.105	3.558
y	24	24.049	-0.203	24.008	-0.035	2.561	1.663
z	-12	-12.030	-0.251	-12.011	-0.093	1.899	1.828
x	21	20.917	0.396	21.073	-0.348	7.215	3.494
y	27	27.020	-0.073	26.918	0.306	2.591	1.832
z	-12	-11.981	0.156	-12.000	-0.001	1.865	1.691
x	21	21.027	-0.131	21.038	-0.180	7.243	3.432
y	31	30.879	0.390	30.973	0.088	2.682	2.199
z	-12	-11.925	0.627	-11.945	0.460	1.674	1.618
x	26	25.931	0.266	25.937	0.241	7.325	2.561
y	24	23.946	0.226	23.998	0.009	2.180	1.653
z	-12	-11.939	0.508	-12.011	-0.089	1.877	1.962
x	26	26.108	-0.415	26.101	-0.387	7.378	2.562
y	27	26.756	0.904	26.912	0.325	2.278	1.908
z	-12	-11.774	1.887	-11.844	1.305	1.748	1.864
x	26	26.230	-0.883	26.047	-0.180	7.354	3.587
y	31	30.764	0.761	30.916	0.272	2.554	2.237
z	-12	-11.821	1.489	-11.912	0.735	1.595	1.633

Elaborado por: Autor.

Se escogen valores aleatorios de las posiciones referenciales, tales valores obtenidos son $x_r = 26$, $y_r = 27$ y $z_r = -1$ y que se muestran en la figura 3.8, 3.9 y 3.10 respectivamente.

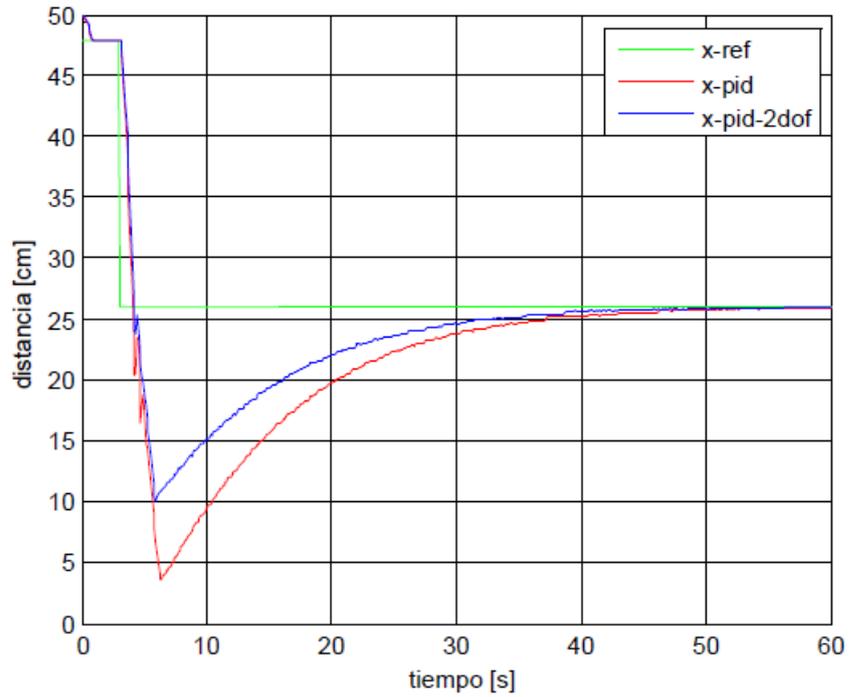


Figura 3. 8: Gráficos del controlador PID y PID de 2 DoF obtenidos para $x=26$.
Elaborado por: Autor.

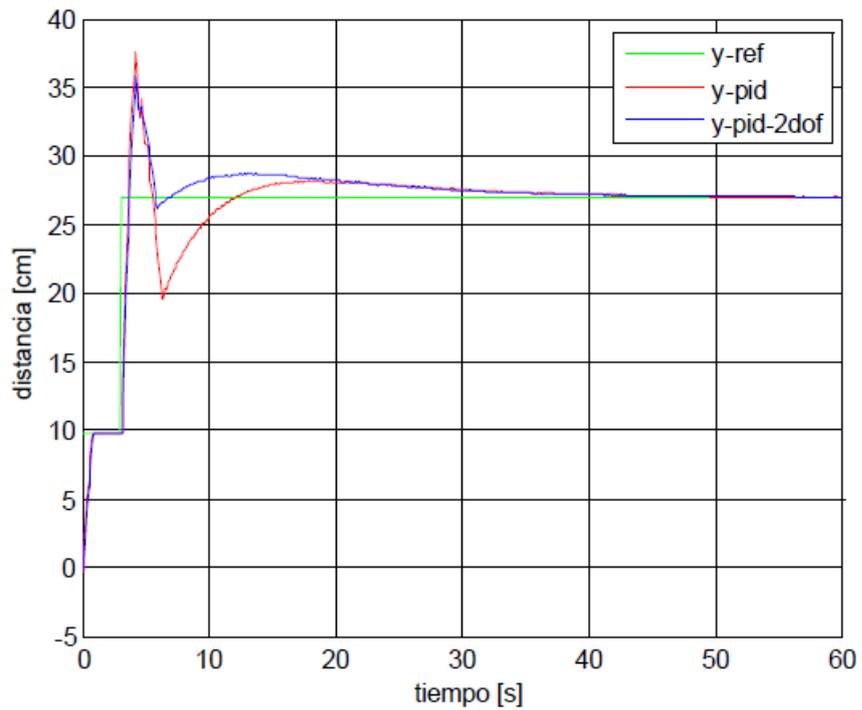


Figura 3. 9: Gráficos del controlador PID y PID de 2 DoF obtenidos para $y=27$.
Elaborado por: Autor.

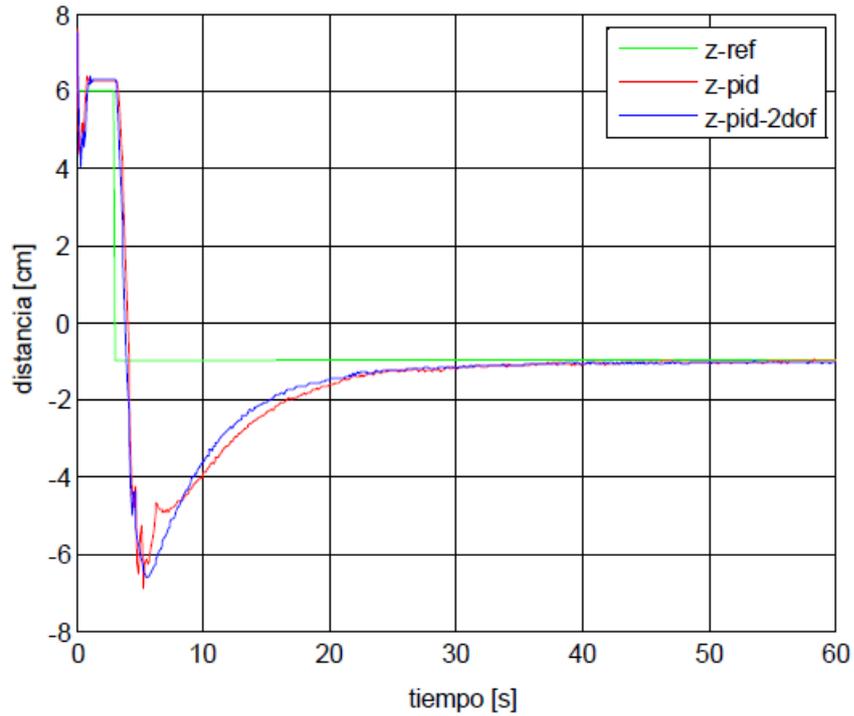


Figura 3. 10: Gráficos del controlador PID y PID de 2 DoF obtenidos para $z=-1$.
Elaborado por: Autor.

Ahora se consideran $x_r = 15$, $y_r = 31$ y $z_r = -3$ y se grafican los datos en las figuras 3.11, 3.12 y 3.13 respectivamente.

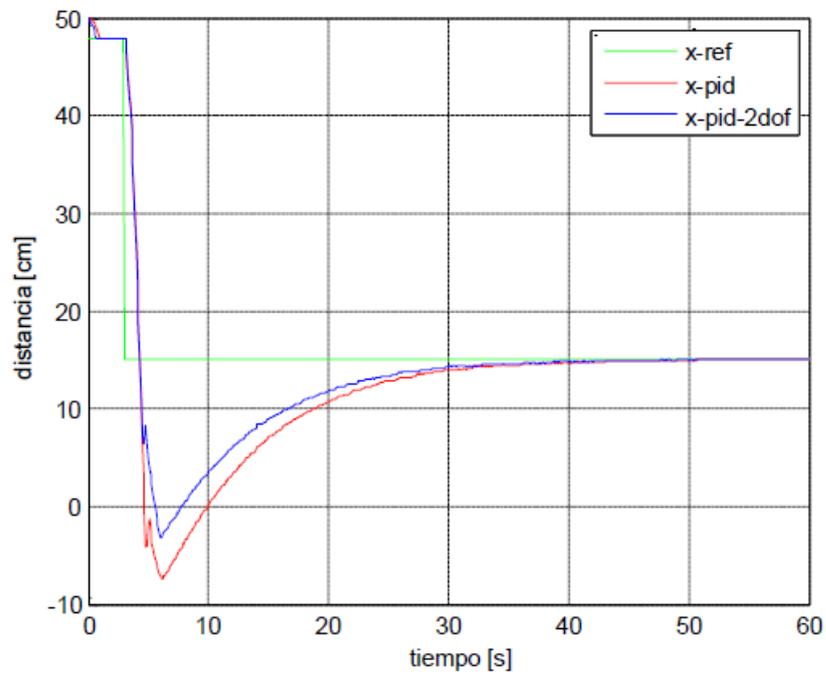


Figura 3. 11: Gráficos del controlador PID y PID de 2 DoF obtenidos para $x=15$.
Elaborado por: Autor.

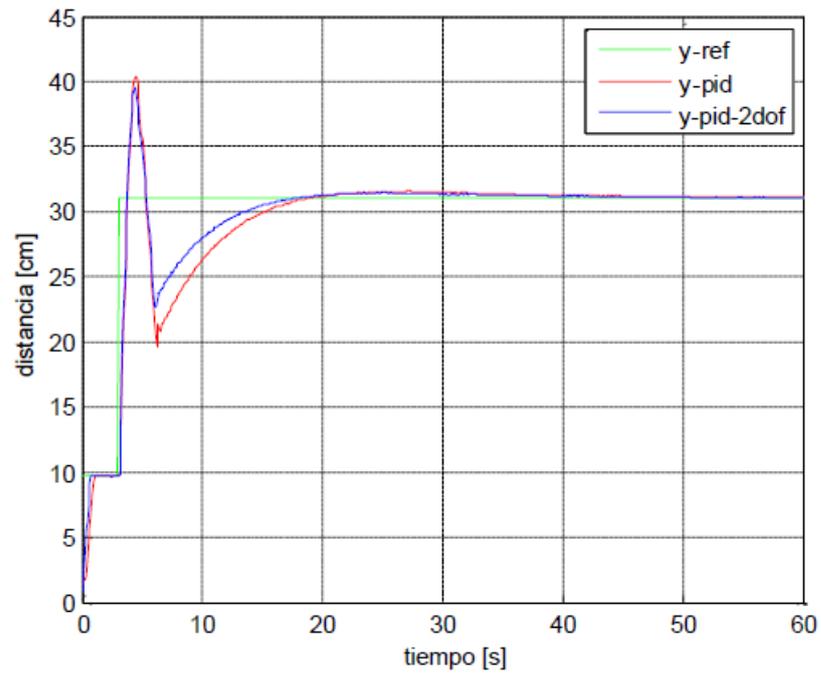


Figura 3. 12: Gráficos del controlador PID y PID de 2 DoF obtenidos para $y=31$.
Elaborado por: Autor.

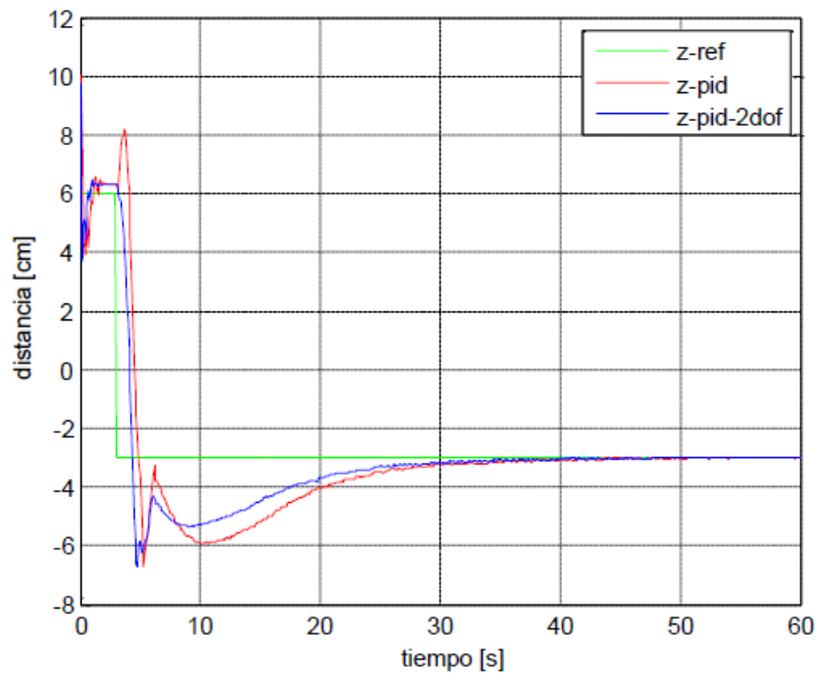


Figura 3. 13: Gráficos del controlador PID y PID de 2 DoF obtenidos para $z=-3$.
Elaborado por: Autor.

En este estudio, se compararon los controladores PID y PID de 2 DoF aplicados al brazo del robot. Los porcentajes de error y los valores RMS (efectivos) totales se calcularon para x-y-z cuando el punto final del brazo robótico captó la referencia. El mayor valor de error entre el punto final y la

referencia es calculado como $x_{2pid} = 0.99\%$, $x_{pid} = 2.12\%$, $y_{2pid} = 0.85\%$, $y_{pid} = 0.96\%$, $z_{2pid} = 3.31\%$ y $z_{pid} = 4.4\%$ para PID de 2 DoF y PID, respectivamente.

Además, los valores RMS totales máximos se obtuvieron como $x_{RMS_{pid}} = 7.6$, $x_{RMS_{pid2}} = 6.3$, $y_{RMS_{pid}} = 3.7$, $y_{RMS_{2pid}} = 2.6$ y $z_{RMS_{pid}} = 2.1$, $z_{RMS_{2pid}} = 1.9$, respectivamente. Cuando se examinan los gráficos anteriores, el tiempo de establecimiento de PID 2 DoF según PID es 1-8 s para x, 1-4 s para y, 2-7 s para z y el sobreimpulso máximo calculado es $x_{2pid} = 13\%$, $x_{pid} = 106\%$, $y_{2pid} = 21\%$, $y_{pid} = 38\%$ y $z_{2pid} = 42\%$, $z_{pid} = 92\%$. Cuando se examinaron los resultados obtenidos, se determinó que el PID de 2 DoF tuvo más éxito que el PID. Por lo tanto, se eligió el controlador PID de 2 DoF.

3.3.2. Aplicaciones de servo visuales basadas en posición

El método desarrollado en la literatura para utilizar la cámara como sensor y permitir que el robot realice el movimiento deseado se denomina servocontrol visual. En el capítulo 2 se proporciona información detallada sobre el servocontrol visual. En esta y las siguientes aplicaciones, se ha elegido un algoritmo de servocontrol visual basado en la posición ya que la cámara está fijada al cuerpo del robot.

Para calcular matemáticamente la distancia del objeto a la cámara, se deben obtener algunas propiedades del objeto mediante técnicas de procesamiento de imágenes. Para esto, se utiliza el bloque "dispositivo de video" para obtener datos de la cámara conectada a la computadora con la "ToolBox de procesamiento de imágenes" de Matlab-Simulink. En este bloque se eligió a RGB como formato de video y 640x480 como tamaño de imagen, tal como se muestra en la figura 3.14.

Se obtuvo un valor único al multiplicar el valor RGB de la imagen tomada por separado en formato RGB de la cámara de video con ciertos coeficientes. Aquí, los números de piso se eligen para separar el color verde

del fondo. Se obtuvo una imagen en blanco y negro pasando el valor obtenido por encima del valor umbral apropiado. Se realizó un proceso morfológico de apertura para eliminar los efectos distorsionadores de la imagen en blanco y negro obtenida y obtener una imagen de mayor calidad. Así, se obtuvo una imagen en blanco y negro utilizando técnicas de procesamiento de imágenes a partir de la imagen en color obtenida con la cámara, tal como se muestra en la figura 3.14.

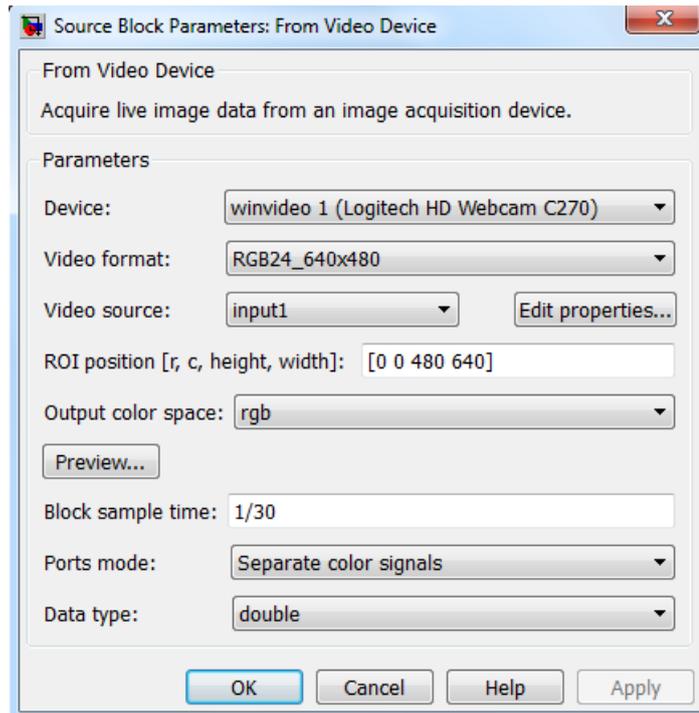


Figura 3. 14: Bloque de dispositivo de vídeo Matlab/Simulink.
Elaborado por: Autor.

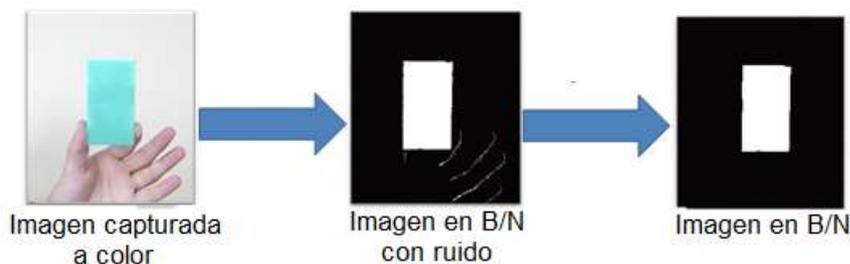


Figura 3. 15: Obtención de una imagen en B/N a partir de una imagen a color
Elaborado por: Autor.

Para calcular las coordenadas de posición del objeto con respecto al robot, se obtuvo la imagen deseada mediante procesamiento de imágenes. De acuerdo con la ley de la lente, la relación entre el objeto y la imagen se muestra en la figura 3.16. Aquí, el eje z es el eje óptico de la cámara.

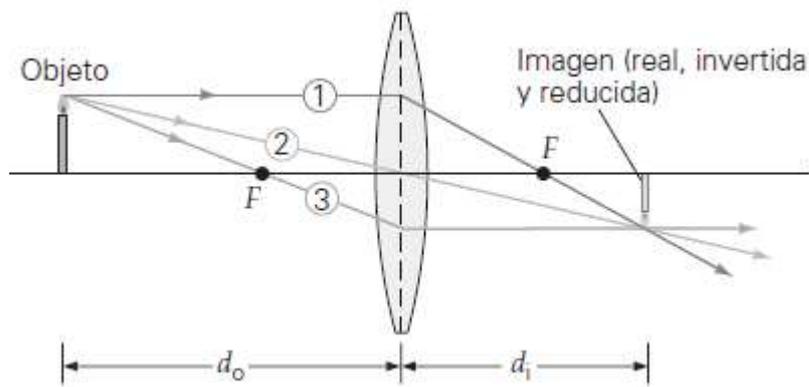


Figura 3. 16: Descripción gráfica de la proyección de imagen del objeto en la cámara.

Fuente: (Young et al., 2018)

De acuerdo con la ley de la lente, las variables se expresan de la siguiente manera.

$$\frac{1}{d_o} + \frac{1}{d_i} = \frac{1}{f}$$

Donde,

d_o , es la distancia al objeto

d_i , es la distancia a la imagen

f , es la distancia focal

La altura de la imagen y_i se relaciona con la altura del objeto y_o de la siguiente manera:

$$M = \frac{y_i}{y_o} = \frac{f}{f - d_o}$$

Utilizando el objeto de la figura 3.15, las relaciones entre el sistema de coordenadas (x, y, z) del objeto y $x_i - y_i$ en el plano de la imagen se expresan de la siguiente manera:

$$x_i = \frac{fx}{f - d}$$

$$y_i = \frac{fy}{f - d}$$

Usando las ecuaciones anteriores, se calcula d_o de la siguiente manera:

$$d_o = f \left(1 - \frac{1}{M} \right)$$

Con esta ecuación, la distancia del objeto a la cámara se puede calcular matemáticamente.

Las coordenadas de posición del centro del objeto (x_c, y_c, z_c) se calculan a partir de la imagen en B/N obtenida mediante técnicas de procesamiento de imágenes con las ecuaciones anteriores. Luego, las coordenadas calculadas según la cámara se mueven al eje básico del brazo $(x_r - y_r - z_r)$, tal como se muestra en la figura 3.17.

$$T_c^r = T_c^k T_k^r$$

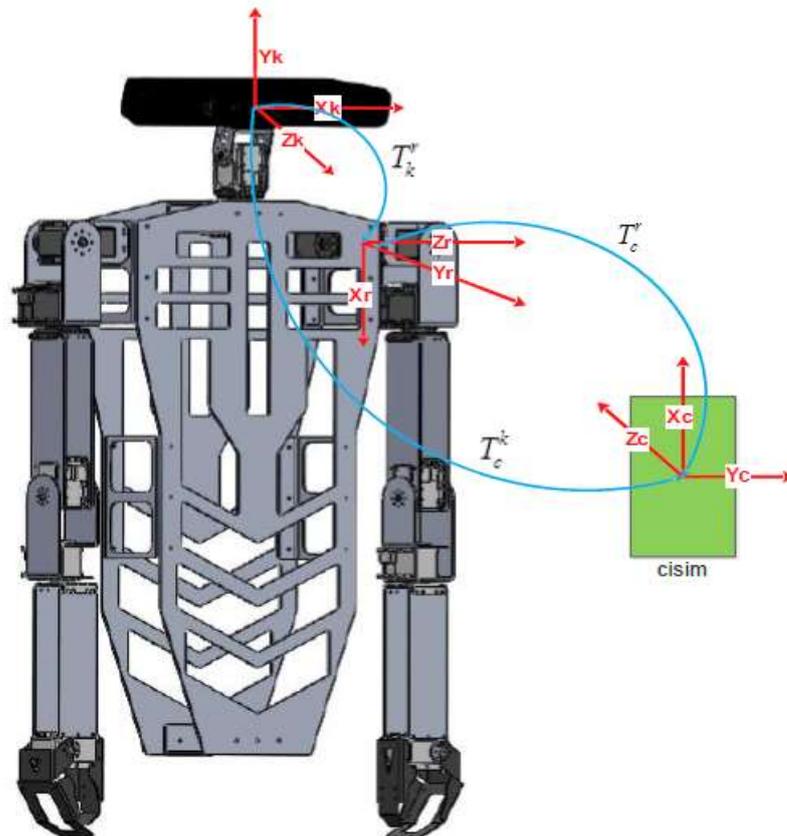


Figura 3. 17: Conjuntos de ejes de objeto-cámara-brazo.
Elaborado por: Autor.

Los 3 elementos de la última 4ª columna de la matriz anterior muestran las coordenadas de posición del brazo en relación con el eje fundamental. Así, se calculan las coordenadas de posición del objeto según el punto base del brazo. Estas coordenadas obtenidas se utilizan en el control del brazo.

Los ejes del objeto, la cámara y el brazo del robot se toman tal cual como en la figura 3.17.

Cuando las expresiones matemáticas obtenidas anteriormente se aplican al robot, se puede calcular la distancia del objeto a la cámara. Sin embargo, la distancia del objeto a la cámara se calcula de manera diferente dependiendo de su posicionamiento en el lado opuesto, derecho e izquierdo del robot. La razón de esto es el deterioro de la imagen del objeto como resultado del movimiento de la cabeza del robot, tal como se muestra en la figura 3.18.

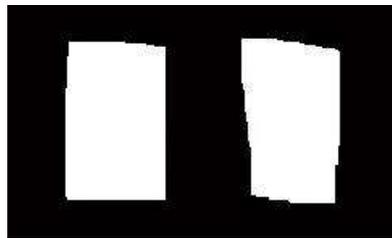


Figura 3. 18: Imagen que ha sido distorsionada respecto a la imagen original.
Elaborado por: Autor.

Para solucionar este inconveniente, se ha obtenido el apoyo de redes neuronales artificiales. La relación entre la altura requerida para el objeto a la misma distancia y la altura obtenida se define como sigue. Pixeles

$$k_d = \frac{h_o (\text{pixeles})}{h_{ao} (\text{pixeles})}$$

Donde: k_d es el parámetro de corrección, h_o es la altura del objeto, y h_{ao} es la altura actual del objeto.

La entrada de la red neuronal artificial se selecciona como la altura del objeto (píxeles), y la salida se elige el parámetro de corrección k_d . En este contexto, la cabeza del robot se coloca de forma que mire hacia delante y el objeto se sitúa a una distancia de 30-35-40 cm, respectivamente. Los valores obtenidos en este momento se aceptaron como el valor de la altura requerida del objeto. A continuación, se colocó el objeto a determinados intervalos, primero hacia la izquierda y luego hacia abajo, y se obtuvieron los valores de altura del objeto actual calculados por la cámara. El parámetro k_d

se calculó con los valores de altura obtenidos. Así pues, el conjunto de datos se preparó de forma que los valores de altura del objeto necesarios para el entrenamiento de las redes neuronales artificiales eran de entrada y el parámetro de corrección era la salida Kd.

Este conjunto de datos se utilizó para el entrenamiento de redes neuronales artificiales. Los resultados obtenidos se observan en la figura 3.19. Aquí se recopilaron 50 datos para el entrenamiento de Redes Neuronales Artificiales. Mediante el uso del módulo de redes neuronales artificiales del programa Matlab se realizó el entrenamiento de la red y se obtuvo el bloque Simulink.

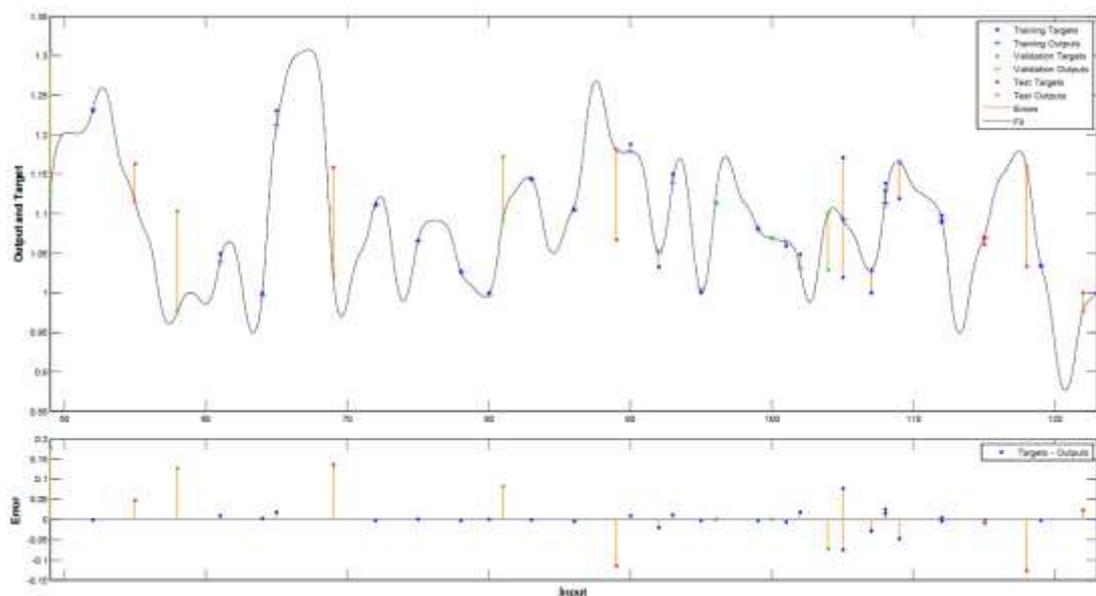


Figura 3. 19: Imagen que ha sido distorsionada respecto a la imagen original.
Elaborado por: Autor.

En esta aplicación, el objeto sostenido por el ser humano está destinado a ser capturado por el soporte izquierdo o derecho del robot. El objeto es de color verde y forma rectangular con una altura de 7-9-11 cm. La información de posición angular obtenida de los servomotores se insertó en las ecuaciones cinemáticas directas y se obtuvo la información de posición real. Con este valor se comparan las coordenadas de posición del objeto reducidas al eje básico del brazo y se calcula el valor de error. Este error se le da al controlador PID 2-DoF, que controla el movimiento del brazo, y se calcula la señal de control requerida para la cinemática inversa. Al utilizar

en el momento en que el robot agarra el objeto y se calcularon los valores de % de error de la posición x-y-z. Estos valores se muestran en la tabla 3.2 para el brazo izquierdo.

Tabla 3. 2: Valores para brazo izquierdo del robot humanoide.

Tiempo (s)	x_r	x_a	y_r	y_a	z_r	z_a	$x_{error}(\%)$	$y_{error}(\%)$	$z_{error}(\%)$
Para objeto de 7 cm de altura									
8.5	10.30	9.97	30.54	31.13	-6.22	-6.25	3.18	-1.94	-0.36
9.4	10.06	10.07	27.68	27.45	-2.35	-2.33	-0.13	0.80	0.74
9.6	13.19	13.47	28.99	28.82	-3.64	-3.60	-2.14	0.62	1.16
9.9	14.22	14.38	30.82	30.85	-4.81	-4.84	-1.07	-0.07	-0.54
9.1	3.79	3.77	30.88	30.90	4.74	4.76	0.57	-0.06	-0.46
9.7	12.40	12.69	31.96	32.14	-11.78	-11.81	-2.27	-0.58	-0.19
8.7	10.27	10.64	32.36	32.95	-8.35	-8.24	-3.59	-1.81	1.26
8.0	6.12	6.34	29.03	30.03	-8.78	-8.85	-3.58	-3.44	-0.77
Para objeto de 9 cm de altura									
9.3	5.23	5.22	29.73	30.04	-10.23	-10.18	0.23	-1.04	0.62
9.7	11.50	11.04	32.11	32.34	-6.60	-6.01	4.04	-0.70	-0.07
9.5	10.06	10.25	30.03	30.18	-3.66	-3.65	-1.89	-0.50	0.17
8.6	5.45	5.50	28.10	28.61	-5.42	-5.45	-0.78	-1.83	-0.61
8.7	7.20	7.26	31.20	31.02	-9.85	-9.79	-0.79	0.61	0.58
9.6	10.85	11.20	27.85	27.94	-4.14	-4.11	-3.17	-0.30	0.66
9.6	9.14	9.17	27.03	27.57	-12.05	-12.09	-0.33	-1.99	-0.37
9.4	17.58	18.66	30.26	31.51	-8.35	-8.28	-6.13	-4.15	0.88
Para objeto de 11 cm de altura									
9.4	8.67	8.71	33.51	33.75	-2.85	-2.80	-0.44	-0.74	1.71
9.1	7.52	7.66	28.65	29.16	-9.17	-9.12	-1.81	-1.78	0.60
9.1	0.81	0.77	27.71	27.33	-9.04	-9.38	4.38	1.39	-3.84
8.7	8.12	7.88	33.44	34.15	-4.82	-4.85	2.94	-2.12	-0.66
8.8	3.37	3.37	29.82	30.49	-4.97	-4.96	-0.09	-2.24	0.18
9.0	6.70	6.75	31.91	32.81	-7.04	-6.99	-0.68	-2.82	0.67
8.2	0.62	0.63	30.86	31.69	0.65	0.65	-2.32	-2.71	0.15
8.9	12.72	13.11	27.16	27.19	-7.67	-7.67	-3.05	-0.11	1.39

Elaborado por: Autor.

Los gráficos obtenidos para la altura de 7 cm y las coordenadas del objetivo $x=1.73$, $y=29.73$ y $z=-10.23$, se muestran en las figuras 3.21, 3.22 y 3.23.

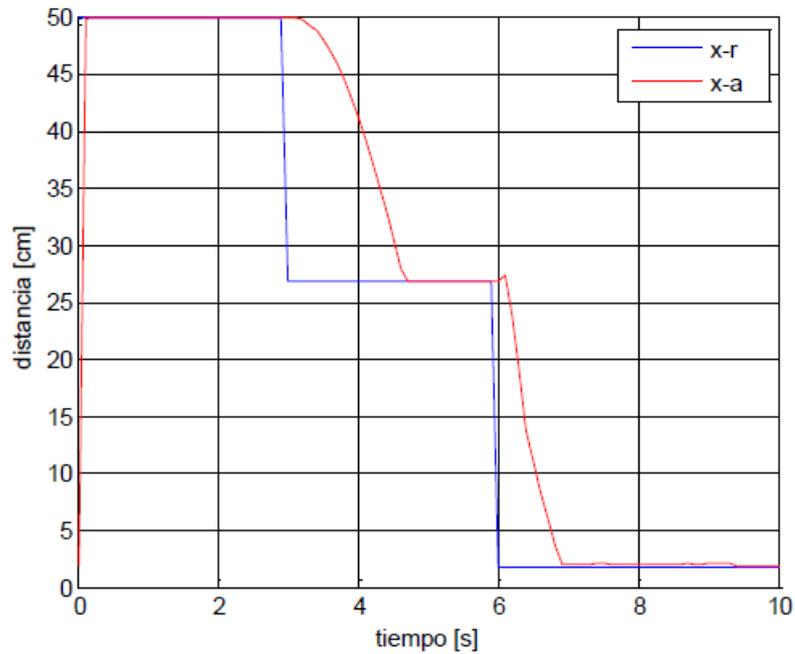


Figura 3. 21: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $x=1.73$.
Elaborado por: Autor.

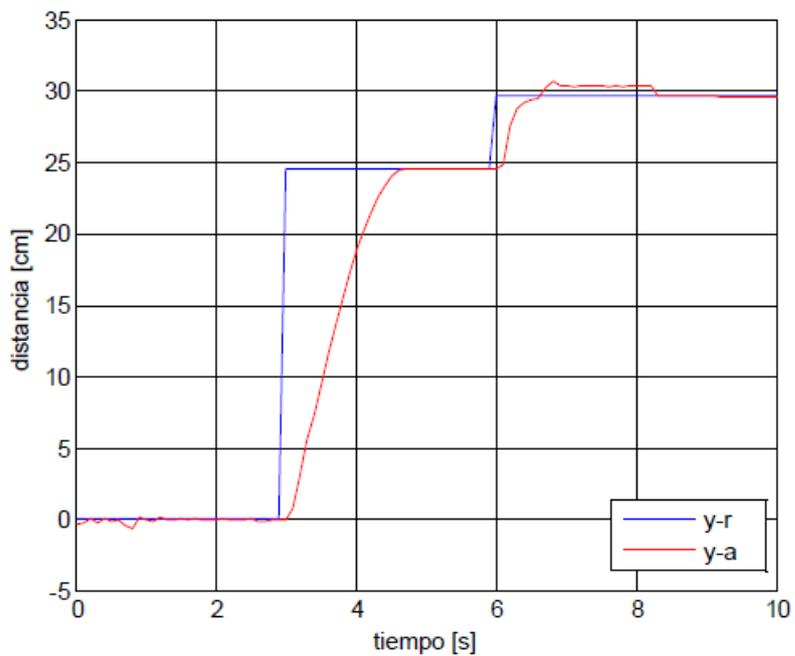


Figura 3. 22: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $y=29.73$.
Elaborado por: Autor.

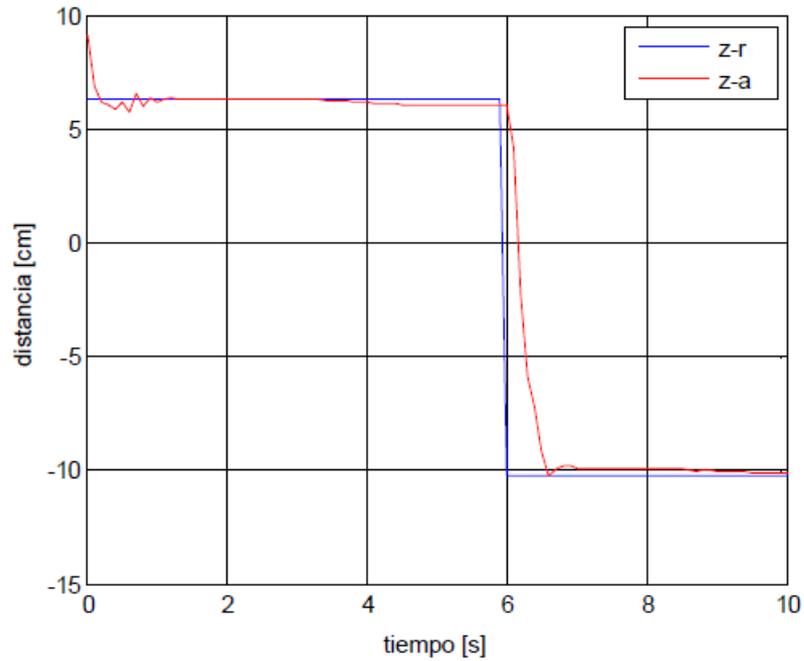


Figura 3. 23: Gráfica de la distancia vs tiempo del objeto de 7 cm obtenido para la posición $z=-10.23$.
Elaborado por: Autor.

Los gráficos obtenidos para la altura de 9 cm y las coordenadas del objetivo $x=10.06$, $y=27.68$ y $z=-2.35$, se muestran en las figuras 3.24, 3.25 y 3.26.

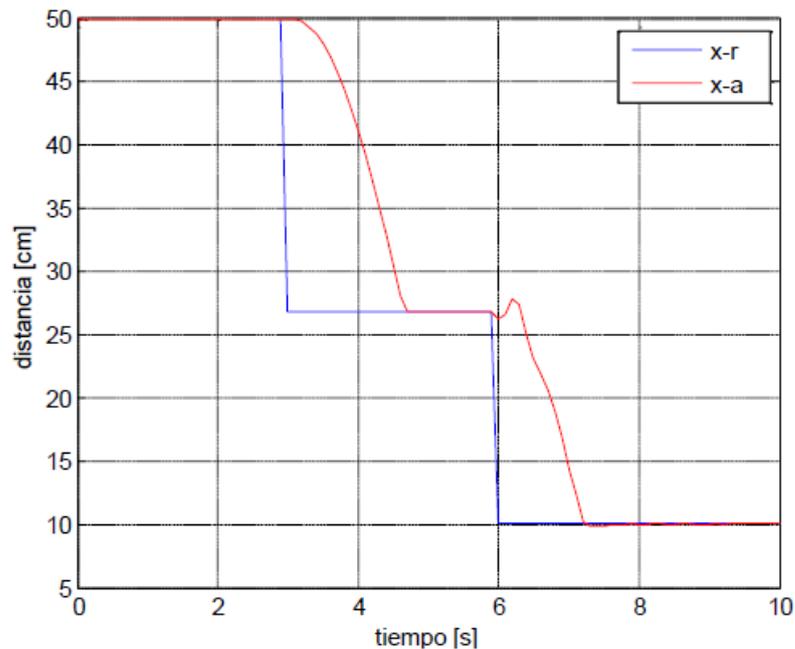


Figura 3. 24: Gráfica de la distancia vs tiempo del objeto de 9 cm obtenido para la posición $x=10.06$.
Elaborado por: Autor.

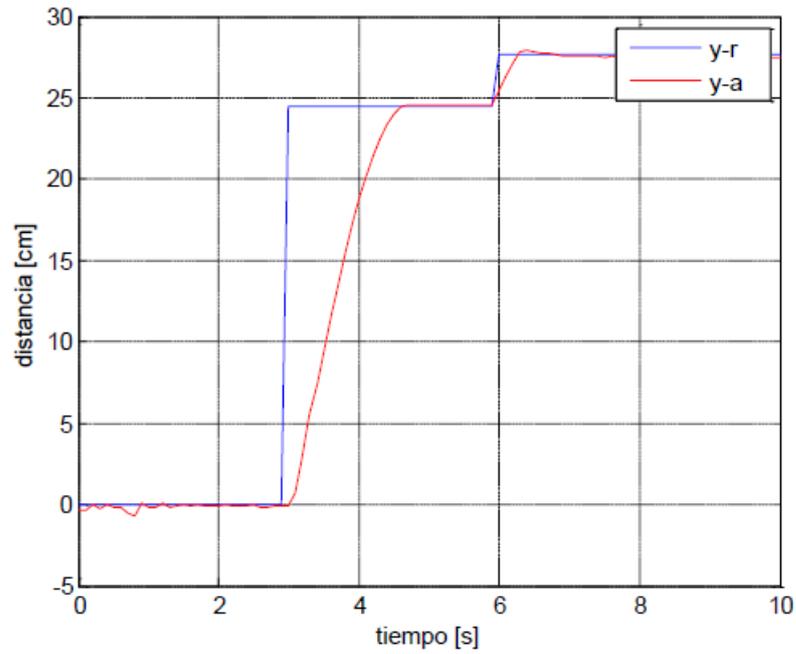


Figura 3. 25: Gráfica de la distancia vs tiempo del objeto de 9 cm obtenido para la posición $y=27.68$.
Elaborado por: Autor.

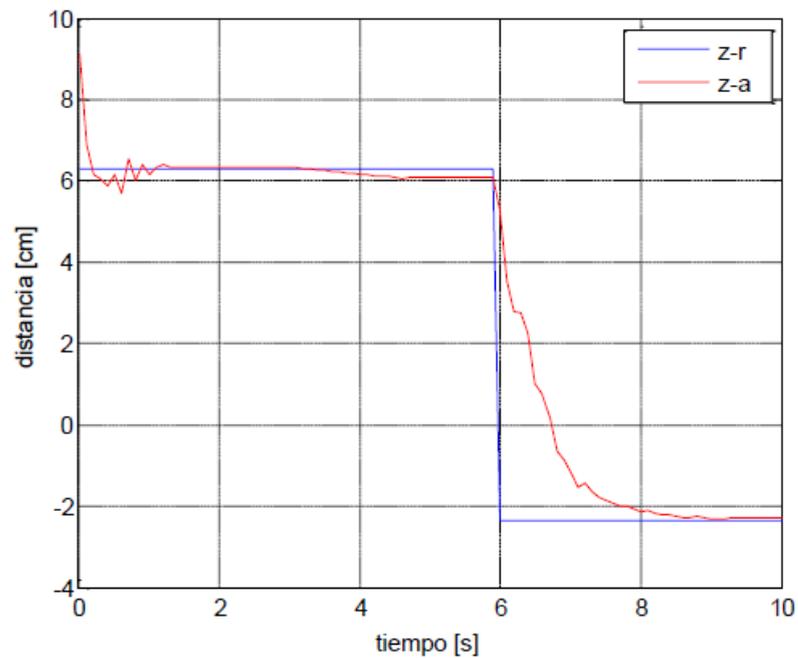


Figura 3. 26: Gráfica de la distancia vs tiempo del objeto de 9 cm obtenido para la posición $z=-2.35$.
Elaborado por: Autor.

Los gráficos obtenidos para la altura de 11 cm y las coordenadas del objetivo $x=8.67$, $y=33.51$ y $z=-2.85$, se muestran en las figuras 3.27, 3.28 y 3.29.

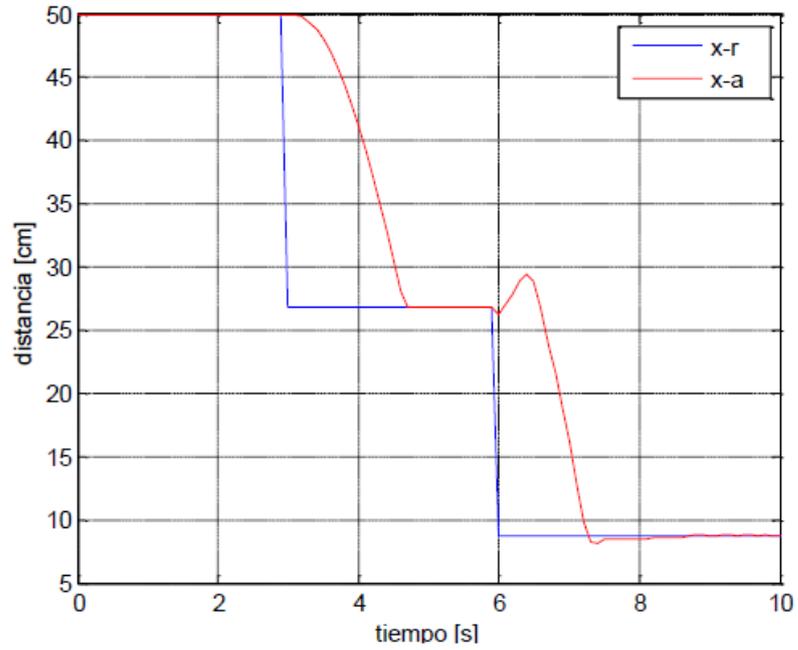


Figura 3. 27: Gráfica de la distancia vs tiempo del objeto de 11 cm obtenido para la posición $x=8.67$.
Elaborado por: Autor.

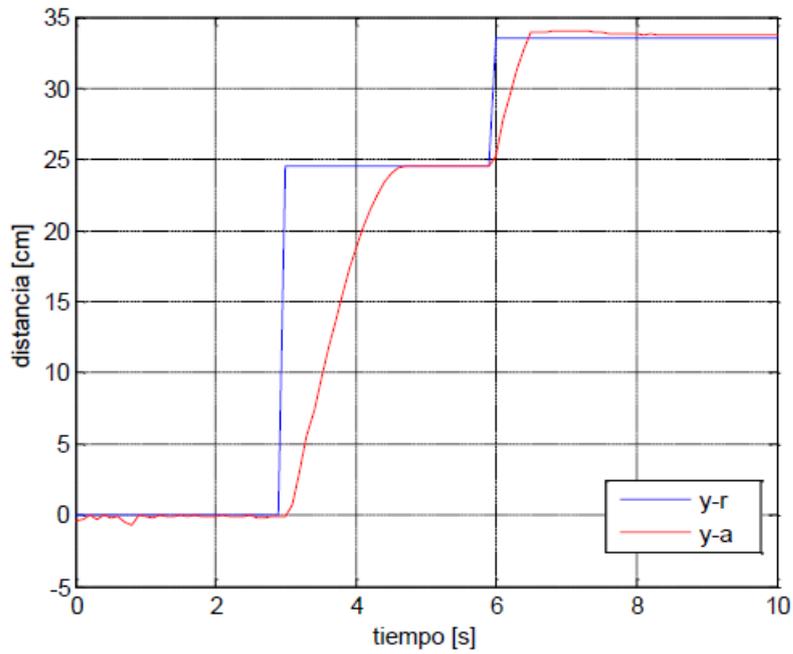


Figura 3. 28: Gráfica de la distancia vs tiempo del objeto de 11 cm obtenido para la posición $y=33.51$.
Elaborado por: Autor.

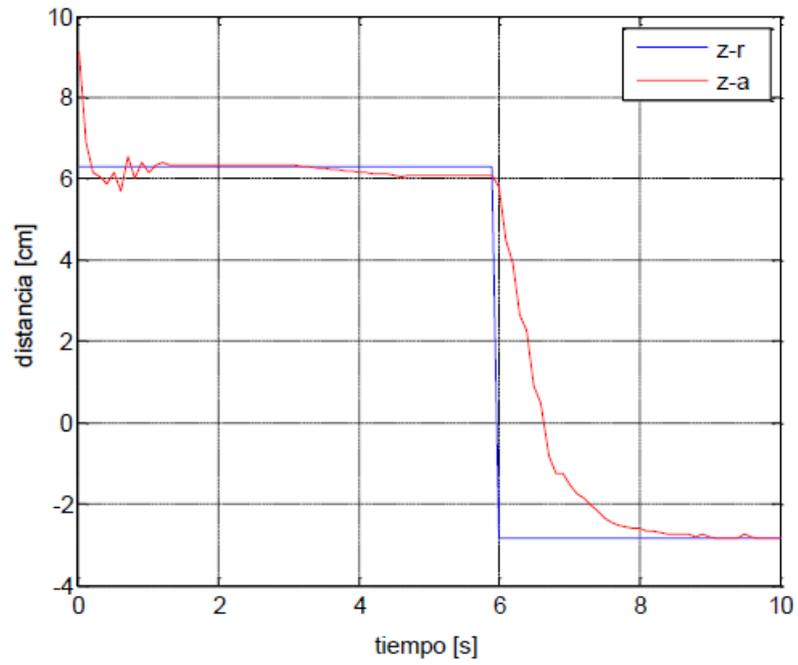


Figura 3. 29: Gráfica de la distancia vs tiempo del objeto de 11 cm obtenido para la posición $z=-2.85$.
Elaborado por: Autor.

Conclusiones

- El regulador PID se adapta bien a la mayoría de los procesos de tipo industrial, es relativamente robusto con respecto a las variaciones en los parámetros del proceso, a pesar de su uso. En este estudio, el algoritmo de servocontrol visual basado en la posición se aplica al brazo robótico de 6 grados de libertad en tiempo real y se realiza un agarre de objeto autónomo. Ha producido robots humanoides móviles adecuados para este propósito.
- Cada brazo del robot tiene un total de 6 grados de libertad, tres en la región del hombro, dos en la región del codo y uno en la pinza. Hay 2 grados de libertad en la cabeza del robot que le permiten girar de arriba hacia abajo y de izquierda a derecha. El robot tiene un total de 14 grados de libertad. El servomotor de la marca Dynamixel se utiliza para cada libertad del robot. Estos motores contienen un microcontrolador y se pueden conectar en serie entre sí. El software del robot se realizó en el programa Matlab/Simulink. Dado que la computadora y los motores están conectados a través de RS-485, y se utiliza el protocolo de comunicaciones RS-485.
- Se necesitan ecuaciones cinemáticas inversas para que los brazos del robot se muevan al punto deseado en el espacio de trabajo. Para ello, en primer lugar, se obtuvieron las ecuaciones cinemáticas directas de los brazos del robot con el método D-H. Luego, las ecuaciones cinemáticas inversas de los brazos se derivan usando el método algebraico.

Recomendaciones.

- En la aplicación de servocontrol visual basado en posición se ha determinado que la imagen obtenida debido al movimiento de la cabeza del robot está distorsionada. Se puede superar este problema, mejorando el coeficiente de corrección diferencial (k_d) con el apoyo de redes neuronales artificiales para mejorar la imagen.
- Se pueden utilizar 2 cámaras para calcular mejor matemáticamente la distancia del objeto al robot. Además, la parte de oruga del robot se puede activar y se puede lograr su movimiento en el suelo. Por lo tanto, también puede capturar objetos de la distancia de un brazo. En aplicaciones donde crece el espacio de trabajo, la cámara existente puede resultar insuficiente, por lo que puede ser necesaria una cámara más profesional.

Bibliografía.

- Corke, P. (2020). *The Blue Book*. Peter Corke. <https://petercorke.com/books/the-blue-book/>
- Kadu, C. B., & Patil, C. Y. (2016). Design and Implementation of Stable PID Controller for Interacting Level Control System. *Procedia Computer Science*, 79, 737–746. <https://doi.org/10.1016/j.procs.2016.03.097>
- Kumar, V., Nakra, B. C., & Mittal, A. (2011). A Review on Classical and Fuzzy PID Controllers. *International Journal of Intelligent Control and Systems*, 16(3), 170–181.
- Mpanza, L. J., & Pedro, J. O. (2021). Optimised Tuning of a PID-Based Flight Controller for a Medium-Scale Rotorcraft. *Algorithms*, 14(6), 178. <https://doi.org/10.3390/a14060178>
- Robotis, Y. (2022). *ROBOTIS Bioloïd GP e-Manual*. ROBOTIS E-Manual. <https://emanual.robotis.com/docs/en/dxl/ax/ax-18a/>
- Schöning, J., Riechmann, A., & Pfisterer, H.-J. (2022). AI for Closed-Loop Control Systems—New Opportunities for Modeling, Designing, and Tuning Control Systems. *arXiv:2201.06961 [cs, eess]*. <http://arxiv.org/abs/2201.06961>
- Young, H. D., Freedman, R. A., Ford, A. L., & Sears, F. W. (2018). *Física Universitaria con física moderna 2*. Pearson. <https://elibro.net/ereader/elibrodemo/113376>



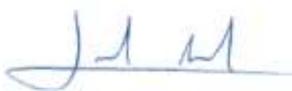
DECLARACIÓN Y AUTORIZACIÓN

Yo, **Ávila Vera, Jhon Alexander** con C.C: # 085018467-2 autor del **Componente Práctico del Examen Complexivo: Diseño e implementación de controladores PID en robots humanoides utilizando MatLab/Simulink** previo a la obtención del título de **INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 10 de marzo del 2022

f. 

Nombre: Ávila Vera, Jhon Alexander

C.C: 085018467-2

REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

TÍTULO Y SUBTÍTULO:	Diseño e implementación de controladores PID en robots humanoides utilizando MatLab/Simulink		
AUTOR(ES)	Ávila Vera, Jhon Alexander		
REVISOR(ES)/TUTOR(ES)	M. Sc. Romero Paz, Manuel de Jesús		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Facultad de Educación Técnica para el Desarrollo		
CARRERA:	Ingeniería Electrónica en Control y Automatismo		
TÍTULO OBTENIDO:	Ingeniero Electrónico en Control y Automatismo		
FECHA DE PUBLICACIÓN:	10 de marzo del 2022	No. DE PÁGINAS:	36
ÁREAS TEMÁTICAS:	Fundamentos de Robótica. Sistemas de Control Digital		
PALABRAS CLAVES/ KEYWORDS:	Robótica, Controladores, Servocontrol, Posicionamiento, Parámetros, Simulación.		
RESUMEN/ABSTRACT (150-250 palabras):	<p>El presente documento presenta el desarrollo del componente práctico del examen complejo denominado "Diseño e implementación de controladores PID en robots humanoides utilizando MatLab/Simulink". En el capítulo 1, se presenta la parte descriptiva de este componente práctico. En el capítulo 2 se discute los fundamentos teóricos de sistemas de control en la robótica y del servocontrol visual. Se ha visto que los servosistemas pueden tener defectos, una precisión insuficiente, una estabilidad demasiado relativa, un tiempo de reacción demasiado lento, un rebasamiento excesivo. En el capítulo 3 se presenta el diseño, simulación y análisis de resultados. Por ejemplo, para la determinación del controlador se basó en las especificaciones que se pueden establecer ya sea en velocidad, precisión, tiempo de reacción, entre otras. En este caso el controlador PID (Proporcional-Integral-Derivativo) se encarga de regular el servomotor utilizado en un robot humanoide. También, se realizó el posicionamiento de objetos mediante una cámara integrada al robot humanoide utilizando el servocontrol visual.</p>		
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
CONTACTO CON AUTOR/ES:	Teléfono: +593-9-89325890	E-mail: john_avila98@hotmail.com	
CONTACTO CON LA INSTITUCIÓN: COORDINADOR DEL PROCESO DE UTE	Nombre: Palacios Meléndez Edwin Fernando		
	Teléfono: +593-9-67608298		
	E-mail: edwin.palacios@cu.ucsg.edu.ec		
SECCIÓN PARA USO DE BIBLIOTECA			
Nº. DE REGISTRO (en base a datos):			
Nº. DE CLASIFICACIÓN:			
DIRECCIÓN URL (tesis en la web):			